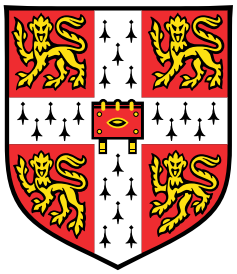
Causal Representation Learning for Latent Space Optimization



Wenlin Chen

Department of Engineering
University of Cambridge

This dissertation is submitted for the degree of
Master of Philosophy in Machine Learning and Machine Intelligence

St Edmund's College August 2021

This thesis is dedicated to my loving parents, who made my studies in Cambridge possible.

Declaration

I, Wenlin Chen of St Edmund's College, being a candidate for the MPhil in Machine Learning and Machine Intelligence, hereby declare that this report and the work described in it are my own work, unaided except as may be specified below, and that the report does not contain material that has already been used to any substantial extent for a comparable purpose.

My co-supervisor Austin Tripp provides me with the code for a SMILES-LSTM auto-encoder, a data loader for molecular datasets, and several drug property functions. The Python code for the IGCI causal discovery algorithm and HSIC conditional/unconditional independence tests is written by myself, based on their official MATLAB implementations. All the other code used in this thesis does not rely on any previously written software other than standard Python packages for machine learning.

Word Count: 12,893

Wenlin Chen
August 2021

Acknowledgements

I would like to thank my supervisors Dr José Miguel Hernández-Lobato and Austin Tripp for their countless ideas, helpful guidance, and insightful feedback on my theoretical and experimental results throughout the entire project. It has been a great pleasure working with them, from which I have learned a great deal about research. I am also thankful to Chaochao Lu and Gregory Flamich for useful discussions about causal inference and variational auto-encoders.

I would also like to express my gratitude to my parents for their unconditional support.

Abstract

In this thesis, we study causal representation learning for latent space optimization, which allows for robust and efficient generation of novel synthetic data with maximal target value. We assume that the observed data was generated by a few latent factors, some of which are causally related to the target and others of which are spuriously correlated with the target and confounded by an environment variable. Our proposed method consists of three steps, which exploits the structure of the causal graph that describes the assumed underlying data generating process. In the first step, we recover the true data representation (i.e., the latent factors from which the observed data originated). We obtain novel identifiability theory, showing that the true data representation can be recovered up to simple transformations by a generalized version of identifiable variational auto-encoders. In the second step, we identify the causal latent factors of the target, for which we propose a practical causal inference scheme that employs (conditional) independence tests and causal discovery algorithms. Our method does not require having access to the true environment variable, which overcomes a major limitation of existing causal representation learning approaches in the literature. In the final step, we query latent points that correspond to data points with high target values by intervening upon the causal latent factors using standard latent space optimization techniques. We empirically evaluate and thoroughly analyze our method on three different tasks, including a chemical design task. We show that our method can successfully recover the true data representation in the finite data regime and correctly identify the causal latent factors of the target, which results in state-of-the-art performance for black-box optimization.

Table of contents

List of figures

List of tables

Nomenclature

Acronyms/Abbreviations

ANM Additive Noise Model

DGM Deep Generative Model

DLVM Deep Latent Variable Model

HSIC Hilbert Schmidt Independence Criterion

ICA Independent Component Analysis

ICE-BeeM Identifiable Conditional Energy-Based Deep Model

ICM Independent Causal Mechanism

IGCI Information Geometric Causal Inference

IRM Invariant Risk Minimization

iVAE Identifiable Variational Auto-Encoder

KL Kullback–Leibler

LSO Latent Space Optimization

LSTM Long Short Term Memory

MC Monte Carlo

MCC Mean Correlation Coefficient

PNLCM Post-Nonlinear Causal Model

ReLU Rectified Linear Unit

SMILES Simplified Molecular Input Line Entry System

VAE Variational Auto-Encoder

Chapter 1

Introduction

1.1 Motivation

Deep neural networks are powerful tools for learning useful representation (Bengio et al., 2013), which is attributed to the successes of various downstream tasks in computer vision (Chen et al., 2020; Krizhevsky et al., 2012), natural language processing (Devlin et al., 2018; Mikolov et al., 2013; Vaswani et al., 2017), computational chemistry (Gómez-Bombarelli et al., 2018; Jumper et al., 2021), and so on. However, the data representation learned by deep neural networks usually only captures statistical associations between variables and completely ignores their causal relationships. Without knowing the underlying causal mechanism, deep learning models tend to exploit easy-to-fit spurious correlations within the training data to solve downstream tasks and thus often fail to generalize to out-of-distribution settings at test time (Lu et al., 2021). To see this, we consider a famous hypothetical task of camel-cow image classification (Beery et al., 2018). Imagine that we are given a set of labelled images of camels and cows. Unfortunately, due to selective biases, most of the pictures of camels were taken in deserts, while many pictures of cows were taken on green pastures (see Figure 1.1). If we train a convolutional neural network on this dataset to solve the classification problem, it turns out that the model would learn to use the spurious correlation between the landscape color in the image and the class label of the image to make predictions. As a results, this model would perform poorly on a test set collected in a different environment (e.g., a test set in which many pictures of cows were taken in beaches).

Arjovsky et al. (2019) formulate a scenario of causal representation learning that takes into account the impact of environment $\mathbf{e}$. In this setting, the joint distribution of the observed data $\mathbf{x}$ and the target of interest y varies across different environments, but there exists a causal relationship between some features of $\mathbf{x}$ and y which remains invariant across dif-


Fig. 1.1 Typical training images for the hypothetical camel-cow classification task.

ferent environments. In the camel-cow classification example, the relationship between the landscape color in the image and the class label of the image changes across different environments, but the causal relationship between the shape of the animal and the class label is invariant across different environments. In this thesis, we aim to learn and identify such invariant causal representation, since it allows for efficient and robust inference, reasoning and prediction in downstream tasks and has the ability to generalize to out-of-distribution settings.

While most works concerning causal representation learning in the literature focus on learning invariant predictors (Arjovsky et al., 2019; Lu et al., 2021; Mitrovic et al., 2020; Muandet et al., 2013), we instead consider black-box optimization as our downstream task. The goal is to generate novel synthetic data with maximal target value, staring from a given initial dataset. We argue that causal representation learning can be naturally combined with latent space optimization which optimizes in the latent space of a deep generative model. We will demonstrate this on a couple of black-box optimization problems, including a chemical design task, on which we achieve robust, efficient, and state-of-the-art optimization performance.

1.2 Thesis Contributions

In this thesis, we present a framework for efficient and robust latent space optimization using causal representation learning, based on the ideas from Lu et al. (2021) and Tripp et al. (2020). Our main contributions are as follows:

- We present novel identifiability theorems for identifiable VAEs with non-factorized conditional priors[1], which enables us to recover the true underlying data representation with reference to the target.

[1]This contributes to the proofs of Theorem 4 and Theorem 5 in Lu et al. (2021), helping address an issue in the first version of this paper.

- We propose a simple yet effective practical causal inference scheme for identifying causal latent factors from the true data representation.
- Our proposed framework does not require having access to the ground truth environment variable, which overcomes a major limitation of existing invariant causal representation learning methods in the literature.
- We demonstrate the identifiability of our generalized identifiable VAEs on a synthetic dataset and perform a thorough analysis of our proposed framework on image optimization and chemical design tasks.

1.3 Thesis Outline

The structure of the remainder of this thesis is as follows.

Chapter 2 establishes the theoretical background for this thesis, introducing the concepts and ideas of 1) variational auto-encoders and its identifiability issues, 2) causal inference and causal representation learning, and 3) latent space optimization for black-box optimization. This chapter is concluded by a formulation of the research problem considered in this thesis, which brings together the three seemingly irrelevant topics discussed in this chapter.

Chapter 3 presents a novel inference and learning scheme for deep latent variable models with non-factorized conditional priors, called generalized identifiable variational auto-encoders. Novel identifiability theorems are developed, showing that this scheme is guaranteed to recover the true latent variable up to simple transformations. The proofs of these theorems can be found in Appendix B.

Chapter 4 describes a framework for latent space optimization with causal representation learning, which employs 1) generalized identifiable variational auto-encoders for recovering the true data representation, 2) a newly proposed practical causal inference scheme for identifying the causal latent factors of the target from the true data representation, and 3) latent space optimization techniques which works with causal latent factors.

Chapter 5 presents empirical evaluation and analysis of our framework on 1) a synthetic dataset, 2) an image optimization task, and 3) a chemical design task, which shows that our method can successfully recover the true data representation in the finite data regime and correctly identify the causal latent factors of the target, which results in robust, efficient, and

state-of-the-art black box optimization performance.

Chapter 6 summarizes the findings in this thesis and point out some interesting directions for future work.

Chapter 2

Background

In this chapter, we introduce some fundamental concepts, ideas and tools in machine learning and statistics, upon which the rest of the thesis will build. We also formulate the research problem considered and state any necessary assumptions needed in this thesis.

2.1 Variational Auto-Encoders

There have been many developments and advances in leveraging probabilistic methods and deep learning for generative modelling and representation learning. In this thesis, we will use variational auto-encoders (VAEs) (Kingma and Welling, 2013; Rezende et al., 2014) as a tool for recovering the true data representation (i.e., the true latent variable from which the observed variable originated). In this section, we give a brief introduction to the VAE framework and discuss the identifiability issue of deep latent variable models (DLVMs) trained by VAEs.

2.1.1 The VAE Framework

Let $\mathbf{x} \in \mathscr{X} \subset \mathbb{R}^d$ be an observed variable and $\mathbf{z} \in \mathscr{Z} \in \mathbb{R}^n$ a latent variable ($n \leq d$). VAEs provide an efficient framework for inference and learning in DLVMs, for which the joint distribution is given by

$$p_\theta(\mathbf{x}, \mathbf{z}) = p_\theta(\mathbf{x}|\mathbf{z})p_\theta(\mathbf{z}), \tag{2.1}$$

where $p_\theta(\mathbf{z})$ is the prior distribution over the latent variable, the likelihood $p_\theta(\mathbf{x}|\mathbf{z}) \coloneqq p_\varepsilon(\mathbf{x} - \mathbf{f}(\mathbf{z}))$, defined by an independent noise distribution $p_\varepsilon(\varepsilon)$ and a mixing function $\mathbf{f} : \mathscr{Z} \to \mathscr{X}$ parameterized by a neural network, specifies the generating process $\mathbf{x} = \mathbf{f}(\mathbf{z}) + \varepsilon$,

and $\theta \in \Theta$ are the parameters of the model. This induces a rich class of flexible models, which allows us to model highly complex marginal distributions over the observed variable:

$$p_\theta(\mathbf{x}) = \int_{\mathscr{Z}} p_\theta(\mathbf{x}|\mathbf{z}) p_\theta(\mathbf{z}) d\mathbf{z}. \tag{2.2}$$

In this model, the true data generating process of a dataset $\mathscr{D} = \{\mathbf{x}^{(1)}, \cdots, \mathbf{x}^{(N)}\}$ can be described as drawing $\mathbf{z}_*^{(i)} \sim p_{\theta^*}(\mathbf{z})$ and $\mathbf{x}^{(i)} \sim p_{\theta^*}(\mathbf{x}|\mathbf{z}_*^{(i)})$ independently for each i, where $\mathbf{z}_*^{(i)}$ can be seen as the true but unobserved underlying representation of $\mathbf{x}^{(i)}$, and θ^* are the true parameters which are unknown to us. It is non-trivial to learn the parameters θ of this model from a set of observed data $\mathscr{D}$, since one has to work with the marginal distribution $p_\theta(\mathbf{x})$ (2.2), which is given by an intractable integral due to the neural network used for modelling the mixing function $\mathbf{f}$ in the likelihood $p_\theta(\mathbf{x}|\mathbf{z})$.

The VAE framework instead works with a tractable variational lower bound $\mathscr{F}$ of the intractable log marginal likelihood:

$$\begin{aligned}
\mathbb{E}_{p(\mathbf{x})}[\log p_\theta(\mathbf{x})] &\geq \mathbb{E}_{p(\mathbf{x})}\left[\log p_\theta(\mathbf{x}) - \mathrm{KL}(q_\phi(\mathbf{z}|\mathbf{x})||p_\theta(\mathbf{z}|\mathbf{x}))\right] && (2.3)\\
&= \mathbb{E}_{p(\mathbf{x})}\left[\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}[\log p_\theta(\mathbf{x}|\mathbf{z})] - \mathrm{KL}(q_\phi(\mathbf{z}|\mathbf{x})||p_\theta(\mathbf{z}))\right] && (2.4)\\
&:= \mathscr{F}(\theta, \phi), && (2.5)
\end{aligned}$$

where $p(\mathbf{x})$ is the (unknown) true data distribution, $q_\phi(\mathbf{z}|\mathbf{x})$ is a variational approximation to the intractable posterior distribution over latent variable $p_\theta(\mathbf{z}|\mathbf{x})$, and the inequality in (2.3) holds by the non-negativity of KL divergence. The KL divergence term in (2.4) regularizes the variational posterior to be close to the prior, and the expected likelihood term measures the (negative) expected reconstruction error. Note that the VAE framework employs amortized inference, since the variational parameters ϕ have a fixed size and are shared across all data points. Also, a mean-field approximation is used for the variational posterior, which is defined by a factorized Gaussian distribution $q_\phi(\mathbf{z}|\mathbf{x}) = \mathscr{N}(\mathbf{z}; \mu_\phi(\mathbf{x}), \mathrm{diag}(\sigma_\phi(\mathbf{x})^2))$ with mean and variance parameterized by neural networks.

In practice, we resort to Monte Carlo (MC) methods to estimate the variational lower bound (2.4). In each iteration, $p(\mathbf{x})$ is estimated by the empirical data distribution given by a mini-batch $\tilde{\mathscr{D}}$ sampled from $\mathscr{D}$, and the expectation over $q_\phi(\mathbf{z}|\mathbf{x})$ is estimated using a single

sample $\mathbf{z} \sim q_\phi(\mathbf{z}|\mathbf{x})$. This gives us an unbiased estimator of the variaitonal lower bound:

$$\mathscr{F}(\theta,\phi) \approx \frac{1}{|\tilde{\mathscr{D}}|} \sum_{\mathbf{x}\in\tilde{\mathscr{D}}} \left[\log p_\theta(\mathbf{x}|\mathbf{z}) - \log q_\phi(\mathbf{z}|\mathbf{x}) + \log p_\theta(\mathbf{z})\right], \quad \mathbf{z} \sim q_\phi(\mathbf{z}|\mathbf{x}). \tag{2.6}$$

The parameters θ and ϕ are learned jointly by maximizing (2.6) using stochastic gradient-based optimization techniques such as Adam (Kingma and Ba, 2014). In order to be able to compute the gradient of (2.6) with respect to ϕ, we employ the reparameterization trick (Kingma and Welling, 2013; Rezende et al., 2014) to generate samples from $q_\phi(\mathbf{z}|\mathbf{x})$, which considers $\mathbf{z} \sim q_\phi(\mathbf{z}|\mathbf{x})$ as being transformed from a random noise $\eta \sim p(\eta)$ through a differentiable and invertible deterministic function $\mathbf{g}_\phi(\eta,\mathbf{x})$, where η is independent of ϕ, θ and $\mathbf{x}$. This enables us to compute the gradient with respect to ϕ through samples of $q_\phi(\mathbf{z}|\mathbf{x})$. The resulting gradient estimator turns out to be unbiased and of low variance. For the factorized Gaussian variational posterior, we use a standard Gaussian noise $\eta \sim \mathscr{N}(\mathbf{0},\mathbf{I})$ with a location-scale transformation $\mathbf{z} = \mu_\phi(\mathbf{x}) + \sigma_\phi(\mathbf{x}) \odot \eta$, where $\odot$ denotes the element-wise product operator.

2.1.2 Identifiability

Although the VAE framework allows us to learn a full generative model $p_\theta(\mathbf{x},\mathbf{z}) = p_\theta(\mathbf{x}|\mathbf{z})p_\theta(\mathbf{z})$ and a variational posterior $q_\phi(\mathbf{z}|\mathbf{x}) \approx p_\theta(\mathbf{z}|\mathbf{x})$, Khemakhem et al. (2020a) points out that it is only guaranteed to give a good estimate of the true marginal distribution over the observed variable after optimization:

$$p_\theta(\mathbf{x}) \approx p_{\theta^*}(\mathbf{x}). \tag{2.7}$$

In other words, all the other learned distributions are meaningless in general, and there is no guarantee that the true latent variable from which the observed variable originated can be recovered. This is known as lack of identifiability in DLVMs (Hyvärinen and Pajunen, 1999), as the true joint distribution $p_{\theta^*}(\mathbf{x},\mathbf{z})$ cannot be identified when an unconditional prior is used. We illustrate this by a simple example here and refer to Khemakhem et al. (2020a) for proofs of general unidentifiability results. Consider the prior $p_\theta(\mathbf{z}) = \mathscr{N}(\mathbf{z};\mathbf{0},\mathbf{I})$ often used in the VAE framework, which is invariant to rotation. In this case, arbitrarily rotating the prior, which can be done in the first layer of the neural network that parameterizes the mixing function $\mathbf{f}$, will not change the marginal distribution $p_\theta(\mathbf{x})$ but will change the posterior $p_\theta(\mathbf{z}|\mathbf{x})$, since each value of $\mathbf{x}$ now comes from a different value of $\mathbf{z}$ than before due to the rotation operation. This means that the model is unidentifiable. Formally, identifiability of deep generative models is defined to be some equivalence relation on the parameter space Θ:

Definition 1 (Identifiability)**.** *Let Θ be the domain of the parameters θ. Let $\sim$ be an equivalence relation on Θ. A deep generative model is said to be $\sim$–identifiable if*

$$p_\theta(\mathbf{x}) = p_{\tilde{\theta}}(\mathbf{x}), \quad \forall \mathbf{x} \in \mathscr{X} \implies \theta \sim \tilde{\theta}\,. \tag{2.8}$$

Each element in the quotient space $\Theta \backslash \sim$ is called an identifiability class.

Definition 1 tells us that two different sets of parameters θ and $\tilde{\theta}$ leading to the same marginal distribution over the observed variable should imply that they are equivalent in some sense. For example, if the equivalence relation is equality and a perfect marginal distribution $p_\theta(\mathbf{x}) = p_{\theta^*}(\mathbf{x})$ is learned, then this would imply that the true joint distribution $p_\theta(\mathbf{x},\mathbf{z}) = p_{\theta^*}(\mathbf{x},\mathbf{z})$, true prior $p_\theta(\mathbf{z}) = p_{\theta^*}(\mathbf{z})$, true likelihood $p_\theta(\mathbf{x}\,|\,\mathbf{z}) = p_{\theta^*}(\mathbf{x}\,|\,\mathbf{z})$, and true posterior $p_\theta(\mathbf{z}\,|\,\mathbf{x}) = p_{\theta^*}(\mathbf{z}\,|\,\mathbf{x})$ are all recovered. In the VAE framework, assuming that the variational family covers a large class of distributions including $p_{\theta^*}(\mathbf{z}\,|\,\mathbf{x})$, this would also imply that the variational posterior $q_\phi(\mathbf{z}\,|\,\mathbf{x})$ can recover the true data representation $\mathbf{z}_*$.

In practice, it is only possible to learn identifiable models up to simple transformations. Khemakhem et al. (2020a) propose a class of identifiable DLVMs (up to simple affine transformations), which requires a factorized exponential family prior over the latent variable that is conditioned on a concurrently observed and sufficiently informative auxiliary variable $\mathbf{u}$ (Hyvarinen et al., 2019):

$$p_{\mathbf{T},\lambda}(\mathbf{z}\,|\,\mathbf{u}) = \prod_{i=1}^{n} \frac{Q_i(z_i)}{Z_i(\mathbf{u})} \exp(\langle \mathbf{T}_i(z_i), \lambda_i(\mathbf{u}) \rangle), \tag{2.9}$$

where the base measure $Q_i(z_i)$, normalizing constant $Z_i(\mathbf{u})$, sufficient statistics $\mathbf{T}_i(z_i)$, and natural parameters $\lambda_i(\mathbf{u})$ are all factorized. The auxiliary variable $\mathbf{u}$ can be, for instance, time index in a time series (Hyvarinen and Morioka, 2016), data in a previous time step in a time series, some form of (possibly imperfect and noisy) label, and so on. We refer to the corresponding inference and learning scheme as identifiable VAE (iVAE). Unlike Hyvarinen et al. (2019) which resorts to heuristic contrastive learning, iVAE is a more principled scheme in the sense that it performs maximum likelihood learning as in the standard VAE framework. Note that iVAE is also applicable to normalizing flows (Rezende and Mohamed, 2015) ($\mathrm{Var}(\varepsilon) = 0$) and is closely related to independent component analysis (ICA) (Comon, 1994; Hyvarinen et al., 2019) ($n = d$). ICE-BeeM (Khemakhem et al., 2020b) is a similar but slightly more general class of identifiable DLVMs, which assumes that the conditional prior

has a general non-factorized base measure and a factorized exponential component:

$$p_{\mathbf{T},\lambda}(\mathbf{z}\,|\,\mathbf{u}) = \frac{Q(\mathbf{z})}{Z(\mathbf{u})} \exp\left(\sum_{i=1}^{n} \langle \mathbf{T}_i(z_i), \lambda_i(\mathbf{u})\rangle\right). \tag{2.10}$$

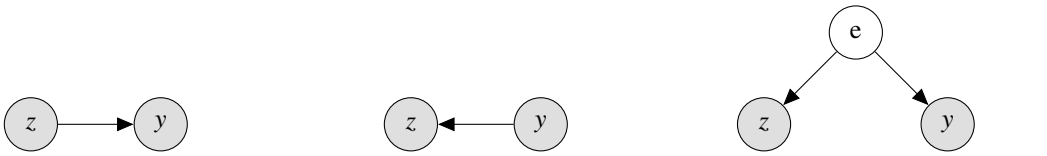
In Chapter 3, we will extend these identifiability results to a more general setting where general non-factorized conditional priors are considered, in order to be able to learn true data representation under the assumptions made in this thesis.

Intuitively, the auxiliary variable $\mathbf{u}$ effectively specifies a particular way to partition the latent space $\mathscr{Z}$. In this way, the given auxiliary information will be used to identify the cluster to which each observed data point belongs in the latent space, which is where the identifiability comes from. Following this idea, Willetts and Paige (2021) propose an empirical approach to obtaining identifiability without observing any auxiliary variable. The idea is to learn a clustering $\mathbf{u}$ in the latent space rather than rely on a given one. This ends up being a standard DLVM with a Gaussian mixture model for the prior over $\mathbf{z}$, with the learned auxiliary variable $\mathbf{u}$ partitioning the latent space in some way. Note that we cannot control the kind of auxiliary information $\mathbf{u}$ that will be learned by this model, but the model can obtain identifiability as long as it always learns the same $\mathbf{u}$-clustering in the latent space.

Identifiability Metric. We can quantitatively measure the identifiability of a DLVM using the mean correlation coefficient (MCC) score between samples of the true latent variable and samples of the latent variable recovered by the DLVM. MCC scores can be obtained by calculating the correlation coefficient between all pairs of true and recovered latent factors and then solving a linear sum assignment problem by assigning each recovered latent factor to the true latent factor with which it best correlates (Khemakhem et al., 2020a). If the true latent variable is unknown to us, then we compute an average MCC score using latent variables recovered by the DLVM trained with different random initializations (Khemakhem et al., 2020b). By definition, higher MCC scores indicate stronger identifiability (up to pointwise transformations).

2.2 Causality for Machine Learning

Causality has recently caught the attention of the machine learning community (Schölkopf, 2019). Causal representation learning, a promising class of methods combining causal inference and deep learning, is the central topic of this thesis. In this section, we introduce



(a) z causes y. (b) y causes z. (c) z and y are confounded by e.

Fig. 2.1 Three possible causal relationships between two dependent variables z and y.

the concept of causal models, contrast it with statistical learning, and describe the general idea of causal representation learning.

2.2.1 Causal Models

“Correlation does not imply causation" is a well-known statement in statistics, which tells us that one cannot identify the causal relationships between variables solely based on their statistical associations calculated from observational data. For example, in the case of two statistically dependent variables z and y, there are at least three possible causal relationships that could explain the observed association between them, as shown in Figure 2.1:

(a) z is the cause of y;

(b) z is the effect of y;

(c) z and y are confounded by another (possibly latent) variable e.

Without additional assumptions it is impossible to distinguish between these three cases from observational data, since the realizations of these different causal models can give the same observational distribution. In general, a causal model reveals the underlying data generating mechanism and thus contains more information than a statistical model which only captures spurious associations between variables and may fail catastrophically under distribution shift.

Intervention is an important concept in causality. Intervening upon a variable means actively setting the variable to some value rather than passively observing the variable takes that value. Unlike observation which only gives us a single (and possibly confounded) observational distribution, each intervention gives us a different realization of the underlying causal model in a controllable way. A causal model can thus be thought of as a set of joint distributions induced by all possible interventions (Schölkopf et al., 2021). This view of causal models provides us with great insights about how to deal with distribution shift and out-of-distribution generalization problems in machine learning – it is useful to learn causal models, since they

are robust when generalizing from an observational distribution to interventional distributions.

It is possible to learn causal models using the information obtained from interventions. That is, if we are allowed to intervene upon the variables of interest, we may be able to learn the underlying data generating mechanism. To see this, imagine that we collected a set of data that consists of average annual temperatures t and altitudes a for many different cities in a certain country (Peters et al., 2017). This dataset gives us an observational distribution over t and a. Using this dataset, we can easily confirm that these two variables are statistically dependent but cannot say anything about the causal relationship between them without any additional assumptions. Now imagine that we hypothetically intervene upon these variables. If we increased the altitude of a city, according to our knowledge of the physical world, this would change the temperature of that city. Conversely, if we increased the temperature of a city, this would not change the altitude of that city. These (hypothetical) interventions confirm that the underlying data generating mechanism is altitude a causing temperature t.

In practice, we are often only given a set of observational data and therefore unable to perform intervention. In order to identify causal relationships between variables from observational data, we will need to make additional assumptions and/or have additional information of the underlying data generating mechanism.

The causal inference problem we face in this thesis can be summarized as follows. We are given a set of latent factors $z_1, \cdots, z_n$ recovered by a DLVM, each one of which is either the cause of the target y, the effect of the target y, or independent of the target y. We would like to identify the causal latent factors of y using observational samples. In Section 4.2, we will propose a practical causal inference scheme to solve this problem.

2.2.2 Causal Representation Learning

Traditional causal inference methods are symbolic approaches which assume that the random variables of interests connected according to a causal graph are given. This assumption is not practical for real-world problems with structured data. For example, in the camel-cow classification task, the observed variables are images represented by pixel values, which cannot be directly fitted into the symbolic causal inference framework.

Causal representation learning (Schölkopf et al., 2021) aims to learn the symbols required by causal inference from structured data by leveraging recent advances in deep learning, which resembles machine learning going beyond symbolic AI. The Independent Causal Mechanism

(ICM) Principle (Peters et al., 2017; Schölkopf et al., 2012, 2021) gives us some high level ideas of what causal representation should be like:

> ***The ICM Principle:*** *The causal generative process of variables in a system is composed of autonomous modules that do not inform or influence each other. In the probabilistic case, this means that the conditional distribution of each variable given its causes (i.e., its mechanism) does not inform or influence the other mechanisms.*

Specifically, a joint distribution $p(z_1, \cdots, z_n)$ has many different factorizations, but there is only one causal factorization

$$p(z_1, \cdots, z_n) = \prod_{i=1}^{n} p(z_i | \mathrm{pa}_i) \tag{2.11}$$

that follows the ICM principle, where pa_i is a subset of $\{z_1, \cdots, z_n\} \backslash \{z_i\}$ which contains the direct causes of z_i in the underlying causal graph. This tells us that the factors (or mechanisms) $\{p(z_i|\mathrm{pa}_i)\}_{i=1}^n$ should be disentangled in causal representation. Disentanglement may be defined by:

1. knowing a factor $p(z_i|\mathrm{pa}_i)$ does not reveal any information about any other factor $p(z_j|\mathrm{pa}_i)$ (Janzing and Schölkopf, 2010);
2. intervening upon a mechanism $p(z_i|\mathrm{pa}_i)$ does not change any other mechanism $p(z_j|\mathrm{pa}_i)$ (Schölkopf et al., 2012).

As an example, let us consider again the temperature-altitude problem, where two possible factorizations of the joint distribution are 1) $p(t,a) = p(a)p(t|a)$ and 2) $p(t,a) = p(t)p(a|t)$. Note that the first one is the causal factorization which is robust to distribution shift. To see this, imagine that we had a second dataset collected in a different country but in the same climate zone as where the first dataset was collected. Essentially, this new dataset was sampled from a different joint distribution $p(t,a)$ where $p(a)$ had changed but $p(t|a)$ remained unchanged. This means that the mechanism $p(t|a)$ learned from the first dataset can robustly generalize to predicting temperatures from altitudes for the new dataset, while neither of the factors in the non-causal factorization is reusable.

In Section 4.1 we will use an iVAE-based method to recover the true data representation, since it is a powerful approach which is guaranteed to recover the true latent variable $\mathbf{z}_*$ (up to simple transformations) and thus achieves a principled form of disentanglement, as required in causal representation learning.

Algorithm 1: Model-based optimization.

Input : Objective $J(\mathbf{x})$, initial dataset $\mathscr{D} = \{(\mathbf{x}^{(i)}, J(\mathbf{x}^{(i)}))\}_{i=1}^{N}$, query budget N_b, surrogate model $h_{\mathscr{X}}(\mathbf{x})$.

for $j \in \{1, \cdots, N_b\}$ **do**

  Train the surrogate model $h_{\mathscr{X}}$ on the current dataset $\mathscr{D} = \{(\mathbf{x}^{(i)}, J(\mathbf{x}^{(i)}))\}_{i=1}^{N+j-1}$

  Optimize $h_{\mathscr{X}}$ to obtain a new query data point $\mathbf{x}^{(N+j)}$

  Evaluate the objective function J at the new query data point to obtain $J(\mathbf{x}^{(N+j)})$

  Update the dataset $\mathscr{D} \leftarrow \mathscr{D} \cup \{(\mathbf{x}^{(j+1)}, J(\mathbf{x}^{(j+1)}))\}$

end

Output : Augmented dataset $\mathscr{D} = \{(\mathbf{x}^{(i)}, J(\mathbf{x}^{(i)}))\}_{i=1}^{N+N_b}$

2.3 Black-box Optimization

Black-box optimization is the downstream task considered in this thesis, in which we want to maximize a black-box objective function $J : \mathscr{X} \subset \mathbb{R}^d \to \mathbb{R}$ whose analytical form and derivative information are unavailable. Instead, we are only able to evaluate the objective function in its domain $\mathscr{X}$. In real-world problems, it can be quite expensive to evaluate the objective function, so exhaustive search in the input domain $\mathscr{X}$ is prohibitive if the dimensions d of $\mathscr{X}$ is large. In this thesis, we aim to achieve efficient black-box optimization with the help of causal representation learning.

2.3.1 Model-based Optimization

Model-based optimization is a popular approach to solving black-box optimization problems, which directly operates in the input domain $\mathscr{X}$. The idea is to fit a surrogate model $h_{\mathscr{X}} : \mathscr{X} \subset \mathbb{R}^d \to \mathbb{R}$ using the initial dataset $\{(\mathbf{x}^{(i)}, J(\mathbf{x}^{(i)}))\}_{i=1}^{N}$ and then iteratively perform optimization to obtain a new query data point using the surrogate model and update the surrogate model using all available data points so far. The detailed procedure of model-based optimization is summarized in Algorithm 1.

Bayesian optimization (Frazier, 2018) is a kind of model-based optimization, where $h_{\mathscr{X}}$ is chosen to be a flexible probabilistic model, such as a Gaussian process (Rasmussen and Williams, 2006) or Bayesian neural network (Neal, 1996). Bayesian optimization algorithms use acquisition functions to guide exploration in the search space, which incorporates the predictions as well as uncertainty estimates from the probabilistic surrogate model. Expected improvement (Jones et al., 1998) is a popular acquisition function, which determines new query data points by maximizing the expected gain upon the best data points so far. Bayesian

Algorithm 2: LSO with weighted retraining.

Input : Objective $J(\mathbf{x})$, initial dataset $\mathscr{D} = \{(\mathbf{x}^{(i)}, J(\mathbf{x}^{(i)}))\}_{i=1}^{N}$, query budget N_b, surrogate model $h_{\mathscr{Z}}(\mathbf{z})$, generative model $\mathbf{f}(\mathbf{z})$ and corresponding inverse model $\mathbf{q}(\mathbf{x})$, retraining frequency r, weighting function $w(\mathbf{x})$, the number of epochs for retraining in each optimization round N_{re}.

Train the DGM ($\mathbf{f}(\mathbf{z})$ and $\mathbf{q}(\mathbf{x})$) on $\mathscr{D}$ with uniform weighting until convergence

for $j \in \{1, \cdots, N_b/r\}$ **do**

  for $l \in \{1, \cdots, r\}$ **do**

    Obtain latent variable samples $\mathscr{D}_{\mathbf{z}} = \{\mathbf{z} = \mathbf{q}(\mathbf{x}) : \mathbf{x} \in \mathscr{D}\}$

    Train the surrogate model $h_{\mathscr{Z}}(\mathbf{z})$ on $\mathscr{D}_{\mathbf{z}}$ and $\mathscr{D}$

    Optimize $h_{\mathscr{Z}}(\mathbf{z})$ to obtain a new query latent variable $\tilde{\mathbf{z}}$

    Evaluate the objective function J at $\tilde{\mathbf{x}} = \mathbf{f}(\tilde{\mathbf{z}})$ to obtain its objective value $\tilde{y}$

    Update the dataset $\mathscr{D} \leftarrow \mathscr{D} \cup \{(\tilde{\mathbf{x}}, \tilde{y})\}$

  end

  Fine-tune the DGM ($\mathbf{f}(\mathbf{z})$ and $\mathbf{q}(\mathbf{x})$) on $\mathscr{D}$ with importance weighting $w(\mathbf{x})$ for N_{re} epoch

end

Output : Augmented dataset $\mathscr{D}$

optimization is widely used in many fields, such as hyper-parameter tuning in machine learning (Snoek et al., 2012), drug discovery in biological chemistry (Negoescu et al., 2011), materials design in material science (Packwood, 2017), and so on.

2.3.2 Latent Space Optimization

In real-world problems, the input variable $\mathbf{x}$ of the black-box objective function J can be a complex, structured, and high dimensional variable of variable length (e.g., molecules). Therefore, directly performing optimization in the input space $\mathscr{X}$ can be very difficult. Latent space optimization (LSO) (Gómez-Bombarelli et al., 2018) performs model-based optimization in the latent space $\mathscr{Z}$ of a deep generative model (DGM) trained on the dataset $\mathscr{D}$. The surrogate model for LSO is $h_{\mathscr{Z}} : \mathscr{Z} \subset \mathbb{R}^n \to \mathbb{R}$. LSO makes optimization easier, since it simplifies the problem to optimizing in a low dimensional and continuous space. However, naive LSO has two limitations:

1. The latent space $\mathscr{Z}$ of a DGM trained on the initial dataset $\mathscr{D}$ may not be useful for efficient optimization of the objective J, since the global optimum is usually very far away from any data points in the initial dataset.

2. The DGM does not incorporate the information in the new query data points obtained during optimization, which could have adjusted the latent space $\mathscr{Z}$ to be more amenable for optimization of J.

To address these two problems, Tripp et al. (2020) propose to perform periodic weighted retraining for the DGM in LSO. Periodic retraining updates the latent space with new query data points obtained during optimization. These new query data points often have high objective values. Importance-weighted training forces the latent space to focus on the regions that corresponds to data points of high objective values, which makes the DGM more relevant to optimization. Combining these two complementary ideas enables the DGM to actively participate in optimization instead of passively encoding-decoding as in naive LSO.

The procedure of LSO with weighted retraining is summarized in Algorithm 2. For retraining, the hyper-parameter r specifies how many new query data points to be collected in each optimization round j. To implement weighted training, we use a weighted sampler to sample mini-batches during training. In practice, we weight each available data point using rank-based weights when retraining the DGM:

$$w_{J,\mathscr{D},k}(\mathbf{x}) \propto \frac{1}{kN + \text{rank}_{J,\mathscr{D}}(\mathbf{x})}, \quad \text{rank}_{J,\mathscr{D}}(\mathbf{x}) \coloneqq |\{\mathbf{x}' \in \mathscr{D} : J(\mathbf{x}') > J(\mathbf{x})\}|, \tag{2.12}$$

since rank weighting turns out to be robust and independent of the size of the dataset $\mathscr{D}$. The hyper-parameter $k \in (0, \infty)$ controls the degree of weighting. $k = 0$ puts all weights on the data point that has the highest objective value, and $k = \infty$ is equivalent to uniform weighting. In practice, $k = 10^{-3}$ is found to be a good choice for LSO.

We argue that LSO can be naturally combined with causal representation learning. The idea is to optimize the causal latent factors of the target y identified from the latent variable $\mathbf{z}$ which is recovered by an identifiable DLVM. In this thesis, we will show that causal representation learning improves the robustness, efficiency, and performance of LSO.

2.4 Problem Formulation and Assumptions

The assumptions of the data generating process considered in this thesis is encapsulated in the causal graph shown in Figure 2.2. The target variable $y \in \mathbb{R}$ is computed by evaluating the target objective function J at the observed variable $\mathbf{x}$. The latent variable $\mathbf{z}$, which generates $\mathbf{x}$, can be divided into three blocks $\mathbf{z}_c$, $\mathbf{z}_s$, and $\mathbf{z}_p$. The block $\mathbf{z}_c$ contains latent factors that cause y. The block $\mathbf{z}_s$ contains latent factors that are caused by y (i.e., they are

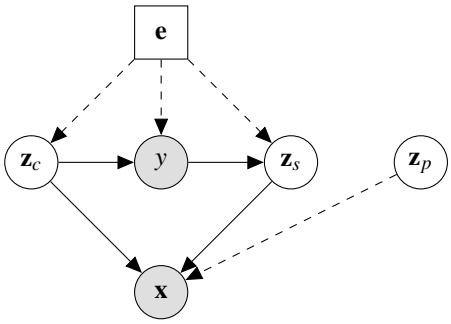


Fig. 2.2 The causal graph of the data generating process considered in this thesis. Nodes represent variables. Shaded nodes represent observed variables and clear nodes represent latent variables. Square nodes represent discrete variables and circle nodes represent continuous variables. Each arrow indicates the causal relationship between the two variables it connects. Dashed arrows may be absent in some cases.

spuriously correlated with y). The block $\mathbf{z}_p$ contains latent factors that are independent of y, which may not exist in some cases. The environment variable $\mathbf{e}$ acts as a hidden confounder.

Our causal graph is more general than those considered in the related works in the literature:

1. The environment variable $\mathbf{e}$ in our causal graph is assumed to be unobserved. This overcomes a major limitation of existing works (Arjovsky et al., 2019; Heinze-Deml et al., 2018; Lu et al., 2021; Peters et al., 2016; Rojas-Carulla et al., 2018) which require $\mathbf{e}$ to be observed, since we usually do not have access to the ground truth environment variable $\mathbf{e}$ in practice.

2. The causal graph considered in Lu et al. (2021) assumes $z_i \perp\!\!\!\perp z_j | \mathbf{e}$ for $z_i, z_j \in \mathbf{z}_c$ and $z_i \perp\!\!\!\perp z_j | (\mathbf{e}, y)$ if z_i and z_j are not both in $\mathbf{z}_c$. In contrast, we only assume $\mathbf{z}_c \perp\!\!\!\perp \mathbf{z}_s | (\mathbf{e}, y)$, $\mathbf{z}_c \perp\!\!\!\perp \mathbf{z}_p$, and $\mathbf{z}_s \perp\!\!\!\perp \mathbf{z}_p$. That is, we allow $z_i \not\perp\!\!\!\perp z_j | \mathbf{e}$ for $z_i, z_j \in \mathbf{z}_c$, $z_i \not\perp\!\!\!\perp z_j | (\mathbf{e}, y)$ for $z_i, z_j \in \mathbf{z}_s$, and $z_i \not\perp\!\!\!\perp z_j$ for $z_i, z_j \in \mathbf{z}_p$. This makes our method applicable to a much wider range of problem scenarios, since such (conditional) dependencies between the latent factors within each block (particularly within $\mathbf{z}_c$) almost always exist in practice.

Our goal is to generate new synthetic data $\mathbf{x}$ with high target value y, starting from a given initial dataset $\mathscr{D} = \{(\mathbf{x}^{(i)}, y^{(i)})\}_{i=1}^{N}$. We will take a causal approach to solve this problem. The idea is as follows. We first recover the true latent variable $\mathbf{z}_*$ using a generalized version of iVAE, based on the assumption that the prior over the latent variable $\mathbf{z}$ given the target y is a non-factorized distribution, induced from the causal graph shown in Figure 2.2. Then we employ a practical causal inference scheme to identify causal latent factors $\mathbf{z}_c$ out of all latent factors $\mathbf{z}$ recovered before. Finally, we perform LSO with weighted retraining to obtain new data points with maximal target values y by intervening upon the causal latent factors $\mathbf{z}_c$.

Chapter 3

Generalized Identifiable VAEs

Before we can identify causal latent factors $\mathbf{z}_c$ of the target y, we need to recover the true data representation (i.e., the true latent variable from which the observed variable originated). In this chapter, we present generalized iVAEs, a novel inference and learning scheme that is guaranteed to learn identifiable DLVMs with a non-factorized conditional prior, which accommodates the assumption that the prior over the data representation given the target is a general non-factorized distribution. Our work can be seen as an extension of iVAE (Khemakhem et al., 2020a) and ICE-BeeM (Khemakhem et al., 2020b).

3.1 Non-factorized Conditional Priors

Let $\mathbf{u} \in \mathscr{U} \subset \mathbb{R}^m$ be an auxiliary variable which is concurrently observed with $\mathbf{x}$, as described in Section 2.1.2. We consider a more general setting than Khemakhem et al. (2020a) and Khemakhem et al. (2020b), in which the prior over the latent variable $\mathbf{z}$ given the auxiliary variable $\mathbf{u}$ is assumed to have a general multivariate strongly exponential family distribution:

$$p_{\mathbf{T},\lambda}(\mathbf{z}\,|\,\mathbf{u}) = \frac{Q(\mathbf{z})}{Z(\mathbf{u})} \exp(\langle \mathbf{T}(\mathbf{z}), \lambda(\mathbf{u}) \rangle), \tag{3.1}$$

where $Q : \mathscr{Z} \to \mathbb{R}$ is the base measure, Z is the normalizing constant, $\mathbf{T} : \mathscr{Z} \to \mathbb{R}^k$ is the sufficient statistics, and the natural parameters $\lambda : \mathscr{U} \to \mathbb{R}^k$ crucially depend on $\mathbf{u}$. The size $k \geq n$ is the dimensions of the sufficient statistics $\mathbf{T}$ and depends on the latent space dimensions n. We do not treat k as a learnable parameter, so k is fixed once we have specified the form of the distribution and the dimensions n of the latent variable $\mathbf{z}$. The definition of strongly exponential family can be found in Appendix A. Note that exponential family distributions have universal approximation power (Sriperumbudur et al., 2017), and all com-

mon multivariate exponential family distributions (e.g., multivariate Gaussian) are strongly exponential. Therefore, these assumptions are not restrictive.

The joint distribution for this new model is then given by

$$p_\theta(\mathbf{x}, \mathbf{z} \mid \mathbf{u}) = p_\mathbf{f}(\mathbf{x} \mid \mathbf{z}) p_{\mathbf{T},\lambda}(\mathbf{z} \mid \mathbf{u}), \tag{3.2}$$

where $p_\mathbf{f}(\mathbf{x} \mid \mathbf{z}) = p_\varepsilon(\mathbf{x} - \mathbf{f}(\mathbf{z}))$ is the likelihood as defined before, and $\theta = \{\mathbf{f}, \mathbf{T}, \lambda\} \in \Theta$ are model parameters. We further assume that the domains $\mathscr{X}$, $\mathscr{Z}$ and $\mathscr{U}$ are open sets, and the mixing function $\mathbf{f} : \mathscr{Z} \to \mathscr{X}$ is bijective so its inverse $\mathbf{f}^{-1} : \mathscr{X} \to \mathscr{Z}$ exists.

3.2 Identifiability Theory

We first define two types of equivalence relations on the parameter space Θ. They correspond to weak identifiability and strong identifiability, respectively.

Definition 2 (Weak identifiability)**.** *Let $\sim_A$ be an equivalence relation on Θ defined by:*

$$(\mathbf{f}, \mathbf{T}, \lambda) \sim_A (\tilde{\mathbf{f}}, \tilde{\mathbf{T}}, \tilde{\lambda}) \iff \exists A, \mathbf{c} \text{ s.t. } \mathbf{T}(\mathbf{f}^{-1}(\mathbf{x})) = A\tilde{\mathbf{T}}(\tilde{\mathbf{f}}^{-1}(\mathbf{x})) + \mathbf{c}, \ \forall \mathbf{x} \in \mathscr{X}, \tag{3.3}$$

where $A \in \mathbb{R}^{k \times k}$ is an invertible matrix and $\mathbf{c} \in \mathbb{R}^k$ is a vector.

Weak identifiability guarantees that we can recover the true data representation up to an invertible affine transformation define by the sufficient statistics $\mathbf{T}$ and $\tilde{\mathbf{T}}$. We can go one step further and define strong identifiability analogous to the one in linear ICA where the true source is recovered up to pointwise scaling and permutation:

Definition 3 (Strong identifiability)**.** *Let $\sim_P$ be an equivalence relation on Θ defined by:*

$$(\mathbf{f}, \mathbf{T}, \lambda) \sim_P (\tilde{\mathbf{f}}, \tilde{\mathbf{T}}, \tilde{\lambda}) \iff \exists P, \mathbf{c} \text{ s.t. } \mathbf{T}(\mathbf{f}^{-1}(\mathbf{x})) = P\tilde{\mathbf{T}}(\tilde{\mathbf{f}}^{-1}(\mathbf{x})) + \mathbf{c}, \ \forall \mathbf{x} \in \mathscr{X}, \tag{3.4}$$

where $P \in \mathbb{R}^{k \times k}$ is a block permutation matrix (each block transforms a factor of $\mathbf{T}$ into a factor of $\tilde{\mathbf{T}}$) and $\mathbf{c} \in \mathbb{R}^k$ is a vector.

We now present two novel identifiability theorems for generalized iVAEs. The proofs of them can be found in Appendix B.1 and B.2.

Theorem 4. *Suppose that we observe data sampled from a DLVM defined according to* (3.1) *and* (3.2) *with parameters $\theta = \{\mathbf{f}, \mathbf{T}, \lambda\}$. Assume that*

(i) the set $\{\omega \in \mathbb{R}^d \,|\, \varphi_\varepsilon(\omega) = 0\}$ *has measure zero, where* φ_ε *is the characteristic function of* $\varepsilon \sim p_\varepsilon(\varepsilon)$*;*

(ii) the mixing function $\mathbf{f}$ *is bijective;*

(iii) there exists $k+1$ *points* $\mathbf{u}_0, \mathbf{u}_1, \cdots, \mathbf{u}_k \in \mathscr{U}$ *such that the matrix*

$$L = [\lambda(\mathbf{u}_1) - \lambda(\mathbf{u}_0), \cdots, \lambda(\mathbf{u}_k) - \lambda(\mathbf{u}_0)] \in \mathbb{R}^{k \times k} \tag{3.5}$$

is invertible.

Then the parameters $\theta = \{\mathbf{f}, \mathbf{T}, \lambda\}$ *are* $\sim_A$*–identifiable.*

Note that Assumption (iii) in Theorem 4 essentially defines a criterion for determining when an auxiliary variable $\mathbf{u}$ is sufficiently informative to guarantee identifiability.

Theorem 5. *Suppose that all assumptions in Theorem 4 hold. Let the sufficient statistics* $\mathbf{T}(\mathbf{z}) = [\mathbf{T}_f(\mathbf{z})^T, \mathbf{T}_{NN}(\mathbf{z})^T]^T$ *be of the form of a concatenation of the sufficient statistics* $\mathbf{T}_f(\mathbf{z}) = [\mathbf{T}_{f_1}(z_1)^T, \cdots, \mathbf{T}_{f_n}(z_n)^T]^T$ *of a factorized strongly exponential family distribution and the output* $\mathbf{T}_{NN}(\mathbf{z})$ *of a neural network with ReLU activation. Let* k' *be the dimensions of* $\mathbf{T}_f$ *and suppose that* $k' \geq 2n$*. Assume that*

(i) the sufficient statistics $\mathbf{T}_f$ *have all second-order own derivatives;*

(ii) the mixing function $\mathbf{f}$ *has all second-order cross derivatives.*

Then the parameters $\theta = \{\mathbf{f}, \mathbf{T}, \lambda\}$ *are* $\sim_P$*–identifiable.*

The use of a ReLU neural network $\mathbf{T}_{NN}(\mathbf{z})$ in Theorem 5 ensures that all its second order derivatives with respect to the input are zero, which is crucial to the proof of this theorem. This design choice is not restrictive, since ReLU neural networks have universal approximation power (Lu et al., 2017) and should be able to capture any dependencies in $\mathbf{z}$ of interest. One common choice for the factorized part $\mathbf{T}_f$ is the sufficient statistics of a factorized Gaussian distribution (i.e., each factor is of the form $\mathbf{T}_{f_i}(z_i) = \left[z_i, z_i^2\right]^T$). Note that the natural parameters corresponding to z_i^2 need to be constrained to be negative in this case.

Overall, Theorem 4 and Theorem 5 provide us with theoretical guarantees of identifiability for DLVMs with a non-factorized conditional prior (up to simple affine transformations). The result in Theorem 5 is particularly desirable, since it guarantees that the true data representation $\mathbf{z}_*$ can be recovered up to an affine transformation of the sufficient statistics $\mathbf{T}_f^*$ and $\tilde{\mathbf{T}}_f$ with a permutation τ:

$$\mathbf{T}_{f_i}^*(z_i^*) = A_i' \tilde{\mathbf{T}}_{f_{\tau(i)}}(z_{\tau(i)}) + \mathbf{c}_i \,. \tag{3.6}$$

3.3 Hybrid Training Objective

Generalized iVAEs cannot be trained using the variational lower bound

$$\mathscr{F}(\theta,\phi) = \mathbb{E}_{p(\mathbf{x},\mathbf{u})}\left[\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x},\mathbf{u})}[\log p_{\mathbf{f}}(\mathbf{x}|\mathbf{z})] - \mathrm{KL}(q_\phi(\mathbf{z}|\mathbf{x},\mathbf{u})||p_{\mathbf{T},\lambda}(\mathbf{z}|\mathbf{u}))\right], \tag{3.7}$$

since the non-factorized conditional prior

$$p_{\mathbf{T},\lambda}(\mathbf{z}|\mathbf{u}) = \frac{1}{Z_{\mathbf{T},\lambda}(\mathbf{u})}\exp\left(\langle \mathbf{T}_f(\mathbf{z}), \lambda_f(\mathbf{u})\rangle + \langle \mathbf{T}_{NN}(\mathbf{z}), \lambda_{NN}(\mathbf{u})\rangle + \log Q(\mathbf{z})\right) \tag{3.8}$$

$$:= \frac{1}{Z_{\mathbf{T},\lambda}(\mathbf{u})} p^*_{\mathbf{T},\lambda}(\mathbf{z}|\mathbf{u}) \tag{3.9}$$

is essentially an energy-based model (LeCun et al., 2006; Song and Kingma, 2021) and has an intractable normalizing constant $Z_{\mathbf{T},\lambda}(\mathbf{u})$ that depends on the parameters $\mathbf{T}$ and λ. Therefore, the KL term in the variational lower bound (3.7) is also intractable.

Score matching (Hyvärinen, 2005) is a popular algorithm for training enenery-based model, which minimizes the Fisher divergence between the target distribution and the distribution given by an energy-based model. Here, score refers to the gradient of the log-density with respect to $\mathbf{z}$, and Fisher divergence is defined to be the mean squared distance between the score of the target distribution and the score of the distribution given by an energy-based model. One nice thing about Fisher divergence is that it only requires computing the score $\nabla_{\mathbf{z}} \log p^*_{\mathbf{T},\lambda}(\mathbf{z}|\mathbf{u}) = \nabla_{\mathbf{z}} \log p_{\mathbf{T},\lambda}(\mathbf{z}|\mathbf{u})$ of the energy-based model which is independent of the intractable normalizing constant. Following the idea of score matching, we replace the term $\mathrm{KL}(q_\phi(\mathbf{z}|\mathbf{x},\mathbf{u})||p_{\mathbf{T},\lambda}(\mathbf{z}|\mathbf{u}))$ in (3.7) by a Fisher divergence and propose a hybrid training scheme which jointly optimizes the following two objectives:

1. the prior parameters $\mathbf{T}$ and λ are learned by minimizing the score matching objective, which is a Fisher divergence between $q_\phi(\mathbf{z}|\mathbf{x},\mathbf{u})$ and $p_{\mathbf{T},\lambda}(\mathbf{z}|\mathbf{u})$):

$$\min_{\mathbf{T},\lambda} \mathscr{L}_s(\mathbf{T},\lambda) = \mathbb{E}_{p(\mathbf{x},\mathbf{u})}\left[\mathbb{E}_{q_{\hat{\phi}}(\mathbf{z}|\mathbf{x},\mathbf{u})}\left[\left\|\nabla_{\mathbf{z}} \log q_{\hat{\phi}}(\mathbf{z}|\mathbf{x},\mathbf{u}) - \nabla_{\mathbf{z}} \log p^*_{\mathbf{T},\lambda}(\mathbf{z}|\mathbf{u})\right\|^2\right]\right]. \tag{3.10}$$

2. the variational parameters ϕ and likelihood parameters $\mathbf{f}$ are learned by maximizing the pseudo variational lower bound:

$$\max_{\phi,\mathbf{f}} \mathscr{F}_v(\phi,\mathbf{f}) := \mathbb{E}_{p(\mathbf{x},\mathbf{u})}\left[\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x},\mathbf{u})}\left[\log p_{\mathbf{f}}(\mathbf{x}|\mathbf{z}) + \log p^*_{\hat{\mathbf{T}},\hat{\lambda}}(\mathbf{z}|\mathbf{u}) - \log q_\phi(\mathbf{z}|\mathbf{x},\mathbf{u})\right]\right], \tag{3.11}$$

This gives us the joint training objective for generalized iVAEs:

$$\max_{\phi,\theta} \; \mathscr{F}_l(\phi,\theta) \coloneqq \mathscr{F}_v(\phi,\mathbf{f}) - \mathscr{L}_s(\mathbf{T},\lambda). \tag{3.12}$$

Note that we only optimize the parameters without a hat and set the parameters with a hat to be the current values of the corresponding parameters without a hat. The true data distribution $p(\mathbf{x},\mathbf{u})$ is again estimated by the empirical distribution of a randomly sampled mini-batch at each iteration, and the expectation over $q_\phi(\mathbf{z}\,|\,\mathbf{x},\mathbf{u})$ is estimated by MC using samples $\mathbf{z} \sim q_\phi(\mathbf{z}\,|\,\mathbf{x},\mathbf{u})$ with the reparameterization trick. Sometimes it might be useful to scale the score matching objective $\mathscr{L}_s$ and the (negative) pseudo KL divergence $\mathbb{E}_{q_\phi(\mathbf{z}\,|\,\mathbf{x},\mathbf{u})}[\log p^*_{\mathbf{T},\lambda}(\mathbf{z}\,|\,\mathbf{u}) - \log q_\phi(\mathbf{z}\,|\,\mathbf{x},\mathbf{u})]$ in the pseudo variational lower bound $\mathscr{F}_v$ by a coefficient, so as to balance between reconstruction and prior regularization, analogous to the technique used in β-VAE (Higgins et al., 2016).

Finally, we present a theorem for the consistency guarantee of generalized iVAEs. The proof of this theorem can be found in Appendix B.3.

Theorem 6. *Assume that*

(i) The family of variational distributions $q_\phi(\mathbf{z}\,|\,\mathbf{x},\mathbf{u})$ contains the posterior $p_{\mathbf{f},\mathbf{T},\lambda}(\mathbf{z}\,|\,\mathbf{x},\mathbf{u})$ which is non-zero everywhere and non-degenerate;

(ii) The model (3.2) *is trained by maximizing $\mathscr{F}_l(\phi,\theta)$* (3.12) *with respect to θ and ϕ;*

(iii) All assumptions in Theorem 5 hold.

In the limit of infinite data, generalized iVAEs learn the true parameters $\theta^* = \{\mathbf{f}^*, \mathbf{T}^*, \lambda^*\}$ up to the equivalence class defined in Definition 3 and thus recover the true data representation $\mathbf{z}_*$ up to simple transformations defined in Equation (3.6).

In Section 5.1, we will use a synthetic dataset to show that the assumptions in Theorem 6 are realistic, and the true latent variable $\mathbf{z}_*$ can be recovered in the finite data regime.

Chapter 4

Latent Space Optimization with Causal Representation Learning

In this chapter, we present a framework for latent space optimization (LSO) with causal representation learning, which consists of three steps. In step I, we use the generalized iVAE described in Chapter 3 to recover the true data representation. In step II, we propose a practical causal inference scheme to identify causal latent factors $\mathbf{z}_c$ of the target y from the data representation $\mathbf{z}$ recovered in step I. In step III, we perform LSO by intervening upon the causal latent factors identified in step II to obtain new data points with maximal target values. The first two steps are summarized in Figure 4.1, which is based on the ideas in Lu et al. (2021) but accommodates a more general setting. The last step follows Tripp et al. (2020).

4.1 Step I: Recovering True Data Representation

In the first step, we use generalized iVAE to recover the true latent variable $\mathbf{z}_*$ from which the observed variable $\mathbf{x}$ originated. For black-box optimization tasks, we choose the auxiliary variable $\mathbf{u}$ to be the target $y = J(\mathbf{x})$ (i.e., the objective function J evaluated at the observed variable $\mathbf{x}$). The non-factorized conditional prior $p_{\mathbf{T},\lambda}(\mathbf{z}|y)$ over the latent variable is modelled by a general exponential family distribution with sufficient statistics defined by $\mathbf{T}_f(\mathbf{z}) = [\mathbf{z}, \mathbf{z} \odot \mathbf{z}]$ and a ReLU neural network $\mathbf{T}_{NN} : \mathbb{R}^n \to \mathbb{R}^{n(n-1)/2}$, which simulates a full Gaussian distribution. The joint distribution of the DLVM that corresponds to this non-factorized conditional prior is given by $p_\theta(\mathbf{x}, \mathbf{z}|y) = p_{\mathbf{f}}(\mathbf{x}|\mathbf{z}) p_{\mathbf{T},\lambda}(\mathbf{z}|y)$. We train this DLVM by maximizing the hybrid objective (3.12). Under the assumptions in Theorem 6, the true data representation $\mathbf{z}_*$ can be recovered up to simple transformations defined in (3.6).

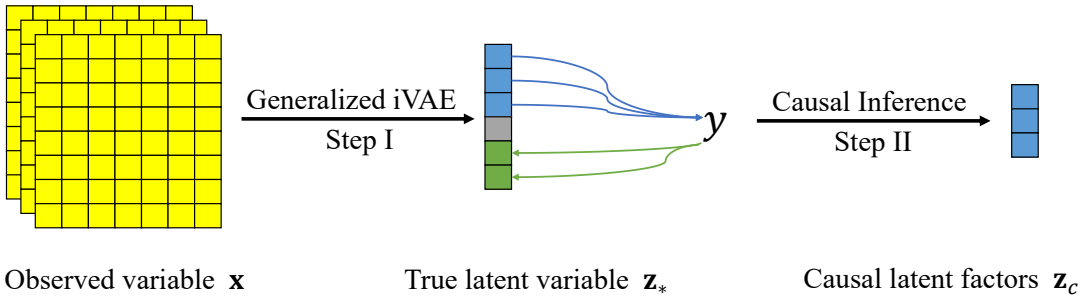


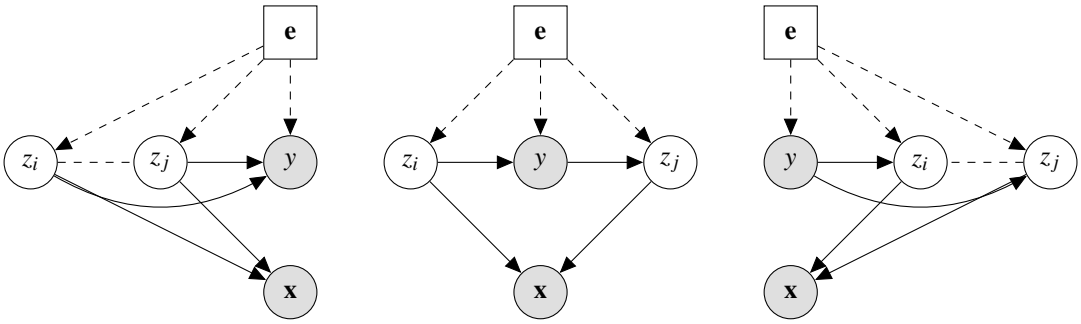
Fig. 4.1 The diagram of causal representation learning (steps I and II).

4.2 Step II: Identifying Causal Latent Factors

After recovering the true data representation by generalized iVAE in Step I, the next step is to identify causal latent factors of the target y. In this section, we propose a practical causal inference scheme to solve this problem.

Recall that a latent factor z_i can be either the cause of y, the effect of y, or independent of y. We first perform independence test between y and each z_i to exclude the latent factors that are independent of y. In practice, we can perform HSIC kernel independence test (Gretton et al., 2007) to infer statistical dependencies from samples.

After removing the latent factors which are independent of y, we can perform conditional independence test to identify the causal relationships between y and each of the remaining z_i. Conditional independence test is a general and powerful method for causal inference that exploits the conditional independence structures of causal graphs. For the causal inference problem considered here, one observation is that a pair of latent factors (z_i, z_j) are both the causes of y if and only if the dependency between them increases after conditioning on y (see Figure 4.2). To implement this idea, we can perform independence test and conditional independence test for each (z_i, z_j) pair and look for pairs that have increased dependencies after conditioning on y. This can be done by comparing p-values from these two tests. Note that there are at most $\mathcal{O}(n^2)$ tests to be performed, which can be parallelized in practice. To infer conditional statistical dependencies from samples, we resort to kernel conditional independence test (Zhang et al., 2012), which is a generalization of HSIC.



(a) Two causes. (b) One cause and one effect. (c) Two effects.

Fig. 4.2 Three possible cases for a pair of latent factors z_i and z_j which depend on y.

Note that there are some special cases to which the method described above is not applicable:

- If a latent factor z_i is a deterministic function of y, then conditioning on y makes z_i deterministic and thus z_i will always be independent of any other latent factor z_j. This is because a deterministic variable is always independent of any other variables. Information Geometric Causal Inference (IGCI) (Daniusis et al., 2012) is a practical method for inferring deterministic (noise-free) causal relationship between two variables (e.g., $y = g(z_i)$), which builds on the idea that the distribution of the effects should be "dependent" on the invertible nonlinear deterministic function g, but the distribution of the causes should be "independent" of g. The "dependence" between a distribution and a function is measured by an information geometry criterion. In practice, we can use Spearman's correlation coefficient (Spearman, 1904) to find out if there are any latent factors that have a deterministic relationship with y (e.g., latent factors whose Spearman's correlation coefficients with y are greater than, say, 0.98) and treat them separately by applying IGCI to them for causal discovery[1].

- In some cases, there could be only one causal latent factor of y. This violates the assumption of our method which looks for pairs of causal latent factors. If we cannot identify any pair of causal latent factors, we can resort to the Additive Noise Model (ANM) (Hoyer et al., 2008) to identify a single causal latent factor[2]. ANM builds on the assumption that the effect is a nonlinear function of the cause plus some additive noise.

[1] If the deterministic function g is linear (e.g., the Pearson's correlation coefficient between z_i and y is greater than, say, 0.98), then the causal relationship between z_i and y will be generally unidentifiable. In practice, we usually just count it as a causal latent factor.

[2] ANM is the most popular, simple and effective method for discovering causal relationship between two variables. There are also other methods available, such as post-nonlinear causal model (PNLCM) (Zhang and Hyvarinen, 2012). The reason why we prefer conditional independence test based method is because ANM and PNLCM make more assumptions about the relationship between z_i and y which may not hold in practice.

Note that the converse is not true due to the nonlinear function. In practice, for each z_i we can fit a nonlinear regression model for each of the two possible causal directions (i.e., $z_i \leftarrow y$ and $z_i \rightarrow y$) and see which direction satisfies the additive noise assumption by performing independence test between the residual and the input variable.

Overall, our causal latent factor identification scheme is summarized in Algorithm 3.

4.3 Step III: Performing Latent Space Optimization

Once we manage to identify the causal latent factors $\mathbf{z}_c$ of the target y in step II, we can then optimize y by intervening upon the causal latent factors $\mathbf{z}_c$ using standard LSO techniques along with weighted retraining as described in Algorithm 2. Our method has three advantages:

- Our latent space $\mathscr{Z}$ is more amenable to optimization, since the generalized iVAE can recover the true latent variable $\mathbf{z}_*$ with reference to y, achieving a principled form of disentanglement.
- We only need to optimize a subset $\mathbf{z}_c$ of the latent variable $\mathbf{z}$, which is easier and more efficient than standard LSO.
- Our surrogate model $h_{\mathscr{Z}_c}$ predicts y from causal latent factors $\mathbf{z}_c$, which gets rid of spurious correlations and is invariant across different environments.

Note that non-causal latent factors do not have any effects on the target y. Hence, we may just set their values to be the same as those for the current best data point in the dataset during optimization.

Algorithm 3: Practical scheme for identifying causal latent factors (Step II).

Input : Samples of true latent factors $\mathbf{z} = \{z_1, \cdots, z_n\}$ recovered by generalized iVAE in Step I and corresponding samples of target y, significance level α.

Initialize $\mathbf{z}_c \leftarrow \emptyset$, $\mathbf{z}_{det} \leftarrow \emptyset$, and $\mathbf{z}_{dep} \leftarrow \emptyset$

for $z_i \in \mathbf{z}$ **do**
  Test dependency between z_i and y to obtain p-value p_i^{uncon} using HSIC
  if $p_i^{uncon} \leq \alpha$ **then**
    Compute Spearman's correlation coefficient ρ_i^s between z_i and y
    if $\rho_i^s > 0.98$ **then**
      $\mathbf{z}_{det} \leftarrow \mathbf{z}_{det} \cup \{z_i\}$
    else
      $\mathbf{z}_{dep} \leftarrow \mathbf{z}_{dep} \cup \{z_i\}$
    end
  end
end
for $z_i \in \mathbf{z}_{det}$ **do**
  Compute Pearson's correlation coefficient ρ_i^p between z_i and y
  if $\rho_i^p > 0.98$ **then**
    $\mathbf{z}_c \leftarrow \mathbf{z}_c \cup \{z_i\}$
  else
    Discover causal relationship between z_i and y using IGCI
    if z_i *is the cause of* y **then**
      $\mathbf{z}_c \leftarrow \mathbf{z}_c \cup \{z_i\}$
    end
  end
end
for $z_i, z_j \in \mathbf{z}_{dep}$ $(i \neq j)$ **do**
  Test dependency between z_i and z_j to obtain p-value $p_{i,j}^{uncon}$ using HSIC
  Test conditional dependency between z_i and z_j given y to obtain p-value $p_{i,j|y}^{con}$
  if $p_{i,j|y}^{con} \leq p_{i,j}^{uncon}$ *and* $p_{i,j|y}^{con} \leq \alpha$ **then**
    $\mathbf{z}_c \leftarrow \mathbf{z}_c \cup \{z_i, z_j\}$
  end
end
if $\mathbf{z}_c$ *is empty* **then**
  for $z_i \in \mathbf{z}_{dep}$ **do**
    Discover causal relationship between z_i and y using ANM or PNLCM
    if z_i *is the cause of* y **then**
      $\mathbf{z}_c \leftarrow \mathbf{z}_c \cup \{z_i\}$
    end
  end
end

Output : Causal latent factors $\mathbf{z}_c$.

Chapter 5

Empirical Evaluations

In this chapter, we empirically evaluate and analyze our proposed framework on 1) a small synthetic dataset, 2) a midsize toy dataset for image optimization, and 3) a large molecular dataset for chemical design.

5.1 Synthetic Dataset

In this section, we demonstrate the identifiability of DLVMs with a non-factorized conditional prior trained by generalized iVAE on a synthetic dataset, similar to the one used in Lu et al. (2021). The underlying ground truth data generating process of this dataset is as follows:

$$e \sim \mathscr{U}\{0.2, 3.0, 6.0, 10.0\}, \tag{5.1}$$

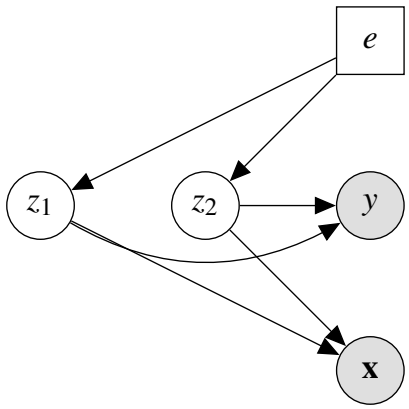
$$z_1 \sim \mathscr{N}(e, 1), \tag{5.2}$$

$$z_2 \sim \mathscr{N}(2e, 4), \tag{5.3}$$

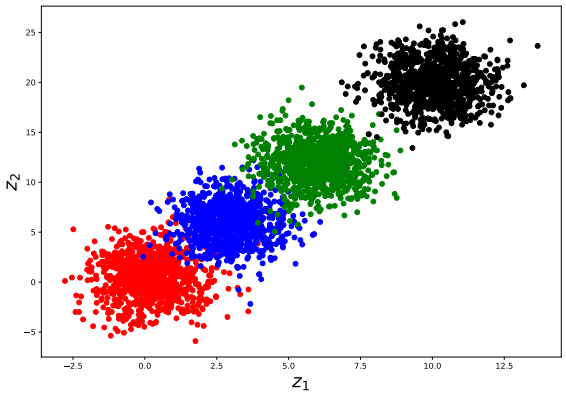
$$y \sim \mathscr{N}(z_1 + z_2 + z_1 z_2, 1), \tag{5.4}$$

$$\mathbf{x} = \mathbf{f}(z_1, z_2), \tag{5.5}$$

where the mixing function $\mathbf{f} : \mathbb{R}^2 \to \mathbb{R}^{10}$ is modelled by a single-hidden-layer neural network whose parameters are randomly preset and fixed. Figure 5.1a shows the causal graph for this data generating process, which fits into the problem setting considered in this thesis, where the latent variable $\mathbf{z} = \mathbf{z}_c = (z_1, z_2)$ causes the target y and the observed variable $\mathbf{x}$, and the two latent factors z_1 and z_2 are confounded by a hidden environment variable e. Note that z_1 and z_2 are conditionally dependent given y (i.e., $z_1 \not\perp\!\!\!\perp z_2 | y$). Therefore, we will need a non-factorized conditional prior $p_{\mathbf{T},\lambda}(\mathbf{z}|y)$.

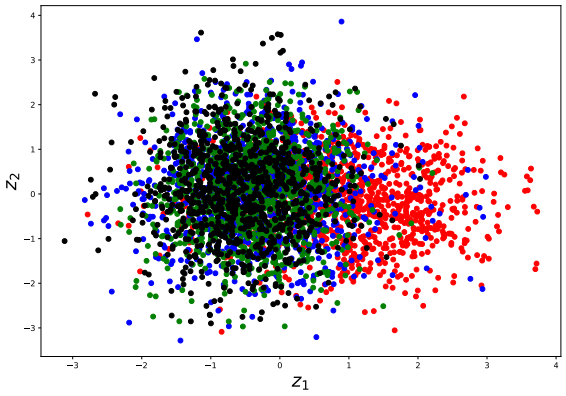


(a) Ground truth causal graph.

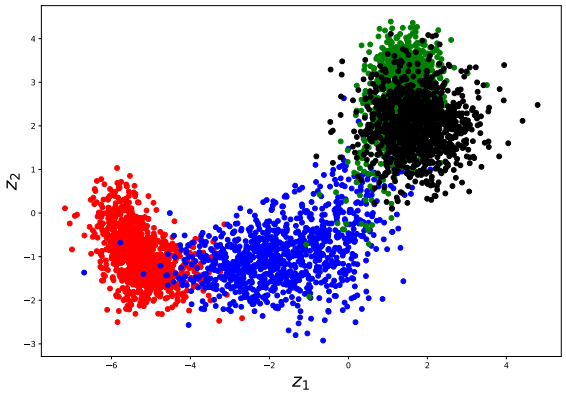


(b) Ground truth latent variable $\mathbf{z}_*$.

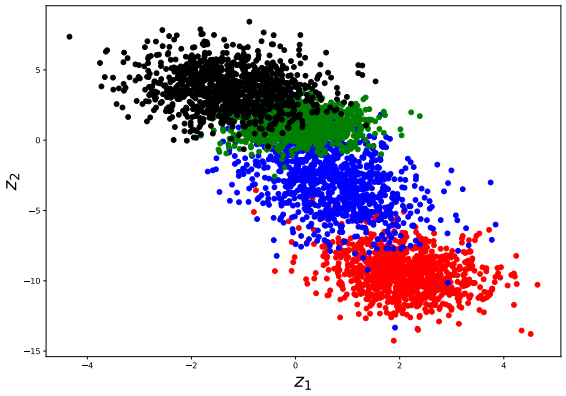
Fig. 5.1 (a) The causal graph that describes the data generating process of the synthetic dataset. (b) The scatter plot of samples of the ground truth latent variable $\mathbf{z}_*$ in the synthetic problem, where colors represent the ground truth environment e.



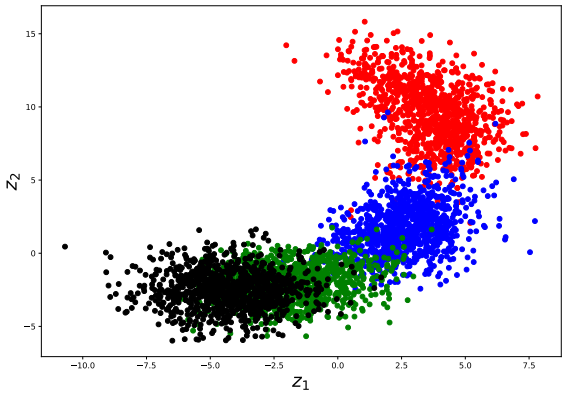
(a) VAE.



(b) iVAE.



(c) Generalized iVAE.



(d) Generalized iVAE (with access to e).

Fig. 5.2 Scatter plots of samples of the latent variable $\mathbf{z} = (z_1, z_2)$ recovered by (a) VAE, (b) iVAE, (c) generalized iVAE, and (d) generalized iVAE with access to e in the synthetic problem. Colors represent the ground truth environment variable e for illustration purpose.

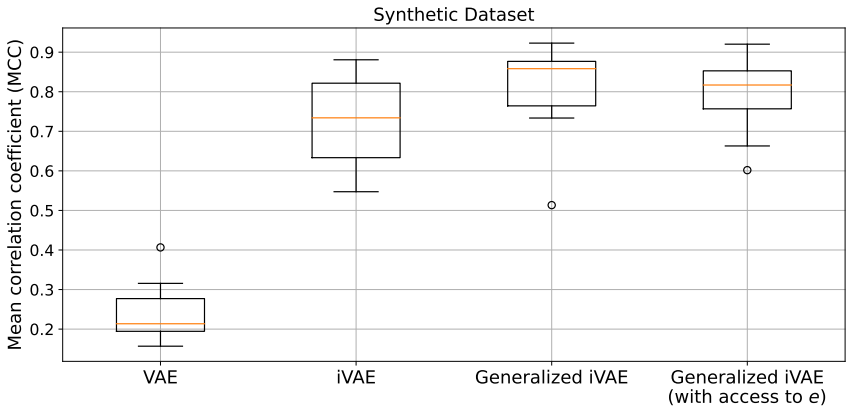


Fig. 5.3 Mean correlation coefficient (MCC) scores for VAE, iVAE, generalized iVAE, and generalized iVAE with access to e on the synthetic dataset.

The synthetic dataset consists of $4,000$ samples generated according to Equations (5.1)–(5.5). The task is to recover the true latent variable $\mathbf{z}_*$ using samples of $\mathbf{x}$ and y in this dataset. This means that we need to solve this problem without access to e or $\mathbf{z}$. We compare the performance of three DLVMs trained using different inference and learning schemes – VAE, iVAE, and generalized iVAE. For iVAE and generalized iVAE, we choose y as the auxiliary variable for the conditional prior $p_{\mathbf{T},\lambda}(\mathbf{z}|y)$. In theory, generalized iVAE should be able to recover the true latent variable up to simple transformations defined in Equation (3.6) (i.e., strong identifiability). We also include an additional result of generalized iVAE with access to samples of the ground truth environment variable e (i.e., including e in the auxiliary variable for the conditional prior $p_{\mathbf{T},\lambda}(\mathbf{z}|y,e)$) as a reference for comparison.

We visualize samples of the ground truth latent variable (as shown in Figure 5.1b) and samples of the latent variable recovered by the VAE, iVAE, and generalized iVAE using 2D scatter plots. It can be seen that generalized iVAE successfully recovers the true latent variable up to simple transformations, as shown in Figure 5.2c. The latent variable recovered by iVAE is not as good as that recovered by generalized iVAE, as there is an overlapping between samples from the red and green environments, as shown in Figure 5.2b. This is because the factorized conditional prior in iVAE is unable to handle dependencies between latent factors when conditioning on y. The latent variable recovered by VAE is clearly not identifiable, as shown in Figure 5.2a. It is worth noting that including e in the auxiliary variable for the conditional prior does not improve the quality of the latent variable recovered by generalized iVAE, as shown in Figure 5.2d.

We train each model ten times with different random seeds and compute the MCC score between the ground truth latent variable and the latent variable recovered with each random seed. Figure 5.3 is a box plot that shows the MCC scores for VAE, iVAE, generalized iVAE, and generalized iVAE with access to e on the synthetic dataset. It can be seen that the median of the MCC score for generalized iVAE is greater than 0.85, indicating very strong identifiability, whereas the median of the MCC score for iVAE is less than 0.75 and that for VAE is less than 0.25. This also confirms that including e in the auxiliary variable for the conditional prior does not improve the identifiability of generalized iVAE. In fact, it even results in a slightly lower median MCC score in this experiment. These results are consistent with the visualizations shown in Figure 5.2.

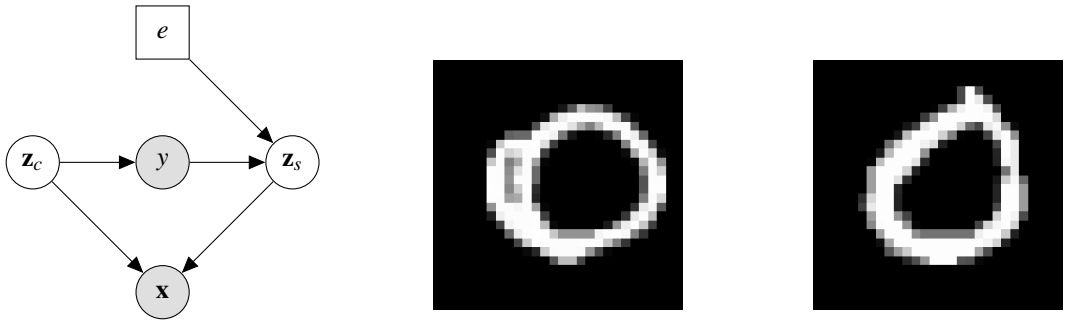
Overall, this experiment shows that the assumptions in Theorem 6 can be met in practice, and the true latent variable $\mathbf{z}_*$ can be recovered in the finite data regime.

5.2 Image Optimization

In this section, we apply our proposed framework to an image optimization task, starting from a toy dataset Colored MNIST. We will analyze our findings in each step.

Dataset

The Colored MNIST dataset is created by coloring the images in the original MNIST dataset (LeCun, 1998), so that the color pixel values are spuriously correlated with the target $y = J(\mathbf{x})$. We will define the objective $J(\mathbf{x})$ later. We color the digit images in a different way to Arjovsky et al. (2019). For each image in the MNIST dataset, we first normalize its pixel values to the interval $[0,1]$. Then we sample an environment variable $e \sim \mathscr{U}\{0,1\}$ and a Bernoulli random variable $b \sim \text{Bernoulli}(p_e)$ with $p_0 = 0.2$ and $p_1 = 0.1$. We color the image by appending two additional channels to it. Each pixel in these two channels is sampled from a Gaussian distribution $\mathscr{N}(\mu_{b,e}, \sigma_e^2)$ with $\sigma_0 = 0.01$ and $\sigma_1 = 0.05$. Note that the mean color $\mu_{b,e}$ depends on both b and e. We set $\mu_{1,e} \sim \mathscr{U}[0,1]$ for both environments $e = 0, 1$, which means that the image is colored randomly if $b = 1$. We set $\mu_{0,0} = \text{Sigmoid}\left(2(y - \hat{\mathbb{E}}[y])/\sqrt{\hat{\text{Var}}[y]}\right)$ and $\mu_{0,1} = \text{Sigmoid}\left(-2(y - \hat{\mathbb{E}}[y])/\sqrt{\hat{\text{Var}}[y]}\right)$, where $\hat{\mathbb{E}}$ and $\hat{\text{Var}}$ are empirical mean and variance over the training set. This means that the image is colored according to the target value y and the environment e if $b = 0$. Finally, we clip the pixel values of the resulting image to the interval $[0,1]$.



(a) Causal graph. (b) Hold-out image $\mathbf{x}^*$. (c) Best image in the training set.

Fig. 5.4 (a) The causal graph for the data generating process of the Colored MNIST dataset. (b) The first channel of the hold-out image $\mathbf{x}^*$ used in the image optimization objective. (c) The first channel of the image in the Colored MNIST training set that has the highest objective value ($y = -1900.48$).

Figure 5.4a shows the causal graph that describes the data generating process of the Colored MNIST dataset, where the causal latent factors $\mathbf{z}_c$ of the target y control the shape of the digit in the image $\mathbf{x}$, and the effect latent factors $\mathbf{z}_s$ of y control the color of the image $\mathbf{x}$. Note that the environment variable e is assumed to be latent, and we only have access to samples of $\mathbf{x}$ and y in the dataset $\mathscr{D}$.

Objective

The black-box objective function to be maximized is defined as the negative Euclidean distance between the first channel of the input colored image $\mathbf{x}$ and the first channel of a hold-out target image $\mathbf{x}^*$ from the test set of the MNIST dataset:

$$J(\mathbf{x}) = -\|\mathbf{h}(\mathbf{x}) - \mathbf{h}(\mathbf{x}^*)\|_2, \tag{5.6}$$

where $\mathbf{x}^*$ is chosen to be the image of digit zero in the test set such that it is the farthest one from all images in the training set, as shown in Figure 5.4b, and the function $\mathbf{h}$ outputs the first channel of an input image in the form of a flattened vector. This objective function implies that the color of an image $\mathbf{x}$ and the latent factors $\mathbf{z}_s$ that control the color are spuriously correlated with the target $y = J(\mathbf{x})$, and therefore predicting y from the color of an image (or $\mathbf{z}_s$) will fail catastrophically when the environment e changes. Hence, this essentially simulates the spurious correlation in the camel-cow classification task. It is also worth noting that although the effect latent factors $\mathbf{z}_s$ are correlated with the target y, they do not cause y. Therefore, intervening upon them will not change the objective value during optimization.

Note that the objective function is assumed to be a black box in this task, which means that we can only evaluate it at different input images but cannot use its analytical form or derivative information during optimization.

Step I: Recover True Data Representation

We recover the true representation of colored MNIST images by training a DLVM $p_\theta(\mathbf{x},\mathbf{z}\,|y) = p_\mathbf{f}(\mathbf{x}\,|\,\mathbf{z})p_{\mathbf{T},\lambda}(\mathbf{z}\,|y)$ using generalized iVAE. To show that this model can recover the true latent variable $\mathbf{z}_*$ that has generated the image $\mathbf{x}$, we quantify its identifiability using MCC scores. Since we do not have access to the true latent variable, we train this DLVM five times with different random seeds and compute the MCC scores between the latent variables recovered by each pair of models (there are $\binom{5}{2} = 10$ pairs of models in total). We also include additional results for DLVMs trained by generalized iVAE with access to the ground truth environment e (i.e., $p_\theta(\mathbf{x},\mathbf{z}\,|y,e) = p_\mathbf{f}(\mathbf{x}\,|\,\mathbf{z})p_{\mathbf{T},\lambda}(\mathbf{z}\,|y,e)$) and VAE (i.e., $p_\theta(\mathbf{x},\mathbf{z}) = p_\mathbf{f}(\mathbf{x}\,|\,\mathbf{z})p(\mathbf{z})$) for comparison. For all three DLVMs, we set the dimensions of the latent space $\mathscr{Z}$ to $n = 8$ and train them on the training set which consists of $60,000$ data points.

It can be seen in Figure 5.5 that the median of the MCC score for generalized iVAE is 0.7, indicating that the latent variable $\mathbf{z}$ recovered by this model has good identifiability. The median of the MCC score for VAE is around 0.47, which shows that the identifiability of VAE is much weaker than generalized iVAE on the Colored MNIST dataset. Interestingly, including e in the auxiliary variable for the conditional prior in generalized iVAE weakens its identifiability. This means that knowing e does not improve the identifiability of generalized iVAE in this problem, which is consistent with the conclusion made for the synthetic dataset.

Step II: Identifying Causal Latent Factors

We use Algorithm 3 to identify causal latent factors $\mathbf{z}_c$ from all latent factors $\mathbf{z}$ recovered by the DLVM $p_\theta(\mathbf{x},\mathbf{z}\,|y) = p_\mathbf{f}(\mathbf{x}\,|\,\mathbf{z})p_{\mathbf{T},\lambda}(\mathbf{z}\,|y)$ trained using generalized iVAE in Step I. We find that the five models trained with different random seeds are of a similar qualitative nature, so here we only show results for one of the five models. We randomly choose $1,000$ observational samples of $(\mathbf{x},y)$ from the dataset $\mathscr{D}$ and encode them to obtain corresponding samples of $\mathbf{z}$. Figure 5.6a shows the scatter plot of y against each z_i using these $1,000$ samples. Applying Algorithm 3 to these $1,000$ samples of $(\mathbf{z},y)$ gives us the following results:

1. All latent factors in $\mathbf{z}$ are dependent on y.
2. $z_2, z_4, z_5, z_6, z_7, z_8$ are the causal latent factors of y.

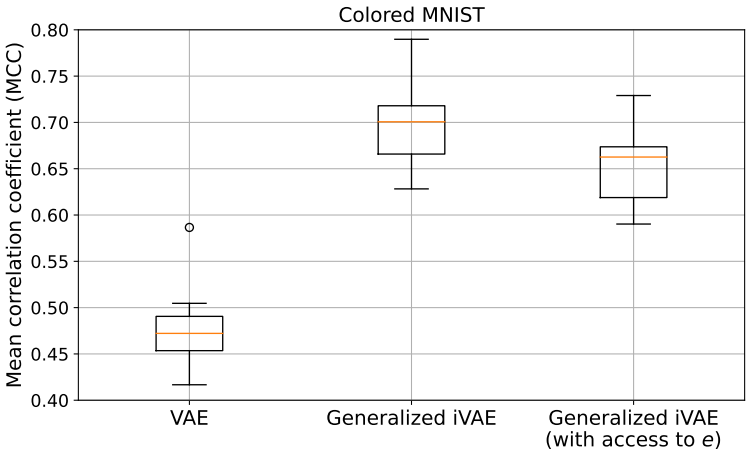
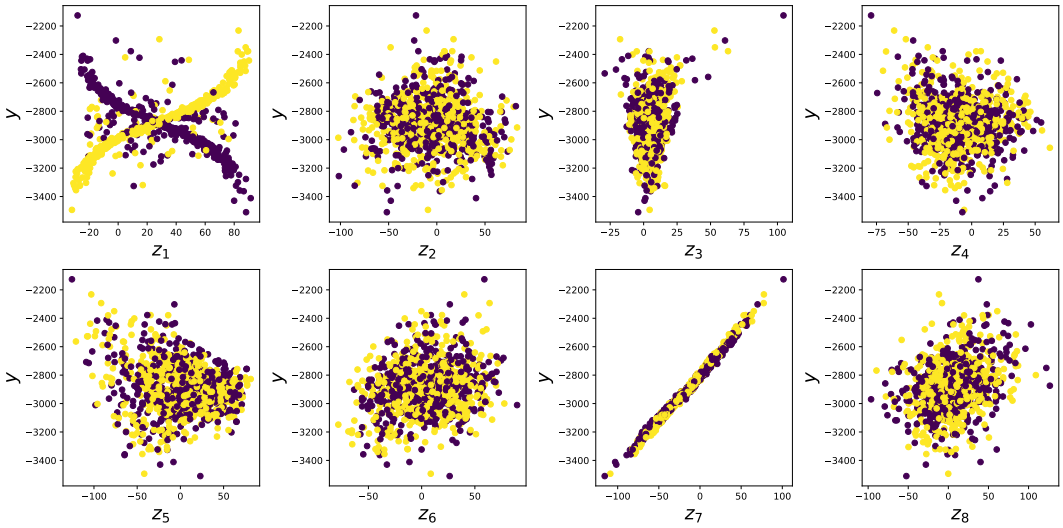


Fig. 5.5 Mean correlation coefficient (MCC) scores for VAE, iVAE, generalized iVAE, and generalized iVAE with access to e on the Colored MNIST dataset.

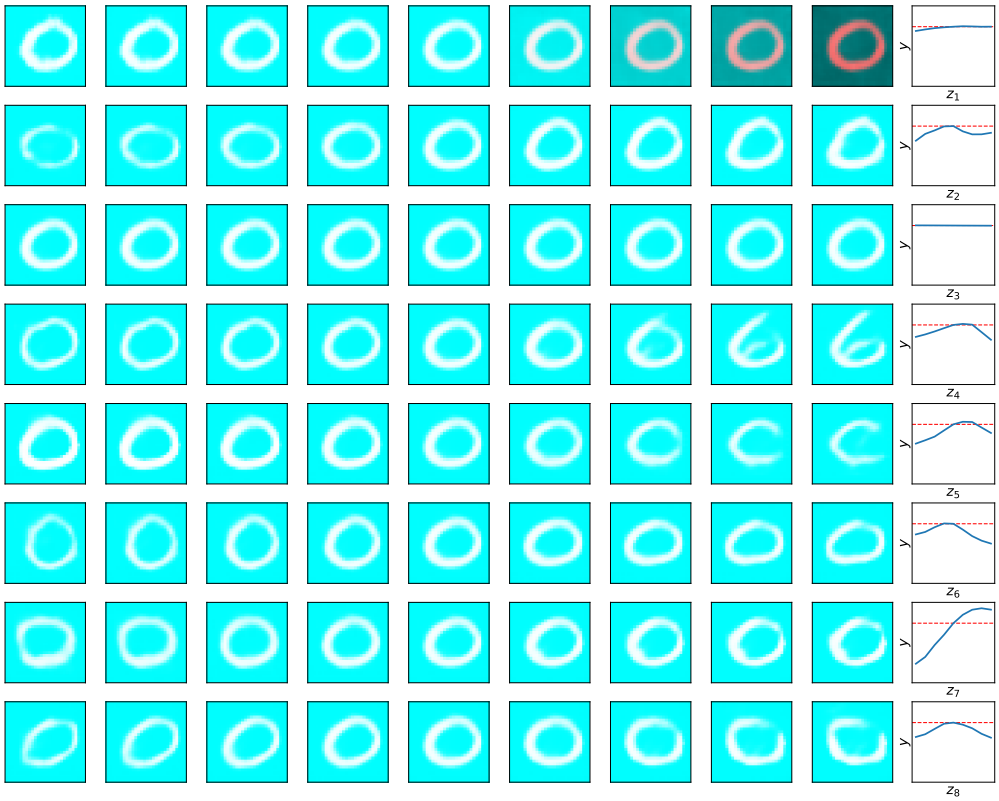
3. z_1 and z_3 are the effect latent factors of y.

We also perform intervention upon each z_i to verify the causal identification results above. Figure 5.6b shows how intervening upon each z_i affects the image $\mathbf{x}$ and the target y. It can be seen that intervening upon each of the latent factors $z_2, z_4, z_5, z_6, z_7, z_8$ affects the shape of the digit in the image and the target y but not the color of the image, which confirms that they are the causes of y. In contrast, intervening upon z_1 affects the color of the image but not the shape of digit in the image or the target y, which confirms that z_1 is the effect of y. This implies that y and z_1 are spuriously correlated and their relationship is not invariant across different environments, which is consistent with the data generating process and the observational samples shown in the leftmost plot in the first row of Figure 5.6a. Interestingly, intervening upon z_3 does not change anything, showing that it is a non-causal latent factor, although it does depend on y according to observational samples.

As a reference, we produce the same plots for generalized iVAE with access to the ground truth environment variable e in Figure 5.7, which is qualitatively similar to Figure 5.6. In this case, Algorithm 3 also identifies six causal latent factors $z_1, z_2, z_3, z_5, z_6, z_7$, showing that including e in the auxiliary variable for the conditional prior does not change our ability to identify causal latent factors using Algorithm 3. We also produce such plots for VAE in Figure 5.8 for comparison. It can be seen in Figure 5.8b that intervening upon z_4 affects the color of the image, the shape of the digit in the image, and the target y. This shows that VAE cannot recover the true latent variable, since the effect latent factor is entangled with some of the causal latent factors.

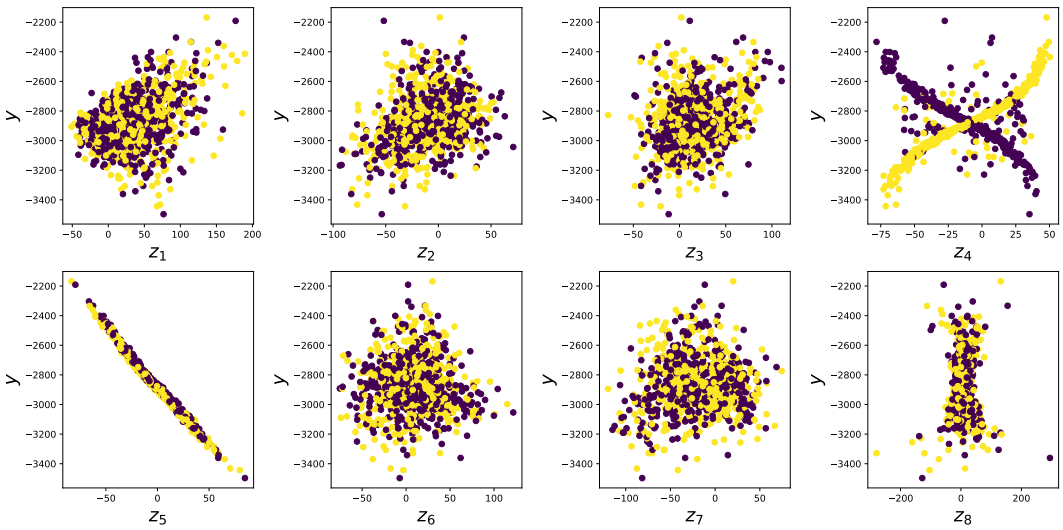


(a) Observational samples of y against each z_i.

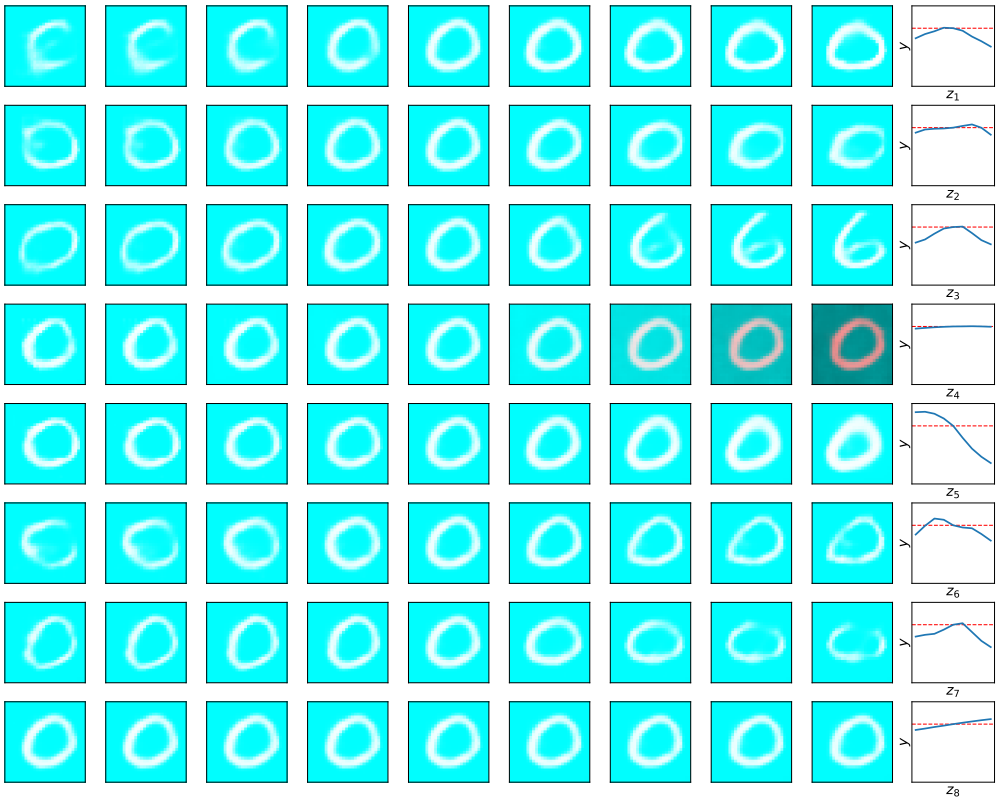


(b) In each row, the rightmost plot shows y against displacement of z_i, and the images show the corresponding changes of $\mathbf{x}$ when displacing z_i. The initial image before intervention is in the middle (column 5) of each row, whose objective y is indicated by the red dashed line in the rightmost plot.

Fig. 5.6 Generalized iVAE: (a) The scatter plot of samples of y against each z_i from observational data, where colors represent the (unknown) ground truth environment variable e for illustration purpose. (b) The effects on the image $\mathbf{x}$ and target y when intervening on each z_i.

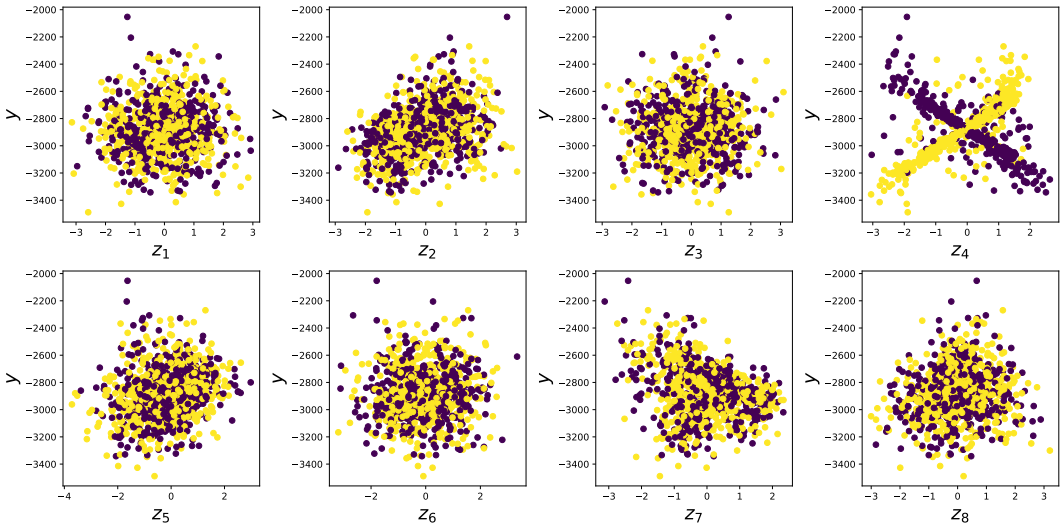


(a) Observational samples of y against each z_i.

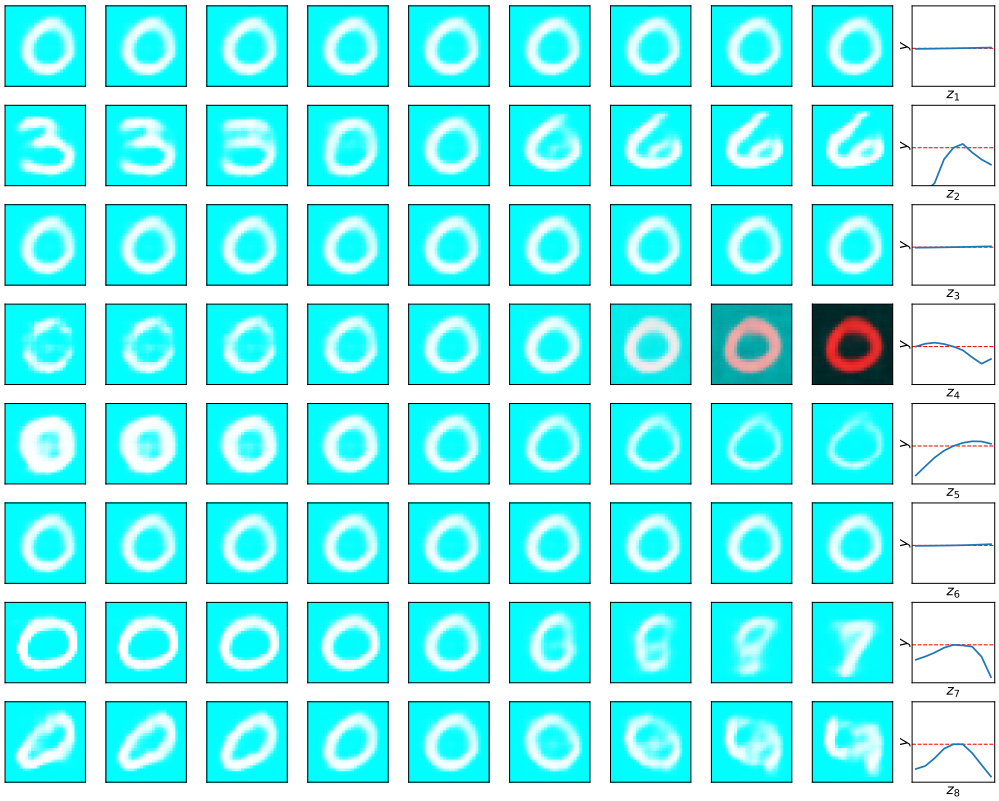


(b) In each row, the rightmost plot shows y against displacement of z_i, and the images show the corresponding changes of $\mathbf{x}$ when displacing z_i. The initial image before intervention is in the middle (column 5) of each row, whose objective y is indicated by the red dashed line in the rightmost plot.

Fig. 5.7 Generalized iVAE with access to e: (a) The scatter plot of samples of y against each z_i from observational data, where colors represent the ground truth environment variable e. (b) The effects on the image $\mathbf{x}$ and target y when intervening on each z_i.



(a) Observational samples of y against each z_i.



(b) In each row, the rightmost plot shows y against displacement of z_i, and the images show the corresponding changes of $\mathbf{x}$ when displacing z_i. The initial image before intervention is in the middle (column 5) of each row, whose objective y is indicated by the red dashed line in the rightmost plot.

Fig. 5.8 VAE: (a) The scatter plot of samples of y against each z_i from observational data, where colors represent the (unknown) ground truth environment variable e for illustration purpose. (b) The effects on the image $\mathbf{x}$ and target y when intervening on each z_i.

Step III: Performing Latent Space Optimization

We perform LSO with weighted retraining by intervening upon the causal latent factors $\mathbf{z}_c$ identified in step II. The values of all non-causal latent factors are set to be the same as those for the current best data point in the dataset $\mathscr{D}$. We find that the underlying mapping from the causal latent factors $\mathbf{z}_c$ to the target y is extremely non-smooth, which makes it very difficult to fit a surrogate model $h_{\mathscr{Z}_c}$. Therefore, for illustration purpose we instead enumerate the causal latent factors $\mathbf{z}_c$ to obtain a coarse optimizer, where we evaluate each latent factor at 7 linearly spaced grid points within its feasible region, resulting in 7^8 evaluations for all latent factors or 7^6 evaluations for causal latent factors. The feasible region of a latent factor is chosen to be between its empirical minimum and maximum from samples. We choose $k = 10^{-3}$ for the rank-based weighting hyper-parameter. We collect $r = 10$ best points and retrain the DLVM for $N_{re} = 1$ epoch in each optimization round. For each of the three DLVMs, we perform LSO for all five models obtained with different random seeds in step I and report the mean top1 optimization performance with standard deviation. For generalized iVAEs, we also compare their optimization performance when using causal latent factors $\mathbf{z}_c$ with that when using all latent factors $\mathbf{z}$.

The LSO performance for the image optimization task is shown in Figure 5.9. It can be seen that the performance of generalized iVAE using causal latent factors (the green curve) is almost identical to that of generalized iVAE using all causal latent factors (the orange curve), indicating that the non-causal latent factors indeed have almost no effect on y during optimization. Also, including e in the auxiliary variable for the conditional prior in generalized iVAE (the red and purple curves) does not improve the optimization performance. In addition, generalized iVAEs significantly outperform VAE (the blue curve). In fact, the mean performance that our method achieved at the first evaluation is already better than that achieved by VAE in the final optimization round.

Overall, we conclude that our method improves both the efficiency and performance of LSO for this image optimization task. The performance gain is due to the ability of generalized iVAEs to recovering the true latent variable $\mathbf{z}_*$ with reference to the target y, achieving a principled form of disentanglement in the latent space $\mathscr{Z}$. The efficiency gain is due to the ability of our causal inference scheme to identifying causal latent factors $\mathbf{z}_c$, which enables us to search a subspace of $\mathscr{Z}$ during optimization without performance loss.

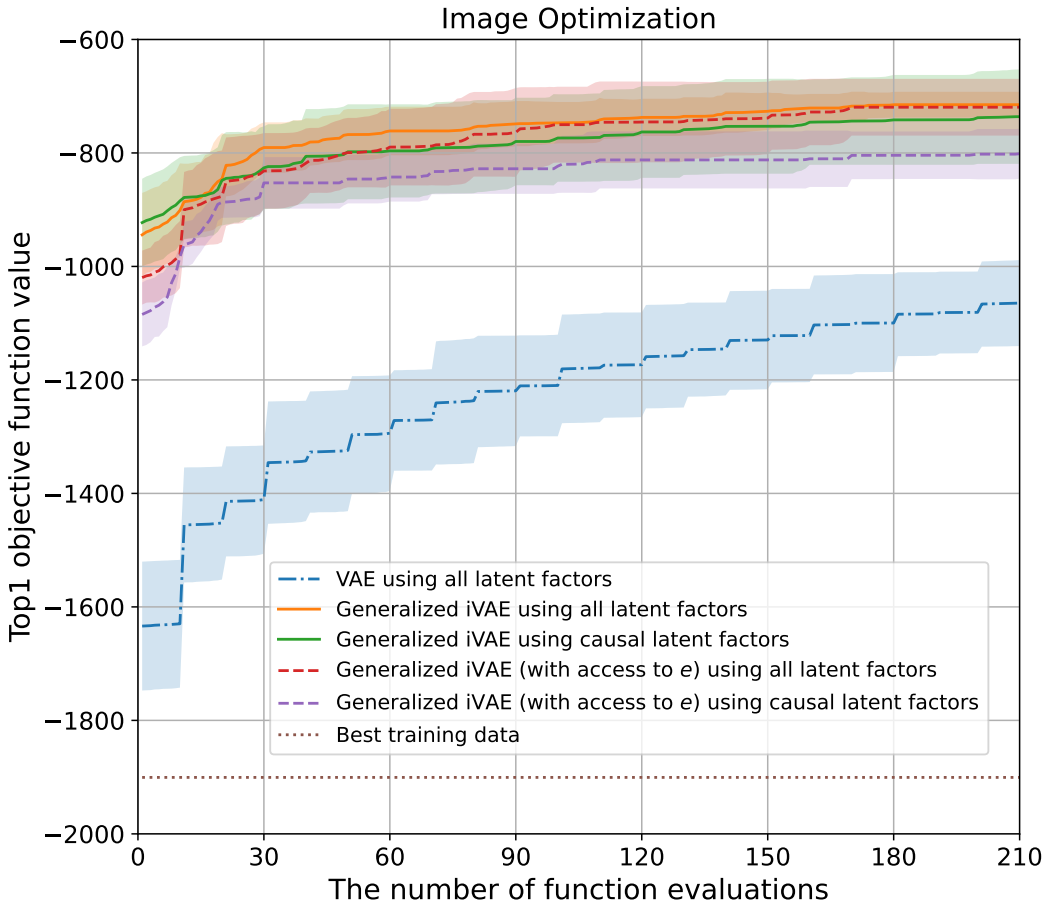


Fig. 5.9 Top1 image optimization performance starting from the Colored MNIST dataset with weighted retraining ($k = 10^{-3}$, $r = 10$ and $N_{re} = 1$) obtained by VAE using all latent factors and generalized iVAEs (with and without access to e) using all latent factors and using causal latent factors. Shaded areas correspond to standard deviation.

5.3 Chemical Design

Chemical design is an important application of LSO (Gómez-Bombarelli et al., 2018), which aims to generate novel molecules with maximal drug properties. In this section, we apply our proposed framework to a chemical design task starting from a large molecular dataset ZINC-250K, following the problem setup considered in Gómez-Bombarelli et al. (2018).

Molecular Dataset and Representation

The ZINC-250K dataset (Irwin et al., 2012) contains $250,000$ molecules for chemical discovery. This is a challenging dataset, since the true data generating process is completely unknown and could be highly complicated. Also, it is unclear what the environment variable $\mathbf{e}$ represents in this case. For this task, we represent molecular graphs in line notation using Simplified Molecular Input Line Entry System (SMILES) (Weininger, 1988, 1990; Weininger et al., 1989). SMILES strings can be easily processed by standard natural language processing models, such as Long Short Term Memory (LSTM) (Hochreiter and Schmidhuber, 1997) neural networks with word embedding (Mikolov et al., 2013). Since a molecule may be represented by multiple different SMILES strings, we use canonical SMILES to create a one-to-one map between molecules and SMILES strings, which specifies a particular ordering of atoms. It is easy for human to understand SMILES strings, although in some cases SMILES may transform short-range dependencies between atoms in a molecular graph into long-range dependencies in the corresponding SMILES strings.

Objective

The black-box objective function to be maximized is the penalized water-octanol partition coefficient (penalized logP) property function:

$$J(\mathbf{x}) = \hat{\log\mathrm{P}}(\mathbf{x}) - \hat{\mathrm{SA}}(\mathbf{x}) - \hat{\mathrm{cycle}}(\mathbf{x}), \tag{5.7}$$

where $\log\mathrm{P}(\mathbf{x})$ is the water-octanol partition coefficient property function, $\mathrm{SA}(\mathbf{x})$ measures the synthetic accessibility of a molecule $\mathbf{x}$, $\mathrm{cycle}(\mathbf{x})$ counts the number of rings with lengths greater than 6 in a molecule $\mathbf{x}$, and the hat operator standardizes the raw output with the empirical mean and variance statistics computed from the ZINC dataset. This is a standard chemical design task first proposed by Gómez-Bombarelli et al. (2018), which has since been studied in many papers (Dai et al., 2018; Jin et al., 2018; Kusner et al., 2017; Tripp et al., 2020; You et al., 2018; Zhou et al., 2019). Figure 5.10 shows the molecule with the highest target value ($y = 4.52$) in the ZINC-250K dataset.

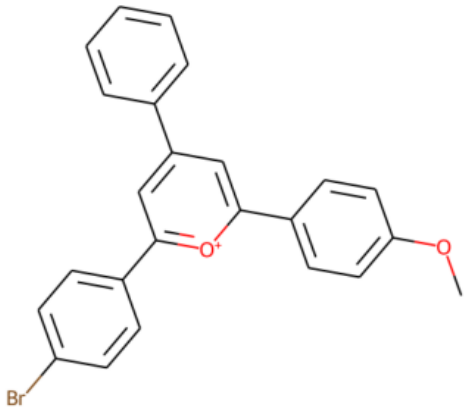


Fig. 5.10 The molecule with the highest penalized logP drug property ($y = 4.52$) in the ZINC-250K dataset.

Results and Discussions

We train the DLVM $p_\theta(\mathbf{x}, \mathbf{z}|y) = p_\mathbf{f}(\mathbf{x}|\mathbf{z})p_{\mathbf{T},\lambda}(\mathbf{z}|y)$ on the ZINC-250K dataset using generalized iVAE, so as to recover the true latent variable from which the molecules originated. This is followed by causal identification and LSO using causal latent factors. We repeat this process five times with different random seeds. Following Tripp et al. (2020), we set the dimensions of the latent space $\mathscr{Z}$ to $n = 56$ and the hyper-parameters of weighted retraining to $k = 10^{-3}$, $r = 50$, and $N_{re} = 1$. To highlight the advantages of our method, we use simple LSTM neural networks with word embedding for the encoder and decoder architectures in generalized iVAE and enumerate a coarse optimizer that only evaluates 5 grid points for each causal latent factor, whereas Tripp et al. (2020) use a more advanced junction tree VAE (Jin et al., 2018) and employ Bayesian optimization with a sparse Gaussian process (Titsias, 2009) and the expected improvement acquisition function (Jones et al., 1998).

Figure 5.11 shows the top1 optimization performance for this chemical design task, in which we compare our method with the start-of-the-art method (Tripp et al., 2020) in the literature. It can be seen that the mean performance of our method significantly outperforms LSO with VAE which uses all 56 latent factors. In fact, the mean performance that our method achieved after the first optimization round (at the 50th evaluation) is already better than that achieved by the other method in the final optimization round. However, the performance of our method also has a significantly larger variance.

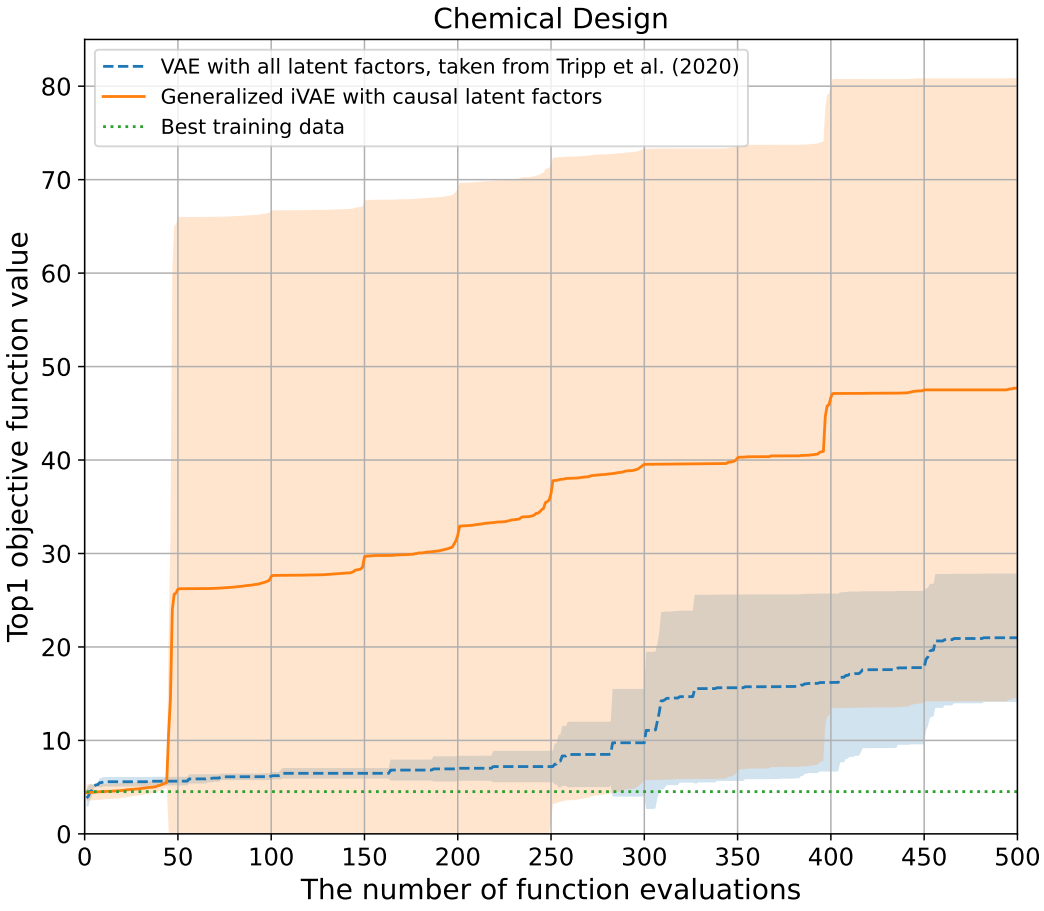


Fig. 5.11 Top1 chemical design performance starting from the ZINC-250K dataset with weighted retraining ($k = 10^{-3}$, $r = 50$ and $N_{re} = 1$) obtained by VAE using all latent factors and generalized iVAE using causal latent factors. Shaded areas correspond to standard deviation.

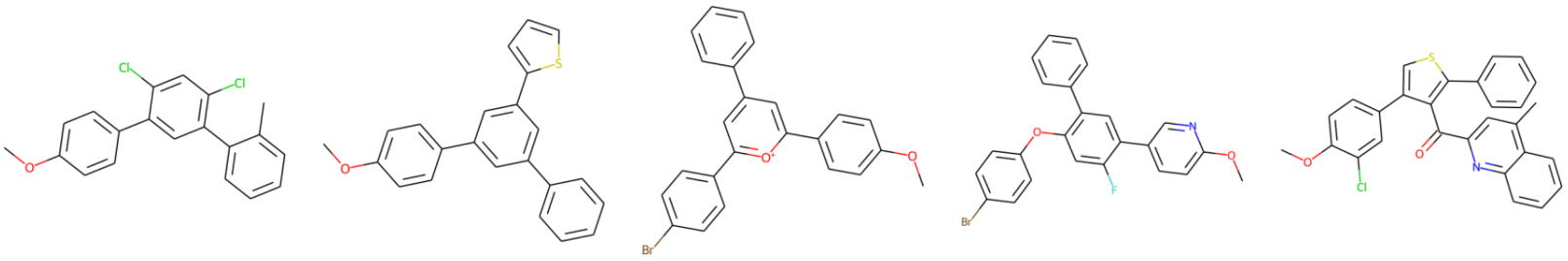
Table 5.1 The summary of causal identification and optimization results obtained by our method for the chemical design task. The dimensions of the latent space $\mathscr{Z}$ are $n = 56$.

Random seed	$\#z_i$ depending on y	$\#z_i$ causing y	highest y value obtained
Seed 1	7	5	62.50
Seed 2	4	4	105.78
Seed 3	6	3	28.92
Seed 4	6	4	13.96
Seed 5	5	3	27.31

Now we take a closer look at our models. Table 5.1 summarizes the causal identification results and optimization performance of each of our five models trained with different random seeds. Interestingly, there are only 4-7 latent factors that are dependent on the target y and only 3-5 latent factors that cause y. We suspect that this is because the logP property function is a sparse objective which is only affected by a few atoms or substructures in the molecules. Therefore, the majority of latent factors are identified to be irrelevant to the target y, which significantly improves the efficiency and effectiveness of LSO.

Finally, we try to interpret the latent factors by performing interventions. Interestingly, when we intervene upon some non-causal latent factors, we find that the molecular structure changes while the target value almost remains constant (see Figure 5.12 for an example). Intervene upon causal latent factors results in changes of both molecular structure and target value. However, we cannot spot any patterns of such changes of molecular structure, although there is one causal latent factor strongly correlated with y, similar to z_7 in Figure 5.6b in the image optimization task. Our conjecture is that intervening upon the causes of the penalized logP target may not necessarily result in obvious patterns of molecular structure changes.

On the other hand, the best molecules obtained with random seeds 1 and 2 are a long chain of carbon atoms ($y = 65.50$) and a long chain that mostly consists of sulfur atoms ($y = 105.78$), respectively. Note that there is no such type of molecules in the ZINC-250K dataset in terms of the molecular structure and size. This means that these models can generate molecules of good penalized logP drug properties that are far away from the initial data distribution, which suggests that our method may actually have learned the underlying causal mechanism for this chemical design task. However, further investigations will be needed in order to confirm this, which is left for future work. The best molecules obtained with random seeds 3-5 consist of many duplicates of the ring structures presented in Figure 5.10, indicating that these models manage to exploit the useful information in the initial dataset.



(a) $y = 4.60$. (b) $y = 4.63$. (c) $y = 4.52$. (d) $y = 4.49$. (e) $y = 4.67$.

Fig. 5.12 The changes of molecular structure and target drug property when intervening upon one of the non-causal latent factors. The molecule in (c) is the initial molecule before intervention.

Chapter 6

Conclusions

6.1 Discussions

In this thesis, we presented and investigated causal representation learning for latent space optimization. We extended iVAEs to a more general case where non-factorized conditional priors are used, for which we obtained novel identifiability theorems. This allowed us to recover the true data representation based on a practical assumption that the prior over the latent variable given the target is a general non-factorized exponential distribution. We also proposed a practical causal inference scheme to identify causal latent factors of the target from the data representation recovered by generalized iVAEs. Our causal representation learning and identification scheme is applicable to a wider range of problem settings than those considered in the existing works in the literature, since we made less assumptions of the causal graph for the data generating process. We also argued that causal representation learning enabled better LSO in terms of robustness, efficiency, and performance.

We demonstrated the identifiability of generalized iVAE on a synthetic dataset. We saw that generalized iVAE achieved a very high median MCC score (> 0.85) and was the only learning and inference scheme that managed to recover the true latent variable from which the observed data had originated up to simple transformations. We also considered an image optimization task, where the target is to generate digits that have minimal distances to a digit in the hold-out image, regardless of color. We saw that our method managed to identify the causal latent factors that control the shape of the digit in an image and discard the non-causal latent factors that control the color of the image. By intervening upon the causal latent factors, our method significantly outperformed LSO with VAE that uses all latent factors. Finally, we applied our method to a standard chemical design task, where the target is to generate molecules that maximize the penalized logP drug property. Our method identified

very few causal latent factors (3-5 out of 56). By intervening upon these causal latent factors, the mean optimization performance of our method was significantly better than that of the state-of-the-art method in the literature which used all 56 latent factors for optimization. We generated novel molecules with high target values that had completely different molecular structures and sizes than the ones in the initial dataset, suggesting that our method might manage to learn the underlying causal mechanism of the target and thus could generate novel molecules that were far away from the initial data distribution.

6.2 Future Work

Although we showed that our proposed method produced impressive results for both causal representation learning and black-box optimization, there are many other interesting things we could investigate if we had more time, which will be left for future work.

Sampling from the prior. An alternative and possibly more efficient way to perform LSO with causal representation learning would be to generate new data points by directly sampling from the conditional prior $p_{\mathbf{T},\lambda}(\mathbf{z}\,|y)$ in generalized iVAEs. If we condition on a high value of y, we should be able to draw samples of $\mathbf{z}$ that can be decoded to $\mathbf{x}$ with high target values. The main difficulty of this approach comes from the fact that the conditional prior is a super flexible energy-based model parameterized by neural networks, which has an intractable normalizing constant. Hence, a proper sampling scheme would need to be devised.

Investigating when the method will work. Our proposed method worked well on the two black-box optimization tasks considered in this thesis. However, we found that it could also fail in some cases. For example, in other chemical design tasks (e.g., when the drug property function is discrete and/or bounded), our method can fail to recover the true latent variable, fail to identify causal latent factors or fail to obtain data points with high target values. It would be useful to perform sensitivity analysis in each step to get a better understanding of when our method will and will not work.

Interpreting the causal latent factors for molecules. Our proposed method achieved great optimization performance on the chemical design task considered in this thesis. However, we found it difficult to interpret the meanings of those causal latent factors identified in this task. It would be interesting to investigate more regarding the interpretability of the causal latent factors for molecules.

References

Arjovsky, M., Bottou, L., Gulrajani, I., and Lopez-Paz, D. (2019). Invariant risk minimization. *arXiv preprint arXiv:1907.02893*.

Beery, S., Van Horn, G., and Perona, P. (2018). Recognition in terra incognita. In *Proceedings of the European Conference on Computer Vision (ECCV)*.

Bengio, Y., Courville, A., and Vincent, P. (2013). Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1798–1828.

Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. (2020). A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR.

Comon, P. (1994). Independent component analysis, a new concept? *Signal processing*, 36(3):287–314.

Dai, H., Tian, Y., Dai, B., Skiena, S., and Song, L. (2018). Syntax-directed variational autoencoder for structured data. *arXiv preprint arXiv:1802.08786*.

Daniusis, P., Janzing, D., Mooij, J., Zscheischler, J., Steudel, B., Zhang, K., and Schölkopf, B. (2012). Inferring deterministic causal relations. *arXiv preprint arXiv:1203.3475*.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Frazier, P. I. (2018). A tutorial on bayesian optimization. *arXiv preprint arXiv:1807.02811*.

Gómez-Bombarelli, R., Wei, J. N., Duvenaud, D., Hernández-Lobato, J. M., Sánchez-Lengeling, B., Sheberla, D., Aguilera-Iparraguirre, J., Hirzel, T. D., Adams, R. P., and Aspuru-Guzik, A. (2018). Automatic chemical design using a data-driven continuous representation of molecules. *ACS central science*, 4(2):268–276.

Gretton, A., Fukumizu, K., Teo, C. H., Song, L., Schölkopf, B., Smola, A. J., et al. (2007). A kernel statistical test of independence. In *Nips*, volume 20, pages 585–592. Citeseer.

Heinze-Deml, C., Peters, J., and Meinshausen, N. (2018). Invariant causal prediction for nonlinear models. *Journal of Causal Inference*, 6(2).

Higgins, I., Matthey, L., Pal, A., Burgess, C., Glorot, X., Botvinick, M., Mohamed, S., and Lerchner, A. (2016). beta-vae: Learning basic visual concepts with a constrained variational framework. In *International Conference on Learning Representations*.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8):1735–1780.

Hoyer, P. O., Janzing, D., Mooij, J. M., Peters, J., Schölkopf, B., et al. (2008). Nonlinear causal discovery with additive noise models. In *NIPS*, volume 21, pages 689–696. Citeseer.

Hyvärinen, A. (2005). Estimation of non-normalized statistical models by score matching. *Journal of Machine Learning Research*, 6(24):695–709.

Hyvarinen, A. and Morioka, H. (2016). Unsupervised feature extraction by time-contrastive learning and nonlinear ica. *Advances in Neural Information Processing Systems*, 29:3765–3773.

Hyvärinen, A. and Pajunen, P. (1999). Nonlinear independent component analysis: Existence and uniqueness results. *Neural networks*, 12(3):429–439.

Hyvarinen, A., Sasaki, H., and Turner, R. (2019). Nonlinear ica using auxiliary variables and generalized contrastive learning. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 859–868. PMLR.

Irwin, J. J., Sterling, T., Mysinger, M. M., Bolstad, E. S., and Coleman, R. G. (2012). Zinc: a free tool to discover chemistry for biology. *Journal of chemical information and modeling*, 52(7):1757–1768.

Janzing, D. and Schölkopf, B. (2010). Causal inference using the algorithmic markov condition. *IEEE Transactions on Information Theory*, 56(10):5168–5194.

Jin, W., Barzilay, R., and Jaakkola, T. (2018). Junction tree variational autoencoder for molecular graph generation. In *International conference on machine learning*, pages 2323–2332. PMLR.

Jones, D. R., Schonlau, M., and Welch, W. J. (1998). Efficient global optimization of expensive black-box functions. *Journal of Global optimization*, 13(4):455–492.

Jumper, J., Evans, R., Pritzel, A., Green, T., Figurnov, M., Ronneberger, O., Tunyasuvunakool, K., Bates, R., Žídek, A., Potapenko, A., et al. (2021). Highly accurate protein structure prediction with alphafold. *Nature*, pages 1–11.

Khemakhem, I., Kingma, D., Monti, R., and Hyvarinen, A. (2020a). Variational autoencoders and nonlinear ica: A unifying framework. In *International Conference on Artificial Intelligence and Statistics*, pages 2207–2217. PMLR.

Khemakhem, I., Monti, R. P., Kingma, D. P., and Hyvärinen, A. (2020b). Ice-beem: Identifiable conditional energy-based deep models based on nonlinear ica. *arXiv preprint arXiv:2002.11537*.

Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Kingma, D. P. and Welling, M. (2013). Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25:1097–1105.

Kusner, M. J., Paige, B., and Hernández-Lobato, J. M. (2017). Grammar variational autoencoder. In *International Conference on Machine Learning*, pages 1945–1954. PMLR.

LeCun, Y. (1998). The mnist database of handwritten digits. *http://yann.lecun.com/exdb/mnist/*.

LeCun, Y., Chopra, S., Hadsell, R., Ranzato, M., and Huang, F. (2006). A tutorial on energy-based learning. *Predicting structured data*, 1(0).

Lu, C., Wu, Y., Hernández-Lobato, J. M., and Schölkopf, B. (2021). Nonlinear invariant risk minimization: A causal approach.

Lu, Z., Pu, H., Wang, F., Hu, Z., and Wang, L. (2017). The expressive power of neural networks: A view from the width. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pages 6232–6240.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Mitrovic, J., McWilliams, B., Walker, J., Buesing, L., and Blundell, C. (2020). Representation learning via invariant causal mechanisms. *arXiv preprint arXiv:2010.07922*.

Muandet, K., Balduzzi, D., and Schölkopf, B. (2013). Domain generalization via invariant feature representation. In *International Conference on Machine Learning*, pages 10–18. PMLR.

Neal, R. M. (1996). *Bayesian Learning for Neural Networks*. PhD thesis, University of Toronto.

Negoescu, D. M., Frazier, P. I., and Powell, W. B. (2011). The knowledge-gradient algorithm for sequencing experiments in drug discovery. *INFORMS Journal on Computing*, 23(3):346–363.

Packwood, D. (2017). *Bayesian Optimization for Materials Science*. Springer.

Peters, J., Bühlmann, P., and Meinshausen, N. (2016). Causal inference by using invariant prediction: identification and confidence intervals. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, pages 947–1012.

Peters, J., Janzing, D., and Schölkopf, B. (2017). *Elements of causal inference: foundations and learning algorithms*. The MIT Press.

Rasmussen, C. and Williams, C. (2006). *Gaussian Processes for Machine Learning*. Adaptive Computation and Machine Learning. MIT Press, Cambridge, MA, USA.

Rezende, D. and Mohamed, S. (2015). Variational inference with normalizing flows. In *International Conference on Machine Learning*, pages 1530–1538. PMLR.

Rezende, D. J., Mohamed, S., and Wierstra, D. (2014). Stochastic backpropagation and approximate inference in deep generative models. In *International conference on machine learning*, pages 1278–1286. PMLR.

Rojas-Carulla, M., Schölkopf, B., Turner, R., and Peters, J. (2018). Invariant models for causal transfer learning. *The Journal of Machine Learning Research*, 19(1):1309–1342.

Schölkopf, B. (2019). Causality for machine learning. *arXiv preprint arXiv:1911.10500*.

Schölkopf, B., Janzing, D., Peters, J., Sgouritsa, E., Zhang, K., and Mooij, J. M. (2012). On causal and anticausal learning. In *ICML*.

Schölkopf, B., Locatello, F., Bauer, S., Ke, N. R., Kalchbrenner, N., Goyal, A., and Bengio, Y. (2021). Toward causal representation learning. *Proceedings of the IEEE*, 109(5):612–634.

Snoek, J., Larochelle, H., and Adams, R. P. (2012). Practical bayesian optimization of machine learning algorithms. *Advances in neural information processing systems*, 25.

Song, Y. and Kingma, D. P. (2021). How to train your energy-based models.

Spearman, C. (1904). The proof and measurement of association between two things. *American Journal of Psychology*.

Sriperumbudur, B., Fukumizu, K., Gretton, A., Hyvärinen, A., and Kumar, R. (2017). Density estimation in infinite dimensional exponential families. *Journal of Machine Learning Research*, 18.

Titsias, M. (2009). Variational learning of inducing variables in sparse gaussian processes. In *Artificial intelligence and statistics*, pages 567–574. PMLR.

Tripp, A., Daxberger, E., and Hernández-Lobato, J. M. (2020). Sample-efficient optimization in the latent space of deep generative models via weighted retraining. *Advances in Neural Information Processing Systems*, 33.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Weininger, D. (1988). Smiles, a chemical language and information system. 1. introduction to methodology and encoding rules. *Journal of chemical information and computer sciences*, 28(1):31–36.

Weininger, D. (1990). Smiles. 3. depict. graphical depiction of chemical structures. *Journal of chemical information and computer sciences*, 30(3):237–243.

Weininger, D., Weininger, A., and Weininger, J. L. (1989). Smiles. 2. algorithm for generation of unique smiles notation. *Journal of chemical information and computer sciences*, 29(2):97–101.

Willetts, M. and Paige, B. (2021). I don't need **u**: Identifiable non-linear ica without side information. *arXiv preprint arXiv:2106.05238*.

You, J., Liu, B., Ying, R., Pande, V., and Leskovec, J. (2018). Graph convolutional policy network for goal-directed molecular graph generation. *arXiv preprint arXiv:1806.02473*.

Zhang, K. and Hyvarinen, A. (2012). On the identifiability of the post-nonlinear causal model. *arXiv preprint arXiv:1205.2599*.

Zhang, K., Peters, J., Janzing, D., and Schölkopf, B. (2012). Kernel-based conditional independence test and application in causal discovery. *arXiv preprint arXiv:1202.3775*.

Zhou, Z., Kearnes, S., Li, L., Zare, R. N., and Riley, P. (2019). Optimization of molecules via deep reinforcement learning. *Scientific reports*, 9(1):1–10.

Appendix A

Strongly Exponential Family

Definition 7 (Strongly exponential)**.** *A multivariate exponential family distribution*

$$p(\mathbf{z}) = \frac{Q(\mathbf{z})}{Z(\theta)} \exp(\langle \mathbf{T}(\mathbf{z}), \theta \rangle) \tag{A.1}$$

is strongly exponential, if

$$(\exists\, \theta \in \mathbb{R}^k \ s.t.\ \langle \mathbf{T}(\mathbf{z}), \theta \rangle = const,\ \forall \mathbf{z} \in \mathscr{Z}) \implies (l(\mathscr{Z}) = 0 \ or \ \theta = \mathbf{0}), \quad \forall\, \mathscr{Z} \subset \mathbb{R}^n, \tag{A.2}$$

where l is the Lebesgue measure.

Essentially, the density of a strongly exponential family distribution almost surely has the exponential component $\exp(\langle \mathbf{T}(\mathbf{z}), \theta \rangle)$ and can only be reduced to the base measure $Q(\mathbf{z})$ on a set of measure zero.

Appendix B

Proofs

B.1 Proof of Theorem 4

Proof. Define $\mathrm{vol}(B) = \sqrt{\det(B^T B)}$ for any full rank matrix B. Suppose that we have two sets of parameters $\theta = (\mathbf{f}, \mathbf{T}, \lambda)$ and $\tilde{\theta} = (\tilde{\mathbf{f}}, \tilde{\mathbf{T}}, \tilde{\lambda})$ such that $p_\theta(\mathbf{x}\,|\,\mathbf{u}) = p_{\tilde{\theta}}(\mathbf{x}\,|\,\mathbf{u})$, $\forall(\mathbf{x},\mathbf{u}) \in \mathscr{X} \times \mathscr{U}$. We want to show $\theta \sim_A \tilde{\theta}$. The proof consists of three steps, the first two of which are similar to those in the proof of Theorem 1 in Khemakhem et al. (2020a). The last step is original and contributes to the proof of Theorem 4 in Lu et al. (2021).

Step I. In this step, we transform the equality of the marginal distributions over observed data into the equality of noise-free distributions. For all pairs $(\mathbf{x},\mathbf{u}) \in \mathscr{X} \times \mathscr{U}$, we have

$$p_\theta(\mathbf{x}\,|\,\mathbf{u}) = p_{\tilde{\theta}}(\mathbf{x}\,|\,\mathbf{u}) \tag{B.1}$$

$$\implies \int_{\mathscr{Z}} p_{\mathbf{f}}(\mathbf{x}\,|\,\mathbf{z}) p_{\mathbf{T},\lambda}(\mathbf{z}\,|\,\mathbf{u}) d\,\mathbf{z} = \int_{\mathscr{Z}} p_{\tilde{\mathbf{f}}}(\mathbf{x}\,|\,\mathbf{z}) p_{\tilde{\mathbf{T}},\tilde{\lambda}}(\mathbf{z}\,|\,\mathbf{u}) d\,\mathbf{z} \tag{B.2}$$

$$\implies \int_{\mathscr{Z}} p_\varepsilon(\mathbf{x}-\mathbf{f}(\mathbf{z})) p_{\mathbf{T},\lambda}(\mathbf{z}\,|\,\mathbf{u}) d\,\mathbf{z} = \int_{\mathscr{Z}} p_\varepsilon(\mathbf{x}-\tilde{\mathbf{f}}(\mathbf{z})) p_{\tilde{\mathbf{T}},\tilde{\lambda}}(\mathbf{z}\,|\,\mathbf{u}) d\,\mathbf{z} \tag{B.3}$$

$$\implies \int_{\mathscr{X}} p_\varepsilon(\mathbf{x}-\bar{\mathbf{x}}) p_{\mathbf{T},\lambda}(\mathbf{f}^{-1}(\bar{\mathbf{x}})\,|\,\mathbf{u}) \,\mathrm{vol}(J_{\mathbf{f}^{-1}}(\bar{\mathbf{x}})) d\,\bar{\mathbf{x}} = \int_{\mathscr{X}} p_\varepsilon(\mathbf{x}-\bar{\mathbf{x}}) p_{\tilde{\mathbf{T}},\tilde{\lambda}}(\tilde{\mathbf{f}}^{-1}(\bar{\mathbf{x}})\,|\,\mathbf{u}) \,\mathrm{vol}(J_{\tilde{\mathbf{f}}^{-1}}(\bar{\mathbf{x}})) d\,\bar{\mathbf{x}} \tag{B.4}$$

$$\implies \int_{\mathbb{R}^d} p_\varepsilon(\mathbf{x}-\bar{\mathbf{x}}) \tilde{p}_{\mathbf{f},\mathbf{T},\lambda,\mathbf{u}}(\bar{\mathbf{x}}) d\,\bar{\mathbf{x}} = \int_{\mathbb{R}^d} p_\varepsilon(\mathbf{x}-\bar{\mathbf{x}}) \tilde{p}_{\tilde{\mathbf{f}},\tilde{\mathbf{T}},\tilde{\lambda},\mathbf{u}}(\bar{\mathbf{x}}) d\,\bar{\mathbf{x}} \tag{B.5}$$

$$\implies (\tilde{p}_{\mathbf{f},\mathbf{T},\lambda,\mathbf{u}} * p_\varepsilon)(\mathbf{x}) = (\tilde{p}_{\tilde{\mathbf{f}},\tilde{\mathbf{T}},\tilde{\lambda},\mathbf{u}} * p_\varepsilon)(\mathbf{x}) \tag{B.6}$$

$$\implies F[\tilde{p}_{\mathbf{f},\mathbf{T},\lambda,\mathbf{u}}](\omega)\varphi_\varepsilon(\omega) = F[\tilde{p}_{\tilde{\mathbf{f}},\tilde{\mathbf{T}},\tilde{\lambda},\mathbf{u}}](\omega)\varphi_\varepsilon(\omega) \tag{B.7}$$

$$\implies F[\tilde{p}_{\mathbf{f},\mathbf{T},\lambda,\mathbf{u}}](\omega) = F[\tilde{p}_{\tilde{\mathbf{f}},\tilde{\mathbf{T}},\tilde{\lambda},\mathbf{u}}](\omega) \tag{B.8}$$

$$\implies \tilde{p}_{\mathbf{f},\mathbf{T},\lambda,\mathbf{u}}(\mathbf{x}) = \tilde{p}_{\tilde{\mathbf{f}},\tilde{\mathbf{T}},\tilde{\lambda},\mathbf{u}}(\mathbf{x}) \tag{B.9}$$

$$\implies p_{\mathbf{T},\lambda}(\mathbf{f}^{-1}(\mathbf{x})\,|\,\mathbf{u}) \,\mathrm{vol}(J_{\mathbf{f}^{-1}}(\mathbf{x})) = p_{\tilde{\mathbf{T}},\tilde{\lambda}}(\tilde{\mathbf{f}}^{-1}(\mathbf{x})\,|\,\mathbf{u}) \,\mathrm{vol}(J_{\tilde{\mathbf{f}}^{-1}}(\mathbf{x})), \tag{B.10}$$

where

- in Equation (B.4), we made changes of variables $\bar{\mathbf{x}} = \mathbf{f}(\mathbf{z})$ on the LHS and $\bar{\mathbf{x}} = \tilde{\mathbf{f}}(\mathbf{z})$ on the RHS and denoted the Jacobian by J;
- in Equation (B.5), we defined $\tilde{p}_{\mathbf{f},\mathbf{T},\lambda,\mathbf{u}}(\mathbf{x}) \triangleq p_{\mathbf{T},\lambda}(\mathbf{f}^{-1}(\mathbf{x})|\,\mathbf{u})\,\mathrm{vol}(J_{\mathbf{f}^{-1}}(\mathbf{x}))\,\mathbb{I}_{\mathscr{X}}(\mathbf{x})$ on the LHS and similarly on the RHS;
- in Equation (B.6), we denoted the convolution operator by $*$;
- in Equation (B.7), we applied Fourier transform F in both sides and used the definition of the characteristic function that $\varphi_\varepsilon(\omega) = F[p_\varepsilon](\omega)$;
- in Equation (B.8), we used assumption (i) that $\varphi_\varepsilon(\omega)$ is non-zero almost everywhere.

Step II. In this step, we remove terms that are functions of $\mathbf{x}$ only. Taking logarithm on both sides of Equation (B.10), we have

$$\begin{aligned}&\log \mathrm{vol}(J_{\mathbf{f}^{-1}}(\mathbf{x})) + \log Q(\mathbf{f}^{-1}(\mathbf{x})) - \log Z(\mathbf{u}) + \left\langle \mathbf{T}(\mathbf{f}^{-1}(\mathbf{x})), \lambda(\mathbf{u}) \right\rangle \\ &\quad = \log \mathrm{vol}(J_{\tilde{\mathbf{f}}^{-1}}(\mathbf{x})) + \log \tilde{Q}(\tilde{\mathbf{f}}^{-1}(\mathbf{x})) - \log \tilde{Z}(\mathbf{u}) + \left\langle \tilde{\mathbf{T}}(\tilde{\mathbf{f}}^{-1}(\mathbf{x})), \tilde{\lambda}(\mathbf{u}) \right\rangle.\end{aligned} \tag{B.11}$$

Let $\mathbf{u}_0, \mathbf{u}_1, \cdots, \mathbf{u}_k \in \mathscr{U}$ be the $k+1$ points defined in assumption (iii). For each $l = 1, \cdots, k$, we evaluate Equation (B.11) at these points to obtain $k+1$ equations, and subtract the first equation from the remaining k equations to obtain:

$$\left\langle \mathbf{T}(\mathbf{f}^{-1}(\mathbf{x})), \lambda(\mathbf{u}_l) - \lambda(\mathbf{u}_0) \right\rangle + \log \frac{Z(\mathbf{u}_0)}{Z(\mathbf{u}_l)} = \left\langle \tilde{\mathbf{T}}(\tilde{\mathbf{f}}^{-1}(\mathbf{x})), \tilde{\lambda}(\mathbf{u}_l) - \tilde{\lambda}(\mathbf{u}_0) \right\rangle + \log \frac{\tilde{Z}(\mathbf{u}_0)}{\tilde{Z}(\mathbf{u}_l)}. \tag{B.12}$$

Let L be defined as in assumption (iii) and $\tilde{L}$ defined similarly for $\tilde{\lambda}$. Note that L is invertible by assumption, but $\tilde{L}$ is not necessarily invertible. Letting $\mathbf{b} \in \mathbb{R}^k$ in which $b_l = \log \frac{\tilde{Z}(\mathbf{u}_0) Z(\mathbf{u}_l)}{\tilde{Z}(\mathbf{u}_l) Z(\mathbf{u}_0)}$, we have

$$L^T \mathbf{T}(\mathbf{f}^{-1}(\mathbf{x})) = \tilde{L}^T \tilde{\mathbf{T}}(\tilde{\mathbf{f}}^{-1}(\mathbf{x})) + \mathbf{b}. \tag{B.13}$$

Left multiplying both sides of Equation (B.13) by L^{-T} gives

$$\mathbf{T}(\mathbf{f}^{-1}(\mathbf{x})) = A \tilde{\mathbf{T}}(\tilde{\mathbf{f}}^{-1}(\mathbf{x})) + \mathbf{c}, \tag{B.14}$$

where $A = L^{-T} \tilde{L} \in \mathbb{R}^{k \times k}$ and $\mathbf{c} = L^{-T} \mathbf{b} \in \mathbb{R}^k$.

Step III. To complete the proof, we need to show that A is invertible. Let $\mathbf{z}_l \in \mathscr{Z},\ \mathbf{x}_l = \mathbf{f}(\mathbf{z}_l),\ l = 0, \cdots, k$. We evaluate Equation (B.14) at these $k+1$ points to obtain $k+1$ equations and subtract the first equation from the remaining k equations to obtain

$$\underbrace{[\mathbf{T}(\mathbf{z}_1) - \mathbf{T}(\mathbf{z}_0), \cdots, \mathbf{T}(\mathbf{z}_k) - \mathbf{T}(\mathbf{z}_0)]}_{\triangleq R \in \mathbb{R}^{k\times k}} = A \underbrace{[\tilde{\mathbf{T}}(\tilde{\mathbf{f}}^{-1}(\mathbf{x}_1)) - \tilde{\mathbf{T}}(\tilde{\mathbf{f}}^{-1}(\mathbf{x}_0)), \cdots, \tilde{\mathbf{T}}(\tilde{\mathbf{f}}^{-1}(\mathbf{x}_k)) - \tilde{\mathbf{T}}(\tilde{\mathbf{f}}^{-1}(\mathbf{x}_0))]}_{\triangleq \tilde{R} \in \mathbb{R}^{k\times k}}. \tag{B.15}$$

We need to show that for a given $\mathbf{z}_0 \in \mathscr{Z}$, there exist k points $\mathbf{z}_1, \cdots, \mathbf{z}_k \in \mathscr{Z}$ such that the columns of R are linearly independent. Suppose, for contradiction, that the columns of R would never be linearly independent for any choice of $\mathbf{z}_1, \cdots, \mathbf{z}_k \in \mathscr{Z}$. Then the function $\mathbf{g}(\mathbf{z}) \triangleq \mathbf{T}(\mathbf{z}) - \mathbf{T}(\mathbf{z}_0)$ would live in a $k-1$ or lower dimensional subspace, and thus we could find a non-zero vector $\lambda \in \mathbb{R}^k$ orthogonal to that subspace. This would imply that $\langle \mathbf{T}(\mathbf{z}) - \mathbf{T}(\mathbf{z}_0), \lambda \rangle = 0$ and thus $\langle \mathbf{T}(\mathbf{z}), \lambda \rangle = \langle \mathbf{T}(\mathbf{z}_0), \lambda \rangle = const,\ \forall \mathbf{z} \in \mathscr{Z}$, which contradicts the assumption that the prior is strongly exponential. Therefore, we have shown that there exist $k+1$ points $\mathbf{z}_0, \mathbf{z}_1, \cdots, \mathbf{z}_k \in \mathscr{Z}$ such that R is invertible. Since $R = A\tilde{R}$ and A is not a function of $\mathbf{z}$, A must be invertible. This completes the proof. Note that in this step we did not work with the Jacobian of the sufficient statistics, so we do not need assumption (iii) in Theorem 1 in Khemakhem et al. (2020a). □

B.2 Proof of Theorem 5

Proof. Let $\mathbf{v} = \tilde{\mathbf{f}}^{-1} \circ \mathbf{f} : \mathscr{Z} \to \mathscr{Z}$. Since all assumptions in Theorem 4 hold, we have

$$\mathbf{T}(\mathbf{z}) = A\tilde{\mathbf{T}}(\mathbf{v}(\mathbf{z})) + \mathbf{c}, \tag{B.16}$$

where $A \in \mathbb{R}^{k\times k}$ is invertible. We want to show that A is a block permutation matrix. The proof consists of two steps, both of which are original and contribute to the proof of Theorem 5 in Lu et al. (2021).

Step I. In this step, we show that $\mathbf{v}$ is a pointwise function. We first differentiate both sides of Equation (B.16) with respect to z_s and z_t $(s \neq t)$ to obtain

$$\frac{\partial \mathbf{T}(\mathbf{z})}{\partial z_s} = A \sum_{i=1}^{n} \frac{\partial \tilde{\mathbf{T}}(\mathbf{v}(\mathbf{z}))}{\partial v_i(\mathbf{z})} \cdot \frac{\partial v_i(\mathbf{z})}{\partial z_s} \tag{B.17}$$

$$\frac{\partial^2 \mathbf{T}(\mathbf{z})}{\partial z_s \partial z_t} = A \sum_{i=1}^{n} \sum_{j=1}^{n} \frac{\partial^2 \tilde{\mathbf{T}}(\mathbf{v}(\mathbf{z}))}{\partial v_i(\mathbf{z}) \partial v_j(\mathbf{z})} \cdot \frac{\partial v_j(\mathbf{z})}{\partial z_t} \cdot \frac{\partial v_i(\mathbf{z})}{\partial z_s} + A \sum_{i=1}^{n} \frac{\partial \tilde{\mathbf{T}}(\mathbf{v}(\mathbf{z}))}{\partial v_i(\mathbf{z})} \cdot \frac{\partial^2 v_i(\mathbf{z})}{\partial z_s \partial z_t}. \tag{B.18}$$

By construction, the second-order cross derivatives of $\mathbf{T}$ and $\tilde{\mathbf{T}}$ are all zero. Therefore, we have

$$\mathbf{0} = A \sum_{i=1}^{n} \frac{\partial^2 \tilde{\mathbf{T}}(\mathbf{v}(\mathbf{z}))}{\partial v_i(\mathbf{z})^2} \cdot \frac{\partial v_i(\mathbf{z})}{\partial z_t} \cdot \frac{\partial v_i(\mathbf{z})}{\partial z_s} + A \sum_{i=1}^{n} \frac{\partial \tilde{\mathbf{T}}(\mathbf{v}(\mathbf{z}))}{\partial v_i(\mathbf{z})} \cdot \frac{\partial^2 v_i(\mathbf{z})}{\partial z_s \partial z_t}. \tag{B.19}$$

Equation (B.19) can be written in the following matrix-vector form:

$$\mathbf{0} = A\, \tilde{\mathbf{T}}''(\mathbf{z})\, \mathbf{v}'_{s,t}(\mathbf{z}) + A\, \tilde{\mathbf{T}}'(\mathbf{z})\, \mathbf{v}''_{s,t}(\mathbf{z}), \tag{B.20}$$

where

$$\tilde{\mathbf{T}}''(\mathbf{z}) \coloneqq \left[\frac{\partial^2 \tilde{\mathbf{T}}(\mathbf{v}(\mathbf{z}))}{\partial v_1(\mathbf{z})^2}, \cdots, \frac{\partial^2 \tilde{\mathbf{T}}(\mathbf{v}(\mathbf{z}))}{\partial v_n(\mathbf{z})^2} \right] \in \mathbb{R}^{k \times n} \tag{B.21}$$

$$\mathbf{v}'_{s,t}(\mathbf{z}) \coloneqq \left[\frac{\partial v_1(\mathbf{z})}{\partial z_t} \cdot \frac{\partial v_1(\mathbf{z})}{\partial z_s}, \cdots, \frac{\partial v_n(\mathbf{z})}{\partial z_t} \cdot \frac{\partial v_n(\mathbf{z})}{\partial z_s} \right]^T \in \mathbb{R}^n, \tag{B.22}$$

and

$$\tilde{\mathbf{T}}'(\mathbf{z}) \coloneqq \left[\frac{\partial \tilde{\mathbf{T}}(\mathbf{v}(\mathbf{z}))}{\partial v_1(\mathbf{z})}, \cdots, \frac{\partial \tilde{\mathbf{T}}(\mathbf{v}(\mathbf{z}))}{\partial v_n(\mathbf{z})} \right] \in \mathbb{R}^{k \times n} \tag{B.23}$$

$$\mathbf{v}''_{s,t}(\mathbf{z}) \coloneqq \left[\frac{\partial^2 v_1(\mathbf{z})}{\partial z_s \partial z_t}, \cdots, \frac{\partial^2 v_n(\mathbf{z})}{\partial z_s \partial z_t} \right]^T \in \mathbb{R}^n. \tag{B.24}$$

Now, by concatenating

$$\tilde{\mathbf{T}}'''(\mathbf{z}) \coloneqq [\tilde{\mathbf{T}}''(\mathbf{z}), \tilde{\mathbf{T}}'(\mathbf{z})] \in \mathbb{R}^{k \times 2n} \tag{B.25}$$

$$\mathbf{v}'''_{s,t}(\mathbf{z}) \coloneqq [\mathbf{v}'_{s,t}(\mathbf{z})^T, \mathbf{v}''_{s,t}(\mathbf{z})^T]^T \in \mathbb{R}^{2n}, \tag{B.26}$$

we further obtain

$$\mathbf{0} = A\, \tilde{\mathbf{T}}'''(\mathbf{z})\, \mathbf{v}'''_{s,t}(\mathbf{z}). \tag{B.27}$$

Finally, we take the rows of $\tilde{\mathbf{T}}'''(\mathbf{z})$ that corresponds to the factorized strongly exponential family distribution part and denote them by $\tilde{\mathbf{T}}'''_f(\mathbf{z}) \in \mathbb{R}^{k' \times 2n}$. By Lemma 5 in Khemakhem et al. (2020a) and the assumption that $k' \geq 2n$, we have that the rank of $\tilde{\mathbf{T}}'''_f(\mathbf{z})$ is $2n$. Since $k \geq k' \geq 2n$, the rank of $\tilde{\mathbf{T}}'''(\mathbf{z})$ is also $2n$. Since the rank of A is k, the rank of $A\, \tilde{\mathbf{T}}'''(\mathbf{z}) \in \mathbb{R}^{k \times 2n}$ is $2n$. This implies that $\mathbf{v}'''_{s,t}(\mathbf{z})$ must be a zero vector. In particular, we have that $\mathbf{v}'_{s,t}(\mathbf{z}) = \mathbf{0}$, $\forall s \neq t$. Therefore, we have shown that $\mathbf{v}$ is a pointwise function.

Step II. To complete the proof, we need to show that A is a block permutation matrix. Without loss of generality, we assume that the permutation in $\mathbf{v}$ is the identity. That is, $\mathbf{v}(\mathbf{z}) = [v_1(z_1), \cdots, v_n(z_n)]^T$ for some nonlinear univariate scalar functions $v_1, \cdots, v_n$. Since $\mathbf{f}$ and $\tilde{\mathbf{f}}$ are bijective, we have that $\mathbf{v}$ is also bijective and $\mathbf{v}^{-1}(\mathbf{z}) = [v_1^{-1}(z_1), \cdots, v_n^{-1}(z_n)]^T$. We denote $\bar{\mathbf{T}}(\mathbf{v}(\mathbf{z})) = \tilde{\mathbf{T}}(\mathbf{v}(\mathbf{z})) + A^{-1}\mathbf{c}$ and plug it into Equation (B.16) to obtain $\mathbf{T}(\mathbf{z}) = A\bar{\mathbf{T}}(\mathbf{v}(\mathbf{z}))$. Applying $\mathbf{v}^{-1}$ to the variables $\mathbf{z}$ at both sides gives

$$\mathbf{T}(\mathbf{v}^{-1}(\mathbf{z})) = A\bar{\mathbf{T}}(\mathbf{z}). \tag{B.28}$$

Let t be the index of an entry in the sufficient statistics $\mathbf{T}$ that corresponds to the the factorized strongly exponential family distribution part $\mathbf{T}_f$. For all $s \neq t$, we have

$$0 = \frac{\partial \mathbf{T}(\mathbf{v}^{-1}(\mathbf{z}))_t}{\partial z_s} = \sum_{j=1}^{k} a_{tj} \frac{\partial \bar{\mathbf{T}}(\mathbf{z})_j}{\partial z_s}. \tag{B.29}$$

Since the entries of $\tilde{\mathbf{T}}$ are linearly independent (if they were not linearly independent, then $\tilde{\mathbf{T}}$ can be compressed into a smaller vector by removing the redundant entries), we have that a_{tj} is zero for any j such that $\frac{\partial \bar{\mathbf{T}}(\mathbf{z})_j}{\partial z_s} \neq 0$. This includes the entries j in the sufficient statistics $\tilde{\mathbf{T}}$ that correspond to 1) the factorized strongly exponential family distribution part which does not depend on z_t; and 2) the neural network part.

Therefore, when t is the index of an entry in the sufficient statistics $\mathbf{T}$ that corresponds to factor i in the factorized strongly exponential family distribution part $\mathbf{T}_f$, the only non-zero a_{tj} are the ones that map between $\mathbf{T}_{f_i}(z_i)$ and $\bar{\mathbf{T}}_{f_i}(v_i(z_i))$, where $\mathbf{T}_{f_i}$ are the factors in $\mathbf{T}_f$ that only depends on z_i and $\bar{\mathbf{T}}_{f_i}$ is defined similarly. Therefore, we can construct an invertible submatrix A_i' with all non-zero elements a_{tj} for all t that corresponds to factor i, such that

$$\mathbf{T}_{f_i}(z_i) = A_i'\bar{\mathbf{T}}_{f_i}(v_i(z_i)) = A_i'\tilde{\mathbf{T}}_{f_i}(v_i(z_i)) + \mathbf{c}_i, \quad i = 1, \cdots, n, \tag{B.30}$$

where $\tilde{\mathbf{T}}_{f_i}$ are the factors in $\tilde{\mathbf{T}}_f$ that only depends on z_i, and $\mathbf{c}_i$ are the corresponding elements of $\mathbf{c}$. This means that the matrix A is a block permutation matrix. For each $i = 1, \cdots, n$, the block A_i' of A affinely transforms $\mathbf{T}_{f_i}(z_i)$ into $\tilde{\mathbf{T}}_{f_i}(v_i(z_i))$. There is also an additional block A_{NN}' which affinely transforms $\mathbf{T}_{NN}(\mathbf{z})$ into $\tilde{\mathbf{T}}_{NN}(\mathbf{v}(\mathbf{z}))$. This completes the proof. □

B.3 Proof of Theorem 6

Proof. This proof builds upon the proof of Theorem 4 in Khemakhem et al. (2020a).

If we maximize the joint training objective (3.12) with respect to ϕ, by Assumption (i) and the properties of variation lower bound and score matching, we will eventually obtain

$$q_\phi(\mathbf{z}|\mathbf{x},\mathbf{u}) = p_\theta(\mathbf{z}|\mathbf{x},\mathbf{u}). \tag{B.31}$$

Hence, the joint training objective will eventually be equivalent to the expected log marginal likelihood up to a constant term. By Assumption (iii), since the identifiability is guaranteed up to the equivalence class defined in Definition 3, the consistency of maximum likelihood estimation means that we will converge to this equivalence class of the true paramteres θ^* in the limit of infinite data if we maximize the objective with respect to θ. Hence, the true data representation $\mathbf{z}_*$ can be recovered up to simple transformations defined in Equation (3.6) in the limit of infinite data. This completes the proof. □