

Advancing Causal Representation Learning: Enhancing Robustness and Transferability in Real-World Applications

by

Shayan Shirahmad Gale Bagi

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Doctor of Philosophy
in
Electrical and Computer Engineering

Waterloo, Ontario, Canada, 2025

© Shayan Shirahmad Gale Bagi 2025

Examining Committee Membership

The following served on the Examining Committee for this thesis. The decision of the Examining Committee is by majority vote.

External Examiner: Graham Taylor
 Professor and Canada Research Chair,
 Machine Learning School of Engineering, University of Guelph

Supervisor(s): Mark Crowley
 Associate Professor, Dept. of ECE, University of Waterloo
 Krzysztof Czarnecki
 Professor, Dept. of ECE, University of Waterloo

Internal Member: Yash Vardhan Pant
 Assistant Professor , Dept. of ECE, University of Waterloo

Internal-External Member: Paul Fieguth
 Professor, Associate Vice President - Academic Operations,
 Dept. of Systems Design Engineering, University of Waterloo

Other Member(s): Alexander Wong
 Professor, Dept. of Systems Design Engineering, University of Waterloo

Author's Declaration

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Abstract

Conventional supervised learning methods heavily depend on statistical inference, often assuming that data is identically and independently distributed (i.i.d). However, this assumption rarely holds in real-world scenarios, where environments or domains frequently shift, posing significant challenges to model robustness and generalization. Moreover, statistical models are typically treated as black boxes, with their learned representations remaining opaque and challenging to interpret. My research addresses these issues through a causal learning perspective, aiming to enhance the interpretability and adaptability of machine learning models in dynamic and uncertain environments.

I have developed innovative methods for learning causal models that are applicable to a wide range of machine learning tasks, including transfer learning, out-of-distribution generalization, reinforcement learning, and action classification. The first method introduces a generative model tailored to learn causal variables in scenarios where the causal graph is known, such as Human Trajectory Prediction. By incorporating domain knowledge, this approach models the underlying causal mechanisms, leading to improved performance on both synthetic and real-world datasets. The results demonstrate that this generative model outperforms traditional statistical models, particularly in out-of-distribution contexts.

The second method targets the more challenging scenario where the causal structure is unknown. I have explored various conditions and assumptions that facilitate the discovery of causal relationships without prior knowledge of the causal graph. This method combines advanced techniques in causal inference and machine learning to uncover the underlying causal graph and variables from observed data. Evaluations on both real-world and synthetic datasets show that this method not only surpasses existing approaches in causal representation learning but also brings AI systems closer to practical, real-world applications by enhancing reliability and interpretability.

Overall, my research contributes significant advancements to the field of causal learning, providing novel solutions that improve model interpretability and robustness. These methods lay a strong foundation for developing AI systems capable of adapting to diverse and evolving real-world conditions, thereby broadening the scope and impact of machine learning across various domains.

Acknowledgements

I would like to express my deepest gratitude to all my colleagues and supervisors at the University of Waterloo and Simon Fraser University. Special thanks to Prof. Mark Crowley, Dr. Zahra Gharaee, Prof. Oliver Schulte, and Prof. Krzysztof Czarnecki for their invaluable guidance and support throughout my PhD journey.

Table of Contents

Examining Committee	ii
Author's Declaration	iii
Abstract	iv
Acknowledgements	v
List of Figures	xi
List of Tables	xiv
List of Abbreviations	xvi
Notation Convention	xix
1 Introduction	1
1.1 Motivation for Causality	1
1.2 Causal Models	3
1.2.1 Graphical Causal Models	3
1.2.2 Structural Causal Models	5
1.2.3 Inputs and Outputs of Causal Models	6
1.2.4 Causal Markov Condition	6

1.2.5	Independent Causal Mechanisms Principle	7
1.2.6	Confounding Bias	8
1.3	Interventions and Counterfactuals	8
1.3.1	Interventions	8
1.3.2	Counterfactuals	10
1.4	Challenges	12
1.4.1	Identifiability	12
1.4.2	Scalability	13
1.5	Downstream Tasks	13
1.5.1	Human Trajectory Prediction	13
1.5.2	Action and Object Recognition	16
1.6	Research Overview and Thesis Structure	17
1.7	Contributions	18
2	Background and Related Work	20
2.1	Causal Domain Adaptation	20
2.1.1	Domain Adaptation by Causal Mechanism Transfer	20
2.1.2	Unsupervised Domain Adaptation by Augmented Directed Acyclic Graphs	21
2.1.3	Markov Blanket Features	22
2.1.4	Domain Generalization by Markov Blanket Features	23
2.1.5	Independent Causal Mechanisms principle in Domain Adaptation	24
2.1.6	Backdoor Adjustment for Confounding Factors	26
2.1.7	Domain Adaptation by Selection Diagrams	28
2.1.8	GCRL for Causal Domain Adaptation	31
2.2	Human Trajectory Prediction	32
2.2.1	Causality for Transfer Learning in Human Trajectory Prediction	32
2.2.2	Counterfactual Analysis in Human Trajectory Prediction	34

2.2.3	Proposed Generative Causal Representation Learning for Human Trajectory Prediction	35
2.3	Causal Representation Learning	35
2.3.1	Sparse Mechanism Shift Hypothesis in Causal Discovery	35
2.3.2	Explicit Models and Acyclicity Constraint	37
2.3.3	Causal Representation Learning by Interventional Data	38
2.3.4	Sparsity as an Inductive Bias	39
2.3.5	Supervised Causal Representation Learning	41
2.3.6	Implicit Latent Causal Models	42
2.3.7	Causal Representation Learning in Time-Series Data	43
2.4	Proposed Implicit Causal Representation Learning via Switchable Mechanisms in Causal Representation Learning	45
3	Generative Causal Representation Learning	46
3.1	Proposed Method	47
3.1.1	Transportability	47
3.1.2	Formalism of Motion Forecasting	48
3.1.3	Causal Formalism	49
3.1.4	Generative Causal Representation Learning	50
3.1.5	Domain Adaptation	56
3.2	Results and Discussion	59
3.2.1	Datasets	59
3.2.2	Metrics	59
3.2.3	Robustness	60
3.2.4	Multi-modality	60
3.2.5	Domain Generalization	61
3.2.6	Domain Adaptation	62
3.2.7	Identifiability	63
3.2.8	Ablation Studies	64
3.3	Summary	66

4	Implicit Causal Representation Learning via Switchable Mechanisms	67
4.1	Implicit and Explicit Latent Causal Models	68
4.2	Hard Interventions vs Soft Interventions	69
4.3	Data Generating Process	71
4.4	Causal Mechanisms Switch Variable	73
4.4.1	Augmented Implicit Causal Model	74
4.4.2	Observability of the Switch Variable	75
4.5	Identifiability Theorem for Augmented Implicit Causal Models with Soft Interventions	77
4.6	Training Objective	86
4.7	Solution Functions	87
4.8	Results and Discussion	87
4.8.1	Datasets	89
4.8.2	Metrics	90
4.8.3	Causal Model Recovery	91
4.8.4	Factors Affecting Causal Disentanglement	92
4.8.5	Action Inference	93
4.8.6	Ablation Studies	95
4.8.7	Scalability	96
4.9	Summary	96
5	Extensions and Future Works	103
5.1	Future Works of Generative Causal Representation Learning (GCRL)	103
5.2	Future Works of Implicit Causal Representation Learning via Switchable Mechanisms (ICRL-SM)	104
6	Conclusion	107
	References	109

APPENDICES	119
A Convergence Plots	120
Glossary	122

List of Figures

1.1	Comparison between statistical (left) and causal models (middle). X represents the observed data, with P_X denoting its distribution. The observational distribution corresponds to the scenario where no interventions have occurred. While statistical models are restricted to a single Identically and Independently Distributed (IID) distribution, causal models can capture multiple distributions by allowing interventions on causal variables [57]. . . .	3
1.2	Chain configuration in graphs.	4
1.3	Fork configuration in graphs.	4
1.4	Collider configuration in graphs.	5
1.5	Comparison of causal graphs for motion forecasting.	9
1.6	Pre and Post-intervention samples from the dataset in [41].	16
1.7	Hypothesized causal graph in action and object recognition task inspired by models used in [41]. z_s are causal variables corresponding to scene properties, c are the potential confounding factors such as camera angle, a are the actions/interventions, the boxes o_i are the set of causal variables z_i^j for object o_i , x are the pre-intervention observations, $\tilde{x}$ are the post-intervention observations, and $\tilde{z}_a$ are the post-intervention causal variables.	17
2.1	Causal graph in context adjustment [69]	27
2.2	Acyclic directed mixed graph in [46]. The orange colored nodes indicate the context variables.	29
2.3	Causal graph for human trajectories in [10].	34

3.1	Proposed causal model for motion forecasting. X , Y , Z , and S represent the past trajectories, future trajectories, invariant features, and variant features, respectively. E is the environment variable which models the changing factors in each environment. The filled circles correspond to the observed variables and the rest are the latent variables.	49
3.2	Overview of GCRL. The encoder, interaction module, and decoder are adopted from the baseline Human Trajectory Prediction (HTP) model [23]. The components within the dotted box are fine-tuned for adaptation to a new domain.	57
3.3	The range of distributions of variant and invariant representations priors. E_i represents environment i	57
3.4	Comparison of ADE/FDE for different values of N	63
3.5	Trajectories of pedestrians in different scenes. The circled trajectories are observed, the dashed trajectories are predicted, and the starred trajectories are groundtruth. In all of the scenes, 100 trajectories are predicted for each pedestrian.	63
3.6	Domain generalization with different style domain shifts. The mean and standard deviation of ADEs are calculated and plotted for 5 seeds.	64
3.7	Domain Adaptation with different number of batches	65
4.1	Difference between hard and soft interventions: As illustrated in the middle row, hard interventions sever the connections with parent variables. This means that, for example, when we intervene on an object’s color, the object’s class can no longer influence its color. In contrast, soft interventions, as depicted in the bottom row, allow these connections to remain, permitting the object’s class to still affect its color even after the intervention.	70
4.2	In the Causal-Triplet dataset [41], visual representations illustrate both pre- and post-intervention scenarios. The first two rows display data samples from <i>Epic-Kitchens</i> dataset, while the third and fourth rows feature samples from <i>ProcTHOR</i> dataset. Each image in the post-intervention condition is labeled with the corresponding action and intervened object. In the images from the first two rows, the agent is performing an action on an object, but the camera angle has also changed. Therefore, we can infer that, for example, the distribution of causal variables, conditioned on the camera angle, has been altered due to the soft intervention.	97

4.3	Generative model	98
4.4	Distribution of observed and causal variables in two causal models, $\mathcal{M}$ and $\mathcal{M}'$, within the same equivalence class up to reparameterization. (a) Ten observed samples with interventions on either Z_1 or Z_2 . (b) For no intervention ($I = 0$), the distributions of causal variables in both models are identical, though their value ranges differ and can be aligned using $\phi_{\mathcal{Z}}$. (c) Intervention on Z_1 ($I = 1$). (d) Intervention on Z_2 ($I = 2$). While the distributions for $I = 1$ and $I = 2$ are consistent between models, they vary with different intervention targets as soft interventions adjust the conditional distribution of causal variables (conditioned on their parents). For each intervention target I , the distributions in $\mathcal{M}$ and $\mathcal{M}'$ should shift consistently, given the known targets.	99
4.5	String diagrams for connections between E and E' . The triangle indicates sampling variables from their corresponding distributions.	100
4.6	The architecture processes pre-intervention observations X , post-intervention observations $\tilde{X}$, and their differences $X - \tilde{X}$ (intervention displacement), encoding them into latent representations. Each encoder outputs the mean (M) and variance (V) of a probability distribution function. By sampling from these distributions, we obtain pre-intervention exogenous variables E , post-intervention exogenous variables $\tilde{E}$, and the causal mechanism switch variable V . The exogenous variables are derived from the corresponding pre- and post-intervention encodings, while V is obtained from the encoding of the differences between X and $\tilde{X}$. The pre- and post-intervention exogenous variables are then passed through two fully connected (FC) layers, which predict the scale and location parameters. These predicted scale and location parameters, together with the post-intervention exogenous variables and the causal mechanism switch variable, are utilized in the solution function (4.36) to compute the post-intervention causal variables $\tilde{Z}$. Here, N denotes the total number of causal variables.	101
4.7	Causal disentanglement for different number of causal variables.	102
A.1	ADE	120
A.2	FDE	121
A.3	Predictions	121
A.4	Regularizations	121

List of Tables

1.1	Comparison between different models [57]	3
3.1	Comparison of GCRL with other recent HTP models. Heterogeneity pertains to the presence of diverse data forms, while multi-modality refers to the generation of multiple trajectories.	58
3.2	Robustness of different methods in the ETH-UCY dataset with controlled spurious correlation.	61
3.3	GCRL performance in cases with multiple changing factors.	62
3.4	Motion forecasting results on different domains of ETH-UCY.	62
3.5	MCC of S and Z after and before applying an affine transformation.	64
3.6	The effect of different components on performance with the synthetic dataset	65
3.7	The effect of different components on performance when experimenting with the ETH-UCY dataset.	66
4.1	Comparison of ICRL-SM with other recent related work on causal learning from interventional data	88
4.2	Actions and objects present in the Causal-Triplet images (ProcTHOR Dataset).	90
4.3	Actions and objects present in the Causal-Triplet images (Epic-Kitchens Dataset).	90
4.4	Comparison of identifiability results in synthetic dataset	92
4.5	Table shows the comparison of ICRL-SM performance on different configurations of $G3$ in Table 4.4.	93

4.6	Table comparing action and object accuracy across various methods on Causal-Triplet datasets under different settings. z and z_i show whether all causal variables (z), or only the intervened casual variable (z_i) are used for the prediction task. R_{64} denote images with resolutions 64×64	94
4.7	Action and object accuracy of the explicit models with experiments conducted applying an image with resolution of R_{64} as the input to the Resnet50 encoder with the intervened causal variable (z_i). Fixed-order denotes that the causal variables' order in the adjacency matrix remains unchanged. . . .	94
4.8	Table comparing action and object accuracy across various methods on Causal-Triplet datasets using ResNet18 model.	95
4.9	Action and object accuracy of three explicit models are compared with ICRL-SM. Experiments are conducted applying image with resolution of R_{64} as the input to the Resnet18 encoder with the intervened casual variable (z_i).	95

List of Abbreviations

ADE Average Displacement Error [59](#), [60](#), [62](#), [120](#)

AICM Augmented Implicit Causal Model [74](#), [78](#)

CAE Causal AutoEncoder [23](#), [24](#)

CGAN Conditional Generative Adversarial Network [22](#)

CONTA CONText Adjustment [27](#)

CRL Causal Representation Learning [2](#), [12](#), [13](#), [15](#), [17–19](#), [35](#), [67](#), [69](#)

CSG Causal Semantic Generative [25](#), [26](#), [31](#)

DAG Directed Acyclic Graph [3](#), [12](#), [21](#), [24](#), [30](#), [31](#), [36–39](#), [42](#), [44](#)

EID Encoder-Interaction-Decoder [56](#), [60](#), [103](#)

ELBO Evidence Lower BOund [25](#), [26](#), [31](#), [37](#), [41](#), [43](#), [51](#), [86](#)

ELCM Explicit Latent Causal Model [42](#), [43](#), [45](#), [68](#), [69](#)

ERM Empirical Risk Minimization [48](#)

FDE Final Displacement Error [59](#), [60](#), [120](#)

GCRL Generative Causal Representation Learning [ix](#), [xiv](#), [15](#), [18–24](#), [26](#), [28](#), [31](#), [32](#), [35](#), [46](#), [47](#), [53–56](#), [58–66](#), [68](#), [103](#), [104](#), [107](#), [108](#), [120](#)

GMM Gaussian Mixture Model [55](#), [56](#), [61](#), [65](#)

HTP Human Trajectory Prediction [xii](#), [2](#), [13–15](#), [17–19](#), [21](#), [31](#), [32](#), [35](#), [46–48](#), [56–58](#), [103](#), [107](#)

ICA Independent Component Analysis [72](#)

ICM Independent Causal Mechanisms [7](#), [25](#), [26](#)

ICRL-SM Implicit Causal Representation Learning via Switchable Mechanisms [ix](#), [xiv](#), [xv](#), [16](#), [18](#), [19](#), [35](#), [45](#), [66](#), [67](#), [86–88](#), [91–93](#), [95](#), [96](#), [104–108](#)

IID Identically and Independently Distributed [xi](#), [1–3](#), [13](#), [33](#), [35](#), [48](#), [61](#)

ILCM Implicit Latent Causal Model [42](#), [43](#), [45](#), [68](#), [69](#), [74](#), [91](#), [96](#)

IM Invariant-Modular [61](#), [62](#)

IRM Invariant Risk Minimization [32–34](#)

KLD Kullback–Leibler Divergence [41](#), [86](#)

LCM Latent Causal Model [77](#), [78](#), [80](#)

MB Markov Blanket [21](#), [23](#), [24](#)

MCC Mean Correlation Coefficient [63](#)

MEC Markov Equivalence Class [12](#), [31](#), [35](#)

MLP MultiLayer Perceptron [38](#)

MSD Minimum Separation Distance [59](#)

MSE Mean Squared Error [33](#), [52](#)

MSS Mechanism Shift Score [35](#), [36](#)

NLL Negative Log-Likelihood [33](#)

OOD Out-Of-Distribution [2](#), [61](#)

SCM Structural Causal Model [5](#), [27](#), [28](#), [37](#), [41](#), [68](#), [70](#), [71](#)

SMS Sparse Mechanisms Shift [7](#), [33](#), [35](#), [41](#), [58](#)

STGAT Spatial-Temporal Graph ATtention [60](#)

VAE Variational AutoEncoder [37](#), [38](#), [41](#), [43](#), [55](#), [91](#)

Notation Convention

- X : A random variable typically representing an outcome or event from a probabilistic experiment, which can take on different values according to a probability distribution. All random variables are denoted using uppercase letters.
- x : A value of random variable X . All values of random variables are denoted using lowercase letters.
- $\mathcal{X}$: The domain of random variable X . All domain of random variables are denoted using caligraphic letters.
- E_X : The exogenous variable of causal variable X . All exogenous variables are denoted using E .
- e_X : A value of exogenous variable of causal variable X . All exogenous variables' value is denoted using e .
- $Y_{X=x}$: A counterfactual query which is read as what would be the causal variable Y had the causal variable X 's value been x .
- pa_i : Set of parents of i th causal variable.
- $do(X = x)$: Intervention on causal variable X that sets its value to x .
- $p(X)$: The probability distribution of random variable X .

Chapter 1

Introduction

1.1 Motivation for Causality

Conventional supervised learning methods predominantly rely on statistical inference, operating under the assumption that data is [IID](#). However, this assumption is often violated in real-world applications, where environments or domains are dynamic, with various factors constantly changing. For instance, consider an image classification task aimed at distinguishing between images of dogs and cats using a deep learning model. The dataset for this task might include numerous variable factors, such as image brightness, weather conditions during image capture, background settings, and the color of the animals. If the model is trained on a dataset that does not adequately capture these variations, it may struggle to make accurate predictions when confronted with new images that exhibit different conditions. For example, a model trained predominantly on images of cats taken indoors and dogs photographed outdoors might perform poorly when presented with an outdoor image of a cat.

Moreover, statistical models are often perceived as black boxes, making it challenging to understand the internal mechanisms driving their predictions. This opacity raises critical questions: What do the learned features represent? How do these features influence the model's predictions? Are all features equally important, or are some more relevant than others? What relationships exist between the learned features? What would happen if one of the features were altered? These questions underscore the inherent limitations of statistical models in terms of interpretability and transparency. Without a clear understanding of the internal processes, it becomes difficult to trust the model's decisions, particularly in critical applications where interpretability is essential.

Real-world applications also often involve [confounding factors](#), which can lead to biased and unreliable predictions in statistical models. Confounding factors are variables that influence both the input features and the target labels, creating spurious associations that can mislead the model. This can result in problems such as Simpson’s paradox [51], where trends observed in different groups reverse when the groups are combined, or the mistaken association of co-occurring features as being causally related.

Returning to the image classification example, imagine that all images of cats were taken indoors, while all images of dogs were captured outdoors. In this scenario, a statistical model might erroneously learn to associate an indoor background with the label “cat” and an outdoor background with the label “dog.” Consequently, the model would likely produce incorrect predictions if presented with images of cats taken outdoors or dogs photographed indoors. In this case, the background setting (outdoor vs. indoor) acts as a [confounding factor](#), distorting the model’s understanding of the true relationship between the visual characteristics of the animals and the target labels.

This example highlights a broader challenge: without addressing [confounding factors](#), models risk learning spurious correlations that fail to generalize beyond the specific conditions of the training data. Such biases not only degrade the model’s performance but also hinder its ability to adapt to new and diverse environments. Overcoming these limitations requires moving beyond purely statistical approaches and incorporating causal reasoning, enabling models to distinguish between genuine causal relationships and misleading correlations. By doing so, we can enhance the robustness, interpretability, and generalizability of machine learning models in real-world applications.

Once a causal model is established, it allows us to make inferences about [interventions](#), [counterfactuals](#), and potential outcomes which will be discussed in more details in Section 1.3. In contrast, traditional statistical models are limited to reasoning about the outcomes of IID experiments. As illustrated in Figure 1.1, a causal model can adapt to different distributions by intervening on the variables, whereas a statistical model is confined to learning a single distribution.

[Causal Representation Learning \(CRL\)](#) has garnered significant attention in various applications, including transfer learning, [Out-Of-Distribution \(OOD\)](#) generalization, reinforcement learning, action classification, and Human Trajectory Prediction (HTP) [40, 69, 57, 42]. Causality aligns with human understanding of physical processes and facilitates the learning of meaningful representations—where “meaningful” refers to robustness, interpretability, transferability, and other desirable qualities [56]. Table 1.1 provides a comparison of different models used to explain a process.

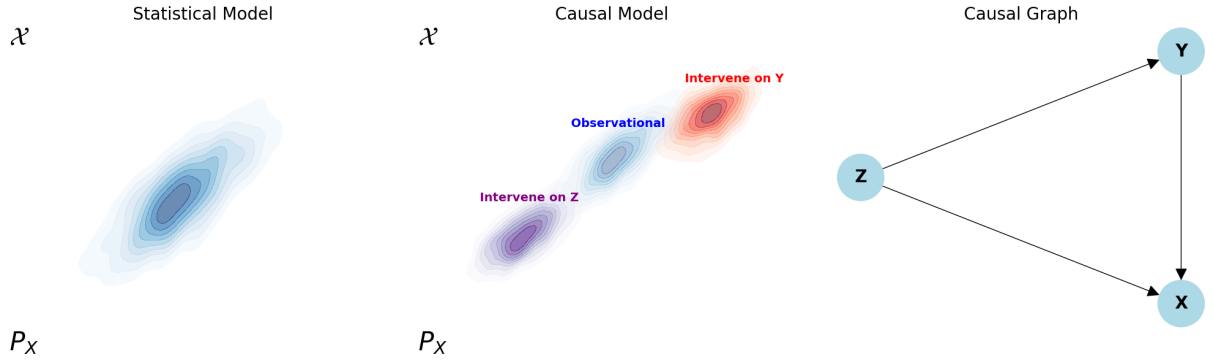


Figure 1.1: Comparison between statistical (left) and causal models (middle). X represents the observed data, with P_X denoting its distribution. The observational distribution corresponds to the scenario where no interventions have occurred. While statistical models are restricted to a single IID distribution, causal models can capture multiple distributions by allowing interventions on causal variables [57].

Table 1.1: Comparison between different models [57]

Model	Predict in i.i.d setting	Predict under intervention	Answer counterfactual questions	Obtain physical insight	Learn from data
Mechanistic/physical	Yes	Yes	Yes	Yes	?
Structural causal	Yes	Yes	Yes	?	?
Causal graphical	Yes	Yes	No	?	?
Statistical	Yes	No	No	No	Yes

1.2 Causal Models

1.2.1 Graphical Causal Models

Bayesian networks are a class of probabilistic graphical models that represent a set of variables and their conditional dependencies using a [Directed Acyclic Graph \(DAG\)](#). These networks are particularly effective for modeling uncertainty and probabilistic relationships. A special type of Bayesian network, known as a causal model, focuses on capturing causal relationships between variables.

In a causal graphical model, causal relationships are depicted by directed edges. For instance, if there is an edge from variable X to variable Y (denoted as $X \rightarrow Y$), X is considered a cause of Y . This means that changes in X can influence the value of Y , but not vice versa [51].

In causal models, there are two primary types of variables: [endogenous variable](#) and [exogenous variable](#). Endogenous variables are those within the causal model that explain the relationships among variables in a dataset. In contrast, exogenous variables are variables that pertain to a single individual or a situation in nature. Unlike endogenous variables, exogenous variables are not influenced by any other variables in the model and are assumed to be mutually independent.

Graphical models feature three main configurations of variables: Chain, Fork, and Collider [\[51\]](#):

- **Chain:** This configuration involves three nodes and two directed edges, with one edge directed into and one edge directed out of the middle variable. In a chain, conditional independence is observed between two variables, X and Y , given a third variable Z , if there is only one unidirectional path connecting X and Y , and Z intercepts this path. See [Figure 1.2](#).



Figure 1.2: Chain configuration in graphs.

- **Fork:** This configuration consists of three nodes, with two edges emanating from the middle variable, which serves as the common cause of the other two nodes. If a variable X is a common cause of variables Y and Z , and there is only one path connecting Y and Z , then Y and Z are conditionally independent given X . In this context, X is referred to as a [confounding factor](#). See [Figure 1.3](#).

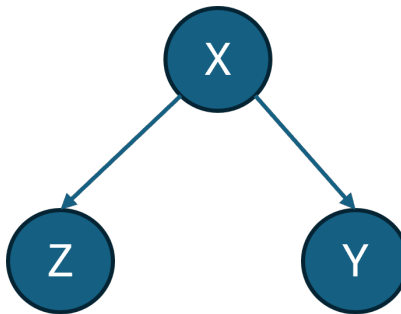


Figure 1.3: Fork configuration in graphs.

- **Collider:** This configuration features three nodes, where the middle node receives directed edges from the other two nodes. The middle node in a collider configuration is a common effect of the other two nodes. In a collider setup, if a variable Z is the collider between variables X and Y , and there is only one path connecting X and Y , then X and Y are unconditionally independent. However, they become dependent when conditioned on Z or any of its descendants. See Figure 1.4.

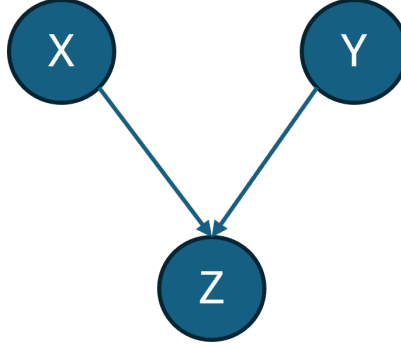


Figure 1.4: Collider configuration in graphs.

1.2.2 Structural Causal Models

Structural Causal Model (SCM)s provide a framework for describing the relationships between features of the world through equations rather than graphical representations. In a **SCM**, we say that X is a direct cause of Y if X appears in the functional equations that determine Y 's value. These functions are usually referred to as causal mechanisms. Conversely, X is considered a cause of Y if X is either a direct cause of Y or is an indirect cause through other variables. An example **SCM** in the context of motion forecasting is given below:

- Observed Trajectory (X): $x = e_X$ No direct causes (exogenous variable).
- Environmental Factors (Y): $y = f(x) + e_Y$. Here, $f(x)$ represents a function of the pedestrian's observed trajectory that influences the environmental factors. e_Y represents an error term capturing any unobserved environmental factors that affect the pedestrian's trajectory.
- Intention (Z): $z = g(x, y) + e_Z$. Here, $g(x, y)$ represents a function of the pedestrian's previous trajectory and environmental factors that influences their intention. e_Z

represents an error term capturing any unobserved factors affecting the pedestrian's intention.

- Future Trajectory (W): $w = h(x, y, z) + e_W$. Here, $h(x, y, z)$ represents a function of the pedestrian's previous trajectory, environmental factors, and intention that determines their future trajectory. e_W represents an error term capturing any unobserved factors affecting the pedestrian's future trajectory.

1.2.3 Inputs and Outputs of Causal Models

Every causal model takes three inputs and produces three outputs. The inputs are [50]:

- A set of qualitative causal assumptions (A) such as faithfulness, causal sufficiency or any other assumption based on domain knowledge.
- A set of queries concerning the causal and counterfactual relations among variables (Q).
- Experimental or non-experimental data (D) presumably consistent with A.

The outputs are [50]:

- A set of logical implications of A (A^*).
- data-dependent claims (C); the magnitude or likelihoods of queries in Q.
- A list of testable statistical implications (T).

1.2.4 Causal Markov Condition

As with all Bayesian networks, causal models benefit from the rule of decomposition. For any model with an acyclic graph, the joint distribution of the variables is given by the product of the conditional distributions $P(\text{child} \mid \text{parents})$ over all the "families" in the graph [51]:

$$P(x_1, x_2, \dots, x_n) = \prod_{i=1}^n P(x_i \mid x_{pa_i})$$

This rule simplifies the estimation problem, particularly in large models. Instead of computing the joint distribution by counting the frequencies of all variables, the problem reduces to estimating a few low-dimensional distributions. Consequently, this approach saves significant processing time and mitigates the curse of dimensionality. Graphical models enable this simplification without always requiring knowledge of the functions relating the variables, their parameters, or the distributions of their error terms.

1.2.5 Independent Causal Mechanisms Principle

"The causal generative process of a system's variables 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."[57]

To illustrate this principle, consider a simple causal model such as $X \rightarrow Y \rightarrow Z$. According to the Causal Markov condition, this model can be factorized as:

$$p(X, Y, Z) = p(X) \cdot p(Y | X) \cdot p(Z | Y)$$

Thus, this model is composed of three independent mechanisms: $p(X)$, $p(Y | X)$, and $p(Z | Y)$. According to the [Independent Causal Mechanisms \(ICM\)](#) principle, any changes in one of these mechanisms do not affect the others. Additionally, information about one mechanism does not provide information about the others.

The former aspect leads to an important hypothesis in causal models known as [Sparse Mechanisms Shift \(SMS\)](#). This hypothesis posits that small changes in a causal model tend to manifest in a sparse manner, affecting only a subset of factors rather than all simultaneously [57]. This principle is particularly useful in transfer learning, where it helps identify which parts of the model need adaptation for a new domain. Given that transfer learning often involves few samples in the new domain, a causal model can be highly effective as it allows for updating only a subset of parameters.

The latter result of the [ICM](#) principle addresses challenges in semi-supervised learning methods. Assuming the underlying causal graph is $X \rightarrow Y$, where X represents inputs and Y represents targets we aim to predict, using additional unlabelled data (i.e., information about $p(X)$) does not provide information about $p(Y | X)$ [57].

1.2.6 Confounding Bias

Confounding bias and sample selection bias are other sources of weak predictions [5]. Sample selection bias is induced by preferential selection of units. Confounding bias arises from confounders in a causal model, such as those represented by forks in causal graphs. The difference between $p(Y \mid \text{do}(X))$ and $p(Y \mid X)$ highlights the impact of confounding bias.

The do-operation, denoted as $\text{do}(X = x)$, is used to capture the causal effect of X on Y by setting X to a specific value x through an intervention, rather than merely observing X in its natural state. Formally, $p(Y \mid \text{do}(X = x))$ represents the probability distribution of Y when X is actively set to x through an [intervention](#), which isolates the causal relationship between X and Y from other confounding influences. This contrasts with $p(Y \mid X = x)$, which may be influenced by [confounding factors](#).

The do-operation is critical for understanding causal relationships because it helps to control for confounding factors and assess the true impact of interventions. To identify and address confounding bias, we use criteria such as the backdoor and front-door criteria [51]. These criteria help determine when and how to adjust for confounders to reveal the true causal effect.

Confounding factors, which are typically unobserved variables that influence both X and Y , can lead to biased predictions. For example, in complex systems with many variables, [confounding factors](#) are likely present. The do-operation and the associated causal criteria enable us to mitigate confounding bias and obtain more reliable causal estimates.

By applying these causal principles, we can refine our models to better handle the complexities and biases inherent in real-world data, leading to more accurate and interpretable predictions.

1.3 Interventions and Counterfactuals

1.3.1 Interventions

In a properly randomized controlled experiment, all factors that influence the outcome variable are either static or vary randomly, except for one. Therefore, any observed change in the outcome variable must be attributable to that single input variable. However, randomized controlled experiments are not always feasible. For instance, we cannot control the weather to study its effect on wildfires; we can only record the data. The challenge with

such observational studies is that it becomes difficult to disentangle causal relationships from mere correlations. Consequently, sometimes we need to hypothetically intervene on variables to study their effects on other variables [51].

Interventions differ from conditioning and are represented by the *do* operation. When we intervene on a variable, we fix its value, which alters the model and the values of other variables. In contrast, conditioning on a variable does not change the model; it merely restricts our focus to a specific subset of the data. In distributional terms, $P(Y = y \mid X = x)$ represents the population distribution of Y among individuals where X equals x . On the other hand, $P(Y = y \mid do(X = x))$ represents the population distribution of Y if every individual in the population had their X value fixed at x [51].

We previously discussed that the difference between $p(Y \mid do(X))$ and $p(Y \mid X)$ is confounding bias. To illustrate how **interventions** can help eliminate confounding bias, consider a causal model in the context of motion forecasting with variables X , Y , and Z , where X represents observed trajectories, Y represents future trajectories, and Z represents invariant features such as physical laws. In this model, Z confounds the relationship between X and Y , and there should be a directed edge from X to Y , as previous locations of pedestrians influence their future trajectories (Figure 1.5a).

If we intervene on X , the edge from Z to X is removed because X is no longer determined by Z (Figure 1.5b). We also know that by intervening on X , the marginal distribution of Z , $p(Z)$, and the conditional distribution $p(Y \mid Z, X)$ will remain unaffected because the conditional distribution refers to the mechanism generating Y from Z and X . Thus, we can write the following:

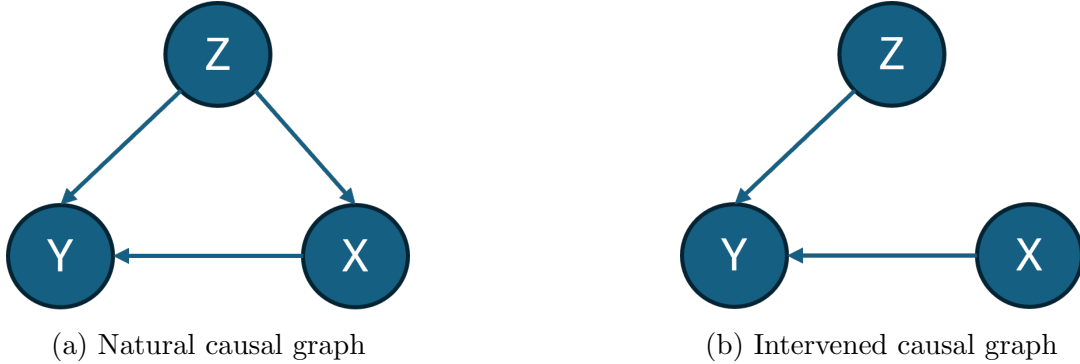


Figure 1.5: Comparison of causal graphs for motion forecasting.

$$p_m(Y = y \mid X = x, Z = z) = p(Y = y \mid X = x, Z = z)$$

$$p_m(Z = z) = p(Z = z)$$

where p_m denotes the distribution after intervention. Using these equalities, we can calculate $p(Y \mid do(X))$ as follows:

$$\begin{aligned} p(Y = y \mid do(X = x)) &= p_m(Y = y \mid X = x) \\ &= \sum_z p_m(Y = y \mid Z = z, X = x), p_m(Z = z \mid X = x) \\ &= \sum_z p_m(Y = y \mid Z = z, X = x), p_m(Z = z) \\ &= \sum_z p(Y = y \mid Z = z, X = x), p(Z = z) \end{aligned} \tag{1.1}$$

The second equality follows from the fact that in the post-intervention graph, there is no edge between X and Z . Equation 1.1 is known as the adjustment formula [51]. This formula indicates that to calculate the causal effect of X on Y , we need to marginalize over the confounding factor Z , which is completely ignored in $p(Y \mid X)$. The adjustment formula is generalized to all causal models through the backdoor and front-door criteria [51].

1.3.2 Counterfactuals

To explain [counterfactuals](#), let's consider a new example involving a marketing campaign. Suppose a company launches a new advertising campaign and wants to assess its impact on sales. After analyzing the data, the company finds that sales increased significantly during the campaign.

Now, the company might ask: "What would have been the sales if we had not run the advertising campaign?" This is a counterfactual scenario because it involves considering an alternative reality where the campaign was not executed.

To express this counterfactual using a do-expression, we face a clash between the sales observed during the campaign and the sales we would have observed without the campaign. The counterfactual in this case is:

$$E(\text{sales} \mid do(\text{campaign} = \text{no}), \text{sales observed} = \text{increased})$$

We use subscripts to address this issue:

$$E(Y_{X=0} \mid X = 1, Y = Y_{X=1} = \text{increased})$$

Here, we have two different worlds: one where the advertising campaign was conducted ($X = 1$) and another where it was not conducted ($X = 0$). In interventions, everything is considered within a single world, so counterfactuals cannot be expressed solely using do-expressions. One might try to estimate the counterfactual by analyzing sales from a similar campaign in the past or comparing with a control group, but this does not capture the exact conditions of the campaign.

Exogenous variables play a critical role in [counterfactuals](#). Consider a causal model with two variables:

$$\begin{aligned} X &= aE_X \\ Y &= bX + E_Y \end{aligned}$$

Let's compute Y_x , i.e., what the sales variable Y would be had the campaign been x , in the situation where $E_Y = e_Y$:

$$Y_x = bx + e_Y$$

Next, compute X_y , i.e., what the campaign variable X would be had the sales been y , in the situation where $E_X = e_X$:

$$X_y = \frac{y - E_Y}{b}$$

The value of Y_x is expected and straightforward. The value of X_y is also expected if we consider this hypothetical condition stemming from an [intervention](#) on Y , since X causes Y and not the other way around. If we do not invoke the [intervention](#) metaphor, and treat the value of Y as a spontaneous change, the value of X_y is less intuitive. This observation reflects the idea that hypothesizing about the future does not alter the past. Another explanation is that hypothesizing about a variable does not change the value of an [exogenous variable](#), and since X only depends on E_X , it remains unchanged [\[51\]](#).

In [interventions](#), we manipulate a variable externally, which affects the entire population. In contrast, [counterfactuals](#) consider specific individuals or situations and hypothesize about [endogenous variables](#). Therefore, the $do(x)$ -operator captures the behavior of a population under [intervention](#), while $Y_x(u)$ describes the behavior of a specific individual $E = e$

under such [interventions](#). This difference is significant and allows us to define probabilities of concepts such as credit, blame, and regret, which the *do*-operator cannot capture. Joint probabilities over multiple-world [counterfactuals](#) can be expressed using the subscript notation, as in $p(Y_1 = y_1, Y_2 = y_2)$, but not using the *do*(x) notation [\[51\]](#).

1.4 Challenges

1.4.1 Identifiability

Identifiability is one of the main challenges in [CRL](#). Identifiability can be defined as follows:

“A statistical model is specified by a parameter space Θ , a family of distributions P , and a mapping $\pi : \theta \rightarrow P$; i.e. $\pi(\theta) \in P$ for each $\theta \in \Theta$. In more conventional notation, we define $P = \{p_\theta : \theta \in \Theta\}$, in which case $p_\theta = \pi(\theta)$. A statistical model is called identifiable if the parameter mapping π is one-to-one (injective).” [\[31\]](#)

$$\pi(\theta) = \pi(\theta') \rightarrow \theta = \theta'$$

We can infer the joint distribution of variables from a graph, but inferring the graph structure from distributions requires identifiability. For example, suppose that we have two datasets: one collected in Austria and one in Switzerland. Suppose that there are two variables in these datasets: altitude A and average annual temperature T of weather stations. The problem is to learn a graph from the joint distribution of these two variables $p(A, T)$. This joint distribution can be factorized as either $p(A)p(T|A)$ or $p(T)p(A|T)$, and both would be correct [\[56\]](#). Considering the Markov condition in graphs, there will be two graphs that can describe this joint distribution: one with an edge from A to T and one with an edge from T to A . Therefore, without further assumptions, the true [DAG](#) is not identifiable from this data. That is, it is possible to learn either of the graphs from the same joint distribution. In general, the set of graphs that have the same skeleton (same edges regardless of direction) and the same v-structures are said to belong to the same [Markov Equivalence Class \(MEC\)](#) [\[57\]](#). Incorrect [DAGs](#) in the [MEC](#) set may contain non-causal factorizations. Without further assumptions, there is no guarantee that we can learn the true causal model from data. We could learn any of the graphs in the [MEC](#). It has been shown that in an unsupervised setting, it is impossible to identify the true causal model with only observational data [\[43\]](#). Therefore, additional assumptions on model parameters, graph structure, distributions of causal variables, and availability of interventional data are required to ensure identifiability.

As assumptions are inseparable parts of the causal models (See Section 1.2.3), researchers have tried to identify causal structures from data using various assumptions on data, causal structure, causal variables, functional form of causal mechanisms, and so on, which sometimes may not be practical [39, 6, 64, 1, 28]. Depending on the application and available data, we should use different assumptions to guarantee identifiability.

1.4.2 Scalability

Scalability is a significant challenge in CRL, especially in complex systems with a large number of variables. As the number of variables increases, so does the complexity of causal relationships, which can be further compounded by confounding and mediating effects [51]. This complexity often leads to ambiguities in identifying and modeling causal interactions. To address these challenges and enhance scalability, it is crucial to leverage domain knowledge. Explicitly incorporating this knowledge allows for more accurate modeling of causal factors and their relationships, thus improving the effectiveness of causal inference in large-scale systems.

1.5 Downstream Tasks

1.5.1 Human Trajectory Prediction

One of the primary applications of CRL is transfer learning, as discussed in [50, 40, 65]. Transfer learning involves leveraging knowledge gained from a model trained on a source domain or task and adapting it to improve performance on a different target domain or task. This adaptation can take various forms, such as transferring learned features or model parameters. A specific type of transfer learning, known as domain adaptation, occurs when the task remains the same, but the data distribution differs between the source and target domains. The goal of domain adaptation is to adjust the model so that it performs effectively on the target domain despite the differences in data distributions. In many real-world scenarios, the IID assumption is violated, leading to failures in statistical inference across multiple domains. To address this, I leverage causality in domain adaptation. A particular application where I apply causal domain adaptation is HTP, which involves dealing with multiple environments [10].

HTP is critical in various domains, such as infrastructure design, traffic operations, crowd abnormality detection systems, evacuation planning, and autonomous vehicles. In

the context of autonomous vehicles, accurate prediction of human trajectories is essential for enabling the vehicle controller to plan its motion effectively [33]. The [HTP](#) problem can be defined as follows:

Definition 1.5.1 (Human Trajectory Prediction ([HTP](#)) Problem). *Given the states of M pedestrians walking in an environment during the observed T_{obs} timesteps, where the states are the 2D coordinates of the pedestrians, predict the states of each pedestrian in the next T_{pred} timesteps.*

[HTP](#) is challenging due to several factors [33, 21]:

- **Encoding past observations:** Effective capturing of long-term dependencies and interactions is essential.
- **Multi-Modality:** Multiple plausible predictions can arise from the same past observations, given that the destination and intended path are unknown, adding randomness to the trajectories.
- **Presence of social and physical interactions:** Humans interact with other pedestrians and objects in the scene, such as walking in groups, following others, and avoiding collisions. These interactions must be modeled.
- **Physically acceptable outputs:** Predictions must be physically feasible, ensuring, for example, that a predicted trajectory does not result in collisions.
- **Socially acceptable outputs:** Even if trajectories are physically feasible, they must also respect social norms, such as yielding right-of-way and maintaining personal space. Modeling these norms is non-trivial.

In [HTP](#), robustness is critical for training models that generalize well across multiple domains. Domains in [HTP](#) can encompass a variety of environments, such as intersections, campuses, and densely crowded areas. A key challenge in these environments is addressing the multi-modality of human trajectories, which refers to the fact that human motion often has multiple plausible future paths due to varying intentions, social interactions, and physical constraints. For instance, at an intersection, a pedestrian may choose to cross the street, turn back, or wait depending on surrounding traffic and social cues. This inherent uncertainty requires models to predict not just a single trajectory but a distribution of possible future trajectories, capturing the range of plausible outcomes.

Methods to address distribution shifts can be categorized into domain generalization, domain adaptation, and transfer learning [42]. In domain generalization, the goal is to train a model that performs well in unseen but related test domains, though these methods often rely on strong assumptions that may not hold in practice [42]. Domain adaptation allows the model to adapt to a new domain using a set of unlabeled test samples, ensuring consistent feature distributions between test and train samples through divergence minimization. However, this approach is less suitable for [HTP](#), where future trajectories are relatively easy to obtain. In contrast, transfer learning leverages specialized architectures or transfer-oriented learning to achieve generalization [42].

The central objective in robust [HTP](#) can be formulated as:

Definition 1.5.2 (Robust Human Trajectory Prediction Problem). *How can we learn invariant representations of past trajectories and interactions to accurately infer future trajectories?*

This problem involves two key aspects:

1. **Trajectory Prediction:** How can we best model social and physical interactions? How can we address the multi-modality of trajectories? How can we develop efficient and fast models for trajectory prediction? How can we ensure accurate and feasible trajectory predictions?
2. **Robustness:** How can we learn domain-invariant mechanisms? How can we identify the causal model that best explains the data? How can we relax the assumptions necessary for learning causal models? How can we eliminate confounding bias?

The focus of this thesis is on the robustness aspect of [HTP](#). First, I will use domain knowledge to construct a causal model that explains the data generation process in motion forecasting. Next, I will introduce a novel method called [GCRL](#) [3] for learning causal representations using variational inference. This method leverages interventions in causal models to eliminate confounding bias and augments the causal model with an additional selection variable to enable robust generalization across different environments. As a proof of concept, I will apply [GCRL](#) to the [HTP](#) application, though its utility could extend to other domains as well. The results on both synthetic and real-world [HTP](#) datasets demonstrate that incorporating causality can enhance the generalizability and transferability of deep learning models.

In the second proposed method for [CRL](#), I have enhanced [GCRL](#) to learn the causal model itself, rather than relying solely on domain knowledge.

1.5.2 Action and Object Recognition

Constructing causal models based on domain knowledge, particularly in complex systems with numerous variables, is a challenging endeavor. Action and object recognition exemplify such a task. In this context, an agent interacts with a set of objects, performing actions or interventions—such as to *Insert* on an object like *Bread* into toaster. An example of pre- and post-intervention samples is illustrated in Figure 1.6. The goal is to accurately identify the type of action and the class of the object from these pre- and post-intervention images. One might hypothesize that the underlying causal graph for this task resembles the structure depicted in Figure 1.7. As shown, the causal graph is intricate, involving numerous variables and significant confounding. Consequently, there is a need for a model capable of autonomously learning the causal structure.



(a) Pre-intervention sample



(b) Post-intervention sample

Figure 1.6: Pre and Post-intervention samples from the dataset in [41].

To this end, I have proposed a novel method called [ICRL-SM](#) [4], which can uncover the underlying causal graph of the dataset from pre-intervention and post-intervention samples. In this work, I have examined the conditions and assumptions under which the identifiability of causal models can be achieved. Based on these assumptions, I have proven the identifiability of the proposed method. As a proof of concept, I evaluated my method on a synthetic dataset that fully complies with the assumptions and compared it to other methods. Additionally, I have evaluated my method on real-world datasets where some of the assumptions are violated. The results suggest that the proposed method is a promising direction for future research, bringing existing causal representation learning methods one step closer to real-world applications.

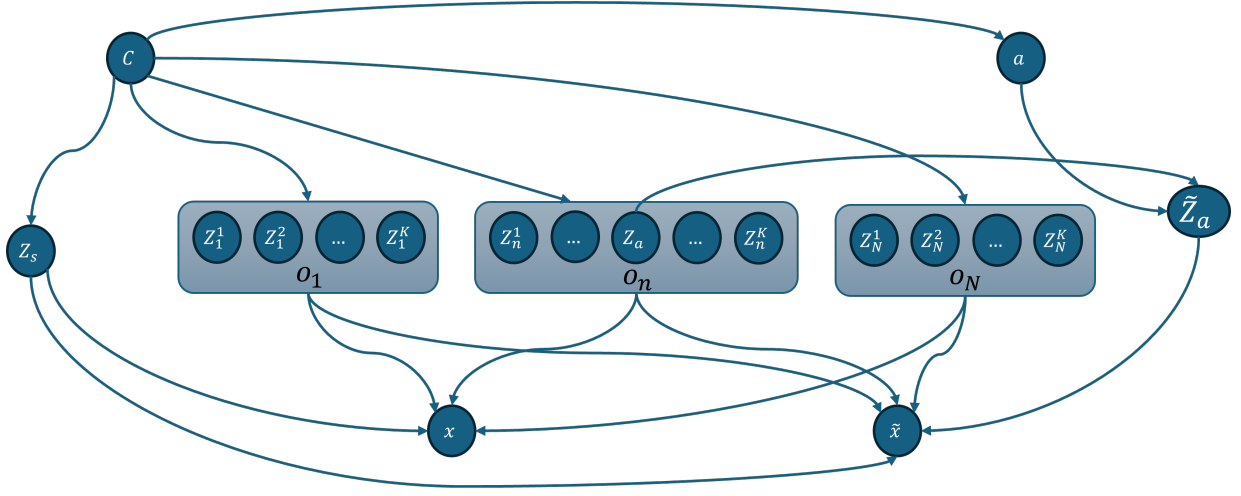


Figure 1.7: Hypothesized causal graph in action and object recognition task inspired by models used in [41]. z_s are causal variables corresponding to scene properties, c are the potential **confounding factors** such as camera angle, a are the actions/**interventions**, the boxes o_i are the set of causal variables z_i^j for object o_i , x are the pre-intervention observations, $\tilde{x}$ are the post-intervention observations, and $\tilde{z}_a$ are the post-intervention causal variables.

1.6 Research Overview and Thesis Structure

The central focus of this thesis is on developing novel methods for causal representation learning, with an emphasis on enhancing the generalizability and robustness of machine learning models across different domains and applications. I have explored two primary applications in this work: **HTP** as well as Action and Object Recognition. These tasks serve as critical testbeds for evaluating the efficacy of the proposed methods in real-world scenarios where traditional machine learning approaches often fall short due to distribution shifts and the complexity of underlying causal mechanisms.

The remainder of this thesis is structured as follows:

- **Chapter 2: Background and Related Work**

This chapter provides a comprehensive review of the current state-of-the-art methods in **HTP**, **CRL**, and causal domain adaptation. It highlights the key challenges associated with these tasks and offers a detailed background on the relevant literature.

- **Chapter 3: Generative Causal Representation Learning**

This chapter introduces [GCRL](#), a novel method for leveraging causal representations to improve the robustness and transferability of deep learning models. As a proof of concept, [GCRL](#) is evaluated in [HTP](#) task. The chapter discusses the domain knowledge used to construct causal models for motion forecasting and details the variational inference approach employed to eliminate confounding bias. The chapter concludes with an evaluation of [GCRL](#) on both synthetic and real-world [HTP](#) datasets.

- **Chapter 4: Implicit Causal Representation Learning via Switchable Mechanisms**

This chapter introduces [ICRL-SM](#), a method designed to uncover the causal structure. The chapter examines the theoretical underpinnings of the method, including conditions for identifiability, and presents empirical results from both synthetic and real-world datasets in action and object recognition task. This chapter also explores the potential of [ICRL-SM](#) as a foundational approach for future research in [CRL](#).

- **Chapter 5: Extensions and Future Work**

This chapter discusses potential extensions of the proposed methods, including the development of more realistic approaches to causal model discovery and the application of these methods to additional domains. It also outlines open questions and future research directions in the field of [CRL](#).

- **Chapter 6: Conclusion**

The thesis concludes with a summary of the key contributions and findings, emphasizing the significance of causality in improving the robustness and generalizability of machine learning models.

1.7 Contributions

The key contributions of this thesis are as follows:

1. **Generative Causal Representation Learning [3] (Accepted in ICML 2023):**

- Developed a novel method for leveraging causal representations to enhance the robustness of [HTP](#) models across different environments.

- Demonstrated the effectiveness of [GCRL](#) in addressing distribution shifts in both synthetic and real-world [HTP](#) datasets, thereby improving generalization and transferability.

2. Implicit Causal Representation Learning via Switchable Mechanisms [\[4\]](#) (Under review):

- Proposed a new approach for learning causal structures, specifically for Action and Object Recognition tasks, without relying solely on domain knowledge for constructing causal graphs.
- Proved the identifiability of the causal model under specific assumptions and validated the method on both synthetic and real-world datasets.
- Showed that [ICRL-SM](#) brings causal representation learning methods closer to practical, real-world applications by relaxing previous assumptions in the literature.

3. Theoretical and Empirical Analysis:

- Conducted a thorough analysis of the conditions and assumptions under which causal models can be identified, contributing to the theoretical foundations of causal representation learning.
- Provided empirical evidence demonstrating the applicability and advantages of the proposed methods in practical scenarios.

4. Bridging Causal Representation Learning and Real-World Applications:

- Established a connection between causal learning methods and their potential impact on real-world applications, particularly in the fields of autonomous vehicles and human-computer interaction.

Through these contributions, this thesis advances the field of [CRL](#) by introducing methods that not only perform well in controlled environments but also hold promise for broader, real-world applications where robustness and generalization are critical.

Chapter 2

Background and Related Work

In this chapter, I will review related work in the field, providing an in-depth explanation of key concepts in causal representation learning. Additionally, I will discuss several papers that offer valuable insights into how causal representation learning methods function.

2.1 Causal Domain Adaptation

This section reviews key concepts that are central to [GCRL](#), many of which have also been explored in prior research.

2.1.1 Domain Adaptation by Causal Mechanism Transfer

In [\[61\]](#), the authors address the challenge of few-shot domain adaptation, where there is limited data available from the target domain. They assume the existence of an invariant mechanism, represented by a nonlinear function f , that relates the distributions of the source and target domains.

The methodology presented in that paper involves several key steps:

1. **Learning the Invariant Mechanism:** Utilizing an invertible neural network f , trained using Generalized Contrastive Learning.
2. **Augmenting Target Domain Data:** Using the learned $\hat{f}$, to extract the Independent Components of the target domain data and augment the target domain samples to increase the sample size.

3. **Empirical Risk Minimization:** Fitting a classifier to the augmented target domain data.

The paper emphasizes the importance of transfer assumptions in domain adaptation which are discussed below [61]:

1. **Parametric:** Assumptions about parametric distribution families or shifts which are mainly used in remote sensing.
2. **Invariant Conditionals and Marginals:** Assumptions about invariance in conditional or marginal distributions which are mainly used in brain computer interface.
3. **Small Discrepancy or Integral Probability Metric:** Methods based on distributional similarities which are mainly used in computer vision tasks.
4. **Transferable Parameter:** Existence of parameterized feature extractors suitable for the target domain. This transfer assumption is mainly used computer vision tasks.
5. **Mechanism:** Assumptions about invariant mechanisms across domains which are mainly used in medical records.

The transferable parameter assumptions is used in GCRL for robust HTP problem.

2.1.2 Unsupervised Domain Adaptation by Augmented Directed Acyclic Graphs

In [71], the authors address domain adaptation in an unsupervised setting where labels are not available in the target domain. Their approach is built on several key assumptions and methods:

- **Transfer Assumption:** The key transfer assumption used in [71] is transferable parameters. The authors assume that the Markov Blanket (MB) features [66] of the class variable (variable that should be predicted) and some external latent variables θ account for variations across domains. The addition of θ , results in the augmented DAGs. These latent variables are assumed to be mutually independent to avoid the curse of dimensionality. Variations in distributions across domains are modeled through $P(\theta)$.

- **Causal Model vs Graphical Model:** The paper [71] critiques the use of graphical models to encode conditional independence relations rather than learning the causal structure. The authors use a graphical model to represent conditional independencies but do not address confounding variables or the distinction between causes and effects due to selection bias and the desire to avoid strong assumptions such as faithfulness.
- **Conditional Generative Adversarial Network:** To learn the augmented Directed Acyclic Graph, the paper uses [71] a non-parametric method Constraint-based causal Discovery from Nonstationary/heterogeneous Data (CD-NOD) [72]. Since no labeled data is available in the target domain, a [Conditional Generative Adversarial Network \(CGAN\)](#) is used to approximate probabilities and generate labeled data.
- **Variational Inference:** Due to the intractability of the involved distributions during training, stochastic variational inference is employed to approximate conditional distribution of θ given observations and labels. The variational expectation maximization algorithm is used to learn parameters in [CGAN](#) and variational parameters for generating θ .
- **Source Domain Distributions:** The reliability of the results improves with the number of source domain distributions. The paper [71] specifically uses at least two source domain distributions.

Similar to θ in [71], [GCRL](#) will also use an external latent variable to model variations in distributions across domains.

2.1.3 Markov Blanket Features

In [66], the authors explore methods for feature selection grounded in causal inference. The paper provides a detailed review of various approaches that leverage causal relationships among variables for feature selection.

The authors emphasize the importance of understanding causal relationships rather than merely relying on correlations. They review techniques for causal discovery that identify relevant features by examining causal dependencies.

Several methods for causal feature selection are discussed, including:

- **Constraint-Based Methods:** These use conditional independence tests to infer causal relationships.

- **Score-Based Methods:** These use scoring functions to evaluate the causal relevance of features.
- **Hybrid Methods:** These combine constraint-based and score-based approaches to enhance accuracy.

The paper [66] evaluates these methods, highlighting their strengths and limitations. It compares causal feature selection techniques with traditional methods, emphasizing that incorporating causal insights can lead to more effective feature selection and improved model performance.

GCRL is a score-based method in which the causal variables are learned by optimizing a novel loss function.

2.1.4 Domain Generalization by Markov Blanket Features

In [65], the authors propose a method called Causal AutoEncoder (CAE) for robust domain adaptation when no data is available in the target domain, and only one source domain distribution is used. Their approach integrates deep learning with structural causal learning and consists of several key components:

- **Domain Adaptation Scenario:** Unlike previous methods, CAE assumes no available data in the target domain, focusing on robust domain adaptation. They rely on a single source domain distribution.
- **Deep Learning and Causal Learning:** The method combines deep learning with causal learning. An autoencoder is first used to reduce the dimensionality of input data.
- **Causal Features vs. Correlation-Based Learning:** Correlation-based learning only captures co-occurrences between inputs and class variables, which may not generalize well. CAE emphasizes learning causal features, categorizing them into task-relevant (MB features) and task-irrelevant features.
- **Invariance of Causal Relations:** CAE relies on the invariance of causal relationships between input variables and class variables. It employs Markov Blanket discovery algorithms [66] to learn causal features without using target domain data.

- **Domain Generalization vs. Domain Adaptation:** CAE works on domain generalization in which no samples are available from target domain. This contrasts with domain adaptation methods, which adapt features using a few target domain examples. Also, CAE uses only one source domain data, distinguishing it from other domain generalization methods.
- **Stable Learning Methods:** Stable learning methods, focusing on selection bias and model misspecification with one source domain, differ from CAE. CAE avoids these issues by learning causal representations instead of using causal feature selection or reweighting samples. Causal feature selection methods are categorized into constraint-based and score-based methods, each with its own limitations [66].
- **Causal Autoencoder Components:** CAE comprises three main parts:
 - **Autoencoder:** Removes noise and reduces the dimensionality of the feature space, enhancing the reliability of independence tests in causal structure learning.
 - **Causal Structure Learning:** Encoded features are categorized into task-relevant (MB features) and task-irrelevant features. Under the faithfulness assumption, MB features are the unique set of features with maximum predictive power. The authors use a DAG over representations and learn the adjacency matrix using an acyclicity constraint.
 - **Classifier:** CAE includes a classifier and cross-entropy loss for predicting labels using MB features.
- **Datasets:** The paper uses commonly used datasets in domain adaptation, including Office-Caltech10, Amazon Review, and Reuters-21578.

GCRL employs a variational autoencoder (VAE) instead of a standard autoencoder, as used in [65], to effectively capture and address the inherent multi-modality of human trajectories.

2.1.5 Independent Causal Mechanisms principle in Domain Adaptation

In [40], the authors focus on learning two latent variables: semantic factors and variation factors. Semantic factors are invariant features, such as the shape of an object, which

remain unaffected by the background or environment. Variation factors, in contrast, depend on the domain’s prior knowledge and can change with different interventions, making them non-invariant across domains. The authors propose a method called **Causal Semantic Generative (CSG)** model based on these assumptions. In their model, x , y , s , and v represent input images, labels, semantic factors, and variation factors, respectively. The joint distribution in this causal model is given by:

$$p(x, y, s, v) = p(s, v) p(x | s, v) p(y | s) \quad (2.1)$$

According to the **ICM** principle (see Section 1.2.5), interventions on one causal mechanism do not affect others. Therefore, changes in domains due to interventions in the prior $p(s, v)$ do not influence other posteriors, ensuring that posterior distributions remain invariant across domains.

When latent factors encompass all changing factors, there is no need for manipulation variables to explicitly model distribution changes. This approach only requires one source domain distribution to train the model, unlike methods requiring multiple source domain distributions.

The paper discusses causal invariance implying inference invariance. While some methods use a shared feature extractor across domains to infer latent variables z from observed variables x , the inference rule according to Bayes’ rule is:

$$p(z | x) = p(z) p(x | z) \quad (2.2)$$

For a given x , only one specific z contributes positively to $p(x | z)$, with others contributing zero. This rule concentrates on this single z determined by $p(x | z)$. However, noisy causal mechanisms can cause multiple z values to generate the same x , making inference dependent on domain-specific priors ($p(z)$) and thus non-invariant.

Given the data distribution $p^*(x, y)$ in the training domain, fitting a **CSG** model $p := \langle p(s, v), p(x | s, v), p(y | s) \rangle$ by maximizing the likelihood $E_{P^*(x, y)}[\log p(x, y)]$ is challenging due to difficulties in estimating $p(x, y, s, v)$. Therefore, the authors use the **Evidence Lower BOund (ELBO)** for fitting the **CSG**:

$$L_{p, q(s, v | x, y)}(x, y) := E_{q(s, v | x, y)} \left[\log \frac{p(s, v, x, y)}{q(s, v | x, y)} \right] \quad (2.3)$$

ELBO drives $q(s, v | x, y)$ towards the posterior $p(s, v | x, y)$ and serves as a tight lower bound of $\log p(x, y)$. Since $q(s, v | x, y)$ does not estimate $p(y | x)$ directly, an auxiliary model $q(s, v, y | x)$ is used to approximate $p(s, v, y | x)$:

$$q(s, v | x, y) = \frac{q(s, v, y | x)}{q(y | x)} \quad (2.4)$$

where $q(y | x) := \int q(s, v, y | x) ds dv$. This auxiliary model simplifies to $q(s, v, y | x) = q(s, v | x)p(y | s)$. The final [ELBO](#) is:

$$\max_{p, q(s, v | x)} E_{p^*(x, y)} \left[\log q(y | x) + \frac{1}{q(y | x)} E_{q(s, v | x)} \left[p(y | s) \log \frac{p(s, v)p(x | s, v)}{q(s, v | x)} \right] \right] \quad (2.5)$$

where $q(y | x) := E_{q(s, v | x)}[p(y | s)]$. Expectations are estimated using Monte Carlo methods after applying the reparameterization trick.

A variant of their method, [CSG-ind](#), decouples the s and v factors, removing spurious correlations in the test domain. An additional model $q^\perp(s, v | x)$ is introduced, and $q(s, v | x)$ is formulated as:

$$q(s, v | x) = \frac{p(s, v)}{p^\perp(s, v)} \frac{p^\perp(x)}{p(x)} q^\perp(s, v | x) \quad (2.6)$$

ensuring that $q(s, v | x)$ achieves its target if and only if $q^\perp(s, v | x)$ does.

Another variant of their method, [CSG-DA](#), utilizes unsupervised test domain data. Following the [ICM](#) principle, only the new prior $\tilde{p}(s, v)$ is needed, and $p(x | s, v)$ and $p(y | s)$ are transferred to the test domain. The same trick as in [CSG-ind](#) is applied to avoid learning two different auxiliary models.

Similar to [\[40\]](#), [ICM](#) principle is integrated into [GCRL](#) to enhance model efficiency and facilitate effective transfer learning.

2.1.6 Backdoor Adjustment for Confounding Factors

In [\[69\]](#), the authors address the challenge of image segmentation using only image-level labels through the following steps:

- **Multi-label Classification:** They first train a multi-label classification model using only image-level labels.
- **Pseudo-mask Generation:** For each image, class-specific seed areas are inferred using classification activation map [\[75\]](#) to create pseudo-masks.
- **Segmentation Model Training:** These pseudo-masks are used as ground truths to train a semantic segmentation model.

The authors identify three main issues with pseudo-mask generation due to confounding factor *context*:

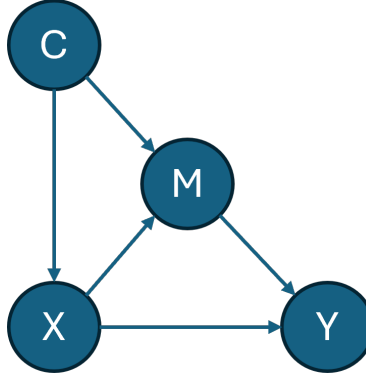


Figure 2.1: Causal graph in context adjustment [69]

- **Object Ambiguity:** Non-causal but correlated pixels may be misclassified (e.g., "floor" as "sofa").
- **Incomplete Background:** Context can lead to incorrect classifications of background areas.
- **Incomplete Foreground:** Context can misclassify foreground areas (e.g., "window" as "non-car").

To address these issues, they propose to model $p(y \mid do(x))$ instead of $p(y \mid x)$ to neutralize confounding effects where Y and X represent the image-level labels and images, respectively. This approach, called **CONText Adjustment (CONTA)**, uses the backdoor criterion and adjustment formula for inference [51].

They develop a **SCM** to reveal the **confounding factors**, as shown in Figure 2.1. In this model:

- M represents image-specific context templates.
- $X \rightarrow Y$ captures the direct causal effect, while $M \rightarrow Y$ accounts for mediation effects.

The adjustment formula is:

$$P(Y \mid do(X)) = \sum_c P(Y \mid X, M = f(X, c))P(c) \quad (2.7)$$

Since C (the **confounding factor**) is not observable, the authors use class-specific average masks to estimate c .

Similar to [69], backdoor adjustment is used in GCRL to eliminate confounding bias in predicted future trajectories.

2.1.7 Domain Adaptation by Selection Diagrams

In [46], the authors propose a method for causal domain adaptation with the following key points:

- **Problem Definition:** They define the causal domain adaptation problem as predicting the distribution of a target variable using measurements from other variables across one or more target domains, with data from one or more source domains. Each distribution in the domains corresponds to an external intervention in the system.
- **Handling Unknowns:** The approach deals with partially unknown causal graphs, intervention types, and intervention targets. It addresses latent confounders by selecting features that are invariant across domains and have maximum predictability (Markov Blanket features [66]).
- **Variables:** They have five types of variables:
 1. Context variables $\{C_i\}_{i \in \mathcal{I}}$ which act as domain identifiers. $\mathcal{I}$ is the domain of context variables indices. In this paper, the usage of context variables is analogous to selection variables in [50].
 2. Causal variables $\{X_j\}_{j \in \mathcal{J}}$. $\mathcal{J}$ is the domain of causal variables indices.
 3. Exogenous variables $\{E_k\}_{k \in \mathcal{K}}$ which are used to model latent causal variables and context variables. $\mathcal{K}$ is the domain of exogenous variables indices.
 4. Target variable Y whose value should be predicted.
 5. Variable V defined as $V = ((X_j)_{j \in \mathcal{J}}, (C_i)_{i \in \mathcal{I}})$.
- **Proposed Structural Causal Model:** The proposed SCM is defined as:

$$\mathcal{M} : \begin{cases} C_i = g_i(E_{pa_i \cap \mathcal{K}}), & i \in \mathcal{I}, \\ X_j = f_j(X_{pa_j \cap \mathcal{J}}, C_{pa_j \cap \mathcal{I}}, E_{pa_j \cap \mathcal{K}}), & j \in \mathcal{J}, \\ p(E) = \prod_{k \in \mathcal{K}} p(E_k). \end{cases} \quad (2.8)$$

This model is represented by an acyclic directed mixed graph, as shown in Figure 2.2. Directed edges indicate direct causation, and bidirected edges represent potential hidden confounders.

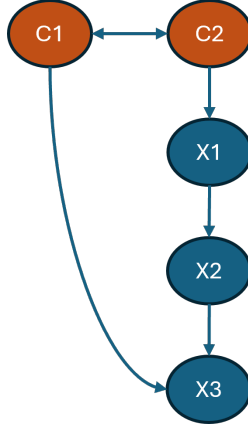


Figure 2.2: Acyclic directed mixed graph in [46]. The orange colored nodes indicate the context variables.

- **Assumptions:** $\mathcal{G}$ denotes the causal graph. The paper lists several assumptions:

1. **Joint Causal Inference Assumptions:**

- No causal variable directly causes any context variable (“exogeneity”):

$$(\forall j \in \mathcal{J}, \forall i \in \mathcal{I} : X_j \rightarrow C_i \notin \mathcal{G}) \quad (2.9)$$

- No causal variable is confounded with a context variable (“randomization”):

$$(\forall j \in \mathcal{J}, \forall i \in \mathcal{I} : X_j \leftrightarrow C_i \notin \mathcal{G}) \quad (2.10)$$

- Every pair of context variables is purely confounded (“genericity”):

$$(\forall i, i' \in \mathcal{I} : C_i \leftrightarrow C_{i'} \in \mathcal{G} \wedge C_i \rightarrow C_{i'} \notin \mathcal{G}) \quad (2.11)$$

2. The joint distribution of causal variables and context variables is Markov and faithful with respect to $\mathcal{G}(\mathcal{M})$. Markov implies [d-separation](#), [d-separation](#) implies conditional independence, and faithfulness implies no conditional independencies beyond those implied by [d-separation](#).
3. Conditional independencies involving Y in source domains hold in target domains.
4. Let C_1 be the source/target domains identifier. C_1 has no direct effect on Y , i.e.:

$$C_1 \rightarrow Y \notin \mathcal{G} \quad (2.12)$$

- **Invariant Features:** If $C_1 \perp\!\!\!\perp Y \mid A$, where $A \subseteq V$, the set A can be used as invariant features for training predictors in the target domain. The predictor $\hat{Y}_A^0$ trained on A in the source domain should have consistent performance in the target domain:

$$C_1 \perp\!\!\!\perp Y \mid A \Rightarrow \mathbb{E}((Y - \hat{Y}_A^0)^2 \mid C_1 = 1) = \mathbb{E}((Y - \hat{Y}_A^0)^2 \mid C_1 = 0) \quad (2.13)$$

- **Bias Analysis:** Using a separating set A can introduce *incomplete information bias* due to loss of relevant information not captured in A . The total bias is expressed as:

$$\hat{Y}_{V \setminus \{Y, C_1\}}^1 - \hat{Y}_A^0 = \underbrace{(\hat{Y}_A^1 - \hat{Y}_A^0)}_{\text{transfer bias}} + \underbrace{(\hat{Y}_{V \setminus \{Y, C_1\}}^1 - \hat{Y}_A^1)}_{\text{incomplete information bias}} \quad (2.14)$$

The transfer bias is minimized as per Equation 2.13.

- **Feature Selection:** To identify set A , they rank feature subsets based on source domain risks and test conditional independence as described in Equation 2.13, while imposing the listed assumptions.

Similarly, in [45], authors introduce an invariant causal representation learning method that leverages data from multiple environments. An environment variable, E , is introduced to account for factors that vary across different environments. Specifically, E serves as an index representing each environment. The goal is to predict the target variable Y using its invariant direct causes. The method is based on the following assumptions:

1. Each variable Z_i depends on either or both Y and E . The variables Z_i are categorized into two sets: Z_p , representing the causal factors of Y , and Z_c , representing non-causal factors. Here, "non-causal" implies that Z_c could either be an effect of Y , independent of Y , or spuriously correlated with Y due to a confounder.
2. The causal graph that includes Z and Y is a DAG.
3. $X \perp\!\!\!\perp Y, E \mid Z$, indicating that $p(X \mid Z)$ remains invariant across all environments, where X represents the observations.
4. $Y \perp\!\!\!\perp E \mid Z_p$, implying that $p(Y \mid Z_p)$ is invariant across all environments.
5. The distribution $p(Z \mid Y, E)$ belongs to a general exponential family, parameterized by an arbitrary function $\lambda(Y, E)$, and has sufficient statistics $T(Z) = [T_f(Z)^T, T_{NN}(Z)^T]^T$. This is a concatenation of: (a) the sufficient statistics $T_f(Z) = [T_1(Z_1)^T, \dots, T_n(Z_n)^T]^T$

of a factorized exponential family, where each $T_i(Z_i)$ has a dimension of at least 2, and (b) the output $T_{NN}(Z)$ from a neural network with ReLU activations. The resulting density function is expressed as $p_{T,\lambda}(Z | Y, E) = Q(Z)/C(Y, E) \exp[T(Z)^T \lambda(Y, E)]$, where Q is the base measure and C is the normalizing constant.

In summary, the first assumption implies that the method addresses covariate shift scenarios, where some causal or non-causal factors might change, but the conditional distributions do not. The method avoids using a factorized prior, meaning it does not condition on the parents of the latent variables but instead captures arbitrary dependencies between latents by learning $T_{NN}(Z)$.

In the first phase, the model parameters are learnt using a variational autoencoder with an [ELBO](#) loss function. Since the normalization constants are unknown, a score matching method is employed within the [ELBO](#) function. Parametric constraints are applied to ensure identifiability up to a permutation of the latent variables.

In the second phase, the method identifies the direct causes of Y . Initially, it employs the PC algorithm [\[66\]](#) to learn a [MEC](#) of the causal [DAG](#). Subsequently, it identifies the direct parents of Y through conditional independence tests.

In the third phase, an invariant predictor is learned using invariant risk minimization [\[2\]](#).

To infer the causal factors of Y in a new environment, the following loss function is optimized:

$$\max_{Z_p, Z_c} \log p(X | Z_p, Z_c) + \lambda_1 \|Z_p\|_2^2 + \lambda_2 \|Z_c\|_2^2 \quad (2.15)$$

Thus, during fine-tuning, the method adjusts the latent space rather than the network parameters. The reasoning behind this is that $p(X | Z)$ is presumed to be invariant across domains. The l_2 regularizers ensure that Z_p and Z_c maintain reasonable scales.

Similar to the environment variable E in [\[45\]](#), I have also used an external variable in [GCRL](#) to model domain variations.

2.1.8 GCRL for Causal Domain Adaptation

Inspired by the [CSG](#) method [\[40\]](#) and selection diagrams [\[50\]](#), I have developed a novel approach, [GCRL](#), to tackle the domain adaptation problem in [HTP](#). In the following sections, I will discuss the [confounding factors](#) associated with the [HTP](#) problem and apply the

backdoor criterion [51] to eliminate confounding bias, as demonstrated in [69]. The transfer assumption in GCRL aligns with the transferable parameters assumption highlighted in [61]. Given that HTP datasets are typically collected from multiple environments, I will utilize data from these various sources. Moreover, since the ground truth future trajectories in HTP are relatively easy to obtain [42], domain adaptation is more suitable for HTP compared to domain generalization [65].

2.2 Human Trajectory Prediction

This section reviews some of the state-of-the-art models in HTP which use causal methods to deal with the distribution shifts in HTP application.

2.2.1 Causality for Transfer Learning in Human Trajectory Prediction

Traditional motion forecasting models face two primary challenges: inefficiency in knowledge transfer and a lack of robustness under distribution shifts. These models typically struggle to generalize effectively when confronted with covariate shift. In [42], these challenges are addressed from a causal perspective by introducing three latent variables: invariant variables, style confounders, and spurious features. Based on these features, the authors propose a causal model for motion forecasting, informed by domain knowledge, and aim to learn these latent variables using Invariant Risk Minimization (IRM) [2]. In this context, invariant features correspond to physical laws, style confounders capture motion styles that vary by location (e.g., differences between an open sports field and an urban intersection), and spurious features represent those whose correlation with future motion varies significantly.

The model in [42] consists of two encoders. The first uses IRM to suppress spurious correlations and learn invariant features, while the second focuses on learning style-specific features. This architecture approximates the proposed causal graph and isolates the invariant subset of parameters, which can be transferred to other domains. The approach seeks to learn causally invariant representations through a structurally informed design based on domain knowledge.

The training data $\mathcal{D}$ is often collected from K environments, which can be utilized in IRM. Traditional approaches merge these environments into a single dataset, where the loss function is based on risk minimization:

$$R(p) = \frac{1}{|\mathcal{D}|} \sum_{(x,y) \in \mathcal{D}} L_{task}(p(x); y) \quad (2.16)$$

Here, p represents the model, x denotes past trajectories, y refers to future trajectories, and L_{task} can be a [Mean Squared Error \(MSE\)](#) or [Negative Log-Likelihood \(NLL\)](#) loss. However, the [IID](#) assumption often does not hold in practice. For instance, the test domain can be markedly different from training domains in widely used datasets like ETH-UCY [\[36, 52\]](#), leading to a lack of robustness in generalization.

The invariant encoder ϕ is designed to learn the invariant features Z , while the variant encoder ψ is responsible for learning style confounders C . The input to ϕ consists of past trajectories, which may lack sufficient information about motion styles, particularly before interactions. In contrast, ψ receives fully observed trajectories. The style modulator f extracts robust features from the variant style features to predict future trajectories.

The foundation of proposed method in [\[42\]](#) lies in the [SMS](#) principle, which posits that small distribution shifts tend to manifest locally within the causal factorization (see [Section 1.2.5](#)). Consequently, the modular architecture and causal factorization enable adjustments to only a small subset of parameters in a new domain.

The domain-invariant predictor, $\phi \circ g$, is trained using [IRM](#) [\[42\]](#):

$$\begin{aligned} \min_{\phi, g} \quad & \frac{1}{|\xi|} \sum_{e \in \xi} R^e(\phi, g) \\ \text{s.t.} \quad & g \in \arg \min_{g^*} R^e(\phi, g^*) \quad \forall e \in \xi \end{aligned} \quad (2.17)$$

where g^* is the optimal predictor for each domain, $R^e(\cdot)$ is the loss function calculated in environment e , and ξ represents the set of environments. Suppressing spurious features using [IRM](#) may inadvertently exclude motion styles that vary across environments. However, the modular design allows different parts of the model to be trained separately.

Similar to selection diagrams [\[50\]](#), the model in [\[42\]](#) can adapt between domains by learning exclusive motion styles for each environment. The style encoder features thus not only contain the necessary information for predicting future trajectories but also capture the relationships between styles across different environments. To facilitate this, a projection head h is employed atop the style encoder ψ , which is trained using supervised contrastive learning. Pairs of scene observations from the same environment are treated as positive samples, while those from different environments are treated as negative. This approach is advantageous as it imposes no assumptions on the number of environments,

which may be unknown in the test domain. Consequently, this component of the model can be applied without modification in the test domain.

Training the network occurs in three phases. First, the networks ϕ and g are trained using [IRM](#). Next, ψ and h are trained with a contrastive loss. Finally, g , ψ , h , and f are jointly trained using risk minimization and contrastive loss. [IRM](#) is excluded in the final phase to preserve style features. Under distribution shifts, only a small subset of parameters, such as f , requires fine-tuning [\[42\]](#).

2.2.2 Counterfactual Analysis in Human Trajectory Prediction

In [\[10\]](#), the elimination of confounding bias using perfect or surgical interventions is addressed within a causal model, as depicted in Figure 2.3, where E , X , and Y represent environmental interactions, past trajectories, and future trajectories, respectively.

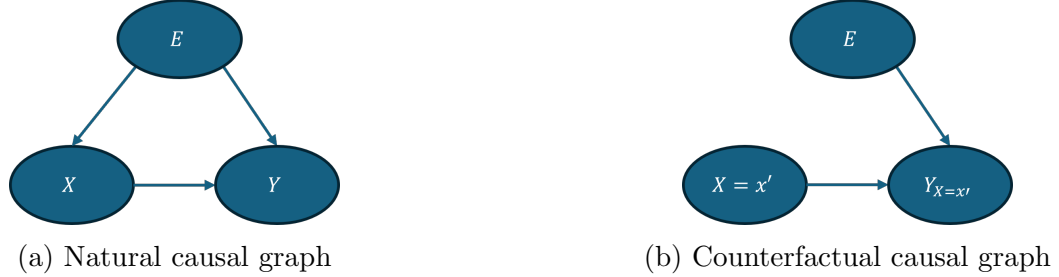


Figure 2.3: Causal graph for human trajectories in [\[10\]](#).

To apply these interventions, the value of the vector representing past trajectories is replaced with uniform rectilinear motion, the mean of trajectories, or a random value, thereby removing the backdoor path from Y to X as per causal inference principles outlined by [\[51\]](#). Consequently, the model output is given by [\[10\]](#):

$$Y_C = Y - Y_{X=x'} \quad (2.18)$$

Here, Y represents the variable for future trajectories as influenced by spontaneous changes in E and X , whereas $Y_{X=x'}$ represents future trajectories resulting from spontaneous changes in E and external intervention on X . The rationale is that in the counterfactual graph, any changes in Y stem solely from changes in E , while in the factual graph, changes in Y arise from both X and E . Thus, by subtracting these two outcomes, the confounding bias can be effectively eliminated.

2.2.3 Proposed Generative Causal Representation Learning for Human Trajectory Prediction

Robust [HTP](#) has been studied from a causality perspective only in [\[42, 10\]](#). Inspired by the results of these methods, I have proposed [GCRL](#) [\[3\]](#), which, in contrast to [\[42, 10\]](#), is a generative approach that better handles the multi-modality of trajectories. Furthermore, the causal graph proposed for [HTP](#) in [\[10\]](#) (See [Figure 2.3](#)) does not encompass all the factors involved in [HTP](#). Unlike the method in [\[42\]](#), which requires multiple training phases, [GCRL](#) can be trained end-to-end in a single phase. More details of [GCRL](#) will be given in [Chapter 3](#).

2.3 Causal Representation Learning

This section reviews key concepts, methods, and assumptions in [CRL](#), where the structure of the causal graph may be unknown. The goal is to learn both the causal variables and the underlying causal graph structure. Several of these assumptions are also incorporated into my second proposed method, [ICRL-SM](#).

2.3.1 Sparse Mechanism Shift Hypothesis in Causal Discovery

In [\[53\]](#), the [Mechanism Shift Score \(MSS\)](#), a tool designed to work with various empirical estimators is introduced. [MSS](#) is capable of identifying the entire causal structure in heterogeneous data, given that the [SMS](#) assumption is satisfied. By relaxing the [IID](#) assumption, [MSS](#) enables the learning of the [MEC](#) of the true causal model. The findings suggest that *sparse* changes in the data are crucial for accurately identifying the causal model.

Causal structures can be identified through both actively specified interventions and natural distribution shifts across multiple environments. Multi-environment methods can discern either the direct causes of a specific variable or the entire causal structure, though their effectiveness is contingent on the number and nature of changes observed.

The paper establishes the identifiability of bivariate causal structures under the [SMS](#) hypothesis without relying on parametric assumptions. It also demonstrates that true (multivariate) causal graphs minimize the [MSS](#).

The work assumes that different environments result from *soft interventions* [\[11\]](#), which do not disrupt the connection between a variable and its parents. A detailed discussion

on soft versus hard interventions will be provided in a subsequent section. Specifically, a causal graphical model can be described by a set of interventional distributions where some causal mechanisms are altered. Consequently, data from multiple environments can be represented as:

$$\tilde{p}_{\mathcal{Z}}^e(Z_1, \dots, Z_d) = \left(\prod_{j \in \mathcal{I}^e} \tilde{p}_{\mathcal{Z}}^e(Z_j \mid PA_j) \right) \prod_{j \notin \mathcal{I}^e} p_{\mathcal{Z}}^e(Z_j \mid PA_j), \quad (2.19)$$

where $\mathcal{I}^e$ denotes the set of intervention targets at environment e , Z_i are the causal variables, and PA_j are the parents of causal variable j .

In the bivariate case, identifiability is achieved if only one mechanism changes. In the multivariate case, any alteration in the mechanisms of the true causal graph G^* results in changes to any DAG over the same variables. Based on this theory, MSS is defined as:

$$MSS_j(G; \mathcal{P}) = \sum_{e' > e}^{n_e} [p_{\mathcal{Z}}^e(Z_j \mid PA_j^G) \neq p_{\mathcal{Z}}^{e'}(Z_j \mid PA_j^G)], \quad (2.20)$$

where $\mathcal{P} := \{p_{\mathcal{Z}}^e\}_{e=1}^{n_e}$, n_e is the number of environments, and PA_j^G represents the parents of causal variable j in the candidate DAG G . The total MSS is then given by:

$$MSS(G; \mathcal{P}) = \sum_{j=1}^d MSS_j(G; \mathcal{P}), \quad (2.21)$$

where $MSS(G; \mathcal{P})$ accounts for the total number of mechanism changes across all variables and pairs of environments according to the Markov factorization of G . Since any change in G^* will result in a change in G , it follows that:

$$MSS_j(G^*; \mathcal{P}) \leq MSS_j(G; \mathcal{P}), \quad (2.22)$$

and consequently:

$$MSS(G^*; \mathcal{P}) \leq MSS(G; \mathcal{P}). \quad (2.23)$$

This indicates that the minimizer of MSS is the true causal DAG. By minimizing MSS, one can asymptotically determine the true causal DAG, with the probability of identifying the true causal DAG increasing with the number of environments. The results also suggest that as mechanisms become denser, the performance in finding the true causal DAG deteriorates quickly. Hence, sparsity in mechanism changes is a critical factor. MSS can thus serve as an inductive bias for discovering the true causal DAG.

2.3.2 Explicit Models and Acyclicity Constraint

In [68], the authors propose DAG-GNN, a model that aims to learn a faithful DAG using a Variational AutoEncoder (VAE). The cost function for this model is the ELBO. DAG-GNN is a score-based method [66], which means it learns a weighted adjacency matrix while imposing combinatorial constraints to ensure the graph remains acyclic. Unlike some methods where the adjacency matrix is a latent structure, DAG-GNN treats it as an explicit parameter, learning it alongside other network parameters. The VAE framework naturally accommodates both continuous and discrete data types, making it versatile. The paper introduces an acyclicity constraint that is particularly suitable for deep learning models.

A linear SCM can be expressed as:

$$Z = A^T Z + E \quad (2.24)$$

where $A^T Z$ corresponds to the parents of Z , and E represents noise.

This can be rewritten as:

$$Z = (I - A^T)^{-1} E = f_A(E), \quad (2.25)$$

where $f_A(E)$ is a parameterized graph neural network that takes E as input and returns Z as a high-level representation. Due to the structure of Equation 2.25, the network $f_A(E)$ can be further expressed as:

$$Z = f_2((I - A^T)^{-1} f_1(E)), \quad (2.26)$$

where f_2 and f_1 are nonlinear transformations. If f_2 were invertible, Equation 2.26 would become:

$$f_2^{-1}(Z) = A^T f_2^{-1}(Z) + f_1(E), \quad (2.27)$$

which generalizes the linear SCM, enabling the proposed method to also learn nonlinear SCMs. The network is trained by maximizing the log evidence given the distribution of E and samples $Z^1, \dots, Z^n$:

$$\frac{1}{n} \sum_{k=1}^n \log p(Z^k) = \frac{1}{n} \sum_{k=1}^n \log \int p(Z^k | E) p(E) dE, \quad (2.28)$$

which is intractable. Therefore, variational Bayes is employed:

$$L_{ELBO} = \frac{1}{n} \sum_{k=1}^n L_{ELBO}^k, \quad (2.29)$$

where

$$L_{ELBO}^k = -D_{KL}(q(E | Z^k) \| p(E)) + \mathbb{E}_{q(E|Z^k)}[\log p(Z^k | E)]. \quad (2.30)$$

The corresponding encoder for this VAE is:

$$E = f_4((I - A^T)f_3(Z)), \quad (2.31)$$

where f_3 and f_4 are nonlinear transformations that conceptually play the inverse role of f_2 and f_1 .

For simplicity, the prior is typically modeled as the standard matrix normal distribution $p(E) = \mathcal{MN}_{m \times d}(0, I, I)$. The authors assume there are m variables, each with dimension d , which is usually set to $d = 1$ in the literature. The functions f_4 and f_1 are considered identity transformations, while f_2 and f_3 are [MultiLayer Perceptron \(MLP\)](#)s.

The likelihood $p(Z | E)$ is a factored Gaussian with mean $M_Z \in \mathbb{R}^{m \times d}$ and standard deviation $S_Z \in \mathbb{R}^{m \times d}$, computed from the decoder.

The constraint for the acyclicity of the [DAG](#) is modified such that the optimization problem becomes:

$$\begin{aligned} \min_{A, \theta} f(A, \theta) &= -L_{ELBO} \\ \text{s.t. } h(A) &= \text{tr}[(I + \alpha A \circ A)^m] - m = 0, \end{aligned} \quad (2.32)$$

where the Augmented Lagrangian method is used to solve this optimization problem.

2.3.3 Causal Representation Learning by Interventional Data

In [\[29\]](#), the authors aim to learn the causal structure from both observational data and unknown interventions. However, as stated in the paper, there is no theoretical guarantee that the structure can be learned in this setting.

The paper is based on the following assumptions:

1. Data is discrete-valued.
2. Data is fully observed.
3. Interventions are sparse.
4. Interventions are soft.
5. Interventions do not stack.

6. No control over interventions.

The proposed method is score-based [66], iterative, and employs continuous optimization to learn the structural parameters γ of the DAG and the functional parameters θ of a set of independent causal mechanisms. Since these two sets of parameters are interdependent, they are trained jointly.

The structural parameters $\gamma \in \mathbb{R}^{M \times M}$ are represented as an adjacency matrix where $\sigma(\gamma_{ij})$ denotes the probability that random variable Z_j is a direct cause of Z_i , with $\sigma(\cdot)$ being the sigmoid function and M representing the number of causal variables.

The functional parameters θ include the parameters θ_i , which model the conditional distribution of Z_i given its parent set pa_i^G . Here, $G \sim \text{Ber}(\sigma(\gamma))$ represents a hypothesized configuration of the DAG.

Training of the network is conducted in three phases:

- **Phase 1:** Configurations of the graph $G_{ij} \sim \text{Ber}(\sigma(\gamma))$ and batches of observational data are sampled. The log-likelihood of the batch under that configuration is then maximized. The use of sampling from the Bernoulli distribution is analogous to dropout.
- **Phase 2:** Multiple graph configurations are sampled based on current edge beliefs, and interventional data samples are collected. The average log-likelihood of each variable Z_i across samples is calculated using the functional parameters fine-tuned in Phase 1. The variable with the greatest deterioration in log-likelihood is identified as the target variable. The loss from interventional data is computed similarly to Phase 1, except the contribution to the total loss from the intervened variable Z_i is ignored and not backpropagated.
- **Phase 3:** The gradient is calculated for backpropagation to update γ . The acyclicity constraint, which penalizes 2-length cycles in the learned adjacency matrix, is given by:

$$J_{DAG}(\gamma) = \sum_{i \neq j} \cosh(\sigma(\gamma_{ij})\sigma(\gamma_{ji})), \quad \forall i, j \in \{0, \dots, M-1\} \quad (2.33)$$

2.3.4 Sparsity as an Inductive Bias

In [34], sparsity is used as an inductive bias in learning the causal model, assuming that the data is explained by unobserved and meaningful factors that are mixed or transformed

by a decoder to generate the observed data. The challenge of disentanglement is then to recover these latent factors from the observations.

The paper introduces an auxiliary variable A , by conditioning on which the latent factors become independent. This auxiliary variable can be an index of time or environment, previous observations in temporal data, or an action in an interactive environment. The authors propose a method to describe a causal model that relates the latent factors and their connection to A (when available). The learned causal graphical model can subsequently be used to perform interventions. These interventions enable the agent to imagine situations that it has never encountered before, which would not be feasible without a disentangled representation.

The observed variables are the d_x -dimensional random variables $\{X^t\}_{t=1}^T$ and d_a -dimensional auxiliary variables $\{A^t\}_{t=0}^{T-1}$ at different timesteps t . The observations $\{X^t\}$ are assumed to be generated by a sequence of d_z -dimensional latent vectors $\{Z^t\}_{t=1}^T$ via the equation $X^t = f(Z^t) + N^t$, where $N^t \sim \mathcal{N}(0, \sigma^2 I)$ are mutually independent across time and independent of all A^t and Z^t :

$$p(z^t \mid z^{<t}, a^{<t}) = \prod_{i=1}^{d_z} p(z_i^t \mid z^{<t}, a^{<t}) \quad (2.34)$$

It is assumed that $d_z \leq d_x$ and f is a diffeomorphism. Additionally, it is assumed that Z_i^t are mutually independent given $Z^{<t}$ and $A^{<t}$.

The theory proposed in this paper is applicable to the exponential family, which has the following form:

$$p(z_i^t \mid z^{<t}, a^{<t}) = h_i(z_i^t) \exp \{T_i(z_i^t)^\top \lambda_i(G_i^z \odot z^{<t}, G_i^a \odot a^{<t}) - \psi_i(z^{<t}, a^{<t})\} \quad (2.35)$$

In the Gaussian case, the sufficient statistic is $T_i(z) := (z, z^2)$, the base measure is $h_i(z) := \frac{1}{\sqrt{2\pi}}$, and the natural parameters output by $\lambda_i(G_i^z \odot z^{<t}, G_i^a \odot a^{<t})$ are $(\lambda_1, \lambda_2) = (\frac{\mu}{\sigma^2}, -\frac{1}{\sigma^2})$. The remaining term $\psi_i(z^{<t}, a^{<t})$ serves as a normalization constant. The binary vectors $G_i^z \in \{0, 1\}^{d_z}$ and $G_i^a \in \{0, 1\}^{d_a}$ act as masks for selecting the parents of z_i^t . Furthermore, the adjacency matrix of the graph is $G := [G^z, G^a]$, where $G^z := [G_1^z, \dots, G_{d_z}^z]^\top \in \mathbb{R}^{d_z \times d_z}$ and $G^a := [G_1^a, \dots, G_{d_a}^a]^\top \in \mathbb{R}^{d_z \times d_a}$. We define $\lambda(z^{<t}, a^{<t}) \in \mathbb{R}^{kd_z}$ as the concatenation of all $\lambda_i(G_i^z \odot z^{<t}, G_i^a \odot a^{<t})$ and similarly define $T(z^t) \in \mathbb{R}^{kd_z}$. Consequently, the learnable parameters are $\theta := (f, \lambda, G)$, which induce a conditional probability distribution $P(X^{\leq T} \mid A^{<T} = a; \theta)$.

The paper imposes specific conditions on the natural parameters, sufficient statistics, and the graph to ensure disentanglement. One important special case occurs when A^{t-1} corresponds to a one-hot vector indexing an intervention with unknown targets. In this scenario, the sparsity of G^a is directly related to the SMS hypothesis, which states that only a few mechanisms change at a time. The theorem presented in this paper provides the exact conditions under which the SMS hypothesis can induce disentanglement.

The authors use an approximate posterior of the form $q(z^{\leq T} | x^{\leq T}, a^{< T}) := \prod_{t=1}^T q(z^t | x^t)$ and optimize the VAE using the ELBO function. To ensure sparsity, they also add the Frobenius norm of the G^z and G^a matrices as a regularizer to the objective function.

2.3.5 Supervised Causal Representation Learning

In [64], a new framework for learning causal representations is presented, arguing against other works that claim data are generated by independent factors which can be learned via Kullback–Leibler Divergence (KLD) regularization and have a factorized prior. Instead, they propose a causal layer in a VAE that takes independent exogenous variables as input and maps them into a space where the variables are causally related. The dependent variables are then taken by the decoder to reconstruct the inputs. The causal layer implements the following SCM:

$$z_i = g_i(A_i \circ z; \eta_i) + e_i, \quad e \sim \mathcal{N}(0, I), \quad (2.36)$$

$$x = f(z) + \xi, \quad (2.37)$$

$$e = h(x, u) + \zeta, \quad (2.38)$$

where $z \in \mathbb{R}^n$ are the dependent latent factors, η_i are the parameters of g_i , x are the observations, ζ, ξ are independent noise vectors, and $e \in \mathbb{R}^n$ are the independent exogenous variables. CausalVAE [64] uses observational data and auxiliary information u as a supervisory signal, which contains the labels of the concepts (generative factors). These labels are utilized to propose a conditional prior $p(z|u)$ and to learn the adjacency matrix A . The adjacency matrix is $A = [A_1, \dots, A_n]$, where $A_i \in \mathbb{R}^n$ is a weight vector that determines the causal strength from other variables to Z_i .

The ELBO function in CausalVAE is defined as:

$$\begin{aligned}
\mathbb{E}_{p_{\mathcal{X}}}[\log p_{\theta}(x|u)] &\leq \\
\text{ELBO} &= \mathbb{E}_{p_{\mathcal{X}}} \left[\mathbb{E}_{q_{\phi}(z,e|x,u)} [\log p_{\theta}(x|z)] \right. \\
&\quad - \text{KLD}(q_{\phi}(e|x,u) \parallel p_{\mathcal{E}}(e)) \\
&\quad \left. - \text{KLD}(q_{\phi}(z|x,u) \parallel p_{\theta}(z|u)) \right]
\end{aligned} \tag{2.39}$$

where $p_{\theta}(z|u) = \prod_{i=1}^n p_{\theta}(z_i|u_i)$, $p_{\theta}(z_i|u_i) = \mathcal{N}(\lambda_1(u_i), \lambda_2^2(u_i))$, and $p_{\mathcal{E}}(e) = \mathcal{N}(0, I)$. To guarantee the identifiability of the causal graph A , they add two additional constraints to the loss function defined as follows:

$$l_u = \mathbb{E}_{p_{\mathcal{X}}} \|u - \sigma(A^{\top}u)\|_2^2 \leq \kappa_1, \tag{2.40}$$

where σ is a logistic function (as u are binary labels) and κ_1 is a small positive constant. The intuition behind this constraint is that A should also be able to explain the causal relations between u . The second constraint is:

$$l_m = \mathbb{E}_{z \sim q_{\phi}} \sum_{i=1}^n \|z_i - g_i(A_i \circ z; \eta_i)\|^2 \leq \kappa_2. \tag{2.41}$$

Similar to l_u , this constraint is also responsible for explaining the causal relations between z . The final constraint is an acyclicity constraint, as in [68], to ensure the acyclicity of the DAG. The final loss function is obtained by using Lagrangian multipliers.

2.3.6 Implicit Latent Causal Models

In [6], a theorem where only pairs of samples before and after random unknown interventions are required is proposed, eliminating the need for labels of intervention targets or active control over interventions. They introduce **Implicit Latent Causal Model (ILCM)**s, which can represent causal structure and variables without requiring an explicit graph representation, unlike the **Explicit Latent Causal Model (ELCM)**s in [68, 64, 34, 29, 20]. This paper requires diffeomorphic causal mechanisms so that exogenous variables can be uniquely inferred from causal variables. They also consider hard interventions, in which the parents of the intervened variable are removed.

The paper presents a theory that guarantees identifiability up to relabeling and element-wise transformation of causal variables based on certain assumptions about the data generation process and the functional form of decoders and causal mechanisms.

ILCMs are **VAEs** with a causally structured prior, similar to **ELCMs**. However, the key difference is that in **ILCMs**, the graph structure is represented through the neural solution function $s(e)$, where e represents the exogenous variables.

The latent variables in **ILCMs** are the exogenous variables E rather than causal variables Z . Therefore, the prior distribution of latents in **ILCMs** is given as $p(e, \tilde{e}, I)$, which encodes the prior distribution of intervention targets $p(I)$ and pre-intervention exogenous variables $p(e)$. The post-intervention exogenous variables prior is given as:

$$p(\tilde{e}|e, I) = \prod_{i \notin I} \delta(\tilde{e}_i - e_i) \prod_{i \in I} p(\tilde{e}_i|e) = \prod_{i \notin I} \delta(\tilde{e}_i - e_i) \prod_{i \in I} \tilde{p}(\tilde{z}_i) \left| \frac{\partial \tilde{z}_i}{\partial \tilde{e}_i} \right|, \quad \tilde{z}_i = \tilde{s}_i(\tilde{e}_i; e_{\setminus i}) \quad (2.42)$$

In the second equality, $p(\tilde{e}_i|e)$ is modeled with normalizing flows. This prior has three properties that make it equivalent to a unique **ELCM**:

1. The exogenous variables are independent of each other.
2. Upon intervening on variable i , the post-intervention causal variable $\tilde{z}_i$ is independent of all exogenous variables except $\tilde{e}_i$.
3. For all other $j \notin I$, $\tilde{e}_j = e_j$.

Additional details on how they implement the infer the intervention targets can be found in [6]. Finally, they use **ELBO** to train the network.

2.3.7 Causal Representation Learning in Time-Series Data

CITRIS [39], is a **VAE**-based framework designed to identify multi-dimensional causal factors from temporal sequences of images, where interventions with known targets may have been performed. This framework relies on both observational data and interventional data with known targets.

The data is assumed to be generated by a dynamic Bayesian network that is first-order Markov, stationary, and free from instantaneous effects. This implies that the parents of each causal factor at time t are the causal factors at time $t - 1$, including its own value at time $t - 1$. Additionally, the structure of the graph is time-invariant, i.e., $\text{PA}(C_i^1) = \text{PA}(C_i^t)$, where $\text{PA}(C_i^t)$ represents the parents of C_i at time t .

CITRIS uses intervention vectors I_i^t , indicating whether causal factor i at time step t has been intervened. These intervention vectors might be confounded by other interventions, represented by an unobserved regime variable R^t . Consequently, an augmented DAG $G = (V', E')$ is constructed, where:

$$V' = \{\{C_i^t\}_{i=1}^K \cup \{I_i^t\}_{i=1}^K \cup R^t\}_{t=1}^T \quad E' = \{\{\text{PA}_G(C_i^t) \rightarrow C_i^t\}_{i=1}^K \cup \{I_i^t \rightarrow C_i^t\}_{i=1}^K \cup \{R^t \rightarrow I_i^t\}_{i=1}^K\}_{t=1}^T \quad (2.43)$$

CITRIS considers soft interventions, where the conditional distributions change, i.e., $p(C_i^t | \text{PA}_G(C_i^t), I_i^t = 1) \neq p(C_i^t | \text{PA}_G(C_i^t), I_i^t = 0)$.

The framework allows the causal factors to be multi-dimensional, so the causal factors space is $\mathcal{C} = \mathcal{D}_1^{M_1} \times \mathcal{D}_2^{M_2} \times \dots \times \mathcal{D}_K^{M_K}$, where $\mathcal{D}$ is $\mathbb{R}$ for continuous variables and $\mathbb{Z}$ for discrete variables. CITRIS assumes a bijective decoder $h(C_1^t, C_2^t, \dots, C_K^t, E_o^t) = X^t$, where E_o^t are noise variables independent of the causal factors, but which influence observations. Hence, the joint dimension of causal factors and noise is limited to the image size.

Since interventions might leave some dimensions of the causal factors unaffected, CITRIS defines an invertible mapping $s_i : \mathcal{D}_i^{M_i} \rightarrow \mathcal{D}_i^{\text{var}} \times \mathcal{D}_i^{\text{inv}}$, which splits the domain $\mathcal{D}_i^{M_i}$ of the causal factor C_i into a part that is invariant to interventions and a part that varies with interventions. To qualify as a split, s must satisfy two conditions:

1. $s_i^{\text{inv}}(C_i^t) \perp I_i^t | \text{PA}(C_i^t)$
2. $s_i^{\text{inv}}(C_i^t) \perp s_i^{\text{var}}(C_i^t) | \text{PA}(C_i^t), I_i^t$

Based on this mapping, minimal causal variables are defined, ensuring that only information truly dependent on interventions is represented by $s_i^{\text{var}}(C_i)$.

CITRIS learns an invertible mapping from observations to latent space, $g_\theta : \mathcal{X} \rightarrow \mathcal{Z}$, and an assignment function $\psi : [1 \dots M] \rightarrow [0 \dots K]$, which assigns each dimension of Z to a causal factor. To enforce disentanglement, the prior of latent variables is conditioned on only one intervention target:

$$p_\phi(z^{t+1} | z^t, I^{t+1}) = \prod_{i=0}^K p_\phi(z_{\psi_i}^{t+1} | z^t, I^{t+1})$$

where z_{ψ_i} indicates the latent variables assigned to causal factor i , and $I_0^{t+1} = 0$.

The objective function in CITRIS is:

$$L_{\text{ELBO}} = -\mathbb{E}_{z^{t+1}} [\log p_\theta(x^{t+1} | z^{t+1})] + \mathbb{E}_{z^t, \psi} \left[\sum_{i=0}^K \text{KLD}(q_\theta(z_{\psi_i}^{t+1} | x^{t+1}) || p_\phi(z_{\psi_i}^{t+1} | z^t, I_i^{t+1})) \right]$$

2.4 Proposed Implicit Causal Representation Learning via Switchable Mechanisms in Causal Representation Learning

As highlighted in [6], [ELCMs](#) such as [64, 68, 20, 29, 1, 59, 62, 37, 70], which rely on an explicit parameterization for learning the causal graph—specifically, the adjacency matrix—are prone to getting stuck in local minima. This is because learning the causal graph without knowing the causal variables is challenging, and learning the causal variables without knowing the causal graph is equally difficult.

Inspired by the results in [6], I proposed a novel implicit method, [ICRL-SM](#) [4], for causal representation learning, drawing on the concepts of augmented networks [49] and switch variables [56]. However, implicit models are still in their infancy, and in this work, I aim to advance [ILCMs](#) by relaxing one of the strict assumptions in [6]—specifically, the requirement for hard interventions. The proposed method introduces a theorem for identifiability through soft interventions and demonstrates superior performance compared to existing explicit and implicit methods on both real-world and synthetic datasets. More details of [ICRL-SM](#) is given in Chapter 4.

Chapter 3

Generative Causal Representation Learning

As demonstrated in [10, 42], real-world datasets in [HTP](#) often consist of multi-environment data, where observations are collected from a range of settings, each with distinct characteristics. This challenge is not unique to [HTP](#), as many real-world applications, such as those in autonomous driving, healthcare, and finance, involve data collected from multiple environments or domains. These domains can exhibit distribution shifts, where the underlying data distribution varies across environments, complicating the development of generalized models that perform well across all domains.

To address this challenge, this chapter introduces Generative Causal Representation Learning ([GCRL](#)), a novel approach designed to effectively handle domain shifts and interventions in real-world applications by leveraging a causal formulation. Specifically, [GCRL](#) separates the learned representations into two key components: invariant representations, which capture aspects of the data that remain consistent across domains, and variant representations, which model domain-specific variations. This distinction enables more robust and transferable models, as the invariant components facilitate generalization, while the variant components account for domain-specific nuances.

Through experimental results on both real-world [HTP](#) datasets and synthetic datasets, the proposed method demonstrates superior robustness to domain shifts. Moreover, it significantly enhances the efficiency of models for transfer learning, making them more adaptable to new, unseen domains and capable of maintaining high performance even when the underlying data distribution changes.

3.1 Proposed Method

In this section, I will review an essential component of GCRL, namely selection variables, in greater detail. Additionally, the formalism of the HTP problem will be introduced, along with the proposed causal structure for HTP, the derivation of the novel loss function, and the proposed method for domain adaptation.

3.1.1 Transportability

In the proposed method, GCRL [3], I am going to leverage concepts in causality which are explained in previous sections and the selection variables first introduced in [50] for transfer learning. Depending on the graph and assumptions integrated in it, the transport formula by which we transport the knowledge about causal effects from source domain to target domain, would vary. Selection variables S are introduced in [50] which represent the variations in each domain. Each member of S corresponds to a mechanism by which the two populations differ. Therefore, for switching between domains we can condition on the values of S .

$$p^*(y \mid do(x)) = p(y \mid do(x), s)$$

The resulting augmented graphs are called selection diagrams. Selection diagrams include every edge in the main graph in addition to the edges $S_i \rightarrow V_i$ where V_i are the variables that can vary across domains. An example selection diagram is shown in Figure 3.1 where E is the selection variable and S is the variable that varies across domains. To illustrate how selection variables can alleviate transfer learning, let us revisit a theorem from [50]:

Theorem 3.1.1. *“Let D be the selection diagram characterizing two populations, Π and Π^* , and S a set of selection variables in D . The relation $R = p^*(y \mid do(x), z)$ is transportable from Π to Π^* if the expression $p(y \mid do(x), z, s)$ is reducible, using the rules of do-calculus, to an expression in which S appears only as a conditioning variable in do-free terms.”*

The reason for this is that every relation that satisfies this property can be written as combination of terms that involve S and terms that do not involve S . The former is estimable from the observations in the target domain and the latter are experimentally identifiable from source domains.

Theorem 3.1.2 [50] gives us the condition to transport a model to another domain using S-admissible variables.

Theorem 3.1.2. “Let D be the selection diagram characterizing two populations, Π and Π^* , and S the set of selection variables in D . The strata-specific causal effect $p^*(y \mid do(x), z)$ is transportable from Π and Π^* if Z d-separates Y from S in the X -manipulated version of D , that is, Z satisfies $(Y \perp\!\!\!\perp S \mid Z, X)_{D_{\bar{X}}}$.”

The set of variables Z that satisfies this property is called S -admissible.

3.1.2 Formalism of Motion Forecasting

Let’s consider a multi-agent environment that contains M agents. Let’s denote the state of agents as $s_t = \{s_t^1, s_t^2, \dots, s_t^M\}$ where $s_t^i = (h_t^i, v_t^i)$ is the 2D coordinates of agent i in the scene. The objective is to predict the future trajectories $y = \{s_{T_{obs}+1}, s_{T_{obs}+2}, \dots, s_{T_{obs}+pred}\}$ given the observed trajectories $x = \{s_1, s_2, \dots, s_{T_{obs}}\}$ for all agents in the scene.

Most of deep-learning based [HTP](#) models are consisted of an encoder, an interaction module, and a decoder [\[33\]](#). Usually, the encoder and interaction module are implemented as a single model. First, I feed the observed timeseries data X into the encoder. Then, I feed the representations of the encoder to the interaction module to obtain an interaction vector which contains information about the human-human or human-space interactions. Finally, I feed the interaction vectors to the decoder of the network to predict the future trajectories Y .

Motion forecasting datasets are usually collected in different environments such as pedestrians walking in an intersection, pedestrians walking in front of a store, pedestrians walking in a university campus, and etc. Therefore, as in the widely used ETH-UCY benchmark dataset, the test domain is significantly different than the training domains [\[10\]](#). Consequently, the [IID](#) assumption is violated and [Empirical Risk Minimization \(ERM\)](#) techniques cannot be used to train the networks.

Selection variables can be useful here to model the changing factors of each environment. From now on, I will refer to the selection variables as environment variables in the context of motion forecasting. Let’s define environment vectors as $e \in \mathbb{R}^n$ where each dimension of e models a changing factor. Subsequently, the set of environments in the dataset can be defined as $\{e_1, e_2, \dots, e_K\}$ such that conditioning on $E = e_1$ will correspond to the samples of the first environment. E will be the environment variable of my proposed causal model.

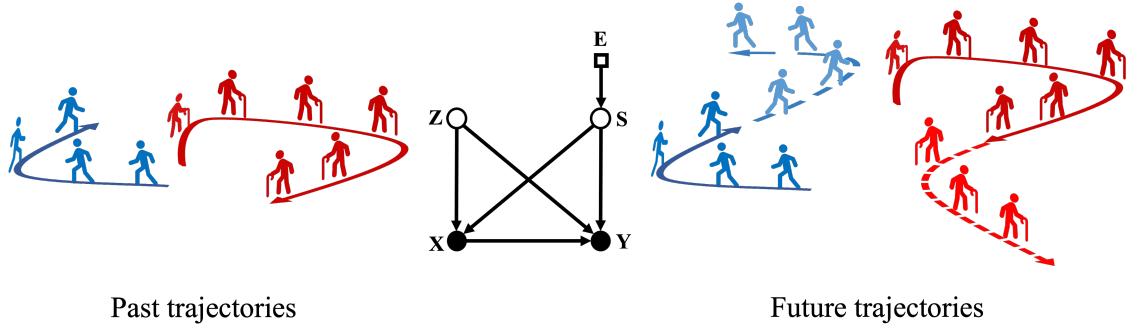


Figure 3.1: Proposed causal model for motion forecasting. X , Y , Z , and S represent the past trajectories, future trajectories, invariant features, and variant features, respectively. E is the environment variable which models the changing factors in each environment. The filled circles correspond to the observed variables and the rest are the latent variables.

3.1.3 Causal Formalism

Based on the domain knowledge, I propose the causal model shown in Figure 3.1. There are two types of variables in the proposed causal model:

- The endogenous variables X , Y , and S
- The exogenous variables E , Z

S represents the variant features which are specific to each environment. This variable can correspond to the motion styles of the pedestrians in each environment. For example, the motion style of pedestrians in an intersection is different than university campus. Obviously, the individuals characteristics and the environmental factors can affect the motions styles of pedestrians in a certain environment. I model all these changing factors in the environment variable E . There is an abstraction level in all of the causal models, and here I am assuming that the abstraction level is the environment variable E . In other words, the individuals characteristics and the environmental factors also have certain causes but I abstract them in E . Z represents the invariant features or the features that are common across domains. These features can correspond to the physical laws and social norms which can affect the walking distance or walking patterns of the pedestrians. X and Y represent the observed and future trajectories, respectively. In the datasets, X and Y are observed and the rest are latent variables.

Note that this observation is context-dependent, as it varies between training and testing phases. Specifically, during training, the target variable Y is observed because the

model is learning from both the past (represented by X) and the future (represented by Y) trajectories. This aligns with common practice in the literature, where models are trained on fully observed data. However, during testing, the future trajectories are not observed. The testing data is composed of the past trajectories only, and the model is tasked with predicting the future trajectories. This segmentation of the trajectory into past and future segments is consistent with temporal prediction models, where the goal is to forecast unseen future data based on past observations.

The rationale behind the edges in the causal graph are as follows:

- There should be an edge from X to Y because the last position of a pedestrian can affect where the pedestrian is going to be in the next time step.
- There should be an edge from S and Z to X and Y because the motion style in each environment and social norms can affect the pedestrians trajectories. Furthermore, since X and Y have same nature, they should both be affected by S and Z .
- There should be an edge from E to S as E is supposed to represent the changing factors in each environment and S is the variant representation.

3.1.4 Generative Causal Representation Learning

As shown in Figure 3.1, S and Z confound X and Y . Therefore, as common in motion forecasting models, predicting Y from X i.e., modeling $p(Y | X)$, will be erroneous. On the other hand, $p(Y | do(X))$ will eliminate the confounding bias. Furthermore, since there are multiple environments, we should calculate the causal effect of X on Y conditioned on E i.e., $p(Y | do(X), E)$.

$$\begin{aligned}
p(Y | do(X), E = e_i) &= \int p(Y | X, S, Z, E) p(S | E) p(Z | E) ds dz \\
&= \int p(Y | X, S, Z) p(S | E) p(Z) ds dz \\
&= \mathbb{E}_{p(S|E), p(Z)} [p(Y | X, S, Z)],
\end{aligned} \tag{3.1}$$

The first equality is the result of backdoor criterion [51]. The second equality follows from the fact that X and Y are colliders, hence, Z will be independent of E , unconditionally. The third equality follows from the Monte-Carlo integration. Equation 3.1 instructs us that in order to calculate the causal effect of X on Y we should stratify over S and Z . Obviously, $p(Y | do(X))$ is different than the trivial $p(Y | X)$. However, there is no

information about S and Z , let alone stratifying over them. Therefore, I am going to learn them via mean-field variational inference.

Normally, the model can be trained using the following Maximum Likelihood estimation:

$$\max_p \mathbb{E}_{p^*(x,y)} [\log p(x,y)] \quad (3.2)$$

where $p^*(x,y)$ is the true distribution of the dataset. Calculating $p(x,y)$ is intractable since $p(x,y) = \sum_{e \in E} \int (p(x,y,s,z,e) ds dz)$. Therefore, I use the well-known [ELBO](#) to maximize the lower bound of $\log p(x,y)$ instead of maximizing itself.

$$\max_{p,q} \mathbb{E}_{p^*(x,y)} \left[\mathbb{E}_{q(s,z,e|x,y)} \left[\log \frac{p(x,y,s,z,e)}{q(s,z,e|x,y)} \right] \right] \quad (3.3)$$

Theoretically, the [ELBO](#) function in Equation 3.3 will drive $q(s,z,e|x,y)$ towards the true posterior distribution $q(s,z,e|x,y)$ and eventually the [ELBO](#) function will become Equation 3.2. However, this [ELBO](#) function is still intractable because of condition y in $q(s,z,e|x,y)$. A model that approximates the posterior of latents from past trajectories alone is preferred. In order to achieve this, the Equation 3.3 should be rephrased.

$$\begin{aligned}
& \max_{p,q} \mathbb{E}_{p^*(x,y)} \left[\mathbb{E}_{q(s,z,e|x,y)} \left[\log \frac{p(x,y,s,z,e)}{q(s,z,e|x,y)} \right] \right] \\
&= \max_{p,q} \mathbb{E}_{p^*(x,y)} \left[\mathbb{E}_{q(s,z,e|x,y)} \left[\log \frac{p(x,y,s,z,e)q(y|x)}{q(s,z,e,y|x)} \right] \right] \\
&= \max_{p,q} \mathbb{E}_{p^*(x,y)} \left[\int \left[\log \frac{p(x,y,s,z,e)q(y|x)}{q(s,z,e,y|x)} \right] q(s,z,e|x,y) ds dz de \right] \\
&= \max_{p,q} \mathbb{E}_{p^*(x,y)} \left[\int \left[\log \frac{p(x,y,s,z,e)}{q(s,z,e,y|x)} \right] q(s,z,e|x,y) ds dz de + \log q(y|x) \right] \\
&= \max_{p,q} \mathbb{E}_{p^*(x,y)} \left[\int \left[\frac{1}{q(y|x)} \log \frac{p(x,y,s,z,e)}{q(s,z,e,y|x)} \right] q(s,z,e,y|x) ds dz de + \log q(y|x) \right] \\
&= \max_{p,q} \int \left[\frac{1}{q(y|x)} \log \frac{p(x,y,s,z,e)}{q(s,z,e,y|x)} \right] p^*(x,y) q(s,z,e,y|x) ds dz de dx dy \\
&\quad + \int [\log q(y|x)] p^*(x,y) dx dy \tag{3.4} \\
&= \max_{p,q} \int \left[\frac{p^*(y|x)}{q(y|x)} \log \frac{p(x,y,s,z,e)}{q(s,z,e,y|x)} \right] q(s,z,e,y|x) p^*(x) ds dz de dx dy \\
&\quad + \int [\log q(y|x)] p^*(y|x) p^*(x) dx dy \\
&= \max_{p,q} \int \mathbb{E}_{q(s,z,e,y|x)} \left[\frac{p^*(y|x)}{q(y|x)} \log \frac{p(x,y,s,z,e)}{q(s,z,e,y|x)} \right] p^*(x) dx \\
&\quad + \int \mathbb{E}_{p^*(y|x)} [\log q(y|x)] p^*(x) dx \\
&= \max_{p,q} \mathbb{E}_{p^*(x)} \left[\mathbb{E}_{p^*(y|x)} [\log q(y|x)] \right] \\
&\quad + \mathbb{E}_{p^*(x)} \left[\mathbb{E}_{q(s,z,e,y|x)} \left[\frac{p^*(y|x)}{q(y|x)} \log \frac{p(x,y,s,z,e)}{q(s,z,e,y|x)} \right] \right]
\end{aligned}$$

The first equality follows from the chain rule of probability. The second equality comes from the definition of the expected value. The third equality is a consequence of the fact that the integral of a probability density function over its entire domain is equal to one. The remaining equalities are straightforward and follow directly from the previous steps.

Assuming that $q(y|x)$ is Gaussian, the first term in Equation 3.4 will become the MSE loss. As for the second term, it will become a lower bound of $p(x)$ which is proven in Theorem 3.1.3.

Theorem 3.1.3. *Let $p(x, y, s, z, e)$ be the joint distribution of latent and observed variables and $q(s, z, y, e | x)$ be the approximate posterior of latent variables and future trajectories given the past trajectories in [GCRL](#), a lower bound on the log-likelihood of past trajectories is:*

$$\mathbb{E}_{q(s, z, e, y | x)} \left[\log \frac{p(x, y, s, z, e)}{q(s, z, e, y | x)} \right] \leq \log p(x)$$

Proof According to the Causal Markov condition [\[57\]](#), we can factorize $q(s, z, e, y | x) = q(y | x, s, z)q(z | x)q(s | x, e)q(e | x)$ and $p(x, s, z, y, e) = p(y | x, s, z)p(x | s, z)p(s | e)p(z)p(e)$. The first approximate model, $q(y | x, s, z)$, can be replaced with $p(y | x, s, z)$ since it is known. Secondly, since $q(s | x, e)$ is an approximation of its true posterior, I assume it to be $q(s | x)$. Therefore:

$$\begin{aligned} & \max_{p, q} \mathbb{E}_{p^*(x)} \left[\mathbb{E}_{q(s, z, e, y | x)} \left[\frac{p^*(y | x)}{p^*(y | x)} \log \frac{p(x, y, s, z, e)}{q(s, z, e, y | x)} \right] \right] \\ &= \max_{p, q} \mathbb{E}_{p^*(x)} \left[\mathbb{E}_{q(s, z, e, y | x)} \left[\log \frac{p(x, y, s, z, e)}{q(s, z, e, y | x)} \right] \right] \\ &= \max_{p, q} \mathbb{E}_{p^*(x)} \left[\int \left[\log \frac{p(x, y, s, z, e)}{q(s, z, e, y | x)} \right] q(s, z, e, y | x) ds dz de dy \right] \\ &= \max_{p, q} \mathbb{E}_{p^*(x)} \left[\int \left[\log \frac{p(y | s, z, x)p(x | s, z)p(z)p(s | e)p(e)}{q(y | s, z, x)q(z | x)q(s | x)q(e | x)} \right] q(y | s, z, x)q(z | x)q(s | x)q(e | x) ds dz de dy \right] \\ &= \max_{p, q} \mathbb{E}_{p^*(x)} \left[\int \left[\log \frac{p(x | s, z)p(z)p(s | e)p(e)}{q(z | x)q(s | x)q(e | x)} \right] p(y | s, z, x)q(z | x)q(s | x)q(e | x) ds dz de dy \right] \\ &= \max_{p, q} \mathbb{E}_{p^*(x)} \left[\int \left[\log \frac{p(x | s, z)p(z)p(s | e)p(e)}{q(z | x)q(s | x)q(e | x)} \right] q(z | x)q(s | x)q(e | x) ds dz de \right] \\ &= \max_{p, q} \mathbb{E}_{p^*(x)} \left[\int \left[\log \frac{p(x | s, z)p(z)p(s)p(e | s)}{q(z | x)q(s | x)q(e | x)} \right] q(z | x)q(s | x)q(e | x) ds dz de \right] \\ &= \max_{p, q} \mathbb{E}_{p^*(x)} \left[\mathbb{E}_{q(s, z | x)} [\log p(x | s, z)] + \mathbb{E}_{q(z | x)} \left[\log \frac{p(z)}{q(z | x)} \right] + \mathbb{E}_{q(s | x)} \left[\log \frac{p(s)}{q(s | x)} \right] + \mathbb{E}_{q(e | x)} \left[\log \frac{p(e | s)}{q(e | x)} \right] \right] \\ &\leq \max_{p, q} \mathbb{E}_{p^*(x)} [\mathbb{E}_{q(s, z | x)} [\log p(x | s, z)]] \\ &\leq \max_{p, q} \mathbb{E}_{p^*(x)} [\log \mathbb{E}_{q(s, z | x)} [p(x | s, z)]] = \mathbb{E}_{p^*(x)} [\log p(x)] \end{aligned} \tag{3.5}$$

Since the natural logarithm is a concave function, the second inequality follows from the Jensen's inequality. The third inequality follows from the Causal Markov condition which states that conditioned on its parents, every causal variable is independent of all its non-descendants.

According to Equation 3.5, the second term in Equation 3.4 will eventually become:

$$\begin{aligned} \max_{p,q} \mathbb{E}_{p^*(x)} & \left[\mathbb{E}_{q(s,z|x)} [\log p(x | s, z)] \right. \\ & + \mathbb{E}_{q(z|x)} \left[\log \frac{p(z)}{q(z | x)} \right] \\ & + \mathbb{E}_{q(s|x)} \left[\log \frac{p(s)}{q(s | x)} \right] \\ & \left. + \mathbb{E}_{q(e|x)} \left[\log \frac{p(e | s)}{q(e | x)} \right] \right] \end{aligned} \quad (3.6)$$

We can see that in Equation 3.6, $q(e | x)$ will be driven towards $p(e | s)$, hence, it is reasonable to assume that $q(e | x) = p(e | s)$. This helps us to avoid modeling $q(e | x)$ and obtain a lighter model. According to Bayes rule, we can rewrite $p(e | s)$ as:

$$q(e | x) = p(e | s) = \frac{p(s | e)p(e)}{p(s)} = \frac{p(s | e)p(e)}{\sum_e p(s | e)p(e)} \quad (3.7)$$

By plugging Equation 3.7 into Equation 3.4 and using Causal Markov condition, the objective function of GCRL can be obtained:

$$\begin{aligned} \max_{p,q} \mathbb{E}_{p^*(x)} \mathbb{E}_{p^*(y|x)} & [\log q(y | x)] + \mathbb{E}_{p^*(x)} \left[\mathbb{E}_{q(s,z,e,y|x)} \left[\frac{p^*(y | x)}{q(y | x)} \log \frac{p(x, y, s, z, e)}{q(s, z, y, e | x)} \right] \right] \\ = \max_{p,q} \mathbb{E}_{p^*(x,y)} & \left[\log q(y | x) + \mathbb{E}_{q(s,z,e|x)} \left[\frac{p(y | s, z, x)}{q(y | x)} \log \frac{p(y | s, z, x)p(x | s, z)p(z)p(s | e)p(e)}{q(y | s, z, x)q(z | x)q(s | x)q(e | x)} \right] \right] \\ = \max_{p,q} \mathbb{E}_{p^*(x,y)} & \left[\log q(y | x) + \frac{1}{q(y | x)} \mathbb{E}_{q(s,z,e|x)} \left[p(y | s, z, x) \log \frac{p(x | s, z)p(z)p(s | e)p(e)}{q(z | x)q(s | x)q(e | x)} \right] \right] \\ = \max_{p,q} \mathbb{E}_{p^*(x,y)} & \left[\log q(y | x) + \frac{1}{q(y | x)} \mathbb{E}_{q(s,z,e|x)} \left[p(y | s, z, x) \log \frac{p(x | s, z)p(z)p(s | e)p(e)p(s)}{q(z | x)q(s | x)p(s | e)p(e)} \right] \right] \\ = \max_{p,q} \mathbb{E}_{p^*(x,y)} & \left[\log q(y | x) + \frac{1}{q(y | x)} \mathbb{E}_{q(s,z,e|x)} \left[p(y | s, z, x) \log \frac{p(x | s, z)p(z)p(s)}{q(z | x)q(s | x)} \right] \right] \end{aligned} \quad (3.8)$$

Furthermore, $q(y | x)$ can be calculated as follows:

$$\begin{aligned}
q(y | x) &= \int q(y | x, s, z) q(s, z | x) ds dz \\
&= \int p(y | x, s, z) q(s | x) q(z | x) ds dz \\
&= p(y | \text{do}(x)) \\
&= \mathbb{E}_{q(s|x), q(z|x)} [p(y | x, s, z)]
\end{aligned} \tag{3.9}$$

In which case, $q(y | x)$ will be equal to $p(y | \text{do}(x))$ according to Equation 3.1 and Equation 3.9. After training, $q(s | x)$ will be driven towards its target $p(s) = \sum_e p(s | e)p(e)$ and $q(z | x)$ will be driven towards its target $p(z)$. Through the objective function in Equation 3.8, GCRL learns:

1. The invariant representations via $\log \frac{p(z)}{q(z|x)}$
2. The variant representations via $\log \frac{p(s)}{q(s|x)}$
3. To minimize the distance between groundtruth and predicted trajectories via $\log q(y | x)$ and eliminate the confounding bias
4. To reconstruct the past trajectories via $\log p(x | s, z)$
5. To tackle the multi-modality of trajectories as it is a generative approach

As a design choice, S is assigned a [Gaussian Mixture Model \(GMM\)](#) prior to effectively capture the variability across multiple environments in the datasets. Since S represents the variant representations and is conditioned on its parent E , its prior can be expressed as $p(s) = \sum_e p(s | e)p(e)$. Examining this equation closely, we can deduce that $p(e)$ are the weights of mixtures and $p(s|e)$ are the distribution of S in environment e . [GMM](#) priors have been used in former popular [VAE](#) models such as [25, 26]. Also, [GMM](#) priors are known to be universal approximators, hence, it is possible to learn variant representations with arbitrary distributions. As discussed before, identifiability is a main challenge in causal models. Furthermore, [VAE](#) models are prone to getting trapped in local minima or experiencing posterior collapse [31]. As proven in [31], a [GMM](#) prior will help to ensure the identifiability of S . In order to achieve the weakest form of identifiability i.e., identifiability up to an affine transformation, the decoder needs to be a piece-wise affine injective function [31]. Making stronger assumption on the decoder or parameter space can achieve stronger identifiability results.

In order to obtain accurate likelihood, I use coupling layers [16] in the priors. Therefore, the priors will be learnable as well and are also referred to as flow priors [63]. The idea behind flow priors is initiated from the following properties in random variables:

$$\begin{aligned} p(z) &= p(d_1) \left| \det \left(\frac{\partial f_{\lambda_1}(z)}{\partial z} \right) \right| \text{ such that } z = f_{\lambda_1}^{-1}(d_1) \\ p(s | e) &= p(d_2 | e) \left| \det \left(\frac{\partial f_{\lambda_2}(s)}{\partial s} \right) \right| \text{ such that } s = f_{\lambda_2}^{-1}(d_2) \end{aligned} \quad (3.10)$$

where d_1 and d_2 are the dummy variables with simple priors such as Standard Normal distribution whereas the priors for S and Z will be richer. The layers to implement f are adopted from [16]. One can also use learnable mean and variance in order to obtain rich priors, however, flow priors are more theoretically sound.

The overview of GCRL is shown in Figure 3.2. As shown in the Figure, GCRL can be implemented with any motion forecasting model that has an Encoder-Interaction-Decoder (EID) architecture. After the encoder and interaction modules, there are fully connected layers that models the mean and variance of latent variables distributions. Then, z and s are samples from $q(s | x)$ and $q(z | x)$ distributions and feed them to fully connected layers to fuse them. Finally, the fused vectors are fed to decoders to reconstruct the past trajectories and predict the future trajectories. The coupling layers are also used to transform the latent variables to the dummy variables. In the experiments, GCRL has not been evaluated in applications beyond HTP, but I expect models with the similar architecture in other applications to be compatible with GCRL as well.

3.1.5 Domain Adaptation

After training is done, it is expected for $q(z | x)$ to be close to $p(z)$ and $q(s | x)$ to be close to $p(s)$. We know that $p(z)$ is a single distribution and all features generated by $q(z | x)$ will be in the range of that distribution. Therefore, they are considered invariant and we do not have to finetune any part of the $p(z)$ or $q(z | x)$. On the other hand, $p(s)$ will form clusters and the features generated by $q(s | x)$ will be in different ranges as illustrated in Figure 3.3. The GMM prior can also be interpreted as the weighted sum of representations learnt from the various environments of the training data. Depending on how related the test domains are to the training domains, we may have to finetune some components of the GMM prior.

The idea for domain adaptation in GCRL is that for significantly different test domains we finetune the components of GMM and obtain new clusters. Subsequently, there will

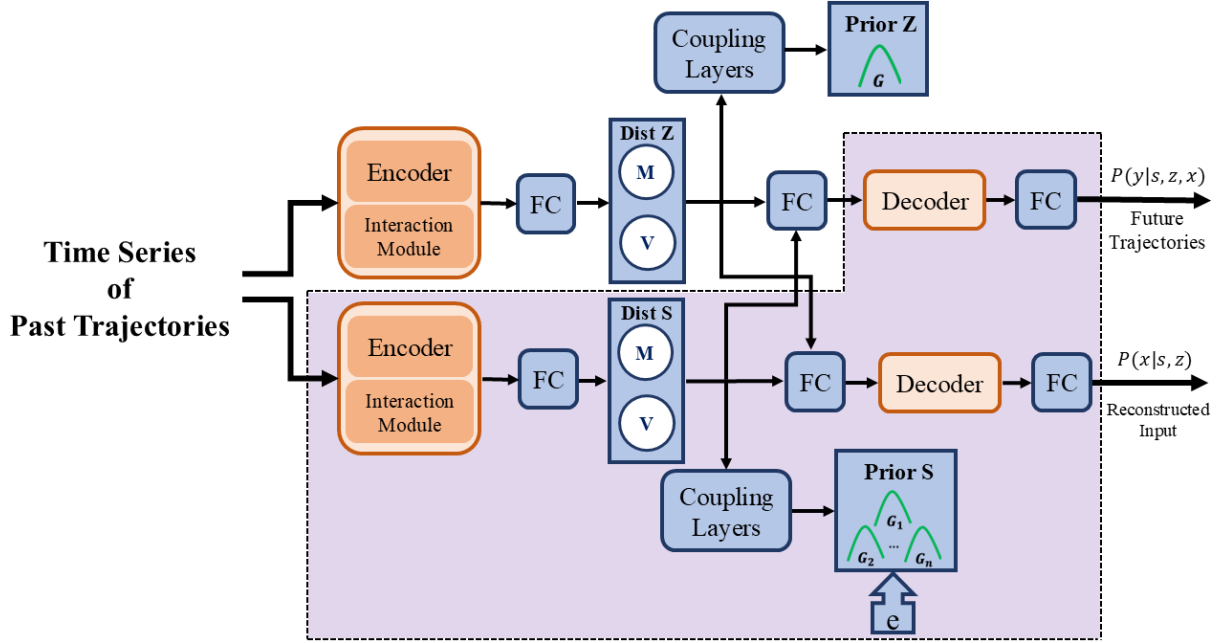


Figure 3.2: Overview of GCRL. The encoder, interaction module, and decoder are adopted from the baseline HTP model [23]. The components within the dotted box are fine-tuned for adaptation to a new domain.

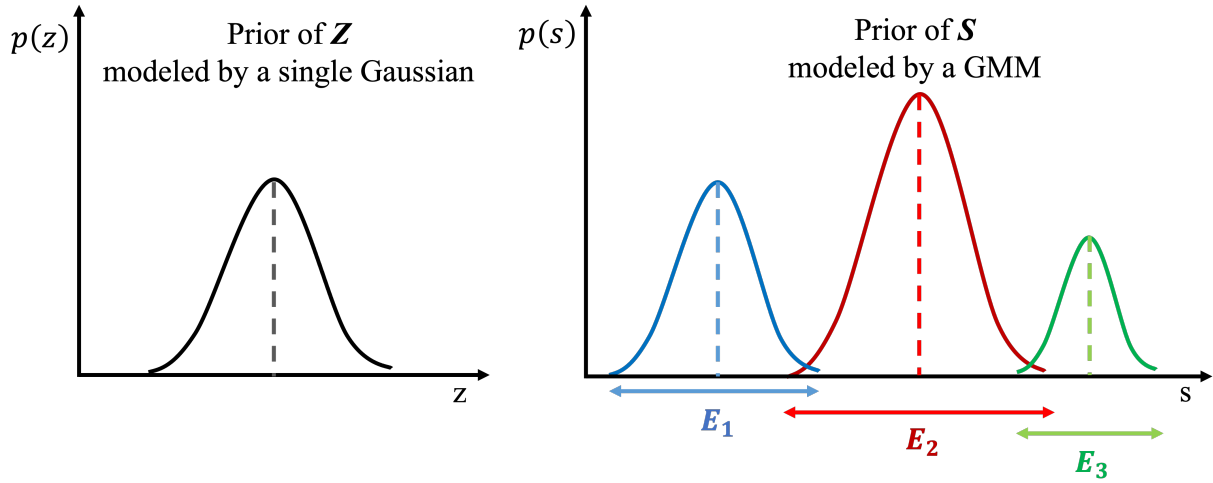


Figure 3.3: The range of distributions of variant and invariant representations priors. E_i represents environment i .

be a gap between $q(s | x)$ and the new prior of S . Therefore, we finetune the model with modified loss function in Equation 3.11 to close the distribution shift. We also need to finetune the decoders as we are going to have a new representations space for S .

$$\max_{p,q} \mathbb{E}_{p^*(x,y)} \left[\log q(y | x) + \frac{1}{q(y | x)} \mathbb{E}_{q(s,z,e|x)} \left[p(y | s, z, x) \log \frac{p(x | s, z)p(s)}{q(s | x)} \right] \right] \quad (3.11)$$

Consequently, we would have to finetune the models for $p(s)$, $q(s | x)$, $p(x | s, z)$, and $p(y | x, s, z)$. It can be seen that how SMS hypothesis helps in transfer learning. In the configuration, the changes are stemming from E which is a sparse change and they manifest themselves in some parts of the model. The changes do not affect all the parameters simultaneously.

A comparison of GCRL with other HTP methods is given in Table 3.1.

Table 3.1: Comparison of GCRL with other recent HTP models. Heterogeneity pertains to the presence of diverse data forms, while multi-modality refers to the generation of multiple trajectories.

Model	Heterogeneous Data	Robustness	Multi-modality	Social Interactions	Physical Interactions	CNN/RNN
Social-GAN [21]	no	no	yes	yes	no	RNN
Social-BiGAT [32]	yes	no	yes	yes	yes	RNN
Trajectron++ [55]	yes	no	yes	yes	yes	RNN
STGAT [23]	no	no	yes	yes	no	RNN
Social-STGCNN [48]	no	no	no	yes	no	CNN
DESIRE [35]	yes	no	yes	yes	yes	RNN
Deep Motion [33]	no	no	no	yes	no	RNN
IM [42]	no	yes	no	yes	no	N/A
Counterfactual [10]	yes	yes	yes	no	yes	N/A
GCRL [3]	no	yes	yes	yes	no	N/A

The next step of GCRL would be to let the model to learn the causal model on its own instead of building it based on domain knowledge. This idea will be discussed in the next sections.

3.2 Results and Discussion

In this section I am going to present the results of [GCRL](#) on a widely used motion forecasting benchmark dataset, ETH-UCY, and a synthetic dataset published in [\[42\]](#).

3.2.1 Datasets

ETH-UCY dataset [\[36, 52\]](#) consists the trajectories of 1536 pedestrians captured in 5 different environments called $\{hotel, univ, zara1, zara2\}$. The trajectories are sampled every 0.4 seconds. As practiced in literature [\[42, 10, 23\]](#), I am going to use leave-one-out approach for testing. Meaning that I am going to train the model with the samples of 4 environments and test it on the samples of the remaining environment. Furthermore, I am going to include 8 frames in the past trajectories (3.2 seconds) and predict the next 12 frames (4.8 seconds). As pointed out in [\[10\]](#), the environments in ETH-UCY dataset have distribution shifts, hence, making it suitable to test [GCRL](#). I am going to use this dataset to evaluate [GCRL](#).

Synthetic dataset published in [\[42\]](#) consists of trajectories of pedestrians in circle-crossing scenarios [\[8\]](#) where the domain shifts are simulated as the [Minimum Separation Distance \(MSD\)](#) of pedestrians in each domain. There are 8 domains in the dataset where the [MSD](#) in each domain is in the set $\{0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8\}$. There are 10,000 trajectories for training, 3,000 trajectories for validation, and 5,000 trajectories for testing in each domain. I am going to use this dataset to evaluate [GCRL](#).

3.2.2 Metrics

In the experiments, I use the [Average Displacement Error \(ADE\)](#) and [Final Displacement Error \(FDE\)](#) to evaluate [GCRL](#) similar to prior works [\[42, 10\]](#). [ADE](#) and [FDE](#) are defined as follows:

$$\text{ADE} = \frac{\sum_{i=1}^n \sum_{t=T_{\text{obs}}+1}^{T_{\text{pred}}} \left[(\hat{h}_i^t - h_i^t)^2 + (\hat{v}_i^t - v_i^t)^2 \right]^{\frac{1}{2}}}{n (T_{\text{pred}} - (T_{\text{obs}} + 1))} \quad (3.12)$$

$$\text{FDE} = \frac{\sum_{i=1}^n \left[(\hat{h}_i^{T_{\text{pred}}} - h_i^{T_{\text{pred}}})^2 + (\hat{v}_i^{T_{\text{pred}}} - v_i^{T_{\text{pred}}})^2 \right]^{\frac{1}{2}}}{n} \quad (3.13)$$

where $\hat{h}_i^t$, $\hat{v}_i^t$, h_i^t , and v_i^t are the *predicted* horizontal and vertical coordinates and actual horizontal and vertical coordinates of the pedestrian at time step t . These are the commonly used metrics in motion forecasting.

3.2.3 Robustness

In this experiment, I evaluate the robustness of [GCRL](#) against spurious correlations. As mentioned before, the presence of observation noise and future trajectories can be a spurious correlation. Furthermore, in real-world datasets such as ETH-UCY, it is difficult to pinpoint the distributions shifts, hence, as practiced in [\[42\]](#), I add synthetic observation noise to the dataset to simulate distribution shifts. This observation noise is added as a third dimension to the coordinates of pedestrians which is modeled as [\[42\]](#):

$$\begin{aligned}\gamma_t &:= (\dot{h}_{t+\delta t} - \dot{h}_t)^2 + (\dot{v}_{t+\delta t} - \dot{v}_t)^2 \\ \sigma_t &:= \alpha(\gamma_t + 1),\end{aligned}\tag{3.14}$$

where α is the noise intensity and $\dot{h}_t = h_{t+1} - h_t$ and $\dot{v}_t = v_{t+1} - v_t$ reflect the velocity of the pedestrians within the temporal window length of $\delta t = 8$. The training environments are $\{hotel, univ, zara1, zara2\}$ and *eth* is the test domain. For training domains, the value of $\alpha \in \{1, 2, 4, 8\}$ and for the test domain the value of $\alpha \in \{8, 16, 32, 64\}$. Consequently, the samples with $\alpha \in \{16, 32, 64\}$ are considered out-of-distribution samples. For a fair comparison with [\[10\]](#) and [\[42\]](#), I use [Spatial-Temporal Graph ATtention \(STGAT\)](#) [\[23\]](#) as the baseline model which also has the [EID](#) architecture. The results are shown in [Table 3.2](#). The performance of [\[23\]](#) and [\[10\]](#) degrades quickly for $\alpha \geq 16$. On the other hand, [GCRL](#) is insensitive to domain shifts as it learns to reconstruct uncorrupted inputs.

By changing α in the *eth* environment I am only changing one factor, however, in real-world there might be multiple changing factors. Therefore, I have tested [GCRL](#) in multiple environments with varying α . As shown in [Table 3.3](#), [GCRL](#) is still insensitive to observation noise in different environments.

The results in [Table 3.4](#) also suggests that [GCRL](#) is able to eliminate the observation noise as the [ADE](#) and [FDE](#) values are same as in [Table 3.3](#).

3.2.4 Multi-modality

In this section, I examine how good [GCRL](#) is in tackling the multi-modality of trajectories. I mentioned before that [GCRL](#) is a generative approach, hence, we can generate multiple

Table 3.2: Robustness of different methods in the ETH-UCY dataset with controlled spurious correlation.

Method	ADE/FDE	ADE/FDE	ADE/FDE	ADE/FDE
	$\alpha = 8$	$\alpha = 16$	$\alpha = 32$	$\alpha = 64$
BASILINE [23]	0.80/1.37	2.15/3.80	2.64/4.44	2.68/4.48
COUNTERFACTUAL [10]	0.80/1.59	1.62/2.68	2.32/3.90	2.68/4.52
INVARIANT $\lambda = 1.0$ [42]	0.94/1.65	1.04/ 1.76	1.52/2.55	1.96/3.35
INVARIANT $\lambda = 3.0$ [42]	0.91/1.67	0.99/1.87	1.18/2.20	1.27/2.33
INVARIANT $\lambda = 5.0$ [42]	0.98/1.79	1.00/1.83	1.06/1.90	1.56/2.58
GCRL	0.97/1.8	0.97/1.8	0.97/1.8	0.97/1.8

trajectories per pedestrians and choose the best of them. It is evident from Figure 3.4 that as the number of generated trajectories per pedestrian increases, the performance of GCRL improves indicating that GCRL can deal with multi-modality of trajectories. Furthermore, as shown in Figure 3.5, the generated trajectories by GCRL are different which further supports the fact that a generative approach can improve the performance of motion forecasting models.

3.2.5 Domain Generalization

In this section, I evaluate GCRL using the synthetic dataset and compare it to [42] which I will refer to as Invariant-Modular (IM). I train GCRL on domains with $MSD \in \{0.1, 0.3, 0.5\}$ and test it using the samples of all domains. In this setting, the samples of domains with $MSD \in \{0.1, 0.3, 0.5\}$ will be IID data. The samples with $MSD \in \{0.2, 0.4\}$ will be OOD-Inter cases as these samples are the interpolation of training domains samples. The samples with $MSD \in \{0.6, 0.7, 0.8\}$ will be OOD-Extra cases. As shown in Figure 3.6, GCRL is robust for IID and OOD-Inter cases, however, for OOD-Extra cases we need to finetune the model. Consequently, for cases with $MSD > 0.5$, the performance of GCRL degrades and the reason is the variant representations of OOD-Inter and IID cases are similar, hence, the clusters produced by GMM prior can be used here. However, for OOD-Extra cases the clusters can no longer be used and they need to be updated.

Table 3.3: GCRL performance in cases with multiple changing factors.

Environment	ADE/FDE	ADE/FDE	ADE/FDE	ADE/FDE
	$\alpha = 8$	$\alpha = 16$	$\alpha = 32$	$\alpha = 64$
ETH	0.97/1.8	0.97/1.8	0.97/1.8	0.97/1.8
HOTEL	0.64/1.3	0.64/1.3	0.64/1.3	0.64/1.3
UNIV	0.53/1.15	0.53/1.15	0.53/1.15	0.53/1.15
ZARA1	0.4/0.86	0.4/0.86	0.4/0.86	0.4/0.86
ZARA2	0.31/0.66	0.31/0.66	0.31/0.66	0.31/0.66

Table 3.4: Motion forecasting results on different domains of ETH-UCY.

Model	ADE/FDE					
	ETH	HOTEL	UNIV	ZARA1	ZARA2	AVG
STGAT	0.8/1.53	0.52/1.08	0.51/1.12	0.39/0.87	0.3/0.64	0.5/1.05
GCRL	0.97/1.9	0.55/1.14	0.51/1.13	0.38/0.84	0.3/0.67	0.54/1.1

3.2.6 Domain Adaptation

In this section, I examine the efficiency of GCRL when transferring it into a new domain. For this purpose, I train both GCRL and IM on domains with $MSD \in \{0.1, 0.3, 0.5\}$ and finetune both using a few samples from the domain with $MSD = 0.6$ and test them on this domain. The batch size in experiments were set to 64, hence, the number of samples used for finetuning is $\{1, 2, 3, 4, 5, 6\} \times 64$. For IM, I use the best setting reported in the paper and for GCRL I finetune the parameters for $p(y | x, s, z)$, $p(x | s, z)$, $p(s)$, and $q(s | x)$. As shown in Figure 3.7, even with one batch GCRL can quickly adapt itself to the new domain. Since the number of samples in the new domain can be very limited, it is crucial for models to adapt efficiently with only a small amount of data. In domain adaptation, the ability to generalize from a source domain to a target domain with minimal labeled data is a key challenge. Models that can rapidly learn transferable representations and finetune effectively with a few samples are particularly valuable, as they reduce the need for extensive labeled data collection in the new domain while maintaining strong performance. Overall, GCRL improves the ADE by 34.3% from IM on average.

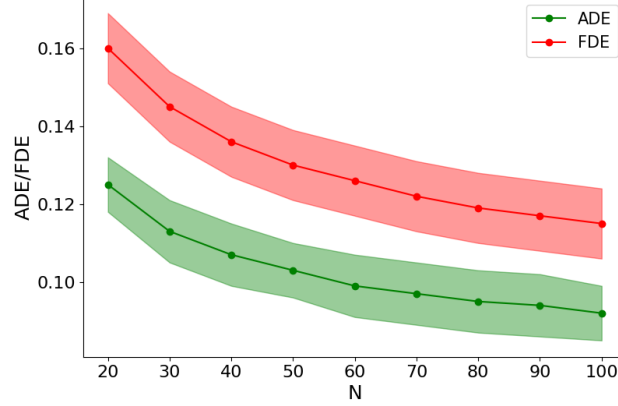


Figure 3.4: Comparison of ADE/FDE for different values of N .

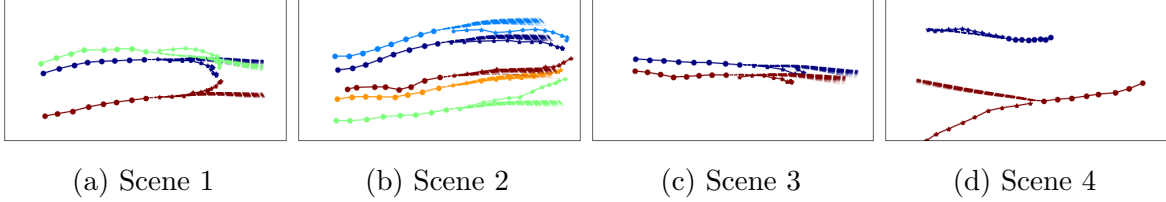


Figure 3.5: Trajectories of pedestrians in different scenes. The circled trajectories are observed, the dashed trajectories are predicted, and the starred trajectories are groundtruth. In all of the scenes, 100 trajectories are predicted for each pedestrian.

3.2.7 Identifiability

In this section, I examine the identifiability of S and Z . Similar to [31], I use [Mean Correlation Coefficient \(MCC\)](#) as a metric. I train [GCRL](#) with 5 different seeds and then calculate the [MCC](#) of latent spaces. I refer to this as Strong [MCC](#). As mentioned before, with a piece-wise affine injective encoder we can achieve identifiability up to an affine transformation. Therefore, I first learn an affine transformation f between two latent spaces using least squares minimization. Then, I calculate the [MCC](#) of the original latent space and the transformed latent space with a different seed. I refer to this as Weak [MCC](#). The results are shown in Table 3.5.

These results suggest that [GCRL](#) is able to identify S up to an affine transformation but, it cannot identify Z . This also aligns with our understanding of the physics and social

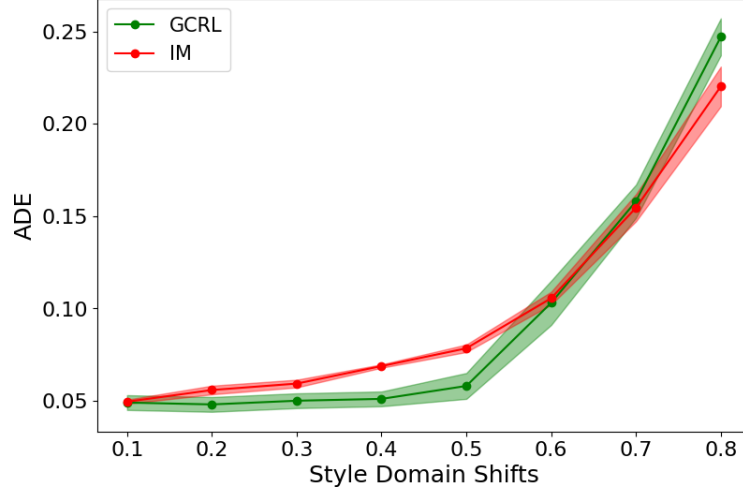


Figure 3.6: Domain generalization with different style domain shifts. The mean and standard deviation of ADEs are calculated and plotted for 5 seeds.

Table 3.5: MCC of S and Z after and before applying an affine transformation.

Weak MCC of S	Weak MCC of Z	Strong MCC of S	Strong MCC of Z
0.956	0.049	-0.16	-0.025

norms as we cannot gain any information about these factors from trajectories alone. Consequently, more information such as images of the environment or interventional data, is required to identify Z . However, the results from ablation studies indicates that Z is indeed an invariant representation.

3.2.8 Ablation Studies

In this section, I change some components of [GCRL](#) to see their effects. Similar to previous experiment, I train [GCRL](#) on domains with $MSD \in \{0.1, 0.3, 0.5\}$ and test it on domain with $MSD = 0.6$. With the ETH-UCY dataset, I train [GCRL](#) on the $\{hotel, univ, zara1, zara2\}$ and test it on *eth* environment. The results in Table 3.6 and

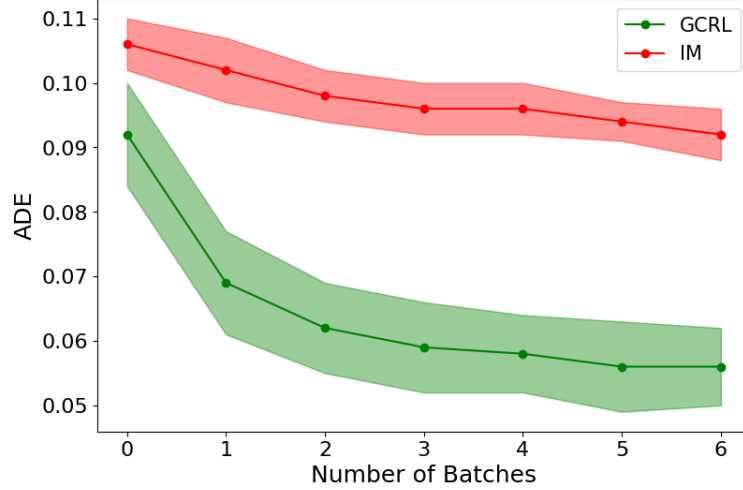


Figure 3.7: Domain Adaptation with different number of batches

Table 3.7 suggest that the addition of coupling layers can improve performance on certain datasets, particularly in complex real-world datasets like ETH-UCH. We can also see that the model trained with Only Z performs better than the model trained with only S . This indicates the invariance of Z variable despite it being unidentifiable. However, the performance of both models are worsened compared to the vanilla GCRL with both S and Z . I also change the values of the dimension of latent variables and the number of clusters in GMM prior, however, no significant changes are observed.

I have also shown the convergence of network in the Figures A.1, A.2, A.3, A.4.

Table 3.6: The effect of different components on performance with the synthetic dataset

Models	ADE	FDE
VANILA GCRL	0.0871	0.1047
NO COUPLING LAYERS	0.0772	0.0916
Z ONLY	0.1054	0.1347
S ONLY	0.2188	0.2418

Table 3.7: The effect of different components on performance when experimenting with the ETH-UCY dataset.

Models	ADE	FDE
VANILA GCRL	0.9728	1.8696
NO COUPLING LAYERS	0.9881	1.8869
LATENT DIMENSION:2	0.9736	1.8459
NUMBER OF GMM CLUSTERS:2	0.9732	1.9534
NUMBER OF GMM CLUSTERS:7	0.9775	1.9169

3.3 Summary

The key findings of [GCRL](#) can be summarized as follows:

- The autoencoder architecture enables the model to filter out noise from observations, making it more robust and less sensitive to observation noise.
- The causal structure of the model enhances its efficiency in transfer learning, allowing it to quickly adapt to new domains.
- Given only trajectories, the invariant representations Z cannot be identified; therefore, additional data types, such as images, are required for identification.

One of the assumptions in [GCRL](#) is that the causal structure is known. As the next step, I have proposed [ICRL-SM](#), which learns both the causal variables and the causal structure from data. This method will be explained in detail in the next chapter.

Chapter 4

Implicit Causal Representation Learning via Switchable Mechanisms

This chapter introduces the second proposed method, [ICRL-SM](#), which extends the previous method by learning both the causal variables and causal structure within a dataset. [ICRL-SM](#) relaxes one of the stringent assumptions in [CRL](#) and provides a theoretical framework that ensures identifiability of the causal structure. This work specifically addresses the problem of learning causal relationships and offers a deeper understanding of how causal variables evolve, particularly in settings involving pre- and post-intervention data.

The problem domain in this chapter revolves around understanding how interventions affect the variables of interest in real-world scenarios, with a focus on both synthetic and real-world datasets. For example, in image-based tasks, the chapter explores how changes made during the intervention phase impact the outcomes, specifically comparing the pre- and post-intervention stages.

[ICRL-SM](#) is compared with existing methods, showcasing its advantages in real-world settings. Results are presented from both synthetic and real-world datasets, demonstrating how the proposed approach outperforms previous techniques in terms of learning causal structure. The chapter concludes with a summary of the key findings, which highlight the effectiveness of the method in various contexts.

4.1 Implicit and Explicit Latent Causal Models

The second contribution to causal discovery in this thesis is inspired by [ILCM](#) [6]. Before delving into the proposed idea, it is essential to understand the distinction between explicit and implicit latent causal models.

In causal representation learning from interventional data, we are provided with a pair of data samples: pre-intervention and post-intervention. Let these samples be denoted by x and $\tilde{x}$, respectively. The underlying assumption is that a causal model with latent variables Z generates the dataset. The variables x and $\tilde{x}$ are produced by a decoder function g , which takes as input the causal variables z and $\tilde{z}$. The challenge here is to identify the latent variables, i.e., the causal model, from the pairs of x and $\tilde{x}$.

In this context, one approach is [ELCM](#), where we explicitly model the causal structure. This involves learning the adjacency matrix of the causal model and incorporating the parent-child relationships of Z in the prior distribution of Z i.e., $p(Z) = \prod_i p(Z_i | PA_i)$.

The [GCRL](#) [3] method introduced in the previous chapter is a simple form of [ELCM](#) since we assumed the causal graph is provided. In other methods, such as [68, 34, 64, 65], the causal graph is learned by estimating the adjacency matrix.

In contrast, [ILCM](#) does not rely on parent-child relationships, nor does it attempt to learn the graph via an adjacency matrix. Instead, in [ILCM](#), the latent variables are the exogenous noise variables E , rather than the endogenous causal variables Z . Therefore, the prior of latents can be conveniently factorized as $p(E) = \prod_i p(E_i)$ since exogenous variables are usually assumed to be mutually independent.

[ILCM](#) [6] introduces a solution function $s_i : \mathcal{E}_{anc_i} \rightarrow \mathcal{Z}_i$, which maps the ancestral exogenous variables E_{anc_i} to the causal variables Z_i . Solution functions and the details of their implementation will be further discussed in Section 4.7. To illustrate how the solution functions operates, let us consider a simple [SCM](#):

$$\begin{aligned} X &= e_X \\ Y &= aX + e_Y \\ Z &= bY + e_Z \end{aligned}$$

In this [SCM](#), e_X , e_Y , and e_Z are the exogenous noise variables and the rest are the causal variables. If we considered this [SCM](#) as a system of equations where the known variables are the exogenous variables. Then, the solution functions would be:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ a & 1 & 0 \\ ab & b & 1 \end{bmatrix} \begin{bmatrix} e_X \\ e_Y \\ e_Z \end{bmatrix} = \begin{bmatrix} s_X(e_X) \\ s_Y(e_X, e_Y) \\ s_Z(e_X, e_Y, e_Z) \end{bmatrix}$$

This example illustrates a simple linear case of a solution function. However, in general, the function s is nonlinear. Solution functions capture the dependencies between causal variables and exogenous variables, ensuring that the solution function for a given causal variable depends only on its ancestral exogenous variables.

In implicit [CRL](#), the model is not constrained by a learned adjacency matrix, providing greater flexibility. As a result, the model is less likely to become trapped in local minima due to incorrect updates of the adjacency matrix during optimization. Although the solution functions do not directly learn the edges of the causal graph, methods described in [\[6\]](#) can be employed post-training to infer the edges based on the dependencies observed in the solution functions.

The reason why [ILCM](#) may be generally preferred over [ELCM](#) is that causal model discovery is like the chicken and egg problem: it is hard to learn the causal variables when the causal graph is not yet learnt, and it is hard to learn the graph without knowing the variables. [ILCM](#) gets around this problem by not explicitly modeling the adjacency matrix.

Despite its advantages over [ELCM](#), [ILCM](#) [\[6\]](#) relies on several strict assumptions to ensure identifiability, which limits its applicability to real-world scenarios. These assumptions include diffeomorphic causal mechanisms, the availability of hard interventions, and the observation of interventions on all causal variables, among others. *In this work, I aim to relax one of the most critical assumptions in [\[6\]](#)—the availability of hard interventions—by employing soft interventions to achieve identifiability in [ILCM](#).* Therefore, understanding the difference between soft and hard interventions is crucial for this work, and this distinction will be explained in the next section.

4.2 Hard Interventions vs Soft Interventions

In a causal model, an intervention refers to an intentional action taken to alter one or more variables to observe its effects on other variables within the system. Interventions allow researchers to examine how changes in one variable directly cause changes in another, thereby uncovering causal relationships.

Interventions can be classified as either soft or hard, based on the degree of control and manipulation involved. A hard intervention involves directly altering the variables of

interest in a controlled manner, as seen in randomized controlled trials [51]. Specifically, a hard intervention sets the value of a causal variable Z to a specific value, denoted as $do(Z = z)$ [51].

In contrast, a soft intervention involves more subtle or less controlled manipulation of variables, leading to a change in the conditional distribution of the causal variable, represented as $p(Z | PA) \rightarrow \tilde{p}(Z | PA)$. This change can manifest in three aspects within a SCM: the causal mechanisms, the set of parents, and the exogenous variable corresponding to the intervened causal variable. Consequently, we have $\tilde{z}_i = \tilde{f}_i(\tilde{z}_{pa_i}, \tilde{e}_i)$, where $\tilde{z}_{pa_i}$ represents the altered set of parents for the causal variable z_i [11]. From a graphical perspective, a

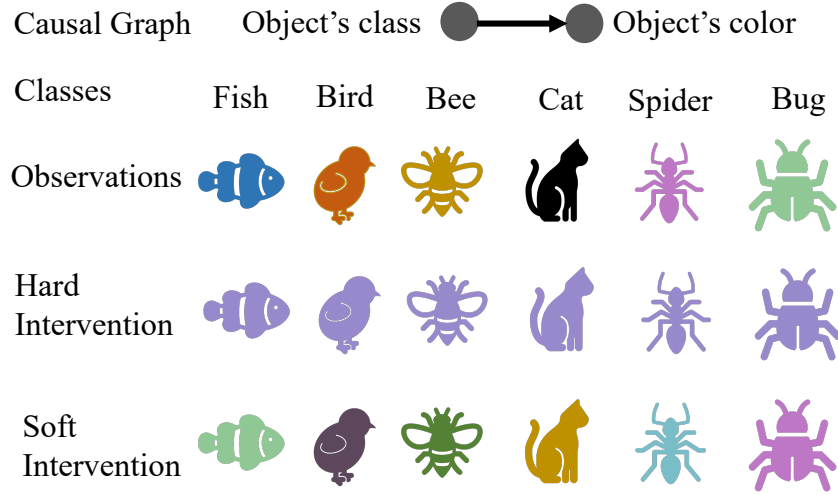


Figure 4.1: Difference between hard and soft interventions: As illustrated in the middle row, hard interventions sever the connections with parent variables. This means that, for example, when we intervene on an object's color, the object's class can no longer influence its color. In contrast, soft interventions, as depicted in the bottom row, allow these connections to remain, permitting the object's class to still affect its color even after the intervention.

hard intervention ensures that the intervened node is influenced solely by the intervention itself, with no input from its ancestral nodes. In contrast, under a soft intervention, the intervened node can still be influenced by its parent nodes in addition to the intervention. For example, consider the graph in Figure 4.1. In the case of a hard intervention, the object's class would no longer affect the object's color. A hard intervention such as $do(\text{object's color} = \text{purple})$ would sever the connection between the object's class and its color, setting the color to a fixed value of purple. On the other hand, a soft intervention

would result in more subtle changes, altering the causal mechanism by which the object's class influences its color, rather than completely severing the connection. As shown in Figure 4.1, the effects of causal variables in soft interventional data can be ambiguous, as they closely resemble those in observational data. This similarity makes it difficult to observe the specific changes we are *interested in*. To address this, we need a method to isolate the effects of interest and utilize them to identify the causal variables. In the following section, I will introduce the *causal mechanism switch variable*, which is designed to achieve this. Another example of interventional data in a real-world example is shown in Figure 4.2.

4.3 Data Generating Process

The formal definition for **SCMs** is given in Definition 4.3.1.

Definition 4.3.1. (*Structural Causal Models*) A **SCM** is a tuple $\mathcal{C} = (\mathcal{F}, \mathcal{Z}, \mathcal{E}, \mathcal{G})$ with the following components:

1. The domain of causal variables $\mathcal{Z} = \mathcal{Z}_1 \times \mathcal{Z}_2 \times \dots \times \mathcal{Z}_n$.
2. The domain of exogenous variables $\mathcal{E} = \mathcal{E}_1 \times \mathcal{E}_2 \times \dots \times \mathcal{E}_n$.
3. The domain of directed acyclic graph $\mathcal{G}(\mathcal{C})$ over the causal and exogenous variables.
4. A causal mechanism $f_i \in \mathcal{F}$ which maps an assignment of parent values for the parents Z_{pa_i} plus an exogenous variable value for E_i to a value of causal variable Z_i .

SCMs provide a framework for understanding and describing how variables interact and influence each other through causal mechanisms. A **mixing function**:

$$g(\mathbf{z}) = \mathbf{x}, \quad (4.1)$$

maps a vector of latent causal variables $\mathbf{z}$ to observed values $\mathbf{x}$. Since the causal variables $\mathbf{Z}$ are not directly observed, the goal is to infer them from interventional data. For each causal variable, a **solution function** in the data generation process is defined as:

$$s_i : \mathcal{E}_{anc_i} \rightarrow \mathcal{Z}_i \quad (4.2)$$

which deterministically maps the values of ancestral exogenous variables e_{anc_i} to a value for the causal variable $z_i \in \mathbb{R}$. In this work, solution functions are assumed to be a

diffeomorphism. Diffeomorphism definition is given in Definition 4.3.2. s_i is a deterministic map because all the uncertainties are modeled. In implicit modeling, we focus on learning these solution functions s_i directly, rather than defining them through local mechanisms f_i .

Definition 4.3.2. (*Diffeomorphism*) A diffeomorphism between smooth manifolds $\mathcal{M}$ and $\mathcal{N}$ is a bijective map $f : \mathcal{M} \rightarrow \mathcal{N}$, which is smooth and has a smooth inverse. Diffeomorphisms preserve information as they are invertible transformations without discontinuous changes in their image.

Similar to causal representation learning, in nonlinear *Independent Component Analysis (ICA)*, the generative process also involves a mixture function g that maps latent causal variables $Z \in \mathbb{R}^n$ to observed variables $X \in \mathbb{R}^n$ [34, 74]. However, a key difference between causal representation learning and nonlinear ICA is that in the former, causal variables Z may exhibit complex dependencies. The goal of this work is to recover E from X and subsequently map E to Z using solution functions.

Identifying a causal model from observational data alone is complex and necessitates assumptions about the model parameters [1]. In unsupervised settings, identifying causal models from observational data is also known to be impossible [43]. Incorporating interventional data, in addition to observational data, helps in identifying causal variables by showing how changes in a causal variable affect observed variables. An interventional data point $(\mathbf{x}, \tilde{\mathbf{x}}, i)$ consists of the pre-intervention observation $\mathbf{x}$, the post-intervention observation $\tilde{\mathbf{x}}$, and the intervention target $i \in I$, where $I \subset \{1, 2, \dots, n\}$ is the set of intervention targets selected from the causal variables. The post-intervention data $\tilde{\mathbf{x}}$ is generated through a *soft intervention* on one of the causal variables. To achieve identifiability up to reparameterization, we rely on the following assumptions about the data generation process:

Assumption 4.3.3. (*Data Generating Assumptions*)

- **Atomic Interventions:** Each sample $(\mathbf{x}, \tilde{\mathbf{x}}, i)$ targets only one causal variable for intervention.
- **Complete Intervention Targets Set:** All possible interventions are observed, so $I = \{1, 2, \dots, n\}$.
- **Known Targets:** The targets of soft interventions are known.

- **Post-intervention Exogenous Variables:** The values of exogenous variables change only for the causal variable being intervened upon, while other exogenous variables retain their pre-intervention values. Thus, $e_i \neq \tilde{e}_i$ if $i \in I$, and $e_i = \tilde{e}_i$ otherwise.
- **Sufficient Action Variability:** Soft interventions should alter solution functions to introduce sufficient variability in actions [34]. These interventions must modify solution functions to ensure non-overlapping conditional distributions of causal variables (see Figure 4.4).
- **Diffeomorphic Decoder and Solution Functions:** Diffeomorphism ensures no loss of information and prevents abrupt changes in the function's image.

The **Known Targets** assumption can be relaxed in scenarios where such data is unavailable; in such cases, the approach outlined in [6] can be used to infer the intervention targets. In our real-world experiments, where intervention targets are not directly available, we hypothesize that the causal variables correspond to object attributes, and actions represent the intervention targets.

Without these assumptions, changes observed in $\tilde{\mathbf{x}}$ can become ambiguous, as overlapping effects from multiple interventions or exogenous variable changes might obscure the exact influence of each. For example, if both a heater and lights are changed simultaneously, it becomes difficult to determine each one's individual effect on room temperature. Thus, the data generation process for an interventional data point $(\mathbf{x}, \tilde{\mathbf{x}}, i)$ can be described as follows:

Pre-intervention data:

$$\begin{cases} \forall j & e_j \sim p_{\mathcal{E}_j} \\ \forall j & z_j = s_j(e_{anc_j}) \end{cases} \rightarrow \mathbf{x} = g(\mathbf{z}),$$

Post-intervention data:

$$\begin{cases} i \in I & \tilde{e}_i \sim p_{\tilde{\mathcal{E}}_i} \\ \forall j \neq i & e_j \sim p_{\mathcal{E}_j} \\ i \in I & \tilde{z}_i = \tilde{s}_i(\tilde{e}_{anc_i}) \\ \forall j \neq i & \tilde{z}_j = s_j(\tilde{e}_{anc_j}) \end{cases} \rightarrow \tilde{\mathbf{x}} = g(\tilde{\mathbf{z}})$$

4.4 Causal Mechanisms Switch Variable

As mentioned in Section 4.2, the key distinction between soft and hard interventions is that in the case of soft interventions, the post-intervention causal variable $\tilde{Z}_i$ remains connected to its parents. This connection complicates the identification of causal mechanisms

compared to hard interventions (see Figure 4.1). Soft intervention data provide fewer constraints on the causal graph structure than hard intervention data.

4.4.1 Augmented Implicit Causal Model

To address the effects of soft interventions, I introduce the causal mechanism switch variable V . Utilizing V allows us to revert to pre-intervention causal mechanisms within post-intervention data, thereby enabling the model to focus on identifying changes in the intrinsic properties of each causal variable. These changes are encapsulated within their corresponding exogenous variables, which enhances the model’s ability to accurately learn causal relationships. I propose employing a modulated version of V to capture the effects of soft interventions on each causal variable using a nonlinear function $h_i : \mathcal{V} \rightarrow \mathbb{R}$ as follows:

$$i \in I, \tilde{z}_i = \tilde{s}_i(\tilde{e}_i; \tilde{e}_{\setminus i}) = s_i(\tilde{e}_i; e_{\setminus i}, h_i(v)) \quad (4.3)$$

Since the parental set for each causal variable is unknown in [ILCM](#), we include all the exogenous variables in the solution functions and let the model give higher importance to ancestral exogenous variables. The inclusion of $h_i(v)$ enables the model to account for variations in the parental sets of all causal variables in V . Consequently, each causal variable Z_i is associated with a switch variable $V_i \in \mathbb{R}$. Consequently, the learnt solution functions would be $s_i : \mathcal{E}_i \times \mathcal{E}_{\setminus i} \times \mathcal{V}$. This leads to the concept of an *augmented implicit causal model*.

Definition 4.4.1. (*Augmented Implicit Causal Models*) An *Augmented Implicit Causal Model (AICM)* is defined as $\mathcal{A} = (\mathcal{S}, \mathcal{Z}, \mathcal{E}, \mathcal{V})$. $V \in \mathbb{R}^n$ represents the causal mechanism switch variable which models the effect of soft interventions on solution functions in $\mathcal{S}$ as shown in (4.3).

The use of V for soft interventions parallels the concept of augmented networks discussed in [49], which were primarily designed for hard interventions. Pearl [49] anticipated this possibility by noting:

“One advantage of the augmented network representation is that it is applicable to any change in the functional relationship f_i and not merely to the replacement of f_i by a constant.”

We can expand the solution functions in Equation 4.3 using Taylor’s expansion as follows:

$$s_i(\tilde{e}_i; e_{\setminus i}, h_i(v)) = s_i(\tilde{e}_i; e_{\setminus i}, h_i(v_0)) + \sum_{n=1}^{\infty} \frac{1}{n!} \left(\frac{\partial^n s_i}{\partial h_i^n} \Big|_{h_i=h_i(v_0)} (h_i(v) - h_i(v_0))^n \right) = s_i(\tilde{e}_i; e_{\setminus i}, h_i(v_0)) + R_i \quad (4.4)$$

where R_i denotes the remainder term in (4.4). We define the **separable dependence** property for solution functions as follows:

$$\exists h_i(v_0) : s_i(\tilde{e}_i; e_{\setminus i}, h_i(v_0)) = s_i(\tilde{e}_i; e_{\setminus i}). \quad (4.5)$$

An example of this scenario can be found in location-scale noise models [24]:

$$s_i(\tilde{e}_i; e_{\setminus i}, h_i(v)) = \tilde{e}_i + \text{loc}(e_{\setminus i}) + h_i(v) = \tilde{e}_i + \text{loc}(e_{\setminus i}) + v^2 + v, \quad (4.6)$$

where v_0 is set to zero. Ignoring the argument $h_i(v_0)$ in this example equates to setting it to zero. More generally, it could involve setting it to another value.

Assuming the separable dependence property, the solution function in (4.4) can be simplified to:

$$s_i(\tilde{e}_i; e_{\setminus i}, h_i(v)) = s_i(\tilde{e}_i; e_{\setminus i}) + R_i = s_i(\tilde{e}_i; e_{\setminus i}) + \text{soft intervention effect} \quad (4.7)$$

Thus, we can revert to pre-intervention solution functions. By modeling soft intervention effects with $h_i(v)$, we can recover pre-intervention solution functions. During inference, the $h_i(v)$ term is disregarded. However, it is possible to train the prior $p(v)$ to ensure that the separable dependence property is preserved for pre-intervention data.

4.4.2 Observability of the Switch Variable

The rationale for introducing V is to decompose the effect of soft intervention on $\tilde{Z}_i$ into:

1. The effect on causal mechanisms and their parents,
2. The effect on the corresponding exogenous variable E_i .

For instance, in images of objects, causal variables might represent attributes such as shape, color, and size. Actions like "Fold" alter these attributes. Moreover, the camera angle in

an image can affect the perceived shape of an object. In the case of hard interventions, the camera angle remains constant between pre and post-intervention phases. In contrast, for soft interventions, the camera angle changes along with the actions performed, indicating soft intervention effects. If we understand how the camera angle affects object attributes, we can separate the impact of the soft intervention. In other words, if V is observable, we can isolate the effect of the intervention on the causal variable itself. For additional details, refer to Figure 4.2.

Without understanding how soft interventions affect the causal model, a more complex model is required. As a result, the term R_i in (4.4) may involve higher-order terms of $h_i(V)$. Thus, we assume the observability of V .

Assumption 4.4.2. (*Observability of V*) *Given an intervention sample $(x, \tilde{x}, i)$ and linear mixing functions, we approximate the soft intervention effects R_i as follows:*

$$\tilde{z} - z \stackrel{(4.4)}{=} \Delta e_i + R \quad (4.8)$$

$$\tilde{x} - x \stackrel{(4.1)}{=} g(\tilde{z}) - g(z) \stackrel{\text{Linearity}}{=} g(\tilde{z} - z) \stackrel{(4.8)}{=} g(\Delta e_i + R), \quad (4.9)$$

where $R = [R_0, R_1, \dots, R_n]$ and n is the number of causal variables. R and Δe_i represent vectors indicating the soft intervention effects and changes in the exogenous variable's effect of the intervened causal variable, respectively. Note that elements of R will be zero except for the intervened causal variable. With linear mixing functions and appropriate preprocessing (e.g., subtraction), we can observe R_i .

For nonlinear mixing functions, alternative augmentations may be used to extract information about the causal mechanism switch variable V .

Our synthetic data is generated with a linear mixing function, but real-world datasets may not follow a linear mixing function. Consequently, V may not be observed from $\tilde{x} - x$ in real-world datasets. Nonetheless, our findings suggest that incorporating soft interventions through V enhances performance compared to other implicit modeling approaches. A better understanding of the impact of soft interventions on the dataset's generative system would lead to improved results.

Figure 4.3 illustrates our main generative model, which includes a data augmentation step that incorporates the intervention displacement $\tilde{x} - x$ as an observed feature directly representing the effect of a soft intervention in the observation space.

4.5 Identifiability Theorem for Augmented Implicit Causal Models with Soft Interventions

In this work, I aim to identify causal variables up to reparameterization using soft interventions. Let's begin by defining identifiability up to reparameterization (Definition 4.5.3) and then present my main identifiability theorem (Theorem 4.5.5).

My approach establishes a mapping between the causal variables $\mathcal{Z}$ and $\mathcal{Z}'$ of two **Latent Causal Model (LCM)s**, denoted as $\mathcal{M}$ and $\mathcal{M}'$, through component-wise transformations (see Definition 4.5.1). Given that our modeling is implicit and lacks explicit knowledge of the causal graph, we incorporate all exogenous variables into the solution functions as shown in equation (4.3). Importantly, **the causal graph structure remains unchanged during the learning process**.

To illustrate the differences between hard and soft interventions in the training process, consider a simple example with a causal model $Z_1 \rightarrow Z_2$ and a location-scale noise model [24] defined by the solution function:

$$\tilde{z}_2 = \frac{\tilde{e}_2 - \widetilde{loc}(e_1)}{\widetilde{scale}(e_1)}. \quad (4.10)$$

The mean of the distribution $p(\tilde{Z}_2)$ can be expressed as:

$$\frac{1}{\widetilde{scale}(e_1)} \times \mathbb{E}[\tilde{E}_2] - \frac{\widetilde{loc}(e_1)}{\widetilde{scale}(e_1)}. \quad (4.11)$$

In the case of *hard interventions*, we assume that $p(\tilde{Z}_2|Z_1 = z_1) = p(\tilde{Z}_2) = \mathcal{N}(0, 1)$ because the parental effects are absent. Consequently, the location and scale functions within the solution tend to suppress parental influences, aligning with the ground truth data where such effects do not exist.

Conversely, *soft interventions* acknowledge parental influence in the ground truth data, leading to $p(\tilde{Z}_2|Z_1 = z_1) \neq \mathcal{N}(0, 1)$. Due to the lack of explicit parental knowledge in implicit modeling, I represent this as:

$$p(\tilde{Z}_2|Z_1 = z_1) = p(\tilde{Z}_2|E_2 = e_2), \quad (4.12)$$

where E_2 serves as a known parent of $\tilde{Z}_2$. However, this approach causes parental effects to propagate to E_i (the exogenous variable corresponding to each causal variable), which violates identifiability up to reparameterization. By introducing the switch variable V , we allow these parental effects to be absorbed by V instead of E_i , preserving identifiability.

Definition 4.5.1. (*Component-wise Transformation*) Let ϕ be a transformation (1-1 onto mapping) between product spaces:

$$\phi : \Pi_{i=1}^n \mathcal{X}_i \rightarrow \Pi_{i=1}^n \mathcal{Y}_i. \quad (4.13)$$

If there exist local transformations ϕ_i such that:

$$\forall i, j, \forall x, \phi(x_1, x_2, \dots, x_n)_i = \phi_i(x_j), \quad (4.14)$$

then ϕ is a component-wise transformation.

Definition 4.5.2. (*Pushforward measure*) Given a measurable function $f : \mathcal{A} \rightarrow \mathcal{B}$ between two measurable spaces $\mathcal{A}$ and $\mathcal{B}$, and a measure p defined on $\mathcal{A}$, the pushforward measure f_*p on $\mathcal{B}$ is defined for measurable sets E in $\mathcal{B}$ as:

$$(f_*p)(E) = p(f^{-1}(E)), \quad (4.15)$$

where $*$ denotes the pushforward operation. In other words, the pushforward measure f_*p assigns a measure to a set in $\mathcal{B}$ by measuring the pre-image of that set under f in the space $\mathcal{A}$.

Definition 4.5.3. (*Equivalence up to Component-wise Reparameterization*) Consider two LCMs $\mathcal{M} = (\mathcal{A}, \mathcal{X}, g, \mathcal{I})$ and $\mathcal{M}' = (\mathcal{A}', \mathcal{X}, g', \mathcal{I})$, both based on AICMs $\mathcal{A}$ and $\mathcal{A}'$, sharing the same observation space $\mathcal{X}$ and intervention targets $\mathcal{I}$, with respective decoders g and g' . We say that $\mathcal{M}$ and $\mathcal{M}'$ are equivalent up to component-wise reparameterization, denoted as $\mathcal{M} \sim_r \mathcal{M}'$, if there exist component-wise transformations (as defined in Definition 4.5.1) ϕ_Z and $\phi_{\mathcal{E}}$ such that:

- **Index Preservation:** For each i , $\phi_i(z_i) = z'_i$ and $\phi_i(e_i) = e'_i$. Corresponding edges in the causal graphs are preserved, i.e., $Z_i \rightarrow Z_j$ exists in G if and only if $Z'_i \rightarrow Z'_j$ exists in G' . Additionally, edges from exogenous to endogenous variables ($E_i \rightarrow Z_i$) are preserved.
- **Measure Preservation on Exogenous Variables:** The transformation $\phi_{\mathcal{E}}$ preserves the probability measure on exogenous variables, i.e., $p_{\mathcal{E}'} = (\phi_{\mathcal{E}})_*p_{\mathcal{E}}$, following the pushforward measure definition in Definition 4.5.2.
- **Measure Preservation on Causal Variables:** Similarly, ϕ_Z preserves the probability measure on causal variables, i.e., $p_{Z'} = (\phi_Z)_*p_Z$.

Lemma 4.5.4. *The transformation $\phi_Z : Z \rightarrow Z'$ between the causal variables of two LCMs $\mathcal{M}$ and $\mathcal{M}'$, as defined in Definition 4.5.3, is a component-wise transformation if $\forall i, j, i \neq j \quad \tilde{E}'_i \perp\!\!\!\perp \tilde{E}'_j$ and the causal variables follow a multivariate normal distribution conditional on the pre-intervention exogenous variables, where $\tilde{E}'_i$ denotes the post-intervention exogenous variable of causal variable i in $\mathcal{M}'$.*

Proof. We consider the case where the exogenous variables are mapped to causal variables by a location-scale noise model such that:

$$\tilde{z}_i = \tilde{s}_i(\tilde{e}_i; e_{\setminus i}) = \frac{\tilde{e}_i - \widetilde{loc}(e_{\setminus i})}{\widetilde{scale}(e_{\setminus i})}, \quad \forall i, j, i \neq j \quad \tilde{E}'_i \perp\!\!\!\perp \tilde{E}'_j \rightarrow \mathbb{E}[\tilde{E}'_i \tilde{E}'_j] = \mathbb{E}[\tilde{E}'_i] \mathbb{E}[\tilde{E}'_j],$$

where $\mathbb{E}[\cdot]$ denotes the expected value. The role of $h_i(v)$ will become clear in the full proof of the theorem where I use it to satisfy $\forall i, j, i \neq j \quad \tilde{E}'_i \perp\!\!\!\perp \tilde{E}'_j$. Therefore, I refrain from using it in this Lemma.

Let's add the following constants:

$$-\mathbb{E}[\tilde{E}'_i] \widetilde{loc}'_j(e'_{\setminus j}), \quad -\mathbb{E}[\tilde{E}'_j] \widetilde{loc}'_i(e'_{\setminus i}), \quad \widetilde{loc}'_i(e'_{\setminus i}) \widetilde{loc}'_j(e'_{\setminus j})$$

to both sides of the equality and then divide both sides by $\widetilde{scale}'_i(e'_{\setminus i}) \widetilde{scale}'_j(e'_{\setminus j})$:

$$\begin{aligned} & \mathbb{E} \left[\frac{\tilde{E}'_i \tilde{E}'_j - \tilde{E}'_i \widetilde{loc}'_j(e'_{\setminus j}) - \tilde{E}'_j \widetilde{loc}'_i(e'_{\setminus i}) + \widetilde{loc}'_i(e'_{\setminus i}) \widetilde{loc}'_j(e'_{\setminus j})}{\widetilde{scale}'_i(e'_{\setminus i}) \widetilde{scale}'_j(e'_{\setminus j})} \right] \\ &= \frac{\mathbb{E}[\tilde{E}'_i] \mathbb{E}[\tilde{E}'_j] - \mathbb{E}[\tilde{E}'_i] \widetilde{loc}'_j(e'_{\setminus j}) - \mathbb{E}[\tilde{E}'_j] \widetilde{loc}'_i(e'_{\setminus i}) + \widetilde{loc}'_i(e'_{\setminus i}) \widetilde{loc}'_j(e'_{\setminus j})}{\widetilde{scale}'_i(e'_{\setminus i}) \widetilde{scale}'_j(e'_{\setminus j})} \\ &\rightarrow \mathbb{E} \left[\left(\frac{\tilde{E}'_i - \widetilde{loc}'_i(e'_{\setminus i})}{\widetilde{scale}'_i(e'_{\setminus i})} \right) \left(\frac{\tilde{E}'_j - \widetilde{loc}'_j(e'_{\setminus j})}{\widetilde{scale}'_j(e'_{\setminus j})} \right) \right] = \left(\frac{\mathbb{E}[\tilde{E}'_i] - \widetilde{loc}'_i(e'_{\setminus i})}{\widetilde{scale}'_i(e'_{\setminus i})} \right) \left(\frac{\mathbb{E}[\tilde{E}'_j] - \widetilde{loc}'_j(e'_{\setminus j})}{\widetilde{scale}'_j(e'_{\setminus j})} \right) \\ &\rightarrow \mathbb{E}[\tilde{Z}'_i \tilde{Z}'_j | E'] = \mathbb{E}[\tilde{Z}'_i | E'] \mathbb{E}[\tilde{Z}'_j | E'] \\ &\rightarrow \mathbb{E}[\tilde{Z}'_i \tilde{Z}'_j | E'] - \mathbb{E}[\tilde{Z}'_i | E'] \mathbb{E}[\tilde{Z}'_j | E'] = 0 \\ &\rightarrow \mathbb{E}[\tilde{Z}'_i \tilde{Z}'_j | E'] - \mathbb{E}[\tilde{Z}'_i | E'] \mathbb{E}[\tilde{Z}'_j | E'] - \mathbb{E}[\tilde{Z}'_i | E'] \mathbb{E}[\tilde{Z}'_j | E'] + \mathbb{E}[\tilde{Z}'_i | E'] \mathbb{E}[\tilde{Z}'_j | E'] = 0 \\ &\rightarrow \mathbb{E}[\tilde{Z}'_i \tilde{Z}'_j | E'] - \mathbb{E}[\tilde{Z}'_j \mathbb{E}[\tilde{Z}'_i | E'] | E'] - \mathbb{E}[\tilde{Z}'_i \mathbb{E}[\tilde{Z}'_j | E'] | E'] + \mathbb{E}[\tilde{Z}'_i | E'] \mathbb{E}[\tilde{Z}'_j | E'] = 0 \\ &\rightarrow \mathbb{E}[(\tilde{Z}'_i - \mathbb{E}[\tilde{Z}'_i | E']) (\tilde{Z}'_j - \mathbb{E}[\tilde{Z}'_j | E']) | E'] = 0 \\ &\rightarrow \text{cov}(\tilde{Z}'_i, \tilde{Z}'_j | E') = 0 \end{aligned}$$

Typically, the aforementioned equalities would be valid for any diffeomorphic solution function $\tilde{s}_i : \tilde{\mathcal{E}}_i \times \mathcal{E}_{\setminus i} \rightarrow \tilde{\mathcal{Z}}_i$. However, in this paper, we specifically focus on solution functions represented by a location-scale noise model.

Assuming that the causal variables follow a **multivariate normal distribution conditional on the pre-intervention exogenous variables**, $\text{cov}(\tilde{Z}'_i, \tilde{Z}'_j | E') = 0$ would imply that $\tilde{Z}'_i \perp\!\!\!\perp \tilde{Z}'_j | E'$.

Let's define $\phi_{\mathcal{E}} = g'^{-1} \circ g : \mathcal{E} \rightarrow \mathcal{E}'$, where g and g' are the decoders in $\mathcal{M}$ and $\mathcal{M}'$. As stated in Assumption 4.3.3, the decoders are diffeomorphisms, hence, $\phi_{\mathcal{E}}$ is a diffeomorphism. Furthermore, let $\tilde{s}(e) = [\tilde{s}_1(e), \tilde{s}_2(e), \dots, \tilde{s}_n(e)]$ be the set of all solution functions post-intervention, which are also diffeomorphisms as stated in Assumption 4.3.3. Consequently:

$$\begin{aligned} & (\phi_{\mathcal{E}}^{-1} \text{ is diffeomorphic}) \forall i, j, i \neq j \quad \tilde{Z}'_i \perp\!\!\!\perp \tilde{Z}'_j | E' \rightarrow \tilde{Z}'_i \perp\!\!\!\perp \tilde{Z}'_j | \phi_{\mathcal{E}}^{-1}(E') \rightarrow \tilde{Z}'_i \perp\!\!\!\perp \tilde{Z}'_j | E \\ & \rightarrow p(\tilde{Z}'_i | E = e) p(\tilde{Z}'_j | E = e) = p(\tilde{Z}'_i, \tilde{Z}'_j | E = e) \\ & (\text{all functions in } \tilde{s} \text{ are diffeomorphism}) \rightarrow p(\tilde{Z}'_i | \tilde{s}(e)) p(\tilde{Z}'_j | \tilde{s}(e)) = p(\tilde{Z}'_i, \tilde{Z}'_j | \tilde{s}(e)) \\ & \rightarrow p(\tilde{Z}'_i | \tilde{Z}) p(\tilde{Z}'_j | \tilde{Z}) = p(\tilde{Z}'_i, \tilde{Z}'_j | \tilde{Z}) \end{aligned}$$

The association between $\tilde{Z}'$ and $\tilde{Z}$ is akin to the relationship between pigeons and pigeonholes. Each causal variable in $\mathcal{M}'$ (a pigeon) must correspond to at least one causal variable in $\mathcal{M}$ (a pigeonhole). If any causal variable in $\mathcal{M}'$ were to depend on more than one causal variable in $\mathcal{M}$, it would be like trying to fit multiple pigeons into the same pigeonhole, creating a dependency between two variables in $\mathcal{M}'$ and violating the equality established earlier. Therefore, no variable in $\mathcal{M}'$ can depend on more than one causal variable in $\mathcal{M}$, ensuring that the transformation $\phi_{\mathcal{Z}}$ is component-wise—each pigeon has its own pigeonhole, or each causal variable in $\mathcal{M}'$ corresponds to exactly one causal variable in $\mathcal{M}$. □

Theorem 4.5.5. (*Identifiability of Latent Causal Models*) Let $\mathcal{M} = (\mathcal{A}, \mathcal{X}, g, \mathcal{I})$ and $\mathcal{M}' = (\mathcal{A}', \mathcal{X}, g', \mathcal{I})$ be two *LCMs* sharing the same observation space $\mathcal{X}$ and intervention targets $\mathcal{I}$. Suppose the following conditions hold:

1. **Data Generating Assumptions:** The assumptions outlined in Assumption 4.3.3 are satisfied.
2. **Observability of Soft Interventions:** Soft interventions adhere to Assumption 4.4.2.

3. **Real-valued Variables:** Both causal and exogenous variables are real-valued.
4. **Multivariate Normality:** The joint distributions of causal and exogenous variables are multivariate normal.

Then, the following statements are equivalent:

1. **Likelihood Equivalence:** The two models assign the same likelihood to both interventional and observational data:

$$p_{\mathcal{M}}^{\mathcal{X}, \mathcal{I}}(x, \tilde{x}, i) = p_{\mathcal{M}'}^{\mathcal{X}, \mathcal{I}}(x, \tilde{x}, i). \quad (4.16)$$

2. **Identifiability:** The models are identified, meaning $\mathcal{M} \sim_r \mathcal{M}'$ as per Definition 4.5.3.

Proof. I will proceed to prove the equivalence between statements 1 and 2 by showing the implication is true in both forward and reverse directions.

Forward Direction: $\mathcal{M} \sim_r \mathcal{M}' \Rightarrow p_{\mathcal{M}}^{\mathcal{X}}(x, \tilde{x}) = p_{\mathcal{M}'}^{\mathcal{X}}(x, \tilde{x})$:

This direction is fairly straightforward. According to Definition 4.5.3, the fact that $\mathcal{M} \sim_r \mathcal{M}'$ implies that $\phi_{\mathcal{E}}$ is measure preserving. Thus:

$$p_{\mathcal{M}'}^{\mathcal{E}'}(e', \tilde{e}') = (\phi_{\mathcal{E}})_* p_{\mathcal{M}}^{\mathcal{E}}(e, \tilde{e}) \quad (4.17)$$

Furthermore, considering that ancestry is preserved, $\phi_{\mathcal{Z}}$ is measure preserving, and that causal variables are obtained from their ancestral exogenous variables in implicit models, we have:

$$p_{\mathcal{M}'}^{\mathcal{Z}'}(z', \tilde{z}') = (\phi_{\mathcal{Z}})_* p_{\mathcal{M}}^{\mathcal{Z}}(z, \tilde{z}) \quad (4.18)$$

Since models are trained to maximize the log likelihood of $p(x, \tilde{x}, \tilde{x} - x)$ and the latent spaces in $\mathcal{M}$ and $\mathcal{M}'$ have the same distribution, the decoders should yield the same observational distributions as:

$$p_{\mathcal{M}}^{\mathcal{X}}(x, \tilde{x}) = p_{\mathcal{M}'}^{\mathcal{X}}(x, \tilde{x}) \quad (4.19)$$

Reverse Direction: $p_{\mathcal{M}}^{\mathcal{X}}(x, \tilde{x}) = p_{\mathcal{M}'}^{\mathcal{X}}(x, \tilde{x}) \Rightarrow \mathcal{M} \sim_r \mathcal{M}'$:

Let's define $\phi_{\mathcal{E}} = g'^{-1} \circ g : \mathcal{E} \rightarrow \mathcal{E}'$. Since we can express $e = s^{-1}(z)$, we can now define $\phi_{\mathcal{Z}}$ as:

$$\phi_{\mathcal{Z}} = s' \circ g'^{-1} \circ g \circ s^{-1} : \mathcal{Z} \rightarrow \mathcal{Z}'. \quad (4.20)$$

Therefore, we have:

$$\phi_{\mathcal{E}} = s'^{-1} \circ \phi_{\mathcal{Z}} \circ s \quad (4.21)$$

Because g and g' are **diffeomorphisms**, $\phi_{\mathcal{E}}$ is a diffeomorphism as well. Furthermore, since $p_{\mathcal{M}}^{\mathcal{X}} = p_{\mathcal{M}'}^{\mathcal{X}}$ and $\phi_{\mathcal{E}}$ is a diffeomorphism, then:

$$p_{\mathcal{M}'}^{\mathcal{E}'} = (\phi_{\mathcal{E}})_* p_{\mathcal{M}}^{\mathcal{E}} \quad (4.22)$$

Consequently, $\phi_{\mathcal{E}}$ is measure-preserving. Similarly, $\phi_{\mathcal{Z}}$ is measure-preserving as well since causal mechanisms are **diffeomorphisms**.

Step 1: Identical correspondence of edges and nodes

Let's define the set U for **atomic** interventions $i \neq j \in I$ as:

$$U = \{(e, \tilde{e}) \mid \text{supp } p_{\mathcal{M}}^{\mathcal{E}, \mathcal{I}}(e, \tilde{e} \mid i) \cap \text{supp } p_{\mathcal{M}}^{\mathcal{E}, \mathcal{I}}(e, \tilde{e} \mid j) \neq \emptyset\} \subset \mathcal{E} \times \mathcal{E} \quad (4.23)$$

Then, assuming **counterfactual exogenous variables**, we have:

$$p_{\mathcal{M}}^{\mathcal{E}, \mathcal{I}}(U \mid i) = p_{\mathcal{M}}^{\mathcal{E}, \mathcal{I}}(U \mid j) = 0 \quad (4.24)$$

We also assume that **all interventions are observed**. Therefore, we can say that:

$$p_{\mathcal{M}}^{\mathcal{E}}(e, \tilde{e}) = \sum_{i \in I} p_{\mathcal{M}}^{\mathcal{E}, \mathcal{I}}(e, \tilde{e} \mid i) p_{\mathcal{M}}^{\mathcal{I}}(i) \quad (4.25)$$

is a discrete mixture of non-overlapping distributions $p_{\mathcal{M}}^{\mathcal{E}, \mathcal{I}}(e, \tilde{e} \mid i)$. Similarly, we can say that $p_{\mathcal{M}'}^{\mathcal{E}'}(e', \tilde{e}')$ is a discrete mixture of non-overlapping distributions. Since $\phi_{\mathcal{E}}$ is measure preserving, $p_{\mathcal{M}}^{\mathcal{E}}(e, \tilde{e}) = p_{\mathcal{M}'}^{\mathcal{E}'}(e', \tilde{e}')$. Therefore,

$$\sum_{i \in I} p_{\mathcal{M}}^{\mathcal{E}, \mathcal{I}}(e, \tilde{e} \mid i) p_{\mathcal{M}}^{\mathcal{I}}(i) = \sum_{i' \in I'} p_{\mathcal{M}'}^{\mathcal{E}', \mathcal{I}'}(e', \tilde{e}' \mid i') p_{\mathcal{M}'}^{\mathcal{I}'}(i') \quad (4.26)$$

It can be concluded that as $\phi_{\mathcal{E}}$ must map between the conditional distributions, there exists a bijection that also induces a permutation $\psi : [n] \rightarrow [n]$. Additionally, since we assume that the **target of interventions are known** and we have a shared domain $\mathcal{I}$ in $\mathcal{M}$ and $\mathcal{M}'$, we have:

$$\sum_{i \in I} p_{\mathcal{M}}^{\mathcal{E}, \mathcal{I}}(e, \tilde{e} \mid i) p_{\mathcal{M}}^{\mathcal{I}}(i) = \sum_{i' \in I'} p_{\mathcal{M}'}^{\mathcal{E}', \mathcal{I}'}(e', \tilde{e}' \mid i') p_{\mathcal{M}'}^{\mathcal{I}'}(i') \quad (4.27)$$

Consequently, the permutation ψ is an identity transformation.

In space $\mathcal{Z}$, the interventions should also be **sufficiently variable** in order to have non-overlapping $p_{\mathcal{M}}^{\mathcal{Z}, \mathcal{I}}(z, \tilde{z} \mid i)$ distributions. In the case of soft interventions, $\tilde{z}$ is affected by all ancestral exogenous variables which could be ancestors of other causal variables as well. Consequently, if the changes in causal mechanisms are not sufficient, the effect of ancestral exogenous variables on causal variables will share some similarities and create overlapping distributions. Similar to space $\mathcal{E}$, we can say that there is a permutation between $p_{\mathcal{M}}^{\mathcal{Z}, \mathcal{I}}(z, \tilde{z} \mid i)$ and $p_{\mathcal{M}'}^{\mathcal{Z}', \mathcal{I}'}(z', \tilde{z}' \mid i')$ as well. Subsequently, as the target of interventions are known, the permutation is an identity transformation.

The effect of soft intervention with known targets on these conditional distributions is shown in Figure 4.4.

Step 2: Component-wise $\phi_{\mathcal{Z}}$

According to Lemma 4.5.4, in order to prove that $\phi_{\mathcal{Z}}$ is a component-wise transformation, we need to prove that $\tilde{E}'_i$ and $\tilde{E}'_j$ are independent $\forall i, j, i \neq j$. In implicit modeling we do not know the parents of each causal variable, hence, we assume the distribution of $\tilde{Z}'_i$ to be conditioned only on E'_i as in Equation 4.35 since E'_i is a known parent of $\tilde{Z}'_i$. The mean of a conditional distribution can be calculated as:

$$\mathbb{E}[\tilde{Z}'_i \mid \tilde{E}'_i = e'_i] = \mu_{\tilde{Z}'_i} + \rho \frac{\sigma_{\tilde{Z}'_i}}{\sigma_{E'_i}} (e'_i - \mu_{E'_i}) \quad (4.28)$$

where ρ and σ are the correlation coefficient and variance of the random variables, respectively. On the other hand, we model $\tilde{Z}'_i$ using switch mechanisms as:

$$\tilde{z}'_i = s_i(\tilde{e}'_i; e'_{\setminus i}, h(v'))$$

By using Taylor's expansion we can write above equation as:

$$\begin{aligned}
s_i(\tilde{e}'_i; e'_{\setminus i}, h_i(v')) &= s_i(\tilde{e}'_i; e'_{\setminus i}, h_i(v'_0)) + \sum_{n=1}^{\infty} \frac{1}{n!} \left(\frac{\partial^n s_i}{\partial h_i^n} \Big|_{h_i=h_i(v'_0)} (h_i(v') - h_i(v'_0))^n \right) \\
&= s_i(\tilde{e}'_i; e'_{\setminus i}, h_i(v'_0)) + R_i
\end{aligned}$$

Furthermore, we assume **separable dependence** such that:

$$\exists v'_0 \text{ such that } \forall i \quad s_i(\tilde{e}'_i; e'_{\setminus i}, h_i(v'_0)) = s_i(\tilde{e}'_i; e'_{\setminus i})$$

An example of such a scenario could be in location-scale noise models, where a soft intervention changes the location parameter of the model as:

$$\begin{aligned}
s_i(e'_i; e'_{\setminus i}) &= e'_i + \text{loc}(e'_{\setminus i}) \rightarrow \tilde{s}_i(\tilde{e}'_i; e'_{\setminus i}) = s_i(\tilde{e}'_i; e'_{\setminus i}, h_i(v')) \\
&= \tilde{e}'_i + \text{loc}(e'_{\setminus i}) + h_i(v') = \tilde{e}'_i + \text{loc}(e'_{\setminus i}) + v'^2 + v'
\end{aligned}$$

In this example, for $v'_0 = 0$, $s_i(\tilde{e}'_i; e'_{\setminus i}, h_i(v'_0)) = s_i(\tilde{e}'_i; e'_{\setminus i})$.

Consequently, we can write the following equality from Equation 4.28:

$$\mathbb{E}[\tilde{Z}'_i | E'_i = e'_i] = \mathbb{E}[s'_i(\tilde{E}'_i; E'_{\setminus i}) + R_i | E'_i = e'_i] = \mu_{\tilde{Z}'_i} + \rho \frac{\sigma_{\tilde{Z}'_i}}{\sigma_{E'_i}} (e'_i - \mu_{E'_i})$$

By taking the partial derivative of both side with respect to $\tilde{E}'_j$ we have:

$$\forall j \neq i \quad \mathbb{E} \left[\frac{\partial s'_i(\tilde{E}'_i; E'_{\setminus i})}{\partial \tilde{E}'_i} \cdot \frac{\partial \tilde{E}'_i}{\partial \tilde{E}'_j} + \frac{\partial s'_i(\tilde{E}'_i; E'_{\setminus i})}{\partial E'_{\setminus i}} \cdot \frac{\partial E'_{\setminus i}}{\partial \tilde{E}'_j} + \frac{\partial R_i}{\partial \tilde{E}'_j} \Big| E'_i = e'_i \right] = 0$$

If we did not have the causal mechanism switch variable $h_i(v')$ (no R_i term), the equation above would only hold if s_i was constant in parents ($E_{/i}$), which is not the case due to the presence of soft interventions, or:

$$\frac{\partial s'_i(\tilde{E}'_i; E'_{\setminus i})}{\partial \tilde{E}'_i} \cdot \frac{\partial \tilde{E}'_i}{\partial \tilde{E}'_j} = - \frac{\partial s'_i(\tilde{E}'_i; E'_{\setminus i})}{\partial E'_{\setminus i}} \cdot \frac{\partial E'_{\setminus i}}{\partial \tilde{E}'_j}, \quad (4.29)$$

then the latter scenario would imply that $\frac{\partial \tilde{E}'_i}{\partial \tilde{E}'_j} \neq 0$, hence, $\tilde{E}'_i \not\perp \tilde{E}'_j$. However, by introducing the causal mechanism switch variable V and assuming it is observed, we can account for the effects of soft interventions through $h_i(v')$. In this case, $\frac{\partial \tilde{E}'_i}{\partial \tilde{E}'_j} = 0$ as exogenous variables are commonly assumed to be independent in practice. Consequently:

$$\forall i, j \quad \tilde{E}'_i \perp \tilde{E}'_j \rightarrow \phi_Z \text{ is a component-wise transformation.}$$

Step 3: Component-wise $\phi_{\mathcal{E}}$

Using the result from previous step that ϕ_Z is a component-wise transformation, the string diagrams for connections between E and E' will be as shown in Figure 4.5. $\phi_{\mathcal{E}}(e)_{i'} = (s'^{-1} \circ \phi_Z \circ s)(e)_{i'}$ will only depend on E_A , where $A = \text{anc}_i$ is the ancestors of variable i , and e_i . Because:

$$\begin{aligned} & (\text{permutations of indices is identity}) \rightarrow \phi_{\mathcal{E}}(e)_{i'} = \phi_{\mathcal{E}}(e)_i \\ & (s'^{-1} \text{ only depends on } z'_i \text{ and } z'_{\text{anc}_i}) \rightarrow \phi_{\mathcal{E}}(e)_i = s'^{-1}_i(z'_i, z'_{\text{anc}_i}) \\ & (\phi_Z \text{ is a component-wise transformation}) \rightarrow s'^{-1}_i(z'_i, z'_{\text{anc}_i}) = s'^{-1}_i(\phi_Z(z_i)_i, \phi_Z(z_{\text{anc}_i})_{\text{anc}_i}) \\ & (s_i \text{ and } s_{\text{anc}_i} \text{ only depend on } e_i \text{ and } e_{\text{anc}_i}) \rightarrow s'^{-1}_i(\phi_Z(z_i)_i, \phi_Z(z_{\text{anc}_i})_{\text{anc}_i}) = \\ & s'^{-1}_i(\phi_Z(s_i(e_i, e_{\text{anc}_i}))_i, \phi_Z(s_{\text{anc}_i}(e_{\text{anc}_i}))_{\text{anc}_i}) \end{aligned}$$

Please note that the arguments of the solution functions presented above reflect their dependencies, rather than the precise inputs used in the implementation. The first equality in Figure 4.5 follows from the definition of $\phi_{\mathcal{E}_i}$. The second equality holds when we first apply ϕ_{Z_A} and then apply the causal mechanisms. It can be concluded from the most right-hand side diagram in Figure 4.5 that the transformation from $\mathcal{E}'_i \times \mathcal{E}_A \rightarrow \mathcal{E}'_i$ is constant in $\mathcal{E}_A$. Therefore, $\phi_{\mathcal{E}_i}$ is a component-wise transformation. □

The Gaussian assumptions in this theorem are necessary for the theoretical proof, and the synthetic datasets used to generate the results in Table 4.4 adhere to this assumption. However, in real-world dataset experiments (Table 4.6), the Gaussian assumption may not hold, as the distributions of causal variables are unknown. Additionally, other assumptions, such as the presence of diffeomorphic and linear decoders, might be violated, given that the underlying generative processes of these datasets remain unknown.

Despite these violations, Table 4.6 demonstrates that the proposed method performs well empirically, suggesting its robustness to deviations from theoretical assumptions.

4.6 Training Objective

Consequently, the [ICRL-SM](#) model will include three latent variables:

1. A causal mechanism switch variable V .
2. The pre-intervention exogenous variables E .
3. The post-intervention exogenous variables $\tilde{E}$.

Given that the data log-likelihood $\log p(x, \tilde{x}, \tilde{x} - x) \equiv \log p(x, \tilde{x})$ is intractable, we adopt an [ELBO](#) approximation as the training objective:

$$\begin{aligned} \log p(x, \tilde{x}) &\geq \mathbb{E}_{q(e, \tilde{e}, v|x, \tilde{x})} [\log p(x, \tilde{x}|e, \tilde{e}, v)] - \text{KLD}(q(e, \tilde{e}, v|x, \tilde{x}) \| p(e, \tilde{e}, v)) \\ &= \mathbb{E}_{q(v|\tilde{x}-x) \cdot q(e|x) \cdot q(\tilde{e}|\tilde{x})} [\log(p(x|e)p(\tilde{x}|\tilde{e})p(\tilde{x}-x|v))] - \text{KLD}(q(v|\tilde{x}-x) \cdot q(e|x) \cdot q(\tilde{e}|\tilde{x}) \| p(\tilde{e}|e, v)p(v)p(e)). \end{aligned} \quad (4.30)$$

Here, the observations are encoded and decoded independently. The [KLD](#) term regularizes the encodings to share the latent *intervention model* $p(\tilde{e}|e, v)p(v)p(e)$, which is consistent across all data points. The components of this model can be interpreted as follows:

1. $p(e)$ is the prior distribution over exogenous variables e .
2. $p(v)$ is the prior distribution over switch variables v .
3. $p(\tilde{e}|e, v)$ is a transition model that describes how the exogenous variables change as a function of the intervention.

We factorize the posterior using a mean-field approximation:

$$q(v, e, \tilde{e}|x, \tilde{x}) = q(v|\tilde{x} - x) \cdot q(e|x) \cdot q(\tilde{e}|\tilde{x}). \quad (4.31)$$

According to our data generation model [\(4.3\)](#), the reconstruction probability is factorized as:

$$p(x, \tilde{x}|e, \tilde{e}, v) = p(x|e)p(\tilde{x}|\tilde{e})p(\tilde{x} - x|v). \quad (4.32)$$

The prior over latent variables is factorized as:

$$p(\tilde{e}, e, v) = p(\tilde{e}|e, v)p(v)p(e) \quad (4.3). \quad (4.33)$$

Since the pre-intervention exogenous variables are mutually independent, we have:

$$p(e) = \prod_i p(e_i) \quad \text{and} \quad p(v) = \prod_i p(v_i). \quad (4.34)$$

We assume $p(e_i)$ and $p(v_i)$ to follow standard Gaussian distributions. Furthermore, since $e_i = \tilde{e}_i$ for all non-intervened variables, the distribution $p(\tilde{e}|e, v)$ is given by:

$$p(\tilde{e}|e, v) = \prod_{i \notin I} \delta(\tilde{e}_i - e_i) \prod_{i \in I} p(\tilde{e}_i|e, v) = \prod_{i \notin I} \delta(\tilde{e}_i - e_i) \prod_{i \in I} p(\tilde{z}_i|e_i) \left| \frac{\partial \tilde{z}_i}{\partial \tilde{e}_i} \right|. \quad (4.35)$$

The last equality in (4.35) is derived from the Change of Variable Rule in probability theory, applied to the solution function $\tilde{z}_i = s_i(\tilde{e}_i; e_{\setminus i}, h_i(v))$. Moreover, we express $p(\tilde{z}_i|e, v)$ as $p(\tilde{z}_i|e_i)$ since only e_i is a known parent of $\tilde{z}_i$ in implicit modeling. We assume $p(\tilde{z}_i|e_i)$ to be a Gaussian distribution, with its mean determined by e_i .

4.7 Solution Functions

We implement the solution function using location-scale noise models [24], as also applied in [6], which define an invertible diffeomorphism. For simplicity, in our experiments, we will only modify the *loc* network in the post-intervention phase. Therefore, $h_i(v)$ will be used as follows:

$$\tilde{z}_i = \tilde{s}_i(\tilde{e}_i; e_{\setminus i}, h_i(v)) = \frac{\tilde{e}_i - (\text{loc}_i(e_{\setminus i}) + h_i(v))}{\text{scale}_i(e_{\setminus i})}, \quad (4.36)$$

where $\text{loc}_i : \mathbb{R}^{n-1} \rightarrow \mathbb{R}$ and $\text{scale}_i : \mathbb{R}^{n-1} \rightarrow \mathbb{R}$ are fully connected networks calculating the first and second moments, respectively. The general overview of the model is illustrated in Figure 4.6.

A Comparison of ICRL-SM with other methods is given in Table 4.1.

4.8 Results and Discussion

1. **Recovering Causal Variables:** I validate my theorem using synthetic datasets to ensure its theoretical soundness. These experiments serve as a *proof of concept* by evaluating our proposed model under conditions where all assumptions are met. Specifically, I assess the model’s performance using a Gaussian data distribution and a linear mixing function. In these experiments, causal disentanglement is employed to identify the true causal graph from pairs of observations $(x, \tilde{x}, i)$.

Table 4.1: Comparison of **ICRL-SM** with other recent related work on causal learning from interventional data

Methods	Causal Mechanism	Mixing Function	Intervention	Learning Method	Identifiability
CausalDisc. [70]	Nonlinear	Full row rank polynomial	Soft	Explicit	Permutation & Affine
CauCA [37]	Nonlinear	Diffeomorphism	Soft	Explicit	Different based on assumptions
Linear-CD [59]	Linear	Linear	Hard	Explicit	Permutation
Scale-I [62]	Nonlinear	Linear	Hard/Soft	Explicit	Scale/Mixed
ILCM [6]	Nonlinear	Diffeomorphism	Hard	Implicit	Permutation & reparameterization
dVAE [44]	Nonlinear	Diffeomorphism	Hard	Implicit	Permutation & reparameterization
ICRL-SM [4]	Nonlinear	Diffeomorphism	Soft	Implicit	Reparameterization

2. **Action and Object Inference:** I apply my model to real-world datasets, namely Epic-Kitchen and ProcTHOR from the Causal-Triplet benchmark [41]. These datasets may not fully satisfy all assumptions, such as the diffeomorphic decoder and the observability of the switch variable (V). For these experiments, I utilize a ResNet-based decoder. Since the data distribution in real-world scenarios is often unknown and may not follow a Gaussian distribution, I use action and object accuracy as proxies to evaluate the effectiveness of our method in uncovering causal variables compared to other approaches. I employ two classifiers that take as input the pre- and post-intervention causal variables to predict the action and object class.

Additionally, we conducted further experiments designed as an ablation study, with results presented in Section 4.8.6. All models were trained under the same settings and with data where the intervention targets are known. In our real-world experiments, however, the intervention targets are unknown, and we hypothesize them. The results from [6] suggest that inferring intervention targets is relatively straightforward, and a similar procedure can be applied in applications where these targets are not known.

4.8.1 Datasets

Synthetic Dataset

I generate simple synthetic datasets with $\mathcal{X} = \mathcal{Z} = \mathbb{R}^n$. For each value of n , we create ten random DAGs, a random location-scale SCM, and then generate a random dataset from the parameterized SCM. To generate these random DAGs, each edge is sampled in a fixed topological order from a Bernoulli distribution with a probability of 0.5. The pre-intervention and post-intervention causal variables are defined as:

$$z_i = \text{scale}(z_{pa_i})e_i + \text{loc}(z_{pa_i}) \xrightarrow{\text{Soft-Intervention}} \tilde{z}_i = \text{scale}(z_{pa_i})\tilde{e}_i + \widetilde{\text{loc}}(z_{pa_i}), \quad (4.37)$$

where the *loc* and *scale* networks are modified in the post-intervention phase. The pre-intervention *loc* and post-intervention $\widetilde{\text{loc}}$ network weights are initialized with samples drawn from $\mathcal{N}(0,1)$ and $\mathcal{N}(3,1)$, respectively. The *scale* is constant at 1 for both pre-intervention and post-intervention samples. Both e_i and $\tilde{e}_i$ are sampled from a standard Gaussian distribution. The causal variables are mapped to the data space through a randomly sampled $SO(n)$ rotation. For each dataset, we generate 100,000 training samples, 10,000 validation samples, and 10,000 test samples.

Real-world Dataset

The Causal-Triplet datasets, designed for *actionable* counterfactuals [41], feature paired images where various global scene properties, such as camera view and object occlusions, may vary (See Figure 4.2). Thus, these images can be interpreted as outcomes of soft interventions, where actions affect objects alongside subtle alterations. The datasets include:

- **ProcTHOR** [14] images obtained from a photo-realistic simulator of embodied agents. This dataset contains 100k images in which 7 types of actions manipulate 24 types of objects in 10k distinct indoor environments. Examples of actions and objects in this dataset are given in Table 4.2. I use 5000 images of this dataset to train all models. The number of causal variables is set to 7 in this dataset.
- **Epic-Kitchen** [12] images repurposed from a real-world video dataset of human-object interactions. This dataset comprises image pairs, collected under a similar setup from the Epic-Kitchens dataset, with 97 actions manipulating 277 objects. Examples of actions and objects in this dataset are given in 4.3. For this dataset, I randomly select 20 actions from the dataset, as 97 causal variables would be too complex in a VAE setup which results in 2,632 images.

Table 4.2: Actions and objects present in the Causal-Triplet images (ProcTHOR Dataset).

ProcTHOR Dataset							
Object	Television	Bed	Bed	Television	Laptop	Book	Box
Action	Break	Clean	Dirty	Turn off	Turn on	Open	Close

Table 4.3: Actions and objects present in the Causal-Triplet images (Epic-Kitchens Dataset).

Epic-Kitchens Dataset										
Object	Tofu	Rice	Hob	Bag	Cupboard	Garlic	Tap	Wrap	Rice	Cheese
Action	Insert	Pour	Wash	Fold	Open	Pat	Move	Check	Transition	Stretch
Object	Wrap	Skin	Button	Lid	Plate	Egg	Sponge	Oil	Water	Dough
Action	Flip	Gather	Press	Lock	Wrap	Drop	Water	Carry	Smell	Mark

Based on the nature of the actions in this dataset, the causal variables should represent attributes of objects such as shape and color. Therefore, actions in these datasets are considered as interventions. For example, consider that z_1 corresponds to the attributes of an object, e.g. a door, the target of opening or closing (action’s target) is z_1 . We use actions’ labels in these datasets to detect the targets of interventions to determine which causal variable has been intervened upon. Note that informing the model about the target of intervention is not same as informing about the action itself (See Table 4.6). Since the dataset consists of images, I train all methods with a ResNet encoder and decoder.

In this task, action/object accuracy is used as a proxy to evaluate the effectiveness of different methods in identifying the causal structure, as the ground truth causal variables are not available.

4.8.2 Metrics

For the causal disentanglement task, I employ the Disentanglement (D), Completeness (C), and Informativeness (I) scores, commonly referred to as DCI scores [18]. According to [18], Disentanglement (D) and Completeness (C) are sufficient to evaluate the correspondence between the learned latents and true causal variables. Therefore, I utilize the following metrics to assess my model’s performance.

- **Causal Disentanglement:** The causal disentanglement score quantifies the extent to which the modeled causal variable Z_i factorizes or disentangles the ground truth

causal variable Z_i^* . Causal disentanglement D_i for Z_i is calculated as:

$$D_i = (1 - H_K(P_{i.})) = \left(1 + \sum_{k=0}^{K-1} P_{ik} \log_K P_{ik}\right), \quad (4.38)$$

where $P_{ij} = \frac{R_{ij}}{\sum_{k=0}^{K-1} R_{ik}}$, and R_{ij} denotes the probability of Z_i being important for predicting Z_j^* . Total causal disentanglement is the weighted average $\sum_i \rho_i D_i$, where $\rho_i = \frac{\sum_j R_{ij}}{\sum_{ij} R_{ij}}$.

- **Causal Completeness:** Causal completeness quantifies the degree to which each ground truth causal variable Z_i^* is captured by a single modeled causal variable Z_i . Causal completeness is calculated as:

$$C_j = (1 - H_D(\tilde{P}_{.j})) = \left(1 + \sum_{d=0}^{D-1} \tilde{P}_{dj} \log_D \tilde{P}_{dj}\right). \quad (4.39)$$

D in (4.38) and K in (4.39) are equal to the dimensions of Z and Z^* , which is n .

- **Classification Accuracy:** DCI scores [18] require ground truth causal variables, which are not available in the causal-triplet datasets. Therefore, in our experiments with real-world datasets, we use classification accuracy as the evaluation metric for the action inference task.

4.8.3 Causal Model Recovery

I generated a dataset for soft interventions and trained the ICRL-SM [4], ILCM [6], β -VAE [22], and dVAE [44] models using 10 different seeds, each corresponding to a distinct causal graph. We selected four causal variables to represent complex causal structures, including forks, chains, and colliders. Table 4.4 presents the Causal Disentanglement and Causal Completeness scores for all models, evaluated on the test data.

It is important to note that the model proposed in [44] utilizes an averaging function applied to the posterior of their causal variables as a substitute for one of their constraints, which is not applicable in our setting. The naming conventions in [44] are based on this averaging function. Following [6], we also refer to this as disentangled VAE (dVAE).

The results in Table 4.4 demonstrate that our method, ICRL-SM, successfully identifies the true causal graph in most cases. However, the weakest performance is observed for graphs $G5$ and $G10$. As discussed in [34, 70], causal graphs tend to be sparse. In the

Table 4.4: Comparison of identifiability results in synthetic dataset

Graph		Causal Disentanglement				Causal Completeness			
Model	Name	β -VAE [22]	dVAE [44]	ILCM [6]	ICRL-SM	β -VAE [22]	dVAE [44]	ILCM [6]	ICRL-SM
	G1	0.38	0.54	0.71	0.82	0.51	0.69	0.78	0.87
	G2	0.30	0.72	0.75	0.83	0.49	0.77	0.80	0.87
	G3	0.28	0.51	0.68	0.98	0.49	0.56	0.78	0.98
	G4	0.16	0.50	0.65	0.68	0.38	0.69	0.77	0.78
	G5	0.27	0.44	0.53	0.42	0.45	0.54	0.66	0.50
	G6	0.52	0.62	0.71	0.98	0.66	0.69	0.86	0.98
	G7	0.39	0.49	0.71	0.75	0.70	0.73	0.89	0.89
	G8	0.47	0.54	0.50	0.59	0.6	0.63	0.62	0.68
	G9	0.30	0.68	0.83	0.85	0.40	0.76	0.86	0.87
	G10	0.39	0.39	0.52	0.32	0.53	0.56	0.82	0.70

case of $G5$, where the graph is fully connected, our method struggles to accurately identify the causal variables. In the next experiment, we will investigate the factors influencing causal disentanglement, such as the number of edges in the graph and the intensity of the soft intervention effect. These insights may clarify why **ICRL-SM** fails to identify causal variables in $G10$, despite its sparsity.

4.8.4 Factors Affecting Causal Disentanglement

In this experiment, I focus on the graph $G3$, which exhibits the highest identifiability, and vary both the intensity of soft interventions and the number of edges in its data generation process. To adjust the intervention intensity, I initialize the post-intervention $\widehat{loc}$ network weights with samples drawn from $\mathcal{N}(1, 1)$ (nearly identical to loc) and $\mathcal{N}(10, 1)$ (markedly different from loc). To modify the number of edges, I consider both a chain and a fully-connected graph.

The results in Table 4.5 further validate the sparsity of causal graphs, as causal disentanglement significantly deteriorates in the fully-connected graph compared to the default structure of $G3$. The results for markedly different post-intervention causal mechanisms

suggest that the switch variable (V) is insufficient to approximate the intense effects of soft interventions, indicating the need for greater supervision to observe V . On the other hand, post-intervention causal mechanisms that are similar to pre-intervention ones lack the variability needed to disentangle the causal variables, as discussed in Theory 4.5.5.

Table 4.5: Table shows the comparison of ICRL-SM performance on different configurations of $G3$ in Table 4.4.

Edges	Post-intervention causal mechanism	Causal Disentanglement	Causal Completeness
Chain	Default	0.98	0.98
Full	Default	0.89	0.89
Default	Significantly different	0.68	0.73
Default	Almost similar	0.85	0.86

4.8.5 Action Inference

In this experiment, I evaluate the performance of ICRL-SM on the real-world Causal-Triplet datasets. In these datasets, the soft intervention effects, represented by V , are not directly observable. However, our findings indicate that incorporating soft interventions via V results in superior performance compared to other implicit modeling approaches. Clearly, a better understanding of the impact of soft interventions on the generative process of the dataset would lead to improved outcomes.

The results in Table 4.6 show that when all causal variables are used to predict actions, ICRL-SM performs on par with the baseline methods. However, including all causal variables in action or object inference may introduce spurious correlations. Therefore, I also experimented with including only the relevant causal variables in action and object inference. In this scenario, ICRL-SM significantly outperforms the baseline methods, indicating its superior ability to disentangle causal variables. I also compared ICRL-SM with explicit causal representation learning methods. ENCO [38] and DDS [7] utilize a variable topological order of causal variables during training. Additionally, I included a specific setting where the topological order is fixed during training. As shown in Table 4.7, our proposed method outperforms these explicit models as well.

The original images in these datasets had higher resolutions. The resolution was reduced to 64×64 to avoid the challenges associated with higher input resolutions. Specifically,

when using higher resolutions, the latent dimension in an autoencoder framework must also be increased to adequately reconstruct the input images. However, larger latent dimensions often lead to training instability, making the optimization process more difficult. Reducing the image resolution mitigates these challenges, ensuring a more stable training process while maintaining a balance between reconstruction quality and model performance.

Table 4.6: Table comparing action and object accuracy across various methods on Causal-Triplet datasets under different settings. z and z_i show whether all causal variables (z), or only the intervened casual variable (z_i) are used for the prediction task. R_{64} denote images with resolutions 64×64 .

Method	Epic-Kitchens				ProcTHOR			
	Action Accuracy		Object Accuracy		Action Accuracy		Object Accuracy	
	$z; R_{64}$	$z_i; R_{64}$	$z; R_{64}$	$z_i; R_{64}$	$z; R_{64}$	$z_i; R_{64}$	$z; R_{64}$	$z_i; R_{64}$
β -VAE [22]	0.27	0.18	0.19	0.06	0.39	0.30	0.44	0.37
dVAE [44]	0.19	0.69	0.20	0.17	0.35	0.81	0.40	0.78
ILCM [6]	0.21	0.59	0.14	0.14	0.30	0.70	0.41	0.76
ICRL-SM	0.16	0.86	0.16	0.18	0.28	0.93	0.40	0.82

Table 4.7: Action and object accuracy of the explicit models with experiments conducted applying an image with resolution of R_{64} as the input to the Resnet50 encoder with the intervened causal variable (z_i). Fixed-order denotes that the causal variables’ order in the adjacency matrix remains unchanged.

Datasets	Methods	Action Accuracy	Object Accuracy
Epic-Kitchens	ENCO [38]	0.69	0.13
	DDS [7]	0.44	0.09
	Fixed-order	0.79	0.14
	ICRL-SM	0.86	0.18
ProcTHOR	ENCO [38]	0.45	0.53
	DDS [7]	0.64	0.67
	Fixed-order	0.65	0.54
	ICRL-SM	0.93	0.82

4.8.6 Ablation Studies

I evaluated the performance of our models using a simpler backbone, ResNet18, to assess its impact on results. The input images had a resolution of 64×64 , and we utilized the intervened causal variables to predict action and object classes. The results are presented in Table 4.9 and Table 4.8. The findings indicate that the proposed method consistently outperforms both explicit and implicit models, even when using this simpler ResNet18 architecture.

Table 4.8: Table comparing action and object accuracy across various methods on Causal-Triplet datasets using ResNet18 model.

Method	Epic-Kitchens		ProcTHOR	
	Action Accuracy	Object Accuracy	Action Accuracy	Object Accuracy
β -VAE [22]	0.15	0.04	0.20	0.36
dVAE [44]	0.16	0.02	0.15	0.38
ILCM [6]	0.19	0.04	0.15	0.42
ICRL-SM	0.35	0.04	0.40	0.69

Table 4.9: Action and object accuracy of three explicit models are compared with ICRL-SM. Experiments are conducted applying image with resolution of R_{64} as the input to the Resnet18 encoder with the intervened casual variable (z_i).

Datasets	Methods	Action Accuracy	Object Accuracy
Epic-Kitchens	ENCO [38]	0.14	0.03
	DDS [7]	0.16	0.05
	Fixed-order	0.14	0.05
	ICRL-SM	0.35	0.04
ProcTHOR	ENCO [38]	0.16	0.28
	DDS [7]	0.34	0.35
	Fixed-order	0.34	0.38
	ICRL-SM	0.40	0.69

4.8.7 Scalability

While my primary research focus was on addressing identifiability challenges in implicit causal models under soft interventions, I also explored the scalability of our proposed model. To thoroughly evaluate its performance, I conducted experiments with a range of causal graphs containing between 5 and 10 variables. For each variable count, I used 10 different seeds, employing a similar experimental setup as used in our 4-variable causal graph experiments. The results comparing [ICRL-SM](#) [4] and [ILCM](#) [6] are shown in Figure 4.7. As the number of variables in the graph increases, so do the confounding factors and ambiguities in causal relations. This necessitates more supervision on V to effectively disentangle the effects of causal variables on the observed variables.

4.9 Summary

The key findings of [ICRL-SM](#) can be summarized as follows:

- Explicitly modeling the effects of soft intervention using switch variables VV helps achieve identifiability from soft intervention.
- Various factors, such as the number of edges in the graph, the number of variables in the graph, and the effect of soft interventions, can influence disentanglement.
- [ICRL-SM](#) relies on several assumptions about the data and model structure, such as differentiable decoders and causal mechanisms, which can limit its application.



Figure 4.2: In the Causal-Triplet dataset [41], visual representations illustrate both pre- and post-intervention scenarios. The first two rows display data samples from *Epic-Kitchens* dataset, while the third and fourth rows feature samples from *ProcTHOR* dataset. Each image in the post-intervention condition is labeled with the corresponding action and intervened object. In the images from the first two rows, the agent is performing an action on an object, but the camera angle has also changed. Therefore, we can infer that, for example, the distribution of causal variables, conditioned on the camera angle, has been altered due to the soft intervention.

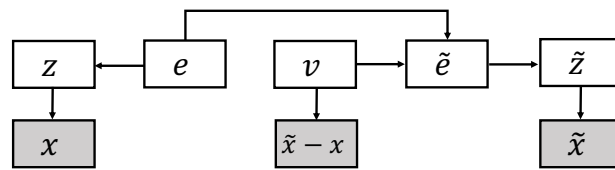


Figure 4.3: Generative model

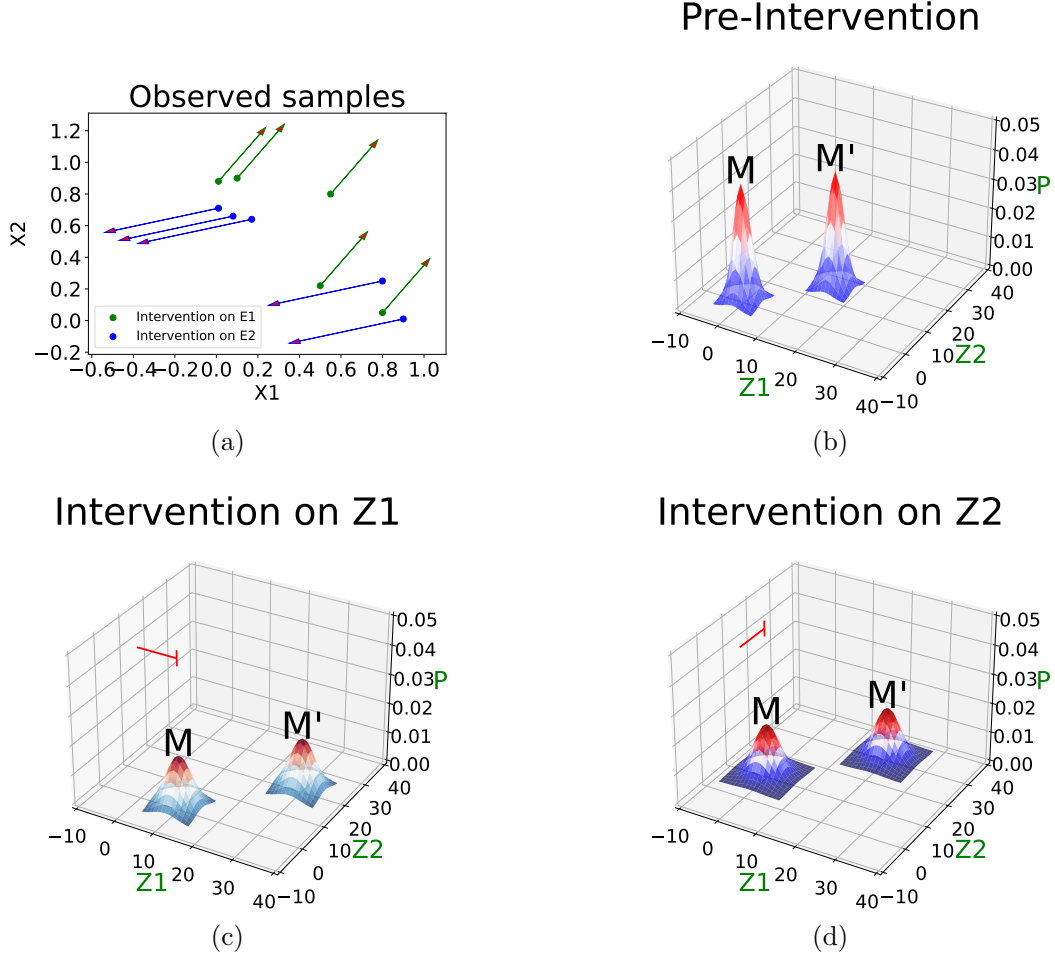


Figure 4.4: Distribution of observed and causal variables in two causal models, $\mathcal{M}$ and $\mathcal{M}'$, within the same equivalence class up to reparameterization. (a) Ten observed samples with interventions on either Z_1 or Z_2 . (b) For no intervention ($I = 0$), the distributions of causal variables in both models are identical, though their value ranges differ and can be aligned using ϕ_Z . (c) Intervention on Z_1 ($I = 1$). (d) Intervention on Z_2 ($I = 2$). While the distributions for $I = 1$ and $I = 2$ are consistent between models, they vary with different intervention targets as soft interventions adjust the conditional distribution of causal variables (conditioned on their parents). For each intervention target I , the distributions in $\mathcal{M}$ and $\mathcal{M}'$ should shift consistently, given the known targets.

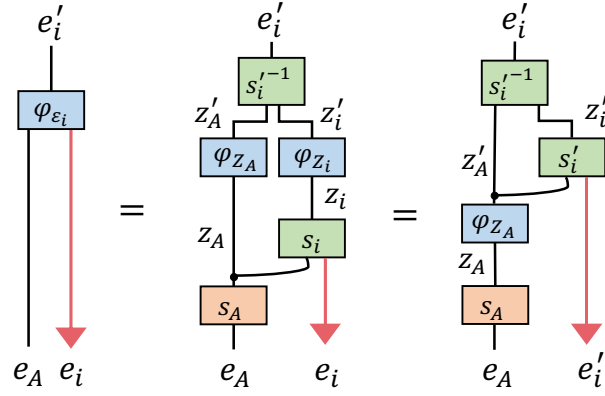


Figure 4.5: String diagrams for connections between E and E' . The triangle indicates sampling variables from their corresponding distributions.

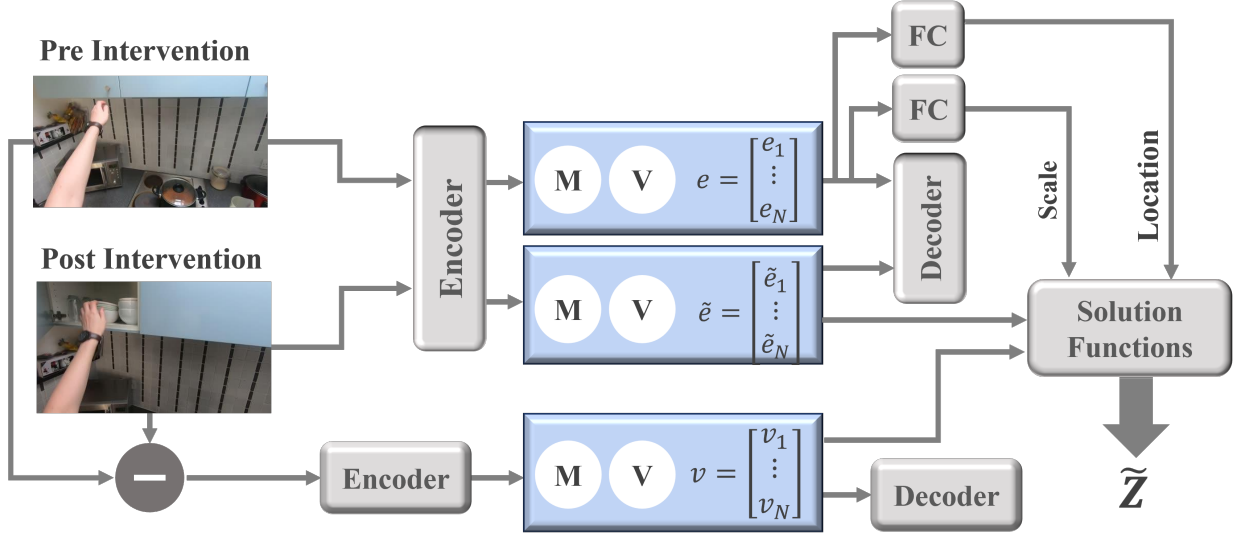


Figure 4.6: The architecture processes pre-intervention observations X , post-intervention observations $\tilde{X}$, and their differences $X - \tilde{X}$ (intervention displacement), encoding them into latent representations. Each encoder outputs the mean (M) and variance (V) of a probability distribution function. By sampling from these distributions, we obtain pre-intervention exogenous variables E , post-intervention exogenous variables $\tilde{E}$, and the causal mechanism switch variable V . The exogenous variables are derived from the corresponding pre- and post-intervention encodings, while V is obtained from the encoding of the differences between X and $\tilde{X}$. The pre- and post-intervention exogenous variables are then passed through two fully connected (FC) layers, which predict the scale and location parameters. These predicted scale and location parameters, together with the post-intervention exogenous variables and the causal mechanism switch variable, are utilized in the solution function (4.36) to compute the post-intervention causal variables $\tilde{Z}$. Here, N denotes the total number of causal variables.

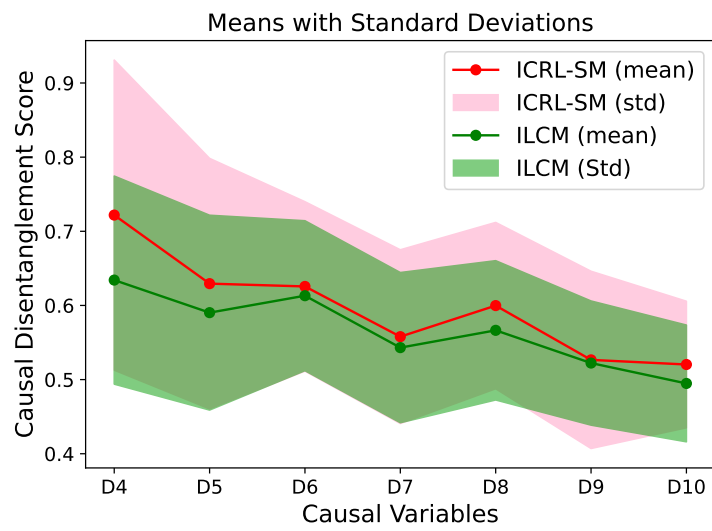


Figure 4.7: Causal disentanglement for different number of causal variables.

Chapter 5

Extensions and Future Works

5.1 Future Works of GCRL

1. Model Architecture: Enhancing Identifiability in GCRL

The current GCRL model architecture employs a piece-wise affine injective function, which is crucial for achieving identifiability. Identifiability refers to the ability of a model to recover the true underlying causal structure or representation uniquely from the observed data. The piece-wise affine injective function ensures that the model can separate different causal mechanisms, thereby facilitating the identification of invariant features.

While GCRL has been applied specifically to models with an EID architecture, future research could explore the performance of GCRL when integrated with other types of decoders and model architectures. Such studies would investigate whether identifiability can still be achieved with alternative architectural choices and under what conditions. This exploration could yield insights into the generalizability and robustness of GCRL across various model configurations.

2. Expanding GCRL to Other Applications

GCRL has primarily been applied within HTP, where the focus is on understanding and predicting phenotypic traits based on genotypic data. While promising, the broader applicability of GCRL to other domains remains unexplored. Future research should investigate how GCRL can be adapted and applied to different fields, such as robotics, healthcare, finance, or any domain where causal representation learning is critical. This exploration could establish the versatility of GCRL and potentially lead to novel applications where the method proves beneficial.

3. Identifiability of Invariant Features (Z)

In Section 3.2.7, the challenge of identifying invariant features, denoted as Z , which represent the underlying causal factors that remain constant across different environments or tasks, is discussed. The identification of Z is crucial for ensuring that the learned representations are robust and generalizable.

It is noted that relying solely on physical laws or trajectories may not be sufficient for identifying these invariant features. Additional information, such as domain knowledge, auxiliary tasks, or specific constraints, may be required to accurately identify Z . Future work should focus on determining the types of additional information necessary and how they can be incorporated into the GCRL framework to improve the identifiability of invariant features. This research could lead to more precise and reliable causal representations, thereby enhancing the overall performance of GCRL in various settings.

4. Heterogeneous Data

GCRL currently relies solely on trajectory data. In the proposed causal graph shown in Figure 3.1, X represents past trajectories. A potential extension of this work involves incorporating additional modalities, such as images, to enhance scene understanding. By introducing image data, new variables would need to be defined in the causal model to represent visual information. This integration would also require utilizing models capable of processing both trajectory and image data effectively.

A promising direction for future work is to combine trajectories with scene images, similar to [32], to provide richer contextual information. This could improve trajectory prediction by capturing spatial semantics and environmental cues that influence movement patterns. Additionally, exploring multi-modal deep learning architectures, such as spatiotemporal transformers or attention-based fusion models, could further enhance predictive accuracy. Investigating how causal relationships evolve when multiple data sources are introduced would also be an interesting research avenue, potentially leading to more robust and interpretable models.

5.2 Future Works of ICRL-SM

ICRL-SM relies on several strict assumptions that could be addressed in future research to bring the method closer to real-world applications.

1. Availability of All Interventions

In the current framework of [ICRL-SM](#), it is assumed that all possible interventions are available for learning the causal representation. However, in real-world scenarios, it is often the case that only a subset of interventions is accessible, leading to partially available interventions. The identifiability of all causal variables under such conditions has yet to be thoroughly studied. Future work could explore methods to achieve identifiability when only partial interventions are available, which would significantly enhance the practicality and applicability of [ICRL-SM](#) in more complex, real-world environments.

2. Multivariate Normal Distribution for Causal Variables

The identifiability results in [ICRL-SM](#) have been proven under the assumption that the causal variables follow a multivariate normal distribution. While this assumption simplifies the theoretical analysis, it may not hold in many real-world cases where causal variables exhibit more complex, non-normal distributions. Future research could investigate the identifiability of causal variables when they follow other types of distributions. Exploring a broader range of distributions could make the [ICRL-SM](#) framework more robust and applicable to a wider variety of real-world data.

3. Diffeomorphic Decoders and Causal Mechanisms

The current framework assumes that both the decoders and the causal mechanisms are diffeomorphic, meaning that they are smooth and invertible functions. While the assumption of diffeomorphic decoders is relatively common, assuming that the causal mechanisms themselves are diffeomorphic is quite strong and may not always be realistic. Future research could investigate the implications of relaxing the diffeomorphic assumption on causal mechanisms. This would involve studying how the model’s performance and identifiability are affected when non-diffeomorphic mechanisms are introduced, potentially leading to a more flexible and realistic modeling approach.

4. Atomic Interventions and Counterfactual Exogenous Variables

The current [ICRL-SM](#) framework assumes atomic interventions, where each intervention affects only one causal variable at a time, and the exogenous variables are considered in a counterfactual context. These assumptions help ensure that there are no overlapping effects in the observed variables, which simplifies the analysis. However, in real-world scenarios, multiple interventions can occur simultaneously, and their effects can overlap. Future work could explore the relaxation of these assumptions by allowing for multiple interventions within a single data sample. This would involve developing new methods to disentangle the overlapping effects and improve the model’s ability to handle more complex intervention scenarios.

5. Observability of Switch Variable V

In the current implementation, the switch variable V is modeled by subtracting post-intervention samples from pre-intervention samples, particularly when using linear decoders. This approach assumes that V can be directly observed or inferred in this manner. However, when dealing with nonlinear decoders, this simple subtraction may not suffice, and other data augmentation techniques may be required to capture the information about V . Future research could focus on developing methods to accurately observe or infer the switch variable V in the context of nonlinear decoders, potentially involving more sophisticated data augmentation or inference strategies. This would enhance the flexibility and applicability of [ICRL-SM](#) across different types of model architectures.

Chapter 6

Conclusion

This thesis has proposed novel methods for causal representation learning, with a focus on increasing the robustness and interpretability of deep learning models in real-world applications. The primary goal was to address the limitations of existing methods by introducing frameworks that leverage domain knowledge and relax some of the strict assumptions typically required in causal modeling. Two novel methods were introduced: [GCRL](#) and [ICRL-SM](#).

GCRL: Generative Causal Representation Learning [3] [GCRL](#) leverages domain knowledge to construct a causal graph of the dataset, followed by learning the invariant and variant features responsible for generating the observed data. A Variational Autoencoder is employed to capture the distribution of these features, which are crucial for robust and interpretable modeling. Experimental results on both synthetic and real-world datasets, specifically within the domain of [HTP](#), demonstrate that [GCRL](#) significantly outperforms baseline methods in terms of robustness and efficacy, particularly in transfer learning tasks.

The analysis reveals that invariant features are highly robust and provide stability across different environments, making them essential for generalization. However, these invariant features cannot be identified solely based on trajectories; additional information is necessary. Conversely, the variant features, though less robust, offer valuable insights that can improve the prediction of future trajectories. The findings suggest that the component of the model responsible for capturing variant features requires fine-tuning when applied to new test domains, ensuring optimal performance across varying conditions.

ICRL-SM: Implicit Causal Representation Learning via Switchable Mechanisms [4] Building upon the foundation laid by [GCRL](#), [ICRL-SM](#) was developed to further

enhance the model’s ability to learn causal structures and variables without relying heavily on predefined causal graphs. This method is particularly beneficial in complex systems characterized by a large number of variables, where constructing a causal graph purely based on domain knowledge may be challenging or impractical.

A key focus of this work was on relaxing the strict assumption of the availability of hard interventions, which are often difficult to obtain in real-world scenarios. [ICRL-SM](#) introduces a method for learning from soft interventions, investigating the necessary assumptions and conditions that must be met to ensure the identifiability of causal models under these circumstances. The results, validated through experiments on both synthetic and real-world datasets, indicate that [ICRL-SM](#) can successfully identify causal models from soft interventions, thereby advancing the applicability of implicit causal models to real-world problems. Additionally, [ICRL-SM](#) consistently outperforms existing baseline methods, highlighting its potential for practical deployment in complex, data-rich environments.

In summary, this thesis has made contributions to the field of causal representation learning by developing methods that improve the robustness and interpretability of models in dynamic and uncertain environments. The proposed frameworks—[GCRL](#) and [ICRL-SM](#)—address critical challenges in current approaches, paving the way for more reliable and generalizable causal models. These advancements are expected to have broad implications, enabling the application of causal learning techniques across diverse domains, from high-throughput phenotyping to complex, high-dimensional systems where traditional methods may falter. Future research will continue to explore and refine these methods, expanding their utility and ensuring their alignment with the needs of real-world applications.

References

- [1] Kartik Ahuja, Divyat Mahajan, Yixin Wang, and Yoshua Bengio. Interventional causal representation learning. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett, editors, *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, volume 202 of *Proceedings of Machine Learning Research*, pages 372–407. PMLR, 2023.
- [2] Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. Invariant risk minimization, 2020.
- [3] Shayan Shirahmad Gale Bagi, Zahra Gharaee, Oliver Schulte, and Mark Crowley. Generative causal representation learning for out-of-distribution motion forecasting. In *International Conference on Machine Learning, ICML*, volume 202 of *Proceedings of Machine Learning Research*, pages 31596–31612. PMLR, 2023.
- [4] Shayan Shirahmad Gale Bagi, Zahra Gharaee, Oliver Schulte, and Mark Crowley. Implicit causal representation learning via switchable mechanisms, 2024.
- [5] Elias Bareinboim and Judea Pearl. Causal inference and the data-fusion problem. *Proceedings of the National Academy of Sciences*, 113(27):7345–7352, 2016.
- [6] Johann Brehmer, Pim de Haan, Phillip Lippe, and Taco S. Cohen. Weakly supervised causal representation learning. In *NeurIPS*, 2022.
- [7] Bertrand Charpentier, Simon Kibler, and Stephan Günnemann. Differentiable DAG sampling. In *The Tenth International Conference on Learning Representations, ICLR*. OpenReview.net, 2022.
- [8] Changan Chen, Yuejiang Liu, Sven Kreiss, and Alexandre Alahi. Crowd-robot interaction: Crowd-aware robot navigation with attention-based deep reinforcement learning.

In *International Conference on Robotics and Automation, ICRA 2019, Montreal, QC, Canada, May 20-24, 2019*, pages 6015–6022. IEEE, 2019.

- [9] Changan Chen, Yuejiang Liu, Sven Kreiss, and Alexandre Alahi. Crowd-robot interaction: Crowd-aware robot navigation with attention-based deep reinforcement learning. In *International Conference on Robotics and Automation, ICRA 2019, Montreal, QC, Canada, May 20-24, 2019*, pages 6015–6022. IEEE, 2019.
- [10] Guangyi Chen, Junlong Li, Jiwen Lu, and Jie Zhou. Human trajectory prediction via counterfactual analysis. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9804–9813, 2021.
- [11] Juan D. Correa and Elias Bareinboim. General transportability of soft interventions: Completeness results. In Hugo Larochelle, Marc’Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin, editors, *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems, NeurIPS*, 2020.
- [12] Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Antonino Furnari, Evangelos Kazakos, Jian Ma, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, and Michael Wray. Rescaling egocentric vision: Collection, pipeline and challenges for EPIC-KITCHENS-100. *Int. J. Comput. Vis.*, 130(1):33–55, 2022.
- [13] Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Antonino Furnari, Evangelos Kazakos, Jian Ma, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, and Michael Wray. Rescaling egocentric vision: Collection, pipeline and challenges for EPIC-KITCHENS-100. *Int. J. Comput. Vis.*, 130(1):33–55, 2022.
- [14] Matt Deitke, Eli VanderBilt, Alvaro Herrasti, Luca Weihs, Kiana Ehsani, Jordi Salvador, Winson Han, Eric Kolve, Aniruddha Kembhavi, and Roozbeh Mottaghi. Proctor: Large-scale embodied ai using procedural generation. *Advances in Neural Information Processing Systems*, 35:5982–5994, 2022.
- [15] Matt Deitke, Eli VanderBilt, Alvaro Herrasti, Luca Weihs, Kiana Ehsani, Jordi Salvador, Winson Han, Eric Kolve, Aniruddha Kembhavi, and Roozbeh Mottaghi. Proctor: Large-scale embodied AI using procedural generation. In *NeurIPS*, 2022.
- [16] Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using real NVP. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017.

- [17] Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using real NVP. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017.
- [18] Cian Eastwood and Christopher K. I. Williams. A framework for the quantitative evaluation of disentangled representations. In *6th International Conference on Learning Representations, ICLR*, 2018.
- [19] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. <http://www.deeplearningbook.org>.
- [20] Olivier Goudet, Diviyani Kalainathan, Philippe Caillou, Isabelle Guyon, David Lopez-Paz, and Michèle Sebag. Causal generative neural networks, 2017.
- [21] Agrim Gupta, Justin Johnson, Li Fei-Fei, Silvio Savarese, and Alexandre Alahi. Social gan: Socially acceptable trajectories with generative adversarial networks. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2255–2264, 2018.
- [22] Irina Higgins, Loïc Matthey, Arka Pal, Christopher P. Burgess, Xavier Glorot, Matthew M. Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework. In *5th International Conference on Learning Representations, ICLR*, 2017.
- [23] Yingfan Huang, Huikun Bi, Zhaoxin Li, Tianlu Mao, and Zhaoqi Wang. Stgat: Modeling spatial-temporal interactions for human trajectory prediction. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 6271–6280, 2019.
- [24] Alexander Immer, Christoph Schultheiss, Julia E. Vogt, Bernhard Schölkopf, Peter Bühlmann, and Alexander Marx. On the identifiability and estimation of causal location-scale noise models. In *International Conference on Machine Learning, ICML*, volume 202 of *Proceedings of Machine Learning Research*, pages 14316–14332. PMLR, 2023.
- [25] Zhuxi Jiang, Yin Zheng, Huachun Tan, Bangsheng Tang, and Hanning Zhou. Variational deep embedding: An unsupervised and generative approach to clustering. In Carles Sierra, editor, *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI 2017, Melbourne, Australia, August 19-25, 2017*, pages 1965–1972. ijcai.org, 2017.

- [26] Matthew J. Johnson, David Duvenaud, Alexander B. Wiltschko, Ryan P. Adams, and Sandeep R. Datta. Composing graphical models with neural networks for structured representations and fast inference. In Daniel D. Lee, Masashi Sugiyama, Ulrike von Luxburg, Isabelle Guyon, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, pages 2946–2954, 2016.
- [27] Matthew J. Johnson, David Duvenaud, Alexander B. Wiltschko, Ryan P. Adams, and Sandeep R. Datta. Composing graphical models with neural networks for structured representations and fast inference. In *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, pages 2946–2954, 2016.
- [28] Jean Kaddour, Aengus Lynch, Qi Liu, Matt J. Kusner, and Ricardo Silva. Causal machine learning: A survey and open problems, 2022.
- [29] Nan Rosemary Ke, Olexa Bilaniuk, Anirudh Goyal, Stefan Bauer, Hugo Larochelle, Bernhard Schölkopf, Michael C. Mozer, Chris Pal, and Yoshua Bengio. Learning neural causal models from unknown interventions, 2020.
- [30] Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, 2014.
- [31] Bohdan Kivva, Goutham Rajendran, Pradeep Ravikumar, and Bryon Aragam. Identifiability of deep generative models without auxiliary information. In *NeurIPS*, 2022.
- [32] Vineet Kosaraju, Amir Sadeghian, Roberto Martín-Martín, Ian Reid, Hamid Reza Tofighi, and Silvio Savarese. Social-bigat: Multimodal trajectory forecasting using bicycle-gan and graph attention networks. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.
- [33] Parth Kothari, Sven Kreiss, and Alexandre Alahi. Human trajectory forecasting in crowds: A deep learning perspective. *IEEE Transactions on Intelligent Transportation Systems*, 23(7):7386–7400, 2022.
- [34] Sébastien Lachapelle, Pau Rodríguez, Yash Sharma, Katie Everett, Rémi Le Priol, Alexandre Lacoste, and Simon Lacoste-Julien. Disentanglement via mechanism sparsity regularization: A new principle for nonlinear ICA. In Bernhard Schölkopf, Caroline Uhler, and Kun Zhang, editors, *1st Conference on Causal Learning and Reasoning*,

CLear 2022, Sequoia Conference Center, Eureka, CA, USA, 11-13 April, 2022, volume 177 of *Proceedings of Machine Learning Research*, pages 428–484. PMLR, 2022.

- [35] Namhoon Lee, Wongun Choi, Paul Vernaza, Christopher B. Choy, Philip H. S. Torr, and Manmohan Chandraker. Desire: Distant future prediction in dynamic scenes with interacting agents. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2165–2174, 2017.
- [36] Alon Lerner, Yiorgos Chrysanthou, and Dani Lischinski. Crowds by example. *Computer Graphics Forum*, 26(3):655–664, 2007.
- [37] Wendong Liang, Armin Kekic, Julius von Kügelgen, Simon Buchholz, Michel Besserve, Luigi Gresele, and Bernhard Schölkopf. Causal component analysis. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine, editors, *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023.
- [38] Phillip Lippe, Taco Cohen, and Efstratios Gavves. Efficient neural causal discovery without acyclicity constraints. In *The Tenth International Conference on Learning Representations, ICLR*. OpenReview.net, 2022.
- [39] Phillip Lippe, Sara Magliacane, Sindy Löwe, Yuki M. Asano, Taco Cohen, and Stratis Gavves. CITRIS: causal identifiability from temporal intervened sequences. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvári, Gang Niu, and Sivan Sabato, editors, *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, volume 162 of *Proceedings of Machine Learning Research*, pages 13557–13603. PMLR, 2022.
- [40] Chang Liu, Xinwei Sun, Jindong Wang, Haoyue Tang, Tao Li, Tao Qin, Wei Chen, and Tie-Yan Liu. Learning causal semantic representation for out-of-distribution prediction. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 6155–6170. Curran Associates, Inc., 2021.
- [41] Yuejiang Liu, Alexandre Alahi, Chris Russell, Max Horn, Dominik Zietlow, Bernhard Schölkopf, and Francesco Locatello. Causal triplet: An open challenge for intervention-centric causal representation learning. In Mihaela van der Schaar, Cheng Zhang, and Dominik Janzing, editors, *Conference on Causal Learning and Reasoning, CLear*

2023, 11-14 April 2023, Amazon Development Center, Tübingen, Germany, April 11-14, 2023, volume 213 of *Proceedings of Machine Learning Research*, pages 553–573. PMLR, 2023.

- [42] Yuejiang Liu, Riccardo Cadei, Jonas Schweizer, Sherwin Bahmani, and Alexandre Alahi. Towards robust and adaptive motion forecasting: A causal representation perspective. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, pages 17060–17071. IEEE, 2022.
- [43] Francesco Locatello, Stefan Bauer, Mario Lucic, Gunnar Raetsch, Sylvain Gelly, Bernhard Schölkopf, and Olivier Bachem. Challenging common assumptions in the unsupervised learning of disentangled representations. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 4114–4124. PMLR, 09–15 Jun 2019.
- [44] Francesco Locatello, Ben Poole, Gunnar Rätsch, Bernhard Schölkopf, Olivier Bachem, and Michael Tschannen. Weakly-supervised disentanglement without compromises. In *Proceedings of the 37th International Conference on Machine Learning, ICML, volume 119 of Proceedings of Machine Learning Research*, pages 6348–6359. PMLR, 2020.
- [45] C. Lu, Y. Wu, J. M. Hernández-Lobato, and B. Schölkopf. Invariant causal representation learning for out-of-distribution generalization. In *10th International Conference on Learning Representations (ICLR)*, April 2022.
- [46] Sara Magliacane, Thijs van Ommen, Tom Claassen, Stephan Bongers, Philip Versteeg, and Joris M Mooij. Domain adaptation by using causal inference to predict invariant conditional distributions. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018.
- [47] Karttikeya Mangalam, Harshayu Girase, Shreyas Agarwal, Kuan-Hui Lee, Ehsan Adeli, Jitendra Malik, and Adrien Gaidon. It is not the journey but the destination: Endpoint conditioned trajectory prediction. In *Computer Vision – ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part II*, page 759–776, Berlin, Heidelberg, 2020. Springer-Verlag.
- [48] Abdullah Mohamed, Kun Qian, Mohamed Elhoseiny, and Christian Claudel. Social-stgcnn: A social spatio-temporal graph convolutional neural network for human tra-

- jectory prediction. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14412–14420, 2020.
- [49] Judea Pearl. *Causality: Models, reasoning, and inference*. Cambridge University Press, 2nd edition, 2009.
 - [50] Judea Pearl and Elias Bareinboim. External Validity: From Do-Calculus to Transportability Across Populations. *Statistical Science*, 29(4):579 – 595, 2014.
 - [51] Judea Pearl, Madelyn Glymour, and Nicholas P. Jewell. *Causal inference in statistics: A primer*. John Wiley and Sons, 2016.
 - [52] Stefano Pellegrini, Andreas Ess, and Luc Van Gool. Improving data association by joint modeling of pedestrian trajectories and groupings. In Kostas Daniilidis, Petros Maragos, and Nikos Paragios, editors, *Computer Vision – ECCV 2010*, pages 452–465, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg.
 - [53] Ronan Perry, Julius von Kügelgen, and Bernhard Schölkopf. Causal discovery in heterogeneous environments under the sparse mechanism shift hypothesis. In *NeurIPS*, 2022.
 - [54] Elan Rosenfeld, Pradeep Kumar Ravikumar, and Andrej Risteski. The risks of invariant risk minimization. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021.
 - [55] Tim Salzmann, Boris Ivanovic, Punarjay Chakravarty, and Marco Pavone. Trajec-tron++: Dynamically-feasible trajectory forecasting with heterogeneous data. In *Computer Vision – ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XVIII*, page 683–700, Berlin, Heidelberg, 2020. Springer-Verlag.
 - [56] Bernhard Schölkopf. Causality for machine learning. In *Probabilistic and Causal Inference*, pages 765–804. ACM, feb 2022.
 - [57] Bernhard Schölkopf, Francesco Locatello, Stefan Bauer, Nan Rosemary Ke, Nal Kalchbrenner, Anirudh Goyal, and Yoshua Bengio. Toward causal representation learning. *Proceedings of the IEEE*, 109(5):612–634, 2021.
 - [58] Xinwei Shen, Furui Liu, Hanze Dong, Qing Lian, Zhitang Chen, and Tong Zhang. Weakly supervised disentangled generative causal representation learning. *J. Mach. Learn. Res.*, 23:241:1–241:55, 2022.

- [59] Chandler Squires, Anna Seigal, Salil S. Bhate, and Caroline Uhler. Linear causal disentanglement via interventions. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett, editors, *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, volume 202 of *Proceedings of Machine Learning Research*, pages 32540–32560. PMLR, 2023.
- [60] Raphael Suter, Djordje Miladinovic, Bernhard Schölkopf, and Stefan Bauer. Robustly disentangled causal mechanisms: Validating deep representations for interventional robustness. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 6056–6065. PMLR, 09–15 Jun 2019.
- [61] Takeshi Teshima, Issei Sato, and Masashi Sugiyama. Few-shot domain adaptation by causal mechanism transfer. In Hal Daumé III and Aarti Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 9458–9469. PMLR, 13–18 Jul 2020.
- [62] Burak Varici, Emre Acarturk, Karthikeyan Shanmugam, Abhishek Kumar, and Ali Tajer. Score-based causal representation learning with interventions, 2023.
- [63] Haowen Xu, Wenxiao Chen, Jinlin Lai, Zhihan Li, Youjian Zhao, and Dan Pei. On the necessity and effectiveness of learning the prior of variational auto-encoder, 2019.
- [64] Mengyue Yang, Furui Liu, Zhitang Chen, Xinwei Shen, Jianye Hao, and Jun Wang. Causalvae: Disentangled representation learning via neural structural causal models. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9588–9597, 2021.
- [65] Shuai Yang, Kui Yu, Fuyuan Cao, Lin Liu, Hao Wang, and Jiuyong Li. Learning causal representations for robust domain adaptation. *IEEE Transactions on Knowledge and Data Engineering*, pages 1–1, 2021.
- [66] Kui Yu, Xianjie Guo, Lin Liu, Jiuyong Li, Hao Wang, Zhaolong Ling, and Xindong Wu. Causality-based feature selection: Methods and evaluations. *ACM Comput. Surv.*, 53(5), sep 2020.
- [67] Kui Yu, Xianjie Guo, Lin Liu, Jiuyong Li, Hao Wang, Zhaolong Ling, and Xindong Wu. Causality-based feature selection: Methods and evaluations. *ACM Comput. Surv.*, 53(5), sep 2020.

- [68] Yue Yu, Jie Chen, Tian Gao, and Mo Yu. DAG-GNN: DAG structure learning with graph neural networks. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 7154–7163. PMLR, 09–15 Jun 2019.
- [69] Dong Zhang, Hanwang Zhang, Jinhui Tang, Xian-Sheng Hua, and Qianru Sun. Causal intervention for weakly-supervised semantic segmentation. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 655–666. Curran Associates, Inc., 2020.
- [70] Jiaqi Zhang, Kristjan H. Greenewald, Chandler Squires, Akash Srivastava, Karthikeyan Shanmugam, and Caroline Uhler. Identifiability guarantees for causal disentanglement from soft interventions. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023.
- [71] Kun Zhang, Mingming Gong, Petar Stojanov, Biwei Huang, Qingsong Liu, and Clark Glymour. Domain adaptation as a problem of inference on graphical models. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS’20, Red Hook, NY, USA, 2020. Curran Associates Inc.
- [72] Kun Zhang, Biwei Huang, Jiji Zhang, Clark Glymour, and Bernhard Schölkopf. Causal discovery from nonstationary/heterogeneous data: Skeleton estimation and orientation determination. In Carles Sierra, editor, *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI 2017, Melbourne, Australia, August 19-25, 2017*, pages 1347–1353. ijcai.org, 2017.
- [73] Xun Zheng, Bryon Aragam, Pradeep Ravikumar, and Eric P. Xing. Dags with NO TEARS: continuous optimization for structure learning. In Samy Bengio, Hanna M. Wallach, Hugo Larochelle, Kristen Grauman, Nicolò Cesa-Bianchi, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pages 9492–9503, 2018.
- [74] Yujia Zheng, Ignavier Ng, and Kun Zhang. On the identifiability of nonlinear ICA: sparsity and beyond. In *NeurIPS*, 2022.
- [75] Bolei Zhou, Aditya Khosla, Àgata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *2016 IEEE Conference on*

Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016, pages 2921–2929. IEEE Computer Society, 2016.

APPENDICES

Appendix A

Convergence Plots

The convergence of the [ADE](#) and [FDE](#) metrics is illustrated in [Figures A.1, A.2](#) for both the training and validation datasets in the experiments discussed in [Section 3.2](#). Additionally, [Figures A.3, A.4](#) present the convergence of different loss function terms in [GCRL](#) for the training set.

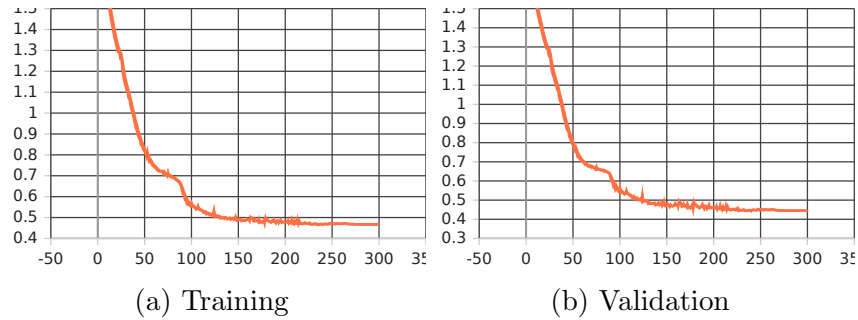


Figure A.1: ADE

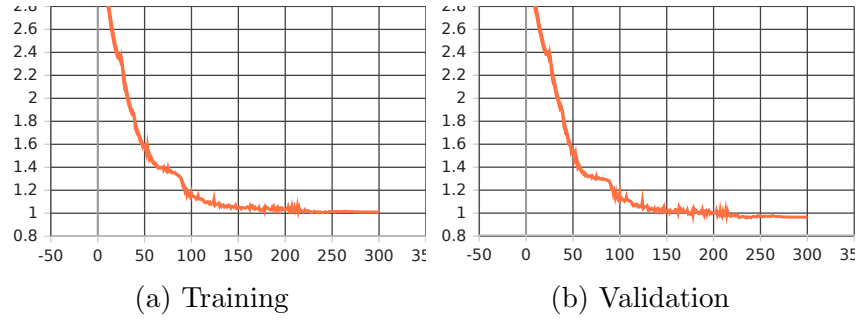


Figure A.2: FDE

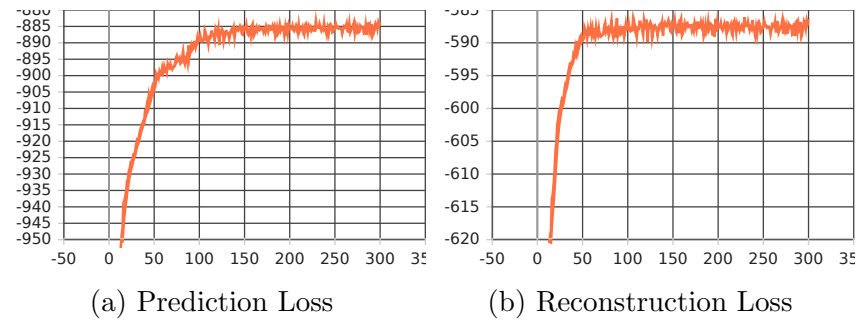


Figure A.3: Predictions

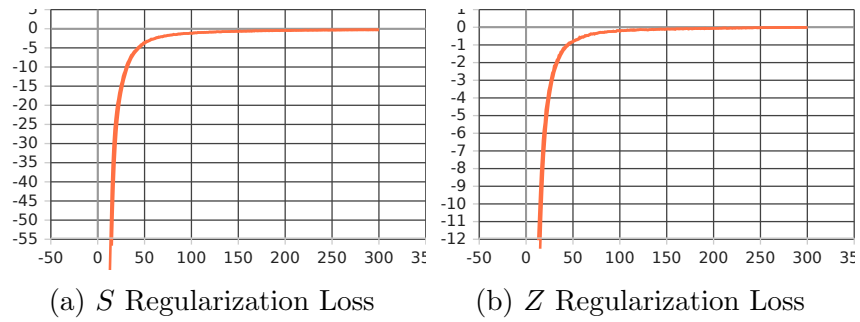


Figure A.4: Regularizations

Glossary

confounding factor A common cause of two causal variables which creates a misleading association between them by distorting the true causal relationship. [xi](#), [2](#), [4](#), [8](#), [17](#), [27](#), [31](#)

counterfactual A hypothetical scenario that represents what would have happened if a causal variable had taken a different value in the past. [2](#), [10–12](#)

d-separation A criterion used in probabilistic graphical models, specifically Bayesian networks, to determine whether two sets of variables are conditionally independent given a third set. Two sets of variables A and B are d-separated by a set C if all paths between A and B are blocked by C , indicating that knowing the state of C provides no additional information about the relationship between A and B . [29](#)

endogenous variable A random variable in a causal graph which has parents. [4](#), [11](#)

exogenous variable A random variable in a causal graph which has no parents. [4](#), [11](#)

intervention An external action or factor that is introduced to alter the behavior of a causal variable in a system or process. [xi](#), [2](#), [8](#), [9](#), [11](#), [12](#), [17](#)