Neural Causal Models with TRAM-DAGs: A framework for modelling causal effects in a flexible and interpretable way and for making subsequent causal queries.

Master Thesis in Biostatistics (STA495)

by

Mike Krähenbühl
Matriculation number: 18-652-149

supervised by

Prof. Dr. Beate Sick
Prof. Dr. Oliver Dürr, HTWG Konstanz

Zurich, July 2025

Neural Causal Models with TRAM-DAGs:

Applied on real-world data and used for ITE estimation.

Mike Krähenbühl

Version June 23, 2025

Contents

Preface

In the introduction part, my aim is to give a summary of important concepts of causal inference and causal models. Further, I motivate the importance for methods that allow to draw causal conclusions from observational data in contrast to randomized controlled trials (RCTs) and that the proposed framework of tram-dags (cite sick duerr) can be used as such a tool.

In the methods section, I give a detailed description of the tram-dag framework and how it works by illustrating it on a simple simulated example. I also show for what kind of causal queries the model can be used for. Although it is not a topic for observations data, I discuss how the model can be used for estimating the individualized treatment effect (ITE).

In the results section, I will first show results from simulation studies where the ground truth is known. Second, I will show results on a real-world example in the setting of climate data. Third, I present the results of the ITE estimation.

In the final section, I will discuss the results and give a conclusion about the advantages and limitation of this framework and provide an outlook.

Mike Krähenbühl
July 2025

Chapter 1

Introduction

1.1 Motivation

The important questions that studies want to answer are usually not associational but causal (Pearl, 2009a). For example, these are questions that ask for effects when making a certain intervention, like the effect of a treatment. They can ask for reasons that lead to the observed outcome, like which disease caused the given symptoms. Or what would have been different if another action was taken, like what would the GDP have been, if the interest rates were incereased only by 25 instead of 75 Bps. To answer such questions, a causal reasoning must be applied that aims to understand the underlying data genearting mechanism, sole associational reasoning directly from the data is not sufficient.

The gold standard to measure causal relationships between an intervention and an outcome is the randomized controlled trial (RCT) (Hariton and Locascio, 2018). The key concept of this prospective study design, is that the participants are randomly allocated due either the treatment or control group. Due to this randomization, the influence of potential confounding variables is eliminated and study groups are balanced with respect to baseline characteristics allowing for an unbiased cause-effect estimation. Disadvantages of RCTs include but are not limited to often high cost, the time for planning and executing the trial, and generalisability to the population of interest. Furthermore, RCTs typically aim to estimate an average treatment effect on a sample, which is the difference in the averages accross the treatment groups (Nichols, 2007). However, patients have individual responses to the treatment, depending on their characteristics. In personalized medicine, such individual treatment effects are crucial. Another central limitation of RCTs is that in many scenarios they can not be conducted due to ethical or practical reasons. For example, an RCT is only ethical in the case of clinical equipoise, which means that there is uncertainty about the (superiority) of one of the two treatment arms (Freedman, 1987). It is not acceptable to treat one group with the assumed inferior treatment. The same is true for obviously harmful interventions, like smoking or drinking alcohol. In these cases, it is not possible to conduct an RCT to estimate the causal effect of smoking on lung cancer.

Therefore, much of the research aims to make causal inference from observational data in a non-experimental or quasi-experimental design. In an observational setting, there are usually confounding variables that make it challenging to measure the effect between exposure and outcome. Methods for causal infreence on observational data for example aims to correctly adjust or control for confounders. Sick and Dürr (2025) proposed the framework of TRAM-DAGs to estimate the causal relationships in an observational setting and make subsequent queries. The aim of this thesis is to further analyze this method and apply it in a real-world scenario.

Background on causality

Causal relationships can be represented by a directed acyclic graph (DAG) as, for example, shown in Figure 1.1(a). The variables, or nodes, are connected by directed edges which indicate the path of causal dependence.

Usually we want to answer questions that can be assigned to one of the groups in pearls hierarchy of causation Pearl (2009b). Visual examples are presented in Figure 1.1(a)-(c). Level 1 are observational queries which are conditional probabilities $P(Y \mid X)$ and can be answered directly from the joint distribution $P(Y \cap X)$. Level 2 are interventional queries which are probabilities $P(Y \mid do(X))$ that result by a taken action do(X). Where observational queries only require to know the joint distribution, for interventional queries an additional understanding of the causal mechanism is necessary. Level 3, the analysis of counterfactuals, poses the biggest challenge. These are what-if questions. An example of this would be if a sick patient was treated with a certain treatment and then died. Death would therefore be the observed and therefore factual outcome. The counterfactual outcome is the outcome that would have occurred if the patient had received a different medication. Such counterfactual questions are often labelled as metaphysical because they can never be tested directly. However, there are important practical questions that require the analysis of such counterfactuals.

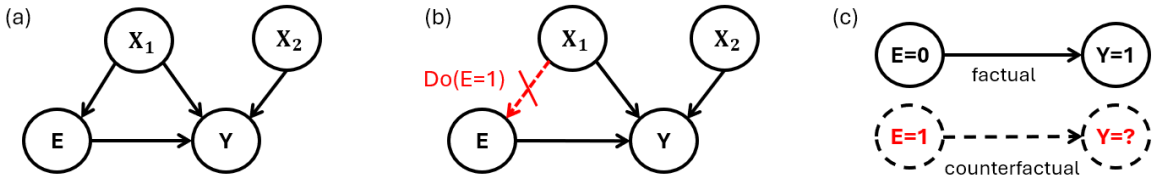

Figure 1.1: Example for the three levels of Pearl's hierarchy of causation. (a) DAG for observational data. (b) DAG when making a do-intervention by fixing the variable E at a certain value. c) Observed factual outcome and corresponding counterfactual query.

To illustrate Pearl's three levels of causality, I consider a simplified example involving the exposure Exercise (E), the outcome Heart Disease (Y), the confounder Age (X_1) and the additional covariate Smoking (X_2). I assume that exercise reduces the risk of heart disease, but both variables are also influenced by age. Figure 1.1(a)-(c) illustrates the corresponding scenarios.

Level 1: Observational ("seeing"). We observe the joint distribution of variables without intervention. Example: What is the probability of heart disease given that a person exercises?

$$P(Y \mid E = 1)$$

This can be estimated directly from data (by filtering for E=1 and calculating the fraction of Y), but does not account for confounding.

Level 2: Interventional ("doing"). We consider the effect of an intervention on the system. Example: What is the probability of heart disease if everyone were made to exercise, regardless of age or smoking?

$$P(Y \mid \text{do}(E = 1))$$

This requires assumptions about the causal structure.

Level 3: Counterfactual ("imagining"). We ask what would have happened under different circumstances, which means imagine an alternative reality. Example: For a person who does not exercise and has heart disease, would they still have heart disease if they had exercised?

$$P(Y_{E=1} \mid E = 0, Y = 1)$$

Here $Y_{E=1}$ is the outcome under the positive exposure. Counterfactual queries require a structural causal model and cannot be answered from data alone.

What is a Structural Causal Model? To answer questions from Pearl's ladder of causation, the concept of DAG's can be extended to structural causal model (SCM). A set of

structural equations of the from $x_i = f_i(pa(x_i), z_i)$, $i = 1, ..., n$ forms a structural causal model (Pearl, 2009b). $pa(x_i)$ are the direct causal parents of X_i and therefore directly determe its value. Z_i are errors that follow exogeneous noise distributions $P(Z_i)$. They can be understood as latent variables that represent unmodeled factors which can not be observed or measured directly. By convention, the Z_i are assumed to be mutually independent. The potentially non-linear function f_i determines the functional form of the parents and the noise that represents the data generating mechanism of the dependent variable X_i. Hence, in a SCM a source node X_j can be represented as $X_j = f_j(z_j)$ since it does not depend on any other variables in the system. Once all components of the structural equations are known (or assumed to be known), it is fully deterministic.

This representation makes it practical to determine interventional distributios and counterfactual queries. This will be discussed in detail in Section XXX.

In this thesis, the focus is not on finding the causal structure of a system. Such a structure can be found by structure finding algorithms, or determided by expert knowledge, for example. Instead, I assume the DAG to be known and focus on the estimation of the functional form of the relationships.

There is a variety of methods that are applied to quantify these structural equations that form the resulting SCM.

The simplest method might be the linear regression which assumes gaussian error terms Z_i and a linear f_i. Similarly other classical statistical methods can be applied they have the advantage of being typically well defined and having interpretable parameters. However, they require to make strong assumptions about the underlying data generating mechanism which can make them susceptible to bias if these assumptions are violated.

Then there is also the possibility to model the relationships with various methods based on neural networks. Because they can be very flexible, they can also model complex underlying distributions with practically no bias. But this comes at the cost of less interpretability. And often they are limited to continuous data.

The framework of TRAM-DAGs proposed by Sick and Dürr (2025) builds a bridge between these two approaches of classical statistical and deep learning methods. It allows us to model the causal relationships with interpretable or fully-flexible parts, depending on what is regarded more important at the moment. The basis of this model is to construct the structural equations as transformation models as introduced by Hothorn *et al.* (2014), which is a flexible distributional regression method. To make them even more customizable, these transformation models (TRAMS) were extended to deep TRAMS (Sick *et al.*, 2021). Applied in a causal context, these deep TRAMS form the framework of TRAM-DAGs that can be fitted on observational data and allow to tackle causal queries on all three levels of Pearl's causal hierarchy. This framework will be explained in detail in the Section XXX.

1.2 Background on Individualized Treatment Effects (ITE)

An application where causal inference is of particular importance is the estimation of individualized treatment effects (ITE) or also referred to conditional average treatment effect (cATE) or uplift modelling often in a marketing context. They can be understood in the difference in outcomes under different treatments, on an individual or subgroup level. The idea is that they can be used for example in personnalized medicine or for targeted direct marketing campaigns. Each individual patient might respond differently to a specific treatment, depending on his unique characteristics. RCTs traditionally are used to estimate an average treatment effect (ATE) which might indicate the trend in the analyzed population, but the ITEs might be very different for individual patients. In the marketing context this individual effect is of high interest because marketing campaings can be executed very specific and individualized. For example, in the assessment wheter a certain customers should receive a push notification (treatment) or not.

Here each customer could be attributed to one of 4 categories, the persuadables (if receiving a notification, they will buy a product or service), the sure thing (they will buy the product either way), the lost causes (they will not buy, regardless of receiving a notification or not), or the sleeping dogs (they will eventually buy, but not if they were treated). Therefore it is crucial to know for each (potential) customer, how he might respond to the treatment, the persuadables should definetely be treated so that we gain them as a customer and the sleeping dogs shall not receive one as we would loos them as customers. There exists many approaches and methods to estimate these individualized treatment effects. However, this is more difficult compared to sole predictive modelling. Chen *et al.* (2025) showed that all causal machine learning models that were trained on a train set failed to generalize to a test set.

1.3 Goals of this thesis

The first goal of this thesis is to further analyze and extend TRAM-DAGs. This includes the application of TRAM-DAGs in different scenarios, such as different datatypes, level of complexity or Neural Network structures (activation function, batchnormalization, dropout). Most analyses are performed in simulation studies, but the model was also applied on real-world climate data.

Analyzing what could be the reason for this behaviour and potential solutions constitute the second goal of this thesis. With the help of simulaition studies we evalueate scenarios when the estimation of ITE fails. Furthermore we show that TRAM-DAGs can be used to estimate the ITE, also in a non-RCT setting, as long as the data can be described by a fully observed DAG.

Chapter 2

Methods

In this section I will explain the necessary background needed to understand the TRAM-DAGs. Once the framework of tram dags is explained, I will present how the experiments of the simulation, the application on real data and the ITE estimation are conducted. Following research questions are addressed:

- How can TRAM-DAGs be applied under different scenarios (datatypes, DAG structure, complexity, scaled varaibles) and how does this influence the interpretation of parameters?
- Why does the estimation of Individualized Treatment Effects (ITE) fail in some cases for most causal ML methods when validating them out of sample?
- How can TRAM-DAGs be used to estimate the Individualized Treatment Effect (ITE) in a RCT and in a observational setting with confounding and mediating variables?

2.1 TRAM-DAGs

The goal of TRAM-DAGs is to estimate the structural equations according to the causal order in a given DAG in a flexible and possibly still interpretable way in order to sample observational and interventional distributions and to make counterfactual statements. The estimation requires data and a DAG that describes the causal structure. It must be assumed that there are no hidden confounders. TRAM-DAGs estimate for each variable X_i a transformation function $Z_i = h_i(X_i \mid pa(X_i))$, where Z_i is the noise value and $pa(X_i)$ are the causal parents of X_i. The important part here is that we can rearrange this equation to $X_i = h_i^{-1}(Z_i \mid pa(x_i))$ to get to the structural equation. The transformation functions h are monotonically increasing functions that are a representation of the conditional distribution of X_i on a latent scale. They are based on the idea of transformation models as introduced by Hothorn *et al.* (2014) but were extended to deep trams by Sick *et al.* (2021). In the following sections I review the most important ideas of these methods as they are the essential components of TRAM-DAGs.

2.2 Transformation Models

Transformation models are a flexible distributional regression method for various data types. They can be for example specified as ordinary linear regression, logistic regression or proportional odds logistic regression. But Transformation models further allow to model conditional outcome distributions that do not even need to belong to a known distribution family of distributions by model it in parts flexibly. This reduces the strength of the assumptions that have to be made.

The basic form of transformation models can be described by

$$F(y|\mathbf{x}) = F_Z(h(y \mid \mathbf{x})) = F_Z(h_I(y) - \mathbf{x}^\top \boldsymbol{\beta}) \tag{2.1}$$

, where $F(y|\mathbf{x})$ is the conditional cumulative distribution function of the outcome variable Y given the predictors $\mathbf{x}$. $h(y \mid \mathbf{x})$ is a transformation function that maps the outcome variable y onto the latent scale of Z. F_Z is the cumulative distribution function of a latent variable Z, the so-called inverse-link function that maps $h(y \mid \mathbf{x})$ to probabilities. In this basic version, the transformation function can be split into an intercept part $h_I(y)$ and a linear shift part $\mathbf{x}^\top \boldsymbol{\beta}$, where the vector $\mathbf{x}$ are the predictors and $\boldsymbol{\beta}$ are the corresponding coefficients.

If the latent distribution Z is chosen to be the standard logistic distribution, then the coefficient β_i can be interpreted as log-odds ratios when increasing the predictor x_i by one unit, holding all other predictors unchanged. This means that an increase of one unit in the predictor x_i leads to an increase of the log-odds of the outcome Y by $\boldsymbol{\beta}$. The additive shift of the transformation function means a linear shift on the latent scale (herer log-odds). The following transformation to probabilities by F_Z potentially leads to a non-linear change in the conditional outcome distribution on the original scale. This means not only is the distribution shifted, also its shape can change to some degree based on the covariates. More details about the choice of the latent distribution and the interpretation of the coefficients are provided in the appendix XXX.

For a continuous outcome Y the intercept h_I is represented by a bernstein polynomial, which is a flexible and monotonically increasing function

$$h_I(y) = \frac{1}{M+1} \sum_{k=0}^{M} \vartheta_k \, \mathrm{B}_{k,M}(y) \tag{2.2}$$

, where ϑ_k are the coefficients of the bernstein polynomial and $\mathrm{B}_{k,M}(y)$ are the Bernstein basis polynomials. More details about the technical implementation of the bernstein polynomial in the context of TRAM-DAGs is given in the appendix XXX.

For a discrete outcome Y the intercept h_I is represented by cut-points, which are the thresholds that separate the different levels of the outcome. For example, for a binary outcome Y there is one cut-point and for an ordinal outcome with K levels there are $K-1$ cut-points. The transformation model is given by

$$P(Y \leq y_k \mid \mathbf{X} = \mathbf{x}) = F_Z(\vartheta_k + \mathbf{x}^\top \boldsymbol{\beta}), \quad k = 1, 2, \ldots, K-1 \tag{2.3}$$

A visual representation for a continuous and discrete (ordinal) outcome is provided in Figure 2.1.

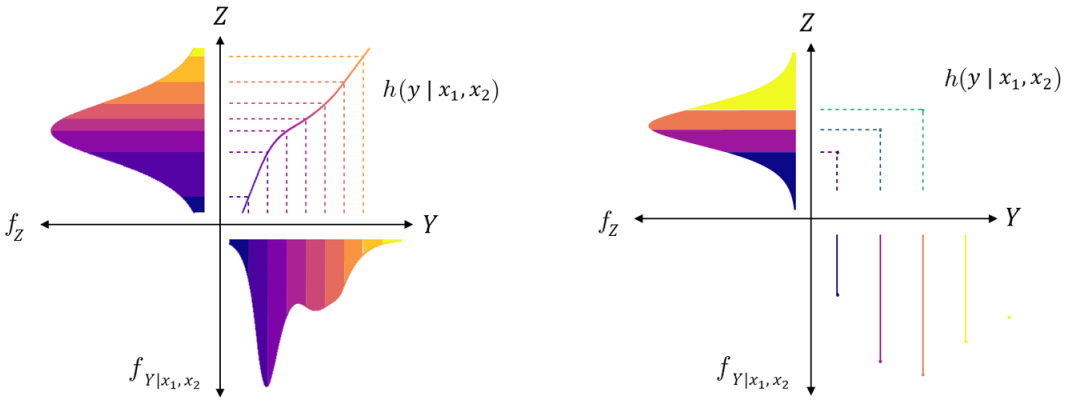

Figure 2.1: Left: Example of a transformation model for a continuous outcome Y with a smooth transformation function. **Right:** Example of a transformation model for an ordinal outcome Y with 5 levels. The transformation function consists of cut-points that separate the probabilities for the levels of the outcome. In both cases the latent distribution Z is the standard logistic and the predictors $\mathbf{x}$ induce a linear (vertical) shift of the transformation function.

To estimate the parameters $\boldsymbol{\beta}$ and $\boldsymbol{\vartheta}$ the negative log likelihood (NLL) is minimized. The NLL is defined as

$$\text{NLL} = -\frac{1}{n}\sum_{i=1}^{n} l_i(\boldsymbol{\beta}, \boldsymbol{\vartheta}) = -\frac{1}{n}\sum_{i=1}^{n} \log(f_{Y|\mathbf{X}=\mathbf{x}}(y_i)) \tag{2.4}$$

where $l_i(\boldsymbol{\beta}, \boldsymbol{\vartheta})$ is the log-likelihood of the i-th observation, $l_i(\boldsymbol{\beta}, \boldsymbol{\vartheta}) = f_{Y|\mathbf{X}=\mathbf{x}}(y_i)$ is the conditional density function of the outcome variable Y given the predictors $\mathbf{x}$ under the current parameterization. I provide the full derivation in the appendix xxx.

For the remainder of this thesis, I rely on the idea of these transformation models to model the conditional distribution functions represented by the transformation functions of the respective variables. The standard logistic distribution is used as F_Z, which results in a logistic transformation model.

2.3 Deep TRAMs

The transformation models as discussed before were extended to deep TRAMs using a modular neural network (Sick *et al.*, 2021). The goal is to get a parametrized transformation function of the form **??**.Each part, the intercept $h_I(X_i)$, the linear shift $\mathbf{x}_L^\top \boldsymbol{\beta}_L$ and the complex shift $f_C(\mathbf{x}_C)$ are assembled by the outputs of the individual neural networks. The user can specify the level of complexity the parents $pa(X_i)$ have on the transformaiton funciton. Figure 2.2 illustrates the case for a SI-LS-CS model.

$$h(y \mid \mathbf{x}_L, \mathbf{x}_C) = h_I(y) + \mathbf{x}_L^\top \boldsymbol{\beta}_L + f_C(\mathbf{x}_C) \tag{2.5}$$

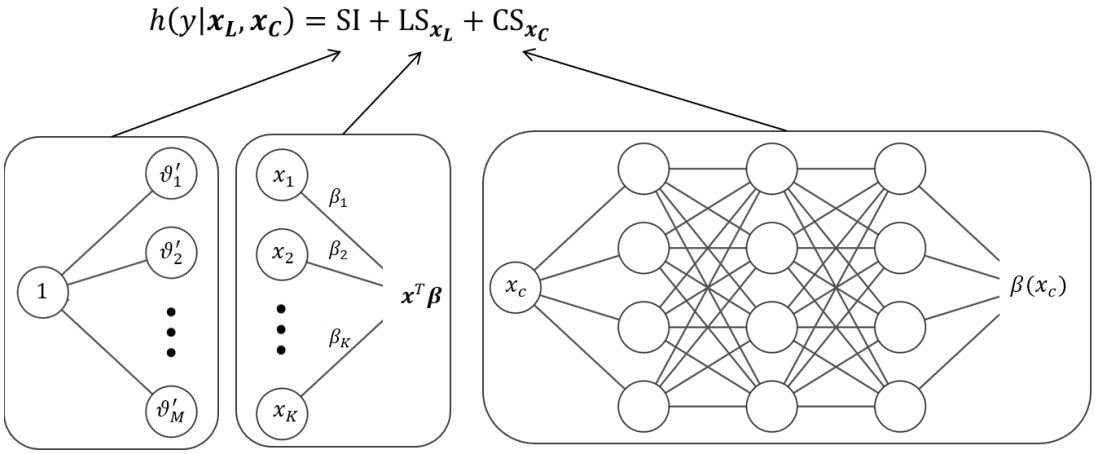

Figure 2.2: Modular deep transformation model. The transformation function $h(y \mid \mathbf{x})$ is constructed by the outputs of three neural networks.

Intercept the shape of the transformation function at the baseline configuration $\mathbf{x}_L^\top \boldsymbol{\beta}_L = 0$ and $f_C(\mathbf{x}_C) = 0$ is determined by the intercept $h_I(y)$. For a continuous outcome the intercept is represented by a smooth bernstein polynomial and in the discrete case by cut-points. In either case the parameters ϑ are obtained as output nodes of the neural network. A simple intercept (SI) is the case where the parameters ϑ do not depend on the any explanatory variables. The neural network thereby only takes a constant as input and directly outputs the parameters ϑ. To make the intercept more flexible, the intercept can also depend on the explanatory variables. In this case the complex intercept (CI) models the intercept $\vartheta(x)$ by taking the predictors x as input to a neural network with some hidden layers. This allows the intercept to change with the value of the predictors. Depending on the assumptions, predictors can be used in the complex intercept, or only a subset of them. A detailed explanation of the construction of the bernstein polynomial is given in appendix XXX.

Linear shift If the predictors should have a linear effect on the transformation function, it can be modelled by a linear shift (LS). For this part the neural network without hidden layers and without biases takes the linear predictors $pa(X_i)$ as input and generates a single output node with a linear activation function. This results in the linear combination $\mathbf{x}_L^\top \boldsymbol{\beta}_L$ and it induces a linear vertical shift of the transformation function. The weights $\boldsymbol{\beta}_L$ are the interpretable coefficients of the linear shift. For the logistic transformation model, they are interpreted as log-odds-ratios. The interpretation is further described in the appendix 6.2.

Complex shift If the transformation function should be allowed to be shifted vertically in a non-linear manner, a complex shift (CS) can be applied. The predictor variables are inputted in a (deep) neural network with at least one hidden layer and non-linear activation functions such as sigmoid or ReLU. A single output node with $f_C(X_C)$ is obtained. With a complex shift, also interactions between predictor variables can be captured by giving the interacting variables into the same neural network.

Level of complexity One practical feature of these modular deep TRAMs is that one can specify, which predictors should have a linear or complex shift effect on the transformation function or that predictors are even allowed to deterimine the shape of the transformation function by a complex intercept. Herzog *et al.* (2023) predicted the ordinal functional outcome three months after stroke by using semi-structured data that included tabular predictors and images. The two data modalities can be included in a single deep TRAM by modeling the part of the images with a CNN.

The estimated distribution function is invariant with respect to the choice of the inverse-link function F_Z (scale of latent distribution) in an unconditional (Hothorn *et al.*, 2018) or fully flexible (CI) setting. However, as soon as restrictions are placed on the influence of the predictors

(LS, CS), this leads to assumptions about the scale of the dependency. Which latent distribution should be chosen depends on following factors: (i) the intended complexity of the model, (ii) the assumptions about the data generating process, (iii) the conventional, widely used, scale of interpretation for the specific problem. If the coefficients β in the linear shift term should be interpreted as log odds ratios, then the standard logistic distribution is appropriate. For log hazard ratios it would be the minimum extreme value distribution. There exist plenty of other alternatives.

(The optimal scale could be found by comparing the likelihoods of the model under different latent distributions.)

Parameter estimation The parameters of the neural networks are learned by minimizing the negative log-likelihood (NLL) of the conditional deep TRAM. The learning process is started with a random parameter configuration and the outputs of the neural networks are used to assemble the NLL of the transformation model. The NLL is then iteratively minimized by adjusting the parameters by the Adam optimizer (Kingma and Ba, 2015) until they eventually converge to the optimum state. Additionally, methods to prevent overfitting — such as dropout, early stopping, or batch normalization — can be applied. These techniques are particularly important in more complex networks to ensure that the model generalizes well to out-of-sample data. In the hidden layers, non-linear activation functions such as ReLU or sigmoid are applied.

2.4 TRAM-DAGs

In TRAM-DAGs these deep transformation models are applied in a causal setting. We assume a pre-specified DAG which defines the causal dependence. Then we estimate the distribution of each node by a transformation model that is conditional on its parents. Figrue 2.3 illustrates the basic idea of a TRAM-DAG where a DAG with 3 variables, without hidden confounder, is assumed to be known. The arrows in the DAG indicate the causal dependencies between the variables. The transformation models are constructed by a modular neural network. The assumed influence from the parent variables has to be specified as SI, LS or CS. In this example, X_1 is a continuous source node that acts as parent of X_2 and X_3. For a source node the transformation function only consists of a simple intercept (SI). X_2 is also continuous and its transformation function can be shifted additively (LS) by the value of X_1. X_3 is an ordinal variable with 4 levels and its transformation function depends on the values of X_1 (LS) and X_2 (CS). The cut-points $h(x_3 \mid x_1, x_2)$ represent the cumulative probabilities on the log-odds scale of the first 3 levels of X_3, where the probability of the last level $K = 4$ is the complement of the previous levels k_{1-3}.

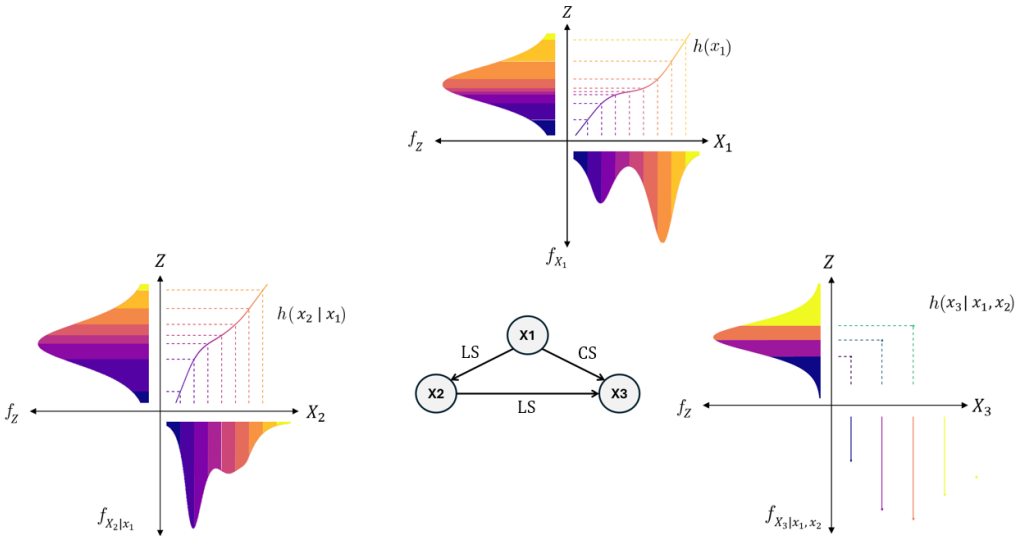

Figure 2.3: Example of a TRAM-DAG with three variables X_1, X_2 and X_3. The transformation functions are represented by the modular neural networks. The arrows indicate the causal dependencies between the variables.

This DAG with the assumed dependencies can be described by an adjacency matrix 2.6, where the rows indicate the source and the columns the target of the effect:

$$\mathbf{MA} = \begin{bmatrix} 0 & \text{LS} & \text{LS} \\ 0 & 0 & \text{CS} \\ 0 & 0 & 0 \end{bmatrix} \tag{2.6}$$

To apply the framework of TRAM-DAGs on this example, we assume to have observational data that follows the structure of the adjacency matrix 2.6. In practice, the DAG is either defined by expert knowledge or by some sort of structure finding algorithm (XXX cite methods). Then we want to estimate the conditional distribution function of each variable by a deep TRAM so that we can sample from the distributions and make causal queries. The conditional distribution functions are given by

$$\begin{aligned} X_1 &\sim F_Z(h_I(x_1)) \\ X_2 &\sim F_Z(h_I(x_2) + \text{LS}_{x_1}) \\ X_3 &\sim F_Z(h_I(x_3) + \text{LS}_{x_1} + \text{CS}_{x_2}) \end{aligned}$$

Construct Modular Neural network

As discussed in the section 2.3, the transformation functions are constructed by a modular neural network. The inputs are the variables in the system as well as the adjacency matrix 2.6 which controls the information flow and assures that only valid connections according to the causal dependence are made. Discrete variables with few categories are dummy encoded, and continuous variables should be scaled before feeding them in the neural network. The encoding and the effect of scaling on the interpretation of parameters is discussed in the appendix (6.3 and 6.4). Scaling the input variables, meaning to bring the variables onto a zero-mean and one-variance, can remove the pattern in marginal variance which some structure learning algorithms rely on (Reisach *et al.*, 2021). However, since our analysis does not require to find the structure and already assumes a known DAG, this is not a problem. Once the input variables are prepared and the structure is defined by the adjacency matrix, the architecture of the neural network for the complex shift and complex intercept has to be specified. These are factors such as depth, width, activation function, and whether dropout or batch normalization should be used. These considerations depend on the assumed complexity of the shifts. The outputs of

the neural networks are the three components for the transformation function (SI, LS, CS) for each variable. These components are assembled to the transformation functions. Finally, the loss is defined as the negative log likelihood, which the model aims to optimize to estimate the optimal parameterization. The estimated parameters ***beta*** in the linear shifts are interpretable as log-odds-ratios when changing the value of the respective parent by one unit, leaving all others unchanged.

2.4.1 Sampling from TRAM-DAGs

Observational sampling Once the TRAM-DAG is fitted on data, it can be used to sample from the observational or interventional distribution or to make counterfactual queries. The structural equations $X_i = f(Z_i, \text{pa}(X_i))$ are represented by the inverse of the conditional transformation functions $h^{-1}(Z_i \mid \text{pa}(X_i))$ because $Z_i = h(X_i \mid \text{pa}(X_i))$. The sampling process from the observational distribution for one iteration (one observation of all variables in the DAG) is described in the pseudocode 1 and illustrated in Figure 2.4. The process is repeated for the desired number of samples.

Algorithm 1 Generate a samples from the TRAM-DAG

Given: A fitted TRAM-DAG with structural equations $X_i = f(Z_i, \text{pa}(X_i))$, where $Z_i = h(X_i \mid \text{pa}(X_i))$
for each node X_i in topological order **do**
    Sample latent value $z_i \sim F_{Z_i}$ ▷ e.g., `rlogis()` in R
    if X_i is continuous **then**
        Compute $x_i = h^{-1}(z_i \mid \text{pa}(x_i))$ by solving $h(x_i \mid \text{pa}(x_i)) - z_i = 0$
    end if
    if X_i is discrete **then**
        Determine x_i such that $x_i = \min\{x : z_i \leq h(x \mid \text{pa}(x_i))\}$
    end if
end for

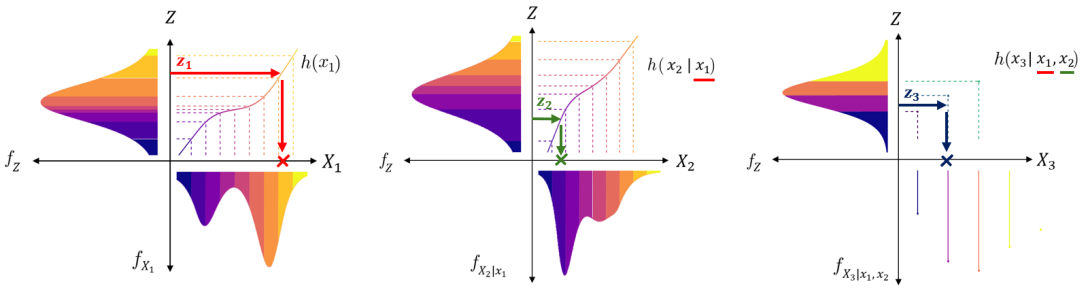

Figure 2.4: One sampling iteration for the three variables from the estimated transformation functions $h(x_i \mid \text{pa}(x_i))$. The latent values z_i are sampled from the standard logistic distribution. The values x_i are determined by applying the inverse of the transformation function for continuous variables or by finding the corresponding category for the ordinal variable.

Interventional sampling To sample from the interventional distribution, we can apply the do-operator as described by Pearl (1995) (Pearl named it set instead of do). The do-operator fixes a variable at a certain value and sample from the distribution of the other variables while keeping the fixed variable constant. For example, if one wants to intervene on X_2 and set it to a specific value α, $\text{do}(x_2 = \alpha)$ and then sample from the interventional-distribution

$$x_3 = \min\{x : z_3 \leq h(x \mid x_1, x_2 = \alpha)\}$$

with the same process as for the observational sampling, with the only difference that the intervened variable X_2 stays constant.

Counterfactual queries In a counterfactual query one wants to know what the value of variable X_i would have been if another variable X_j had a different value than what was acutally observed. Pearl (2009b) describes the three-step process to answer counterfacutal queries as follows: Given a causal model M and observed evidence e (which are the actually observed values of the variables X_i of one sample) one wants to compute the probability of $Y = y$ under the hypothetical condition $X = x$.

Step 1 aims to explain the past (Z) by knowledge of the evidence e; Step 2 amends the past to the hypothetical condition $X = x$ Step 3 predicts the future (Y) based on our new understanding of the past and our newly established condition, $X = x$

Pearl named these three steps, (1) abduction, (2) action and (3) prediction. The procedure is described in the pseudocode **??** and illustrated in Figure.

Algorithm 2 Answer a Single Counterfactual Query

1: **Given:** A structural model $X_k = f(Z_k, \text{pa}(X_k))$, with inverse noise map $Z_k = h(X_k \mid \text{pa}(X_k))$
2: **Input:** Observed sample x, intervention $X_i := \alpha$, target variable X_j
3: **Step 1: Abduction** Infer latent variable $Z_j = h(x_j \mid \text{pa}(x_j))$ using the observed values
4: **Step 2: Action** Replace the value of X_i with α in the set of parent variables
5: **Step 3: Prediction** Compute the counterfactual value $x_j^{cf} = h_j^{-1}(Z_j \mid \text{pa}(x_j)^{cf})$

While the probability of Y under the hypothetical condition $X = x$ can be determined in any case, the actual counterfactual value of Y is only defined for a continuous outcome but not for discrete outcomes.

(What pearl writes: Likewise, in contrast with the potential-outcome framework, counterfactuals in the structural account are not treated as undefined primitives but rather as quantities to be derived from the more fundamental concepts of causal mechanisms and their structure.)

2.5 Individualized Treatment Effect (ITE)

Curth *et al.* (2024) provide a comprehensive overview of the individualized treatment effect (ITE) and its estimation in the context of causal machine learning. They state its importance in comparison to average treatment effects, the assumpitons that need to be fulfilled, what kind of limiations typically are encountered and how models should be validated.

Randomized controlled trials (RCTs) are considered the gold standard for estimating causal effects due to their ability to eliminate confounding through randomization. However, RCTs typically report the *average treatment effect (ATE)*, which summarizes the effect of a treatment across an entire study population. This obscures individual-level variation in treatment response: some individuals may benefit substantially, others not at all, or even be harmed. In personalized medicine and risk-based decision-making, such population-level summaries are insufficient. Instead, the objective is to guide treatment decisions tailored to individual patient characteristics, for which the *individualized treatment effect (ITE)* is a more appropriate target. Where the homogeneous treatment effect refers to the part of the effect that is equal for all patients, the heterogeneous treatment effect describes the non-random variation in treatment effects across individuals or groups.

The Potential Outcomes Framework Causal inference is commonly formalized within the *Rubin Causal Model*, also known as the potential outcomes framework. For each individual i, let $Y_i(1)$ denote the potential outcome under treatment, and $Y_i(0)$ the outcome under control.

The individual treatment effect is defined as

$$\tau_i = Y_i(1) - Y_i(0). \tag{2.7}$$

However, only one of the two potential outcomes can be observed for each individual, which constitutes the *fundamental problem of causal inference*. Related estimands include the *conditional average treatment effect (CATE)*,

$$\tau(x) = \mathbb{E}[Y(1) - Y(0) \mid X = x], \tag{2.8}$$

which reflects the expected effect conditional on covariates $X = x$, and the more general concept of *heterogeneous treatment effects (HTE)*.

In contrast to the mean-based estimands above, the *quantile treatment effect (QTE)* evaluates differences in the distribution of potential outcomes. For instance, the median treatment effect is defined as

$$\tau^{(0.5)}(x) = Q_{Y(1)|X=x}(0.5) - Q_{Y(0)|X=x}(0.5), \tag{2.9}$$

where $Q_{Y(t)|X=x}(q)$ denotes the q-th quantile of the potential outcome under treatment t. QTEs are particularly relevant when treatment effects are not symmetrically distributed or when tail behavior is of interest. We will later perform a simulation study using the median for the QTE estimation.

2.6 Assumptions for Identifiability

To identify treatment effects from observational data, several key assumptions are required. *Consistency* ensures that the observed outcome equals the potential outcome under the received treatment. The *Stable Unit Treatment Value Assumption (SUTVA)* assumes no interference between units and that treatments are well-defined. The most critical assumption is *ignorability* (or unconfoundedness), which posits that, conditional on covariates X, the treatment assignment is independent of the potential outcomes:

$$(Y(1), Y(0)) \perp T \mid X. \tag{2.10}$$

In addition, the *positivity* assumption requires that the probability of receiving each treatment is strictly between 0 and 1 for all covariate strata:

$$0 < P(T = 1 \mid X = x) < 1. \tag{2.11}$$

These assumptions are untestable but are necessary for identifying causal effects from non-randomized data.

2.7 Propensity Score Adjustment in Observational Settings

In the absence of randomization, the *propensity score*, defined as the probability of receiving treatment conditional on covariates X,

$$e(X) = P(T = 1 \mid X), \tag{2.12}$$

can be used to balance treatment groups and reduce confounding rosenbaum1983central. When the ignorability assumption holds, adjusting for the propensity score allows for unbiased estimation of average treatment effects. However, while propensity score methods (e.g., matching, stratification, weighting) are effective for estimating population-level quantities like the ATE, they are often insufficient for ITE estimation. Since ITE requires modeling both potential outcomes at the individual level, direct modeling approaches such as outcome regression, meta-learners,

or Bayesian models are typically more appropriate rubin2007design, nie2021quasi. Moreover, reliance on the propensity score alone may fail to capture fine-grained individual heterogeneity necessary for personalized treatment decisions ali2019addressing.

Rubins potential outcomes framework.

Quantile Treatment effect

- also talk about propensity score Rubin(2007) to basically estimate an RCT...and overcome the problem of confounding. but this might work for ATE but not really for ITE, direct modelling of the outcome is necessary

Models for ITE Estimation

Assessing predictive performance with AUC, calibration slope, and brier score. In Leo presentation, he says that recalibration can either be done with deviance statistics or by leave one out (loo) cross validation the slope of this regression would then be the estimated shrinkage factor. T-learner vs s-learner, metalearner, virtual twins

The ITE for a binary endpoint is estimated as the difference of two probabilities (the risk under treatment minus the risk under control). It is essential that the model used to estimate these probabilities is well calibrated and generalizes to new (unseen) data. (Guo *et al.*, 2017) point out that even though modern neural networks became much more accurate in terms of prediction performance, they are no longer well-calibrated. As it may not be a big problem for the sole purpose of making good predictions, it is very problematic for applications where an accurate quantification of the uncertainty is needed. When using models that are estimated with conventional methods such as ordinary least squares or standard maximum likelihood, they tend to overfit on the training data and make too extreme predictions on the test data. This problem increases with reduced sample size, low event rate or large number of predictor variables. To prevent such overfitting in regression models, penalization (shrinkage) methods are proposed as they shrink the estimated coefficients towards zero to reduce the variance in predictions on new data (Riley *et al.*, 2021).

Logistic regression, penalized logistic regression (shrinkage, lasso Shrinkage methods should provide better predictive perfomance on average (cite articles). Calster *et al.* (2020) analyzed different regression shrinkage methods with a binary outcome in a simulation study. They concluded, although the calibration slope improved on average, shrinkage often worked poorly on individual datasets. With small sample size and low number of events per variable the performance of most of these methods were highly variable and should be used with caution in such settings. Riley *et al.* (2021) obtained to similar results in their simulation study. Problems occur, because tuning parameters are often estimated with large uncertainty on the training data and fail to generalize. In both studies the autors pointed out that these penalization methods are more unreliable when needed most, that is when the risk of overfitting may be large.

(Shrinkage shrinks the coefficients so that the calibration slope is improved on a test set. The shrinkage factor can for example be found with n-fold cross validation, as e.g. done by lasso with L1 penalization)

Explain tuning of random forest, with depth number of trees and mtry (ranger?)

TRAM-DAGs with complex shifts or compmlex intercepts can capture heterogeneity. See appendix XXX for an example of ITE estimation with a complex shift.

Use of instrumental variables (IV) to also estimate CATE in presence of unobserved confounders (Nichols, 2007), (Hartford *et al.*, 2017). Frauen and Feuerriegel (2023) propose a model based on IV that is said to also be applicable on observational data.

2.8 Experiments

2.8.1 TRAM-DAG simulation

Show easy simulation with 3 variables and in the results the plots of the loss function, the coefficient learning, intercepts, shifts, and the sampling results. The sampling results should show that the sampled data matches the DGP very well. Also show the estimated parameters of the linear shifts and the intercepts. The complex shift can be shown by plotting the transformation function of X3 with respect to X2. also some queries for observational, interventional and counterfactual.

- take simple example from intermediate presentation. Make counterfactual analysis on continuous X2. (- ideally an experiment with ordinal predictor, and with interpretability and a complex shift and 4 variables.)

2.8.2 ITE real data

We also want to check if we end up with similar results as Chen *et al.* (2025) in their ITE estimation on the IST stroke trial. We will use the same data preprocessing and apply the TRAM-DAG framework as well as a tuned random forest (comets) to estimate the ITE. Both models are trained on a training set and validated on a hold-out test set. For validation, since the ground truth is not known, we first rely on calibration plots to assess the general prediction power for the probabilities. Second, we will predict the potential outcomes (potentially with the re-calibrated models) to estimate the ITE on the train and test set. For visual validation, we will use ITE-ATE plots and ITE-outcome plots. Due to the binary outcome, the ATE is defined as the risk-difference of the individuals in the respective ITE subgroup (bin j) $ATE_j = \mathbb{E}[Y(1) - Y(0) \mid ITE \in bin_j]$.

The cATE is defined as:

$$cATE = \mathbb{E}[Y \mid T = 1] - \mathbb{E}[Y \mid T = 0] \tag{2.13}$$

Since the IST stroke trial is a randomized controlled trial, the full potential of TRAM-DAGs is not needed, since only the outcome has to be modelled as a function of the baseline patient characteristics. Nevertheless, this is not a reason not to apply it. The TRAM-DAG is specified so that only the loss for the outcome node is optimized, the distributions of the other nodes are not estimated. The transformation function for the outcome is modelled by a complex intercept model $h(Y \mid X) = CI(X)$, with 4 hidden layers of shape $(20, 10, 10, 2)$. The numerical values are further standardized and dropout (0.1) is used to prevent severe overfitting (and batchnorm?). The model is trained for..

The Random Forest is specified accoring to the default version of the comets package.

As benchmark, we also fit a t-learner logistic regression.

Results on IST trial with the interpretation in the discussion part.

show results of different models including tram dag.

- maybe also make propensity score estimation on IST stroke trial to check if possibly confounded.

2.8.3 ITE simulations (RCT) - assess when estimations fail

In this section, we will perform a simulation study to estimate the ITE with different models in an RCT setting under different szenarios. The aim is to identify, in which circumstances the ITE estimation fails and if failure is model agnostic, meaning that the reason for failure lies in external factors like (unobserved variables, sample size, effect size) and not the models itself. It should give potential explanations, why modeles for estimating the ITE failed in a real world application as Chen *et al.* (2025) showed. The simulation is based on the data generating process (DGP) that should resemble an RCT. We assume a binary outcome and a set of covariates that

influence the treatment effect. There further potentially exist treatment covariate interacitons that are responsible for heterogeneity of treatment effect. The Szenarios are laid out in (Table XX)

table with szenarios:

table with models:

logistic t-learner, lasso t-learner, lasso s-learner, random forest, tuned random forest (comets), (TRAM-dag was not applied here due to time reasons, it further would not add much value since we want to mainly get a feeling of the bahaviour of ML models in general under different scenarios)

With these scenarios we want to show, what must be fulfilled that ITE estimation works and what implication this has on non-randomized settings.

In this thesis , I will apply Lasso regression on the IST stroke trial and simulation studies, where the sample size is relatively large.

simulation studies "The setup was such that development and test sets were generated from the same data generating mechanism. In practice, there may be differences between these two settings that are not captured by the models, and the uncertainty that accompanies these unknowns may overshadow relatively small gains realized by more complex models."

"This could include the analysis of individual patient data from multiple randomized trials, or even the use of nonrandomized studies for the estimation of outcome risk under a control condition." this motivates the need for observational modeling.

Problems with ITE: (in an RCT setting) - to estimate the ITE we must assume un-confoundedness. Does this also apply to itneractions (effect modifiers)? Check how this is handled in the literature. - when there are treatment covariate interactions and these covariates are in the DGP but dropped from the dataset (so unobserved), then the ITE Estimation failed in the simulations. At least when there is only 1 strongly interacting variable and we drop this one. An example could be the psychological condition of a patient which might also affect how the treatment works, this is not a confounder but an effect modifier, and i would assume that this variable is rarely recorede or measured.

- Maybe a good conclusion: because this problematic with missing effect modifiers in RCT data can be a motivation to work with observational data where the dag is very detailed specified with all confounders and interactions, then a tram-dag can be applied. However, there we also have the problem, that important variables are probably also not known/measured...

- question still to answer: the estimated ITE on the train vs test set is equally bad (in terms of scatterplot and RMSE), so why does the ITE-ATE plot and the ITE Outcome plot looks like it discriminates good in the train set but not in the test set? Could the answer be, that the model is overfitting, hence tries to really model the observed outcomes and not the true probabilities, hence when an inportant variable is missing, it could still reasonably well predict the outcome (probability) but these are not the causal relationships anymore, so therefore the ITE estimation is bad on the train and the test set. But the ITE-ATE plot still looks good in the train set, because at least the observed outcomes could be predicted very well.??? still not sure if this is the case and how to proof.

- another point is the effect of the correlation of the variables. If the X's are strongly correlated, and one X with interaction effect is dropped, can the info then still be retreived from the other variables? maybe the effect is then attributed to another correlated variable. –> check with simulations and or theroretical proof.

2.8.4 ITE estimation in observational data

We claim that if the assumptions for ITE estimation (identifiability, (unobserved) unconfoundedness etc.) are fulfilled and the DAG is fully known, with the TRAM-DAG framework, the ITE can be estimated under observational data just like with RCT data. We aim to proof this with the DAG as displayed in Figure 2.5. The binary treatment (X4) is the intervention variable and we want to estimate the ITE for the continuous outcome Y.

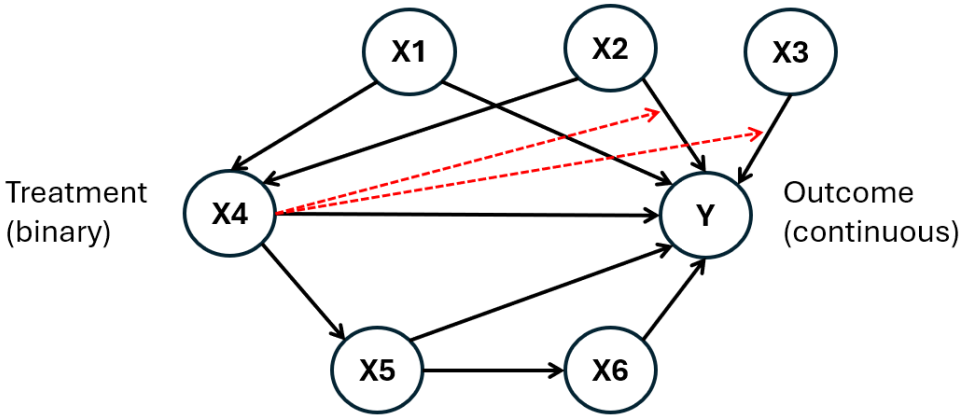

Figure 2.5: DAG used for the experiment to estimate the ITE. DGP: the source nodes X1, X2 and X3 come from a multivariate standard normal distribution with 0.1 correlation. In the observational setting the binary treatment X4 depends on the parents X1 and X2, in the RCT Setting, this dependency is omitted due to randomization. X5 depends on the treatment X4. X6 depends on X5. The outcome Y depends on all variables with additional interaction effects between the treatment and the variables X2 and X3. All variables except the treatment X4 are continuous.

An example scenario that would have the structure of the proposed dag could be the following: A marketing campaign is conducted to increase customer spending. The treatment is the marketing email (X4) that is sent to the customers. If the treatment is not randomized, it depends on the prior total spend (X1) and the customer engagement score (X2). The outcome is the total spend in the next 30 days after receiving the email (X7). The prior total spend and customer engagement score are confounders that influence both the treatment and the outcome. Customer satisfaction score (X3) from a recent survey is another predictor. The time spent on the website after receiving the email (X5) is a mediator that influences the number of product pages viewed (X6), which in turn influences the total spend in the next 30 days.

Data generating mechanism: The standard logistic was chosen as the noise distribution to align with other examples in this thesis. Also any other noise distribution cold be chosen, as we are not interested in interpretability of the coefficients in this experiment. All variables except the binary treatment X4 are continuous. The source nodes X1, X2 and X3 are generated from a multivariate standard normal distribution, where each pair of variables has a correlation of 0.1. These variables represent baseline patient characteristics. In the observational setting, X1 and X2 act as confounders by influencing the treatment allocation X4 and the outcome Y. In the RCT setting, these connections are cut due to randomization. X5 depends on the treatment X4. X6 depends on X5. The log odds for the continuous outcome are linearly depend on all covariates including additional interaction terms between the treatment and X2 and X3. Hence, the log odds of the outcome can be written in terms of a transformation model with linear shift $h(y \mid X) = h_I(y) + \text{LS}$. Equation 2.14 outlines the outcome on the log odds scale

$$h(y \mid \mathbf{X}) = h_I(y) + \boldsymbol{\beta}_X^\top \mathbf{X} + \boldsymbol{\beta}_{TX}^\top \mathbf{X}_{\text{int}} X_4 \tag{2.14}$$

where $h_I(y)$ is the intercept function, $\mathbf{X}$ is the covariate vector including all variables and $\mathbf{X}_{\text{int}} = \{X2, X3\}$ is the vector with the interaction variables that only has an effect if the treatment is present ($X4 = 1$). The intercept $h_I(y)$ has to be a smooth monotonically increasing and function which we defined as $h_I(y) = tan(y/2)/0.2$ in the interval between -2 and 2 and linearly extrapolated the function at the boundaries. The coefficients $\boldsymbol{\beta}_X$ are set to $\boldsymbol{\beta}_X = (-0.5, 0.5, 0.2, 1.5, -0.6, 0.4)$, where 1.5 is the direct effect of the treatment X4 on the outcome. $\boldsymbol{\beta}_{TX}$ are set to $\boldsymbol{\beta}_{TX} = (-0.9, 0.7)$ for the interaction terms.

Experiment: The experiment is conducted with 3 different scenarios of data generating mechanism for the outcome Y accordingly: (1) direct and interaction effect, (2) only direct effect, (3) only interaction effect. Depending on the scenario, the corresponding coefficients in $\boldsymbol{\beta}_X$ and $\boldsymbol{\beta}_{TX}$ are set to zero. The data is generated with a sample size of 20'000 samples for the training set. In both settings, observational and RCT, the TRAM-DAG is first fitted on the data. To allow for full flexibility, all nodes that depend on some parents are modelled by complex intercepts with 3 hidden layers of shape (10, 10, 10). Batch normalization, dropout (0.1) and ReLU activation are used. The model is fitted on the training data consisting of 20'000 samples. To prevent overfitting, an additional validation set with 10'000 samples is used and the model is selected, where the validation loss was is (early stopping).

Once the model is fitted, we obtained the estimated (inverse) transformation functions $X_i = h^{-1}(Z_i \mid pa(X_i))$ that represent the equations $X_i = f(Z_i, pa(X_i))$ in the structural causal model. The process for the ITE estimation is outlined in 3. In a first step to estimate the ITE, we determine the latent values z_{ij} in all observed samples j for the explanatory nodes i - X1, X2, X3, X5 and X6. The latent values are the values of the transformation functions at the observed value of the variable given the observed values of its parents $z_{ij} = h_i(x_{ij} \mid pa(x_{ij}))$. In a second step, these latent values z_{ij} are used to sequentially sample from the two interventional distributions when setting the treatment X4 to either 0 or 1. For each individual, these interventions impact the mediator nodes X5 and X6 as well as the outcome Y. The source nodes X1, X2 and X3 are the same under both treatments. The treatment X4 is the variable which we fix by the do-intervention. X5 and X6 will change according to the treatment. Finally, for each set of samples j (meaning for each individual) we get two distributions for the outcome, one under treatment and one under control. In contrast to the potential outcomes framework, where the potential outcomes are defined as the expected value of the outcome under treatment, we define the potential outcomes as the median of the outcome distribution under treatment - the quantile treatment effect (QTE). For simplicity, we will further refer to the individual treatment effect as ITE even though technically, the QTE is meant. Determining the potential outcomes in terms of the expected values would also be possible, but would require us to repeatedly sample from each resulting potential outcome distribution for each individual and average the results. This was computationally too time consuming and therefore we decided to estimate the QTE instead. In the ITE estimation in the previous examples with binary outcome, this was not necessary, since the potential outcomes were defined as the probabilities of the outcome under treatment and control, hence a single number that represents the expected value.

Maybe visualize the potential outcome transformation funcitons (both funcitons in one plot) and then show that the median Latent value 0 creates the two potential median outcomes on the x axis.

NOTE: in both, the RCT and in the Observational setting, also other models could be applied instead of TRAM-DAG. As long as all confounders are included in the model, we controll for the confounders and can get unbiased results. For example a T-learner Colr($Y \sim X_1 + X_2 + X_3$) (because Colr is basically what we did in the DGP) fitted on both treatment groups separately could be used to estimate the ITE in our proposed experiment. This might only be possible so easily as long as we do not assume additional interactions between the treatment and the mediators X_5 and X_6. If we would assume such interactions, we would have to include these in the model as well, which would make it more complex and possibliy requires to fit and apply multiple models. If there are no interactions with the mediators, they can be omitted, since we are interested in the total treatment effects and not in separating the effect (mediation analysis). But again, we can only omit if these variables do not contain additional information about treatment effect heterogeneity. The reasoning is because to estimate the total effect one should not control for mediators. (check if really true!!!) However, the TRAM-DAG framework is well suited to also deal with mediators and calculate counterfactuals, therefore we think it is a good example to show its capabilities.

Algorithm 3 ITE Estimation (QTE) Using TRAM-DAG in Observational Data

Input: Fitted TRAM-DAG, observational dataset with n samples
for each sample $j = 1$ to n **do**
    Step 1: Encode explanatory nodes
    for each explanatory node $X_i \in \{X_1, X_2, X_3, X_5, X_6\}$ **do**
        Compute latent value: $z_{ij} = h_i(x_{ij} \mid \text{pa}(x_{ij}))$
    end for
    Step 2: Generate potential outcomes under treatment and control
    for $x_4 \in \{0, 1\}$ **do** ▷ Simulate both treatment states
        Fix $X_4 = x_4$ (intervention)
        Sample X_5 and X_6 sequentially using z_{ij} and inverse transformations
        Sample potential outcome $y_j^{(x_4)}$ using $z_{7,i} = 0$ (median of the potential outcome distribution)
    end for
    Step 3: Compute QTE for individual j
    $\text{ITE}_j = \text{median}(y_j^{(1)}) - \text{median}(y_j^{(0)})$
end for
Output: ITE estimates $\{\text{ITE}_j\}_{j=1}^n$

Validation of results: In the data generating mechanism, along with the actually sampled values, the potential values under both treatments are also recorded and used to determine the true QTE (the ITE based on the 50 percent quantiles of the potential outcome distributions of each individual.) The results are displayed by densities of the estimated ITE, the scatterplots of the true vs. estimated ITE, the ITE-ATE plot with the difference in medians as ATE within subgroups to make it comparable to the estimated ITEs. Furthermore the average of all estimated and true (dgp) ITEs are presented in a table (XX) which should be an estimator (?) for the ATE. We further calculate the ATE as the overall difference in medians in the RCT setting and compare it to the estimated values based on the ITEs. If these estimates are comparable, it would support our claim that with TRAM-DAGs it does not matter if the data is from an RCT or observational setting, as long as the assumptions are fulfilled and the DAG is fully known and observed.

2.9 Software

All code was done in R wiht packages xx ussed for yy.

Maybe it is the methods section. Here however, we give a couple hints. Note that you can wisely use *preamble*-chunks. Minimal, is likely:

```
library(knitr)
opts_chunk$set(
    fig.path='figure/ch02_fig',
    self.contained=FALSE,
    cache=TRUE
)
```

Defining figure options is very helpful:

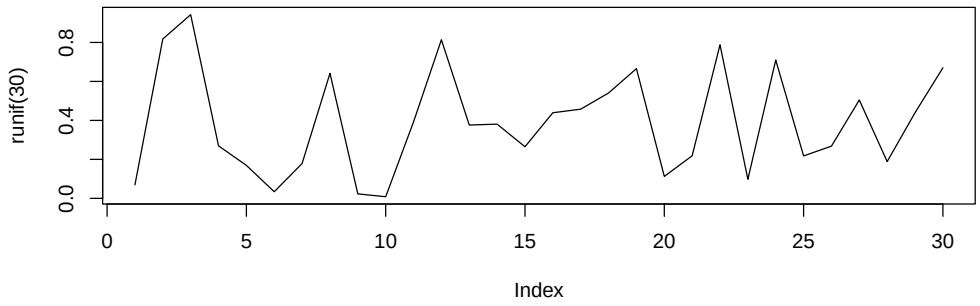

Figure 2.6: Test figure to illustrate figure options used by knitr.

```
library(knitr)
opts_chunk$set(fig.path='figure/ch02_fig',
               echo=TRUE, message=FALSE,
               fig.width=8, fig.height=2.5,
               out.width='\\textwidth-3cm',
               message=FALSE, fig.align='center',
               background="gray98", tidy=FALSE, #tidy.opts=list(width.cutoff=60),
               cache=TRUE
)
options(width=74)
```

This options are best placed in the main document at the beginning. Otherwise a `cache=FALSE` as knitr option is necessary to overrule a possible `cache=TRUE` flag.

Notice how in Figure 2.6 everything is properly scaled.

2.10 Citations

Recall the difference between `\citet{}` (e.g., Chu and George (1999)), `\citep{}` (e.g., (Chu and George, 1999)) and `\citealp{}` (e.g., Chu and George, 1999). For simplicity, we include here all references in the file `biblio.bib` with the command `\nocite{*}`.

Chapter 3

Results

3.1 TRAM-DAGs simple simulation study

Intercepts: show estimates vs. dgp same as in intermediate presentation.

Show the Discrete case with just cutpints (only K-1 parameters of outputs are used) Show the continuous case where the outputs are transformed to monotonically increasing betas for the bernstein polynomial.

Linear and complex shifts:

Here in the first two plots we can see the linear shifts. And in the right plot we have the complex shift of X2 on X3. The estimated shifts match quite well with the DGP.

Complex shift (Interaction example) to show what is also possible:

Here I just want to make a short input from another example. So there the true model was that of a logistic regression with the binary outcome Y and 3 predictors. The binary treatment T and the two continuous predictors X1 and X2. There was also an interaction effect assumed between treatment and X1. So this basically means that the effect of X1 on the outcome is different for the two treatment groups. And here we can show that our TRAM-DAG specified by a complex shift of T and X1 can also capture this interaction effect quite well.

3.2 ITE simulation study - when do causal ML models fail?

In this section, we present the performance of different causal ML models for estimating the ITE under different scenarios. Starting with a favourable scenario where everything is assumed to be known and effect sizes are large, we will sequentially introduce more complexity and uncertainty into the data generating process (DGP) to see how the models perform under less favourable conditions. The scenarios are designed to reflect different real-world situations, such as the presence of (confounding variables - maybe not include, since ITE estimation must assume no unobserved confounding?), interaction effects, and varying treatment effects across individuals and unobserved variables.

The results are presented for all causal ML models per scenario.

In each scenario we present the results of the simple model (glm tlearner, basically dgp) and a complex model (tuned random forest). In the appendix, the results in scenario 1 of a not-tuned random forest are presented as an example that it is crucial to tune the model so that it is appropriate for ITE (and in this case probability calculation) estimation, hence well calibrated.

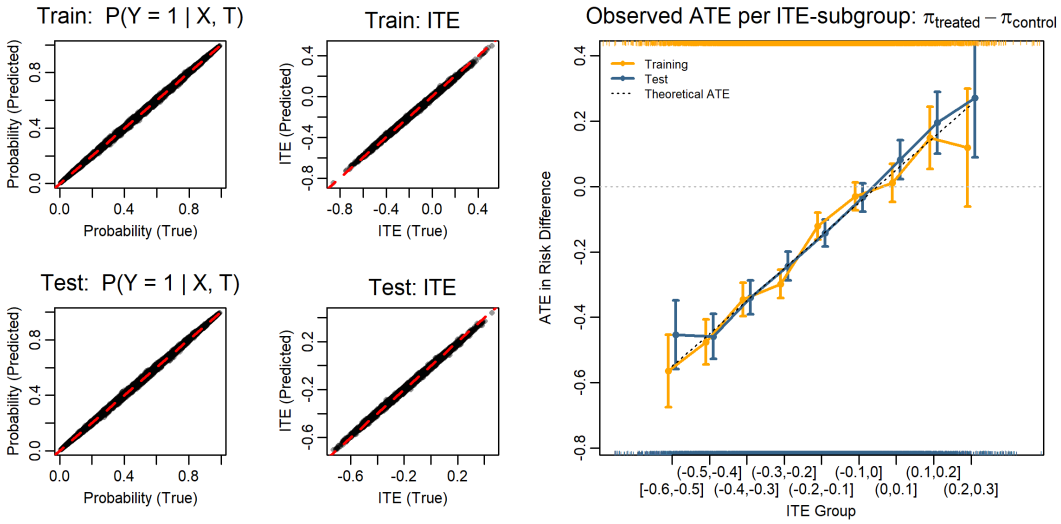

Figure 3.1: Results with the GLM T-learner when the DAG is fully observed, strong effects.

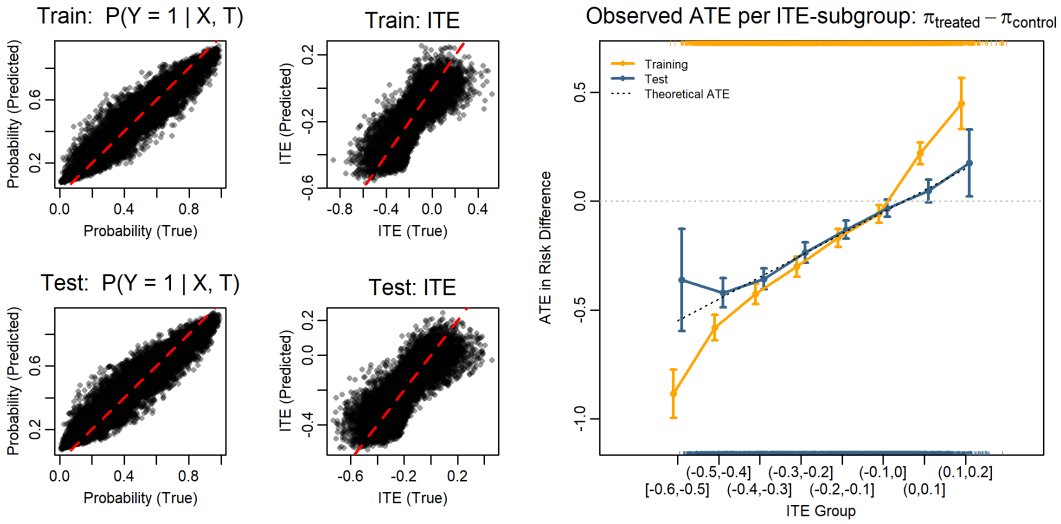

Figure 3.2: Results with the Tuned Random Forest (comets) T-learner when the DAG is fully observed, strong effects.

3.2.1 Scenario (1): Fully observed, large effects

3.2.2 Scenario (2): unobserved interaction

3.3 ITE estimation with TRAM-DAGs

First, we present the results for scenario (1) with a direct and interaction effect. Then, we present the results for scenario (2) with a direct effect but no interaction effects, and finally, scenario (3) with interaction effects but no direct effect of the treatment. For each scenario, we compare the results in an observational setting with confounded treatment allocation and in a randomized controlled trial (RCT) setting without confounders. We also compare the average treatment effect (ATE), which can directly be calculated in the RCT, with the ATE based on the estimated individualized treatment effects. If the estimated ITEs are unbiased, they should be a good estimate of the ATE. All ITEs presented in this section are technically quantile treatment effects (QTEs) based on the 0.5-quantile of the potential outcomes. For simplicity we will refer

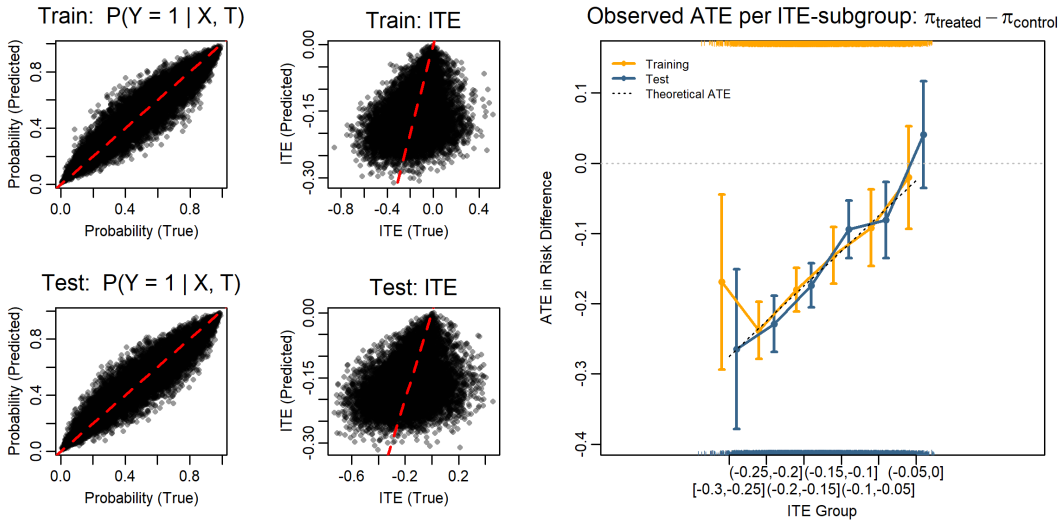

Figure 3.3: Results with the GLM T-learner if the interaction variable is unobserved, strong effects.

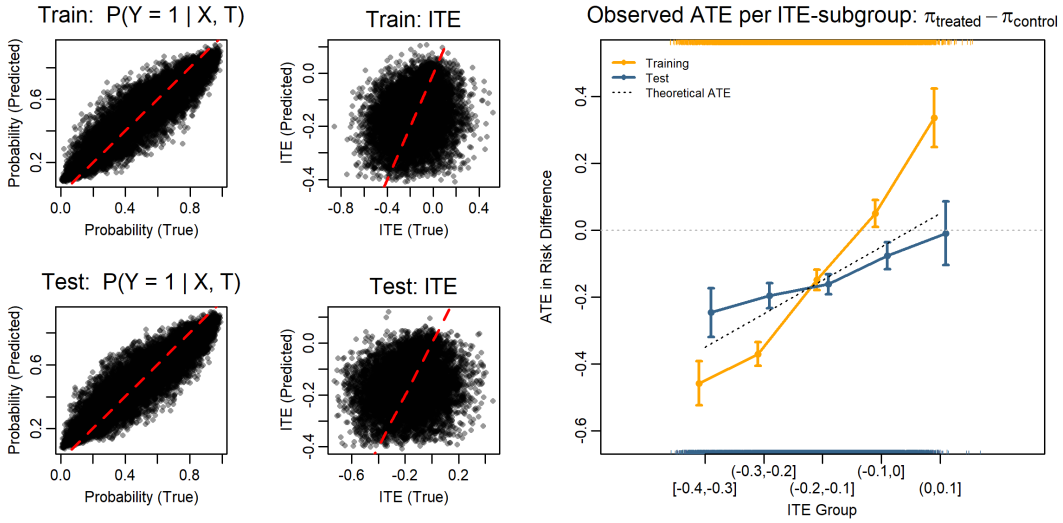

Figure 3.4: Results with the Tuned Random Forest (comets) T-learner if the interaction variable is unobserved, strong effects.

to them as ITEs in the following.

3.3.1 Scenario (1): Direct and interaction effects

Scenario (1) included a direct effect of the treatment on the outcome and an additional interaction effect of the treatment with the covariates X2 and X3. A train and test set were generated with 20'000 observations each. In the observational setting, the treatment allocation was confounded by the covariates X1 and X2. In the train set, 38.6% of patients were in the control group and 61.4% were in the treatment group. This ratio was similar in the test set. In the RCT setting treatment allocation was randomized. In the train set 49.8% individuals were in the control group and 50.2% in the treatment group. In the test set 50.2% were in the control group and 49.8% in the treatment group. Figure 3.5 illustrates the true ITE distribution that resulted from the DGP. Due to the interaction effects, there is some heterogeneity in the ITE distribution. Figure 3.6 shows the marginal distributions of all variables according to the DGP

and the estimates of the fitted TRAM-DAG. Figure 3.7 shows the distribution of the outcome under the do(Tr=0) and do(Tr=1) interventions. The fitted model was applied to estimate the ITEs in terms of the difference in medians of the potential outcomes. The resulting density of the estimated ITEs compared to the true ITEs according to the DGP is shown in Figure 3.8. Across both settings, the densities of the estimated ITEs are close to the true densities in both the training and test datasets. Figure 3.9 shows the scatterplots of true against estimated ITEs. Finally, Figure 3.10 displays the ITE-ATE plot where the ATE is computed as the difference in medians of the observed outcome under the treatments within the respective ITE-subgroups The trends observed in the training and test sets are consistent.

The average treatment effect (ATE) is presented in Table 3.1. In the RCT setting in the training set, the difference in means of the outcomes in the two treatment groups was -0.563 with a confidence interval of -0.582 to -0.543. The ATE in terms of the difference in medians of the observed outcomes was -0.626. Also in the training set, the ATE in terms of the mean of the true ITEs was -0.62 and the ATE in terms of the mean of the estimated ITEs was -0.619. All measures, including the ones from the test datasets, are shown in Table 3.1.

NOTE: also add CIs in the table with the ATEs?

Table 3.1: Scenario (1), including direct and interaction effects: Comparison of ATE measures across train and test sets for the observational and RCT setting.

Measure	**Observational**		**RCT**	
	Train	**Test**	**Train**	**Test**
ATE as mean($Y^{(1)}_{\text{observed}}$) − mean($Y^{(0)}_{\text{observed}}$)	NA	NA	-0.563	-0.563
ATE as median($Y^{(1)}_{\text{observed}}$) − median($Y^{(0)}_{\text{observed}}$)	NA	NA	-0.626	-0.638
ATE as mean(ITE_{true})	-0.62	-0.622	-0.62	-0.622
ATE as mean($\text{ITE}_{\text{estimated}}$)	-0.617	-0.62	-0.619	-0.622

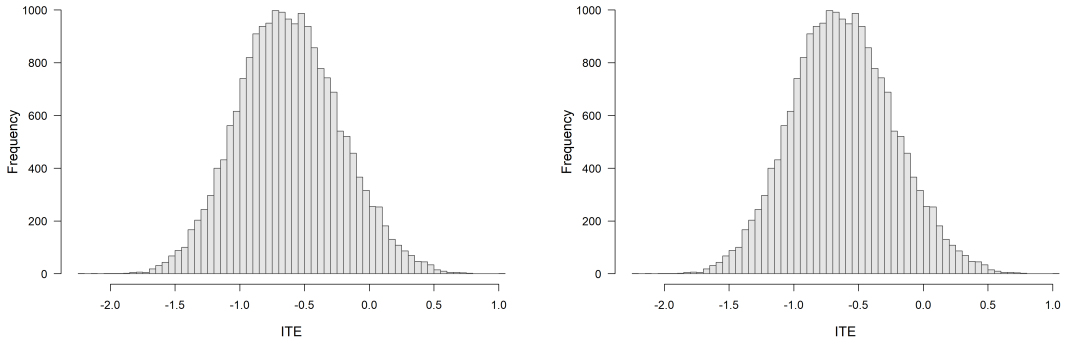

Figure 3.5: True ITE distribution resulting from the DGP for scenario (1) with direct and interaction effects. The true ITEs are identical in the observational and in the RCT setting, since they depend on the potential outcomes under both treatment allocations. Left: Observational; Right: RCT setting.

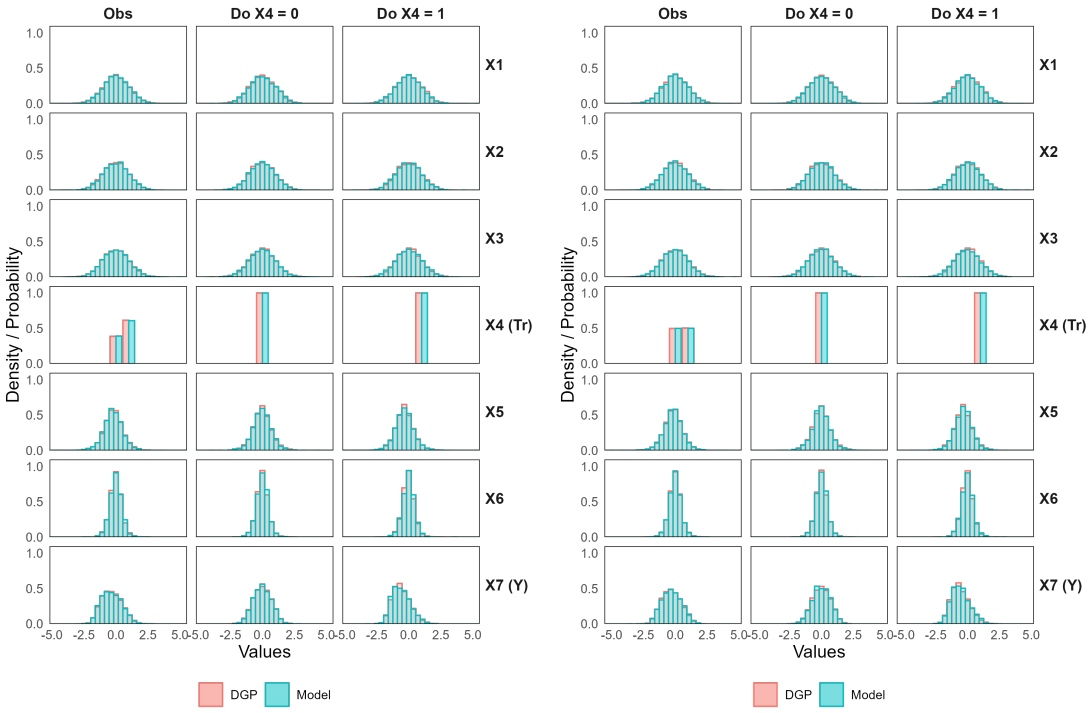

Figure 3.6: Marginal distributions of DGP variables and fitted TRAM-DAG samples for scenario (1) with direct and interaction effects. The distributions shown as observed (Obs), under control intervention (Do $X4 = 0$) and under treatment intervention (Do $X4 = 1$). Left: Observational; Right: RCT setting.

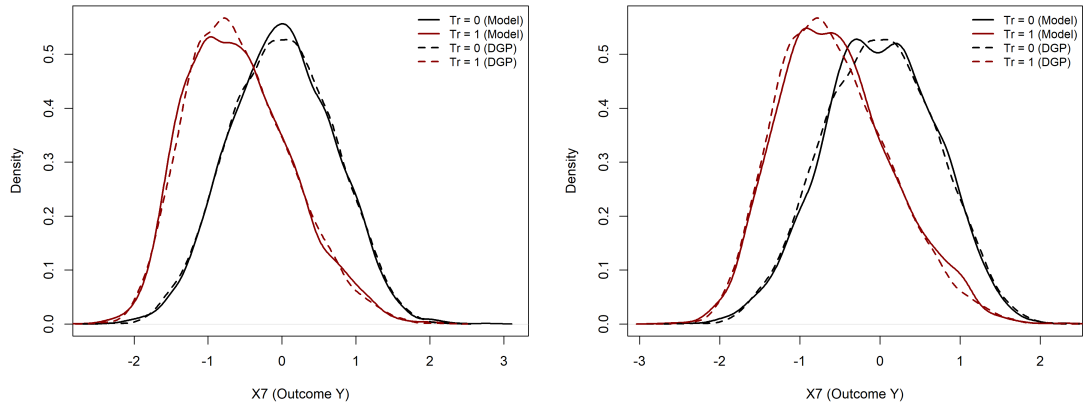

Figure 3.7: Distributions of the outcome variable (X7) under treatment and control interventions for scenario (1), including direct and interaction effects. This plot is a higher resolution view of the X7 panels (Do $X4 = 0$) and (Do $X4 = 1$) from Figure 3.6. Left: Observational; Right: RCT setting.

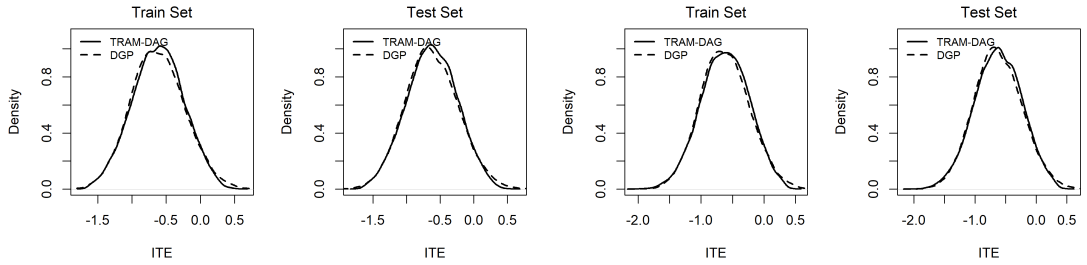

Figure 3.8: Densities of estimated ITEs compared to the true ITEs in the training and test datasets for scenario (1), including direct and interaction effects. Left: Observational; right: RCT setting.

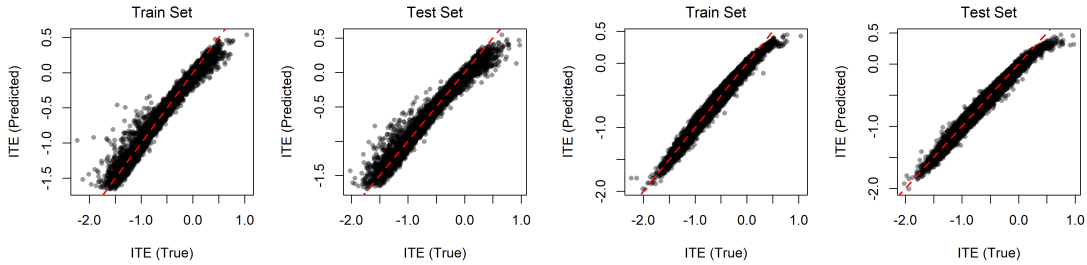

Figure 3.9: Scatterplots of estimated ITEs compared to the true ITEs in the training and test datasets for scenario (1), including direct and interaction effects. Left: Observational; right: RCT setting.

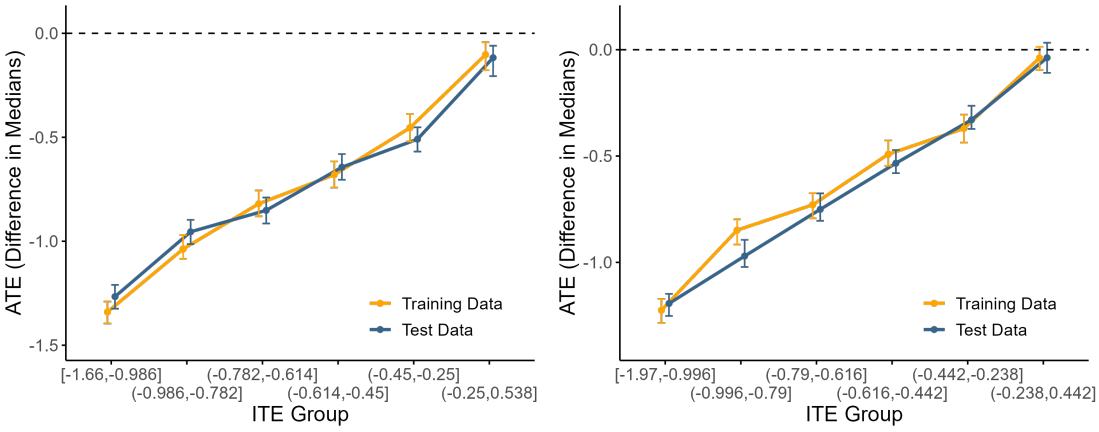

Figure 3.10: ITE-ATE plot for scenario (1), including direct and interaction effects. Individuals are grouped into bins according to the estimated ITE and in each bin the ATE is calculated as the difference in medians of the observed outcomes under the treatments. 95% bootstrap confidence intervals indicate the uncertainty. Left: Observational; right: RCT setting.

3.3.2 Scenario (2): With direct but no interaction effects

Scenario (2) included a direct effect of the treatment on the outcome and coefficients of the interaction effects are set to zero. This results in less heterogeneity of ITE compared to scenario (1) as shown in Figure 3.11. The observational and interventional densities sampled by the fitted TRAM-DAG are aligned with the true densities according to the DGP as illustrated in Figures 3.12 and 3.13. A notable discrepancy in variance exists between the estimated and true ITEs, as illustrated in Figures 3.14 and 3.15. The ITE-ATE plot in Figure 3.16 shows a less informative view compared to scenario (1). Table 3.2 presents the ATE measures for scenario (2). In the test set of the RCT setting, the ATE in terms of the difference in medians of the observed outcomes was -0.639. In contrast, the ATE based on the estimated ITEs in the same dataset was -0.586.

Table 3.2: Scenario (2), including a direct treatment but no interaction effects: Comparison of ATE measures across train and test sets for the observational and RCT setting.

Measure	**Observational**		**RCT**	
	Train	**Test**	**Train**	**Test**
ATE as mean($Y^{(1)}_{\text{observed}}$) − mean($Y^{(0)}_{\text{observed}}$)	NA	NA	-0.569	-0.572
ATE as median($Y^{(1)}_{\text{observed}}$) − median($Y^{(0)}_{\text{observed}}$)	NA	NA	-0.629	-0.639
ATE as mean(ITE_{true})	-0.633	-0.633	-0.633	-0.633
ATE as mean($\text{ITE}_{\text{estimated}}$)	-0.645	-0.644	-0.587	-0.586

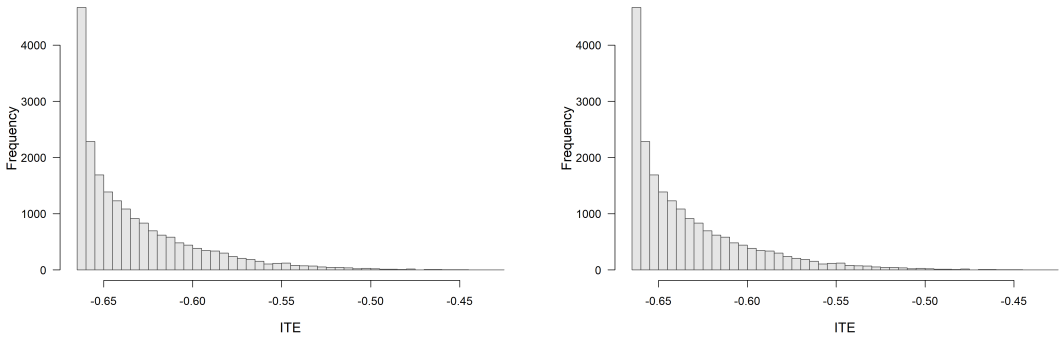

Figure 3.11: True ITE distribution resulting from the DGP for scenario (2), including a direct treatment but no interaction effects. The true ITEs are identical in the observational and in the RCT setting, since they depend on the potential outcomes under both treatment allocations. Left: Observational; Right: RCT setting.

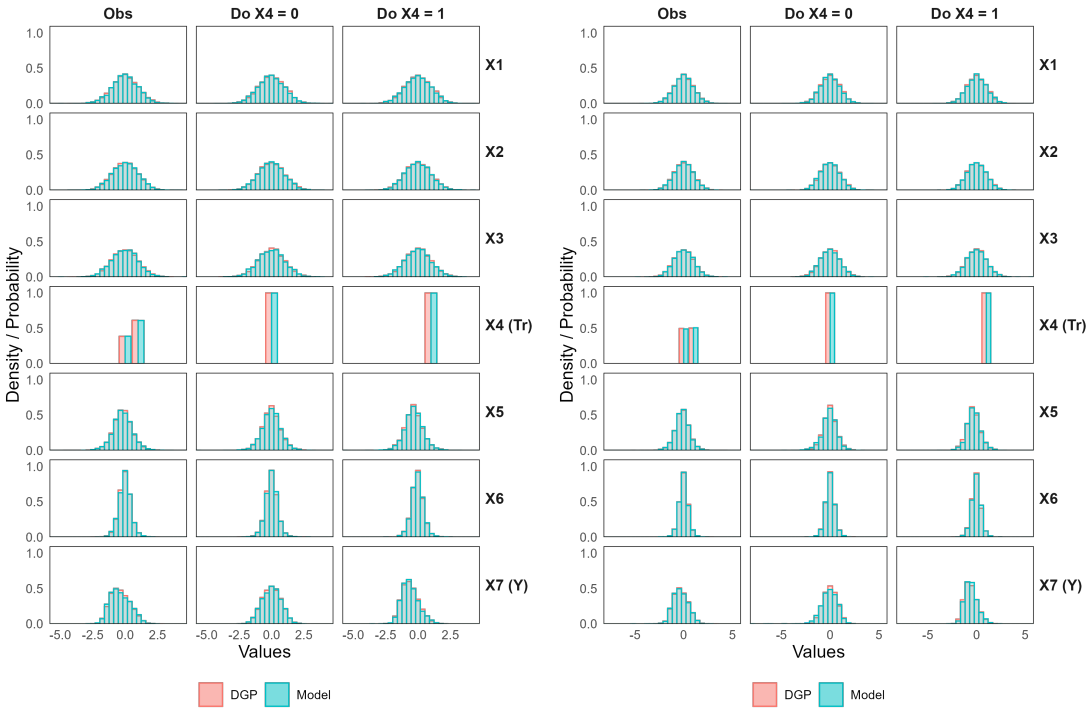

Figure 3.12: Marginal distributions of DGP variables and fitted TRAM-DAG samples for scenario (2), including a direct treatment but no interaction effects. The distributions shown as observed (Obs), under control intervention (Do $X4 = 0$) and under treatment intervention (Do $X4 = 1$). Left: Observational; Right: RCT setting.

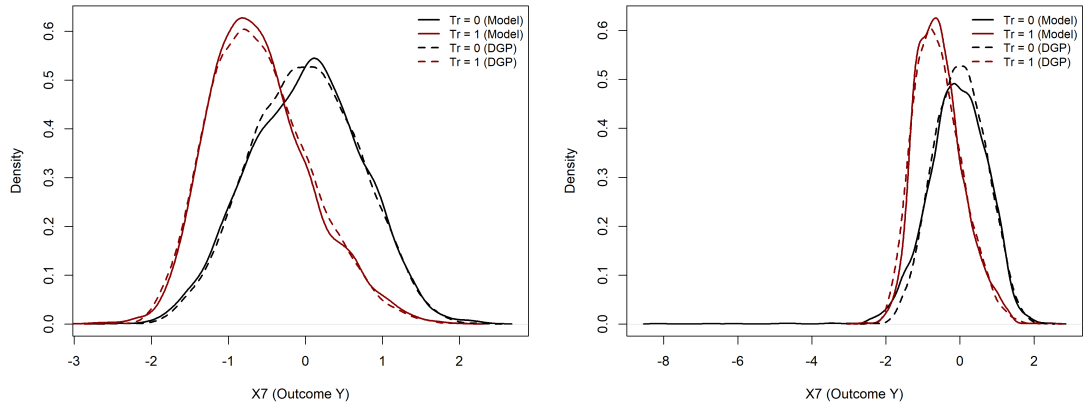

Figure 3.13: Distributions of the outcome variable (X7) under treatment and control interventions for scenario (2), including a direct treatment but no interaction effects. This plot is a higher resolution view of the X7 panels (Do $X4 = 0$) and (Do $X4 = 1$) from Figure 3.12. Left: Observational; Right: RCT setting.

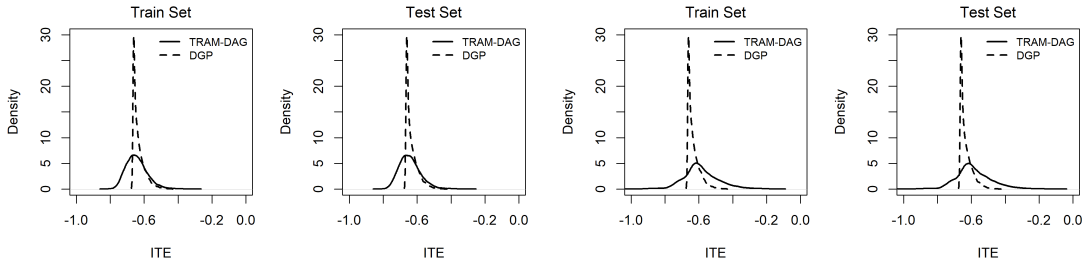

Figure 3.14: Densities of estimated ITEs compared to the true ITEs in the training and test datasets for scenario (2), including a direct treatment but no interaction effects. Left: Observational; right: RCT setting.

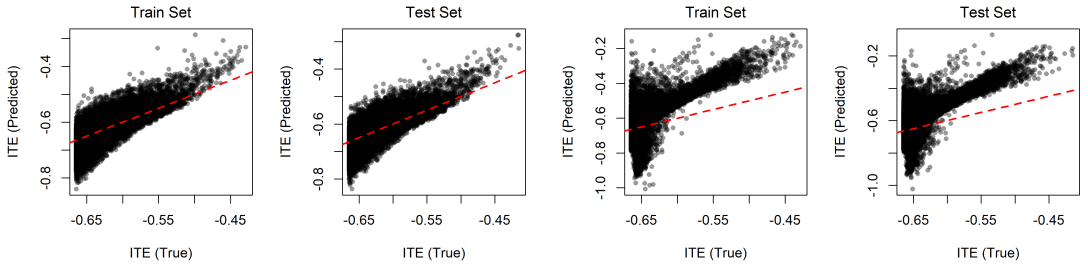

Figure 3.15: Scatterplots of estimated ITEs compared to the true ITEs in the training and test datasets for scenario (2), including a direct treatment but no interaction effects. Left: Observational; right: RCT setting.

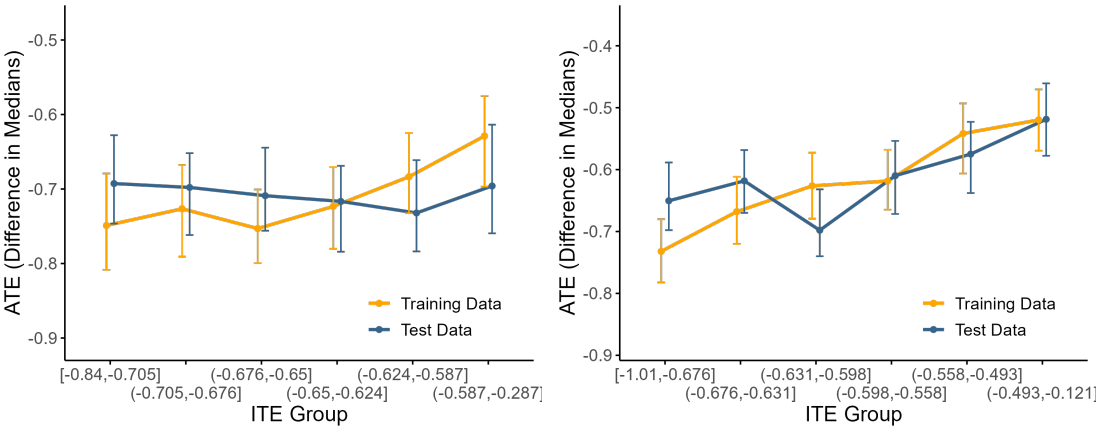

Figure 3.16: ITE-ATE plot for scenario (2), including a direct treatment but no interaction effects. Individuals are grouped into bins according to the estimated ITE and in each bin the ATE is calculated as the difference in medians of the observed outcomes under the treatments. 95% bootstrap confidence intervals indicate the uncertainty. Left: Observational; right: RCT setting.

Â¨

3.3.3 Scenario (3): No direct but with interaction effects

Scenario (3) included no direct effect of the treatment on the outcome but it included interaction effects of the treatment with the covariates X2 and X3. Compared to scenario (1), when excluding the direct effect of the treatment, the distribution of ITEs is more centered as shown in Figure 3.17. The ATE in terms of the mean difference in the test set of the RCT setting is -0.048 with a confidence interval of -0.068 to -0.028.

Table 3.3: Scenario (3), without direct treatment effect but including interaction effects: Comparison of ATE measures across train and test sets for the observational and RCT setting.

Measure	**Observational**		**RCT**	
	Train	**Test**	**Train**	**Test**
ATE as mean($Y^{(1)}_{\text{observed}}$) − mean($Y^{(0)}_{\text{observed}}$)	NA	NA	-0.048	-0.048
ATE as median($Y^{(1)}_{\text{observed}}$) − median($Y^{(0)}_{\text{observed}}$)	NA	NA	-0.048	-0.059
ATE as mean(ITE_{true})	-0.065	-0.068	-0.065	-0.068
ATE as mean($\text{ITE}_{\text{estimated}}$)	-0.059	-0.061	-0.051	-0.053

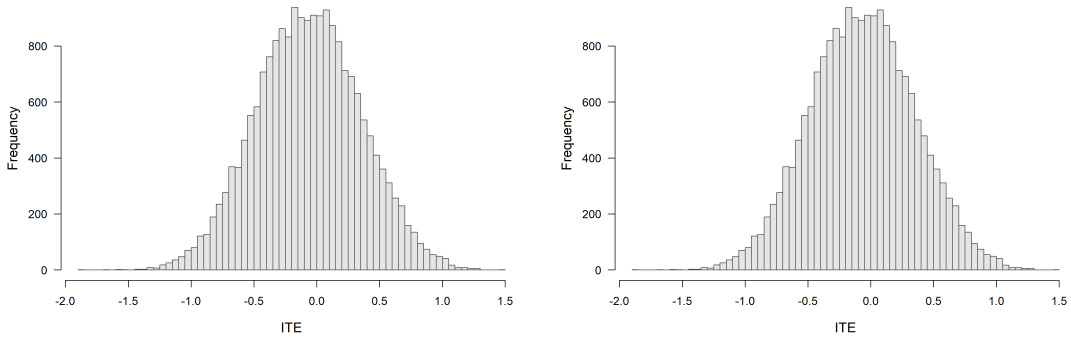

Figure 3.17: True ITE distribution resulting from the DGP for scenario (3), without direct treatment effect but including interaction effects. The true ITEs are identical in the observational and in the RCT setting, since they depend on the potential outcomes under both treatment allocations. Left: Observational; Right: RCT setting.

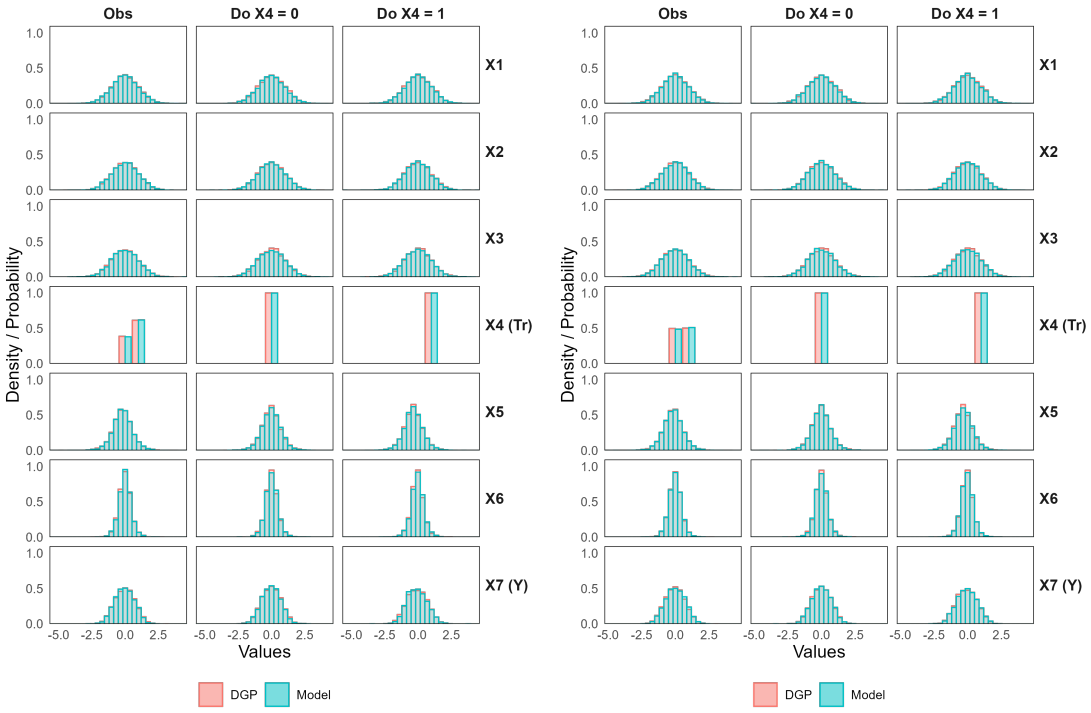

Figure 3.18: Marginal distributions of DGP variables and fitted TRAM-DAG samples for scenario (3), without direct treatment effect but including interaction effects. The distributions shown as observed (Obs), under control intervention (Do $X4 = 0$) and under treatment intervention (Do $X4 = 1$). Left: Observational; Right: RCT setting.

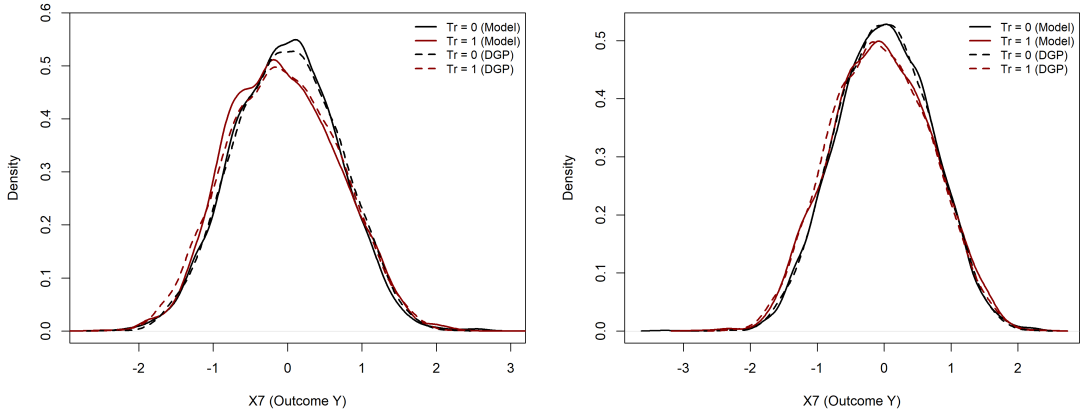

Figure 3.19: Distributions of the outcome variable (X7) under treatment and control interventions for scenario (3), without direct treatment effect but including interaction effects. This plot is a higher resolution view of the X7 panels (Do $X4 = 0$) and (Do $X4 = 1$) from Figure 3.18. Left: Observational; Right: RCT setting.

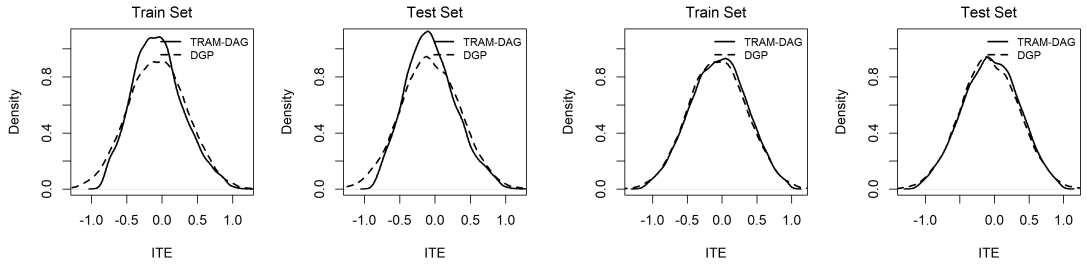

Figure 3.20: Densities of estimated ITEs compared to the true ITEs in the training and test datasets for scenario (3), without direct treatment effect but including interaction effects. Left: Observational; right: RCT setting.

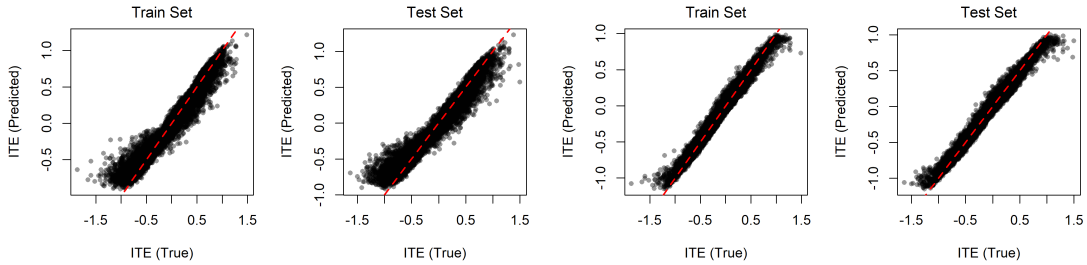

Figure 3.21: Scatterplots of estimated ITEs compared to the true ITEs in the training and test datasets for scenario (3), without direct treatment effect but including interaction effects. Left: Observational; right: RCT setting.

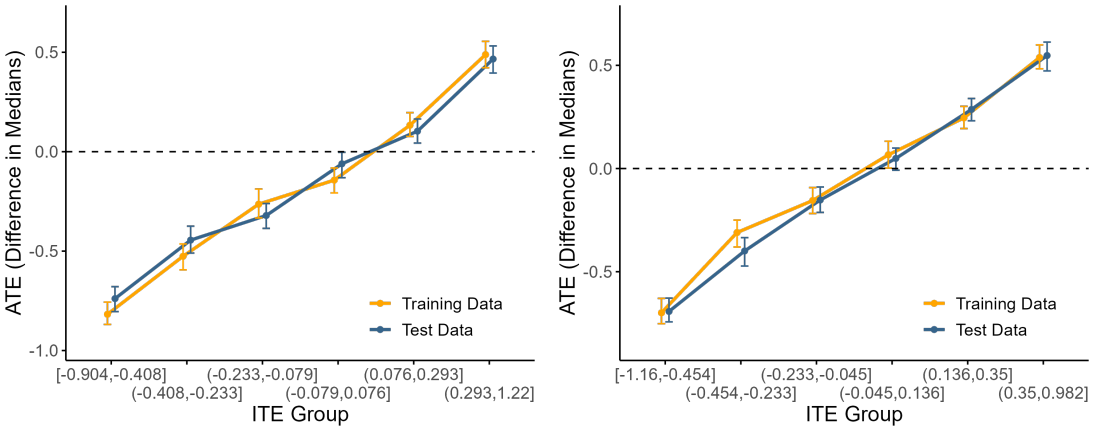

Figure 3.22: ITE-ATE plot for scenario (3), without direct treatment effect but including interaction effects. Individuals are grouped into bins according to the estimated ITE and in each bin the ATE is calculated as the difference in medians of the observed outcomes under the treatments. 95% bootstrap confidence intervals indicate the uncertainty. Left: Observational; right: RCT setting.

Chapter 4

Discussion and Outlook

Check all of the following again when including the final Experiment results in section 3 (here written just from memory):

4.1 Experiment 1: TRAM-DAG simulation

The tram dag can accurately estimate the causal dependencies with interpretable coefficients.

4.2 Experiment 2: ITE estimation - IST stroke trial

We used the same data as was used by **??**. Both models, the tuned RF and the TRAM-DAG did not generalize to the test set. The results are very similar to the ones of the original paper. Calibration seemed to be not bad in both cases...Possible reasons could be small true heterogeneity, low effect size of the treatment, missing important variables (e.g. effect modifiers/interaction variables with treatment). In the the next section (discussion for experiment 3) we are looking into those cases in a simulation study.

4.3 Experiment 3: When do causal ML models fail? (ITE simulation study)

All models achieve good performance as long as the effect sizes are large and the dag is fully observed. Once effect sizes and through that heterogeneity gets smaller, the models become powerful (which is obvious since we can not estimate an effect where there isnt an effect in reality). But in the training sets the complex models still estimate a quite high ITE but this doesnt generalize to the test sets. The largest problem occured when an effect modifier (interaction variable was unobserved), meaning that it was included in the data generating mechanism but not included in the dataset for training the models.

4.4 Experiment 4: TRAM-DAGs in Observational vs. RCT setting (ITE simulation study)

We analyzed ITE estimation under an observational setting (confounded) and under an RCT setting (randomized treatment allocation) in three different scenarios - direct and interaciton treatment effect, only direct but no interaction effect, and no direct but with interaction effect. We noticed that in the first scenario with

What might be surprising is that in scenario 1 where we dont have explicitly included interaction terms in the data generating process, there is still some heterogeneity in the treatment effect

(as shown in figure XX). One might expect that the ITE is constant across all individuals in such a case. However since we used a non linear transformatino function as intercept in the data generating process (as would likely be the case in a real world setting), the treatment effect is not constant across all individuals (that is the ATE). When a linear transformation function would be applied (as for example a linear regression is specified, where the latent noise distribution would be the standard normal and the transformation function would be linear) then the noise term cancels out when calculating the ITE, leading to a constant ITE when no interactions are present: $\text{ITE} = \text{E}[Y(1)] - \text{E}[Y(0)] = (\beta_0 + \beta_t 1 + \beta_x X + \epsilon) - (\beta_0 + \beta_t 0 + \beta_x X + \epsilon) = \beta_t$.

In a model with nonlinear transformation, as in this experiment, the noise term does not cancel out anymore leading to different ITEs for patients with different characteristics.

$$\text{ITE} = \text{E}[Y(1) - Y(0)] = \text{E}[h^{-1}(Z + \beta_t 1 + \beta_x X)] - \text{E}[h^{-1}(Z + \beta_t 0 + \beta_x X)] \tag{4.1}$$

where h is the nonlinear transformation function, Z is the latent noise term, β_t is the direct treatment effect and β_x are the coefficients of the covariates. The state of the covariates X alters the position on the transformation function and thereby affects the difference between the two terms. If the transformation was fixed to be linear, the difference would be constant independent of the state of the covariates X. (This also has to do with non-collapsibility as discussed by susanne and torsten , also check Beates Mail 21.06.2025, and chatgpt discussion)

Chapter 5

Conclusions

We showed how TRAM-DAGS can be applied do estimate the causal relationships in a given fully observed DAG. We pointed out the importance of individualized treatment effects, for example in personnalized medicine or targeted marketing. Calibration of causal ML models is key to achieve an accurate ITE estimation. Also the trade off between complexity and generalizability becomes more important in this application compared to sole predictive modelling. We pointed out potential pitfalls that can emerge in real world settings and should be paid attention towards. These can be for example too little heterogeneity or general poor effect of the treatment, or the fact that there could be unobserved effect modifiers (treatment-covariate interactions). In terms of effect modifiers, methods in literature have already been proposed such as instrumental variables (IV) or Negative Controls (?) where additional variables in a special dependency to the treatment and exposure are used to adjust for unobserved variables (confounders or effect modifiers?). However, it strongly depends on the setting and it is not guaranteed that there exist such supporting variables. We claim that if we know the structure of the DAG, with TRAM-DAGs we can estimate the ITE regardless if we have a RCT or observational data. The only requirement is that the DAG is correct and fully observed, i.e. no unobserved confounders or effect modifiers exist. And since the average treatment effect (ATE) is the average of the individual treatment effects (ITE), we can also estimate the ATE from the ITEs. This implies that running an expensive RCT is not necessary if we have a good observational dataset and know the DAG structure. Our last experiment supports this claim. We used the medians of the potential outcomes to calculate the ITE, however, if the ITE was calculated based on the expected values, it would be directly comparable to the ATE from the RCT in terms of the difference in means, which might be a more classical measure.

Bibliography

Calster, B. V., van Smeden, M., Cock, B. D., and Steyerberg, E. W. (2020). Regression shrinkage methods for clinical prediction models do not guarantee improved performance: Simulation study. *Statistical Methods in Medical Research*, **29**, 3166–3178. 14

Chen, H., Aebersold, H., Puhan, M. A., and Serra-Burriel, M. (2025). Causal machine learning methods for estimating personalised treatment effects – insights on validity from two large trials. 4, 15

Chu, E. and George, A. (1999). *Inside the FFT Black Box: Serial and Parallel Fast Fourier Transform Algorithms*. CRC Press. 20

Collett, D. (2014). *Modelling Survival Data in Medical Research*. Chapman & Hall/CRC Texts in Statistical Science. Taylor & Francis, third edition.

Curth, A., Peck, R. W., McKinney, E., Weatherall, J., and van der Schaar, M. (2024). Using machine learning to individualize treatment effect estimation: Challenges and opportunities. *Clinical Pharmacology & Therapeutics*, **115**, 710–719. 12

Frauen, D. and Feuerriegel, S. (2023). Estimating individual treatment effects under unobserved confounding using binary instruments. Accepted at ICLR 2023. 14

Freedman, B. (1987). Equipoise and the ethics of clinical research. *The New England journal of medicine*, **317**, 141–145. 1

Guo, C., Pleiss, G., Sun, Y., and Weinberger, K. Q. (2017). On calibration of modern neural networks. In Precup, D. and Teh, Y. W., editors, *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, volume 70 of *Proceedings of Machine Learning Research*, 1321–1330. PMLR. 14

Hariton, E. and Locascio, J. J. (2018). Randomised controlled trials - the gold standard for effectiveness research. *BJOG: An International Journal of Obstetrics & Gynaecology*, **125**, 1716 – 1716. 1

Hartford, J., Lewis, G., Leyton-Brown, K., and Taddy, M. (2017). Deep iv: a flexible approach for counterfactual prediction. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, 1414–1423. JMLR.org. 14

Herzog, L., Kook, L., Götschi, A., Petermann, K., Hänsel, M., Hamann, J., Dürr, O., Wegener, S., and Sick, B. (2023). Deep transformation models for functional outcome prediction after acute ischemic stroke. *Biometrical Journal*, **65**, 2100379. 8

Hoogland, J., IntHout, J., Belias, M., Rovers, M. M., Riley, R. D., E. Harrell Jr, F., Moons, K. G. M., Debray, T. P. A., and Reitsma, J. B. (2021). A tutorial on individualized treatment effect prediction from randomized trials with a binary endpoint. *Statistics in Medicine*, **40**, 5961–5981.

Hothorn, T., Kneib, T., and Bühlmann, P. (2014). Conditional transformation models. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, **76**, 3–27. 3, 5

Hothorn, T., Möst, L., and Bühlmann, P. (2018). Most likely transformations. *Scandinavian Journal of Statistics*, **45**, 110–134. 8

Hurrell, J. W., Kushnir, Y., Ottersen, G., and Visbeck, M. (2013). *An Overview of the North Atlantic Oscillation*, 1–35. American Geophysical Union.

Kingma, D. P. and Ba, J. (2015). Adam: A method for stochastic optimization. Published as a conference paper at the 3rd International Conference for Learning Representations, San Diego, 2015. 9

Kratzer, G. and Furrer, R. (2018). *varrank: an R package for variable ranking based on mutual information with applications to observed systemic datasets*. R package version 0.3.

Nichols, A. (2007). Causal inference with observational data. *The Stata Journal*, **7**, 507 – 541. 1, 14

Pearl, J. (1995). Causal diagrams for empirical research. *Biometrika*, **82**, 669–688. 11

Pearl, J. (2009a). Causal inference in statistics: An overview. *Statistics Surveys*, **3**, 96 – 146. 1

Pearl, J. (2009b). *Causality: Models, Reasoning and Inference*. Cambridge University Press, USA, 2nd edition. 2, 3, 12

Porcu, E., Furrer, R., and Nychka, D. (2020). 30 years of space-time covariance functions. *Wiley Interdisciplinary Reviews. Computational Statistics*, **13:e1512**, 1–24.

Reisach, A. G., Seiler, C., and Weichwald, S. (2021). Beware of the simulated dag! causal discovery benchmarks may be easy to game. In *Proceedings of the 35th International Conference on Neural Information Processing Systems*, NIPS '21. Curran Associates Inc., Red Hook, NY, USA. 10

Riley, R. D., Snell, K. I., Martin, G. P., Whittle, R., Archer, L., Sperrin, M., and Collins, G. S. (2021). Penalization and shrinkage methods produced unreliable clinical prediction models especially when sample size was small. *Journal of Clinical Epidemiology*, **132**, 88–96. 14

Sick, B. and Dürr, O. (2025). Interpretable neural causal models with tram-dags. Accepted at the CLeaR 2025 Conference. 1, 3

Sick, B., Hathorn, T., and Dürr, O. (2021). Deep transformation models: Tackling complex regression problems with neural network based transformation models. In *2020 25th International Conference on Pattern Recognition (ICPR)*, 2476–2481. 3, 5, 7

Wang, C. and Furrer, R. (2020). Monte Carlo permutation tests for assessing spatial dependence at different scales. In La Rocca, M., Liseo, B., and Salmaso, L., editors, *Nonparametric Statistics. ISNPS 2018. Springer Proceedings in Mathematics & Statistics*, volume 339, 503–511. Springer.

Chapter 6

Appendix

6.1 Negative Log Likelihood

6.1.1 Continuous Outcome

For a continuous outcome Y the CDF is given by:

$$F_{Y|\mathbf{X}=\mathbf{x}}(y) = F_Z(h(s(y) \mid \mathbf{x})) \tag{6.1}$$

where in our case F_Z is the cumulative distribution function of the standard logistic distribution

$$F_Z(z) = \frac{1}{1+e^{-z}}, \quad z \in \mathbb{R} \tag{6.2}$$

and h is the conditional transformation function that maps the scaled outcome $s(y)$ to the latent scale Z (log-odds).

The outcome y has to be scaled onto the range $[0, 1]$, because the Bernstein polynomial is bounded:

$$s(y) = \frac{y - \min(y)}{\max(y) - \min(y)} \tag{6.3}$$

This scaling also has to be considered when taking the derivative to get the PDF with the change of variables formula:

$$f_{Y|\mathbf{X}=\mathbf{x}}(y) = f_Z(h(s(y) \mid \mathbf{x})) \cdot h'(s(y) \mid \mathbf{x}) \cdot s'(y) \tag{6.4}$$

Where f_Z is the PDF of the standard logistic distribution:

$$f_Z(z) = \frac{e^z}{(1+e^z)^2}, \quad z \in \mathbb{R} \tag{6.5}$$

Finally, the NLL-contributions are then given by the negative log-densities evaluated at the observations.

$$\text{NLL} = -\log(f_{Y|\mathbf{X}=\mathbf{x}}(y)) \tag{6.6}$$

The full formula is given by

$$\begin{aligned} \text{NLL} = -\log f_{Y|\mathbf{X}=\mathbf{x}}(y) = &-h(s(y) \mid \mathbf{x}) - 2\log(1 + \exp(-h(s(y) \mid \mathbf{x}))) \\ &+ \log h'(s(y) \mid \mathbf{x}) - \log(\max(y) - \min(y)) \end{aligned} \tag{6.7}$$

6.1.2 Discrete Outcome

The for a discrete outcome (binary, ordinal, categoric) with categories y_k, $k = 1, \ldots, K$, the CDF is given by:

$$F(Y_k \mid \mathbf{X}) = F_Z(h(y_k \mid \mathbf{x})) \tag{6.8}$$

The likelihood contributions are then given by

$$l_i(y_k \mid \mathbf{x}) = f_{Y_k|\mathbf{X}=\mathbf{x}}(y_k) = \begin{cases} F_Z(h(y_k \mid \mathbf{x})) & k = 1 \\ F_Z(h(y_k \mid \mathbf{x})) - F_Z(h(y_{k-1} \mid \mathbf{x})) & k = 2, \ldots, K-1 \\ 1 - F_Z(h(y_{k-1} \mid \mathbf{x})) & k = K \end{cases} \tag{6.9}$$

from which the NLL-contributions are derived

$$\text{NLL} = -\log(f_{Y_k|\mathbf{X}=\mathbf{x}}(y) \tag{6.10}$$

6.2 Interpretation of Linear Coefficients

The transformation model framework allows for interpretation of the coefficients in the linear shift. Consider the conditional transformation function:

$$F_{X_2|X_1}(x_2) = \text{expit}(h(x_2) + \beta_{12}x_1), \tag{6.11}$$

where $h(x_2)$ is a smooth, monotonic transformation (e.g., a Bernstein polynomial), and β_{12} is a linear coefficient encoding the effect of X_1 on X_2.

Taking the logit (inverse expit) yields:

$$\log\left(\frac{F_{X_2|X_1}(x_2)}{1 - F_{X_2|X_1}(x_2)}\right) = h(x_2) + \beta_{12}x_1. \tag{6.12}$$

This linear additive structure allows the interpretation of β_{12}. The odds ratio when increasing x_1 by one unit is:

$$\text{OR}_{x_1 \to x_1+1} = \frac{\exp(h(x_2) + \beta_{12}(x_1 + 1))}{\exp(h(x_2) + \beta_{12}x_1)} = \exp(\beta_{12}). \tag{6.13}$$

Interpretation: The quantity $\exp(\beta_{12})$ represents the **multiplicative change in odds** for $X_2 \leq x_2$ when increasing X_1 by one unit, holding all else constant.

6.3 Encoding of discrete variables

In the TRAM-DAG a variable X_i can act as a predictor variable for a child node, or as an outcome (child node) that depends on some parent nodes. When X_i is acting as an outcome, the distribution of the variable X_i represented by the transformation function h which estimates a cut-point for each variable. So different form of intercept h_i is used compared to a continuous outcome variable.

If a discrete variable X_i with K categories is used as a predictor variable, it should be dummy encoded. This is done by creating $K - 1$ binary variables, where each variable indicates whether the observation belongs to this specific category/level or not. The first category/level is used as the reference and is not explicitly included in the model.

Example: for an ordinal variable X_i with three levels (1, 2 3), we create two binary variables:

- $X_{i,1}$: 1 if $X_i = 2$, 0 otherwise

- $X_{i,2}$: 1 if $X_i = 3$, 0 otherwise

Assume a continuous outcome Y that depends on the ordinal variable X with 3 levels, the CDF for Y is given by: $F(Y \mid X = 1) = F_Z(h_I(y) + x_1\beta_1 + x_2\beta_2)$

For $X = 1$, the reference level, the CDF simplifies to: $F(Y \mid X = 1) = F_Z(h_I(y))$

For $X = 2$, the CDF becomes: $F(Y \mid X = 1) = F_Z(h_I(y) + \beta_1)$

For $X = 3$, the CDF becomes: $F(Y \mid X = 1) = F_Z(h_I(y) + \beta_2)$

The coefficients β_1 and β_2 can be interpreted as the additive shift in the latent scale $h_I(y)$ when moving from the reference level (1) to levels 2 and 3, respectively.

6.4 Scaling of continuous variables

Neural networks work best when the input variables are standardized. A linear, monotonic and invertible transformation of a predictor variable changes the interpretation of the coefficient. Scaling a predictor variable X as $X_{\text{std}} = (X - mean(X))/sd(X)$ will imply that the coefficient $\tilde{\beta}$ is interpreted as the change in log-odds for a one standard deviation increase in the predictor variable or equivalently, for a one unit increase in the standardized predictor. This is different from the interpretation of the coefficient β in the original scale, which represents the change in log-odds for a one unit increase in the predictor variable.

In contrast, the standardization of the outcome variable has no effect on the interpretation (because the scale invariance of the log-odds). Consider, we standardize the outcome Y as follows:

$$Y_{\text{std}} = \frac{Y - \mu_Y}{\sigma_Y}$$

This transformation is linear, monotonic, and invertible:

$$Y = Y_{\text{std}} \cdot \sigma_Y + \mu_Y$$

Therefore, for any threshold y, we have the equivalence:

$$P(Y < y \mid X) = P\left(Y_{\text{std}} < \frac{y - \mu_Y}{\sigma_Y} \mid X\right)$$

This means that the probability is the identical when evaluating the same quantile in the standardized outcome as in the raw outcome. Furthermore, the interpretation of coefficients in a continuous outcome logistic regression remains unchanged. In particular, the log-odds ratio:

$$\log\left(\frac{P(Y < y \mid X + 1)}{1 - P(Y < y \mid X + 1)}\right) - \log\left(\frac{P(Y < y \mid X)}{1 - P(Y < y \mid X)}\right)$$

is equal to:

$$\log\left(\frac{P\left(Y_{\text{std}} < \frac{y-\mu_Y}{\sigma_Y} \mid X + 1\right)}{1 - P\left(Y_{\text{std}} < \frac{y-\mu_Y}{\sigma_Y} \mid X + 1\right)}\right) - \log\left(\frac{P\left(Y_{\text{std}} < \frac{y-\mu_Y}{\sigma_Y} \mid X\right)}{1 - P\left(Y_{\text{std}} < \frac{y-\mu_Y}{\sigma_Y} \mid X\right)}\right)$$

as long as the same quantile (i.e. probability threshold) is used. Thus, the coefficient β reflects the same change in log-odds for a one-unit increase in the (standardized) predictor, regardless if the outcome is standardized or not. This property is also crucial for the evaluation of the bernstein polynomial, since the outcome has to be scaled on a range between 0 and 1.

The general formula of the transformation model is

$$P(Y < y \mid X = x) = F_z\left(h(Y) + \beta \cdot X\right)$$

but the model is fitted with standardized outcome and predictors

$$P(Y_{\text{std}} < y_{\text{std}} \mid X_{\text{std}} = x_{\text{std}}) = F_z\left(\tilde{h}(Y_{\text{std}}) + \tilde{\beta} \cdot X_{\text{std}}\right)$$

where $\tilde{h}$ and $\tilde{\beta}$ represent the estimated transformation function and coefficients after standardizing the outcome and predictors.

For example, if we want to know the probability $P(Y < 20 \mid X = 3)$ with standardized variables, the model is specified as

$$P\left(\frac{Y - \mu_Y}{\sigma_Y} < \frac{20 - \mu_Y}{\sigma_Y} \,\middle|\, X_{\text{std}} = \frac{3 - \mu_X}{\sigma_X}\right) = F_z\left(\tilde{h}\left(\frac{20 - \mu_Y}{\sigma_Y}\right) + \tilde{\beta} \cdot \frac{3 - \mu_X}{\sigma_X}\right)$$

6.5 Bernstein Polynomial for Continuous Outcomes

In deep TRAMs the intercept for continuous variables is a smooth monotonically increasing function that is represented by a Bernstein polynomial of order K (here the complex intercept case where the Intercept already depends on the predictors x, however, the same principle that follows also applies for the simple intercept case):

$$h_I(y \mid \mathbf{x}) = \sum_{k=0}^{K} b_k(\mathbf{x}) \cdot B_k(s(y)) \tag{6.14}$$

where $B_k(s(y))$ is the Bernstein basis polynomial of order K evaluated at the scaled outcome $s(y)$:

To guarantee that the transformation $h_I(y \mid \mathbf{x})$ is monotonically increasing in y, the coefficients $b_k(\mathbf{x})$ must form a non-decreasing sequence. This is ensured via a *cumulative softmax* parameterization. Instead of learning $b_k(\mathbf{x})$ directly as the outputs of the intercept neural network, we first define unbounded parameters $\theta_k(\mathbf{x}) \in \mathbb{R}$ and then compute the Bernstein parameters using the cumulative softmax transformation:

$$\tilde{b}_k(\mathbf{x}) = \sum_{j=0}^{k} \frac{\exp(\theta_j(\mathbf{x}))}{\sum_{\ell=0}^{K} \exp(\theta_\ell(\mathbf{x}))}, \quad \text{for } k = 0, \ldots, K. \tag{6.15}$$

This transformation produces a vector $\tilde{b}_k(\mathbf{x})$ that is monotonically increasing in k, with values bounded in $[0, 1]$. It ensures that:

- $\tilde{b}_0(\mathbf{x}) \leq \tilde{b}_1(\mathbf{x}) \leq \ldots \leq \tilde{b}_K(\mathbf{x})$, - The sum of increments is 1, - The transformation is smooth and differentiable.

The combination of Bernstein polynomials with cumulative softmax-transformed parameters allows flexible, smooth, and strictly monotonic transformations of continuous outcomes, which are essential properties for distribution estimation and generative sampling within the deep TRAM architecture.

6.5.1 Scaling and Extrapolation of the Bernstein Polynomial

Because the Bernstein polynomial is only defined on the range $[0, 1]$ the outcome y has to be scaled onto the same range. Furthermore, for the sole purpose of estimating the parameters of the Bernstein polynomial it would be sufficient to finish here. However, one has to be able to also evaluate $h(y \mid \mathbf{x})$ for arbitrary values of y, in particular also the ones that are outside of $(\min(y_{train}), \max(y_{train})))$. This is also crucial for sampling. Therefore we extend the Bernstein polynomial by linearly extrapolating the tails of the polynomial. We do this by constructing inside the 5% and 95% quantiles of y by the smooth Bernstein polynomial 6.14 and linearly extrapolating the outside this range using the slope of the polynomial at the boundaries. This

results in a piecewise-defined function that is differentiable, monotonic, and defined for all real values of y, which is essential for evaluating the model at arbitrary outcomes or for generative sampling.

Tho formalize this, let $q_{0.05}$ and $q_{0.95}$ denote the 5% and 95% empirical quantiles of the outcome y, computed on the training data. The scaled outcome is defined as

$$s(y) = \frac{y - q_{0.05}}{q_{0.95} - q_{0.05}}. \tag{6.16}$$

This scaling maps the interval $[q_{0.05}, q_{0.95}]$ to the unit interval $[0, 1]$, which is the domain of the Bernstein basis polynomials. Let $h_I(s(y) \mid \mathbf{x})$ be the original transformation as defined in Equation (6.14). The extrapolated transformation $h^*(y \mid \mathbf{x})$ is then defined as

$$h^*(y \mid \mathbf{x}) = \begin{cases} h_I(0 \mid \mathbf{x}) + h_I'(0 \mid \mathbf{x}) \cdot (s(y) - 0), & \text{if } s(y) < 0 \\ h_I(s(y) \mid \mathbf{x}), & \text{if } 0 \leq s(y) \leq 1 \\ h_I(1 \mid \mathbf{x}) + h_I'(1 \mid \mathbf{x}) \cdot (s(y) - 1), & \text{if } s(y) > 1 \end{cases} \tag{6.17}$$

The function is thus extrapolated beyond the central range using the tangent line at the boundaries. The derivatives $h_I'(0 \mid \mathbf{x})$ and $h_I'(1 \mid \mathbf{x})$ are computed analytically from the Bernstein basis and the learned coefficients $b_k(\mathbf{x})$, and ensure continuous differentiability across the domain (see next subsection).

This construction ensures several desirable mathematical properties. First, the transformation $\tilde{h}(y \mid \mathbf{x})$ is globally defined on $\mathbb{R}$, avoiding undefined regions or discontinuities. Second, it preserves monotonicity due to the use of the cumulative softmax parameterization of the coefficients $b_k(\mathbf{x})$, which guarantees that the Bernstein polynomial is strictly increasing. Finally, the piecewise-linear extrapolation ensures the function is continuously differentiable and smooth at the junctions $s(y) = 0$ and $s(y) = 1$.

6.5.2 Analytical Derivative of the Bernstein Polynomial Transformation

To efficiently compute the gradient of the transformation $h_I(y \mid \mathbf{x})$ with respect to its inputs, we can exploit the analytical structure of the Bernstein basis polynomials. Recall the general form of the transformation:

$$h_I(y \mid \mathbf{x}) = \sum_{k=0}^{K} b_k(\mathbf{x}) \cdot B_k(s(y)), \tag{6.18}$$

where $B_k(s)$ are the Bernstein basis polynomials of order K, and $b_k(\mathbf{x})$ are predictor-dependent coefficients. For fixed $\mathbf{x}$, the derivative with respect to y is needed, for example, to evaluate the density function when h_I is used in a generative model.

Let us denote $s = s(y)$. Using the chain rule, we compute:

$$\frac{d}{dy} h_I(y \mid \mathbf{x}) = \sum_{k=0}^{K} b_k(\mathbf{x}) \cdot \frac{d}{dy} B_k(s) = \sum_{k=0}^{K} b_k(\mathbf{x}) \cdot \frac{dB_k(s)}{ds} \cdot \frac{ds}{dy}. \tag{6.19}$$

The derivative of the scaled outcome $s(y) = \frac{y - q_{0.05}}{q_{0.95} - q_{0.05}}$ is simply

$$\frac{ds}{dy} = \frac{1}{q_{0.95} - q_{0.05}}. \tag{6.20}$$

The derivative of the Bernstein basis polynomial $B_{k,K}(s)$ is known and given by:

$$\frac{d}{ds} B_{k,K}(s) = K \left[B_{k-1,K-1}(s) - B_{k,K-1}(s) \right]. \tag{6.21}$$

Therefore, the full derivative is:

$$\frac{d}{dy} h_I(y \mid \mathbf{x}) = \frac{K}{q_{0.95} - q_{0.05}} \sum_{k=0}^{K} b_k(\mathbf{x}) \left[B_{k-1,K-1}(s) - B_{k,K-1}(s) \right]. \tag{6.22}$$

This expression can be evaluated efficiently and is used both in the likelihood computation (e.g., via change-of-variables) and for constructing tail extrapolations with matching slopes.