Functional Modeling with Neural Causal Models and Personalized Treatment Effect Estimation

Master Thesis in Biostatistics (STA495)

by

Mike Krähenbühl

Matriculation number: 18-652-149

supervised by

Prof. Dr. Beate Sick, University of Zurich & ZHAW

Prof. Dr. Oliver Dürr, HTWG Konstanz

Zurich, July 2025

Functional Modeling with Neural Causal Models and Personalized Treatment Effect Estimation

Mike Krähenbühl

Version July 14, 2025

Contents

Preface

This thesis marks the final part of my Master of Science in Biostatistics at the University of Zurich. I wanted to work on a topic where I could apply my interest and deepen my knowledge in machine learning, especially in relation to causal questions.

The TRAM-DAG framework (Sick and Dürr, 2025), developed by my supervisors Prof. Dr. Beate Sick and Prof. Dr. Oliver Dürr, provided a perfect opportunity to do so. Our initial aim was to apply it to real-world data and potentially include semi-structured data. However, due to some surprising findings by Chen *et al.* (2025), our focus shifted towards the increasingly important topic of individualized treatment effect (ITE) estimation. Towards the end, we then bridged ITE estimation with the TRAM-DAG framework.

I want to thank my supervisors and all the people I had the chance to work and study with, as well as everyone who supported me on this journey.

Mike Krähenbühl
July 2025

Abstract

This thesis investigates the use of TRAM-DAGs (Sick and Dürr, 2025) as a flexible approach for estimating structural equations in known directed acyclic graphs (DAGs). TRAM-DAGs offer several advantages: the model inherently knows when to control for covariates based on the DAG, is highly customizable in terms of flexibility and interpretability, and allows sampling from observational, interventional, and counterfactual distributions. We show how to incorporate ordinal predictors, model interactions, and examine how variable scaling affects interpretability.

A main focus was the estimation of individualized treatment effects (ITEs) using a variety of causal machine learning (ML) models. In simulation studies, we analyzed limitations in ITE estimation and found that unmeasured effect modifiers can severely impact estimation accuracy, and that the ignorability assumption alone may not ensure unbiased results – a concern that has also been noted in prior research (Vegetabile, 2021). The limitations found may also help explain the poor ITE estimation performance observed in the real-world application on the International Stroke Trial dataset (Chen *et al.*, 2025). We further demonstrated that TRAM-DAGs can be used for ITE estimation in relatively complex DAG structures, provided that the DAG is fully known and all variables are observed.

While promising, TRAM-DAGs require training time due to their reliance on neural networks and, when aiming for interpretability, certain assumptions about the model structure. Future work could explore applications to more real-world data, potentially including semi-structured inputs, and further investigate ITE estimation in the presence of unmeasured interactions.

This thesis contributes to the field of causal inference under observational data and to the estimation of personalized treatment effects using causal ML models.

Keywords: TRAM-DAGs, neural causal model, individualized treatment effect, structural causal model, counterfactuals, transformation model, observational data, heterogeneous treatment effect, conditional average treatment effect

Chapter 1

Introduction

1.1 Motivation

The most important questions in research are mostly not associational, but causal (Pearl, 2009a). They concern the effects of interventions – such as the impact of a treatment – or seek explanations for observed outcomes, such as identifying which disease caused certain symptoms. They also include hypothetical scenarios; for example: what would the GDP have been if interest rates had increased by 25 instead of 75 basis points? Answering such questions requires causal reasoning and demands an understanding of the underlying data-generating process. Purely associational approaches are typically not sufficient to draw valid causal conclusions.

The gold standard for estimating the causal effect of an intervention on an outcome is the randomized controlled trial (RCT) (Hariton and Locascio, 2018). In this prospective study design, participants are randomly assigned to either the treatment or control group. Randomization aims to eliminate the influence of potential confounding variables, ensuring that treatment groups are balanced with respect to baseline characteristics. This allows for an unbiased estimation of the causal effect. Despite their strengths, RCTs have several limitations. They are often expensive and time-consuming to plan and execute. Moreover, the results may not generalize well to the population of interest, as individuals who volunteer or are accepted for trials are not always representative of the target group. Another central limitation of RCTs is that in many scenarios they simply cannot 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, such as 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.

For these reasons, much of research aims to make causal inference from observational data, using non-experimental or quasi-experimental designs. Unlike RCTs, these settings do not involve randomization to treatment, which introduces challenges due to confounding. Methods for causal inference from observational data aim to correctly control for such confounders to enable valid causal conclusions. Recently, Sick and Dürr (2025) proposed the TRAM-DAGs framework, which estimates the functional form of causal relationships in a known causal graph based on observational or RCT data, and can then be used to make subsequent causal queries. In this thesis, we further analyze and apply this method.

An application where causal inference is of particular importance is the estimation of personalized treatment effects. RCTs typically estimate an average treatment effect (ATE) on a sample, which is the difference in mean outcomes between the treatment and the control groups (Nichols, 2007). However, individual patients may respond differently to the treatment, depending on their unique characteristics. In personalized medicine, the estimate of treatment effects at the individual level is referred to as the individualized treatment effect (ITE) or conditional average treatment effect (CATE), while in business and marketing contexts, the term uplift modeling is often used (Gutierrez and Gérardy, 2017; Zhao and Harinen, 2020). These concepts refer to the difference in potential outcomes under different treatments, assessed at the level of individuals or subgroups. Such estimates are critical in settings where treatment responses vary significantly between individuals. For clinical decision-making, tailoring therapies to individual characteristics can lead to more effective and efficient care. The importance of estimating individual-level effects is not limited to medicine. It is also of high interest in marketing, where campaigns can be precisely targeted to maximize impact and minimize adverse responses. Consider, for instance, the decision of whether to send a push notification (treatment) to a customer. Some customers might be persuadables, who will respond positively only if treated. Others, in contrast, might have responded positively without the intervention but are negatively affected by receiving it – for example, a customer who is reminded of a forgotten subscription and, as a result, decides to cancel it. In this context, identifying persuadables is valuable, while treating the latter may be counterproductive. This illustrates the need to understand treatment effects at a granular level to guide individualized decisions. Various methods have been proposed to estimate individualized treatment effects, yet this task remains challenging. The fundamental problem is that only one of the two potential outcomes can ever be observed for any given individual (Holland, 1986), making the estimation of treatment effects inherently more difficult than standard predictive modeling.

1.2 Key concepts of causal inference

Causal relationships can be represented by a directed acyclic graph (DAG), as shown in Figure 1.1(a). The variables, or nodes, are connected by directed edges, which represent causal dependencies.

Questions in causal inference are typically classified into one of the three levels of Pearl's hierarchy of causation (Pearl, 2009b). Level 1 corresponds to observational queries, expressed as conditional probabilities $\mathrm{P}(Y \mid X)$, which can be answered directly from the joint distribution $\mathrm{P}(Y \cap X)$. Level 2 involves interventional queries, such as $\mathrm{P}(Y \mid \mathrm{do}(X = \alpha))$, which describe the probability when actively setting a variable X to a particular value. Unlike observational queries, answering interventional questions requires knowledge of the underlying causal structure. Level 3 addresses counterfactual reasoning, which poses the greatest challenge. These are hypothetical what-if questions that require reasoning about outcomes under alternative realities. For example, if a patient received a treatment and died, the factual outcome is death under the received treatment. The counterfactual would be the outcome that would have occurred had the patient received a different treatment.

Some statisticians argue that counterfactuals – being unobservable and untestable – are of limited scientific value and may be regarded as metaphysical (Dawid, 2000). Nevertheless, there are important practical questions that require the analysis of such counterfactuals.

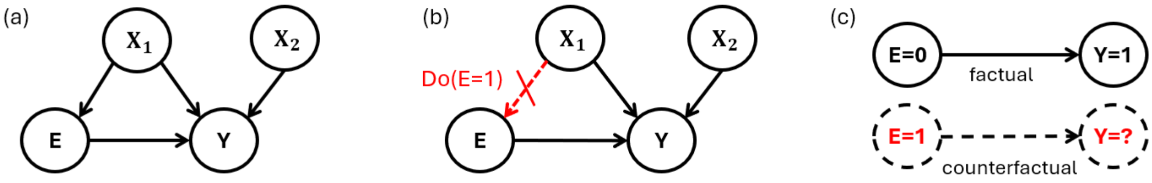


Figure 1.1: Illustration of the three levels of Pearl's hierarchy of causation. (a) Directed acyclic graph (DAG) for observational data. (b) DAG when making a do-intervention by fixing the variable E at a certain value. (c) Observed factual outcome and the corresponding counterfactual query.

To illustrate Pearl's three levels of causality, we 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?

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

This can be estimated directly from data by conditioning on $E = 1$ and computing the frequency of $Y = 1$. However, such an estimate does not account for confounding variables like age.

Level 2: Interventional (”doing”): We consider the effect of actively intervening in the system. Example: What is the probability of heart disease if everyone were made to exercise, regardless of age or smoking status?

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

Answering this requires assumptions about the underlying causal structure.

Level 3: Counterfactual (”imagining”): We ask what would have happened under different circumstances – that is, we imagine an alternative scenario for the same individual. Example: For a person who does not exercise and has heart disease, would they still have had heart disease if they had exercised?

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

Here, $Y_{E=1}$ represents the counterfactual outcome under positive exposure. Counterfactual queries cannot be answered from observational data alone; they require a structural framework that explicitly models the data-generating process.

For this, the concept of DAGs can be extended to structural causal models (SCMs). A set of structural equations of the form $X_i = f_i(\mathrm{pa}(X_i), Z_i), \quad i = 1, \ldots, n$ builds a structural causal model (Pearl, 2009b). $\mathrm{pa}(X_i)$ denotes the direct causal parents of X_i, and Z_i is an exogenous noise variable. These exogenous variables capture latent factors that influence X_i but are not explicitly modeled. By convention, the Z_i are assumed to be mutually independent. This assumption implies that the DAG includes all relevant causal relationships among the variables. Any dependence between variables must be due to a direct or indirect path in the graph; otherwise, dependencies would show up in the noise terms, violating their independence (Pearl, 2009b).

Each function f_i – which may be nonlinear – defines how the value of X_i is generated from its parents and the corresponding noise term. A source node X_j without any parents is modeled as $X_j = f_j(Z_j)$. Once all structural equations and noise variables are specified, the model is fully deterministic in the sense that each variable is a fixed function of its parents and its own exogenous noise. The randomness in the system arises entirely from these independent noise terms, which encode unobserved factors. This functional representation makes it possible to compute interventional distributions and evaluate counterfactual outcomes. These aspects are discussed in detail in Section 2.1.4.

In this thesis, we do not focus on discovering the underlying causal graph. Such a structure may be obtained through structure learning algorithms (see e.g., Zheng *et al.*, 2018) or determined from expert knowledge. Instead, we assume the graph is known and concentrate on estimating the functional form of the relationships between variables – that is, the structural equations that define the SCM.

Various approaches exist for estimating the functions f_i that constitute an SCM, depending on the assumptions made about the data and the model class. These methods are discussed in the next section.

A simple approach to modeling the structural equations is linear regression, which assumes Gaussian error terms Z_i and linear functional forms f_i. Classical statistical methods of this kind are typically well-defined, computationally efficient, and offer interpretable parameters. However, they rely on strong assumptions about the underlying data-generating mechanism – such as linearity and homoscedasticity – which may not hold in practice. Violations of these assumptions can lead to biased or misleading results.

Alternatively, more flexible approaches based on neural networks have gained popularity for estimating structural equations. These models are capable of approximating complex, nonlinear relationships and capturing complicated interactions between variables with minimal bias. Their flexibility, however, often comes at the cost of reduced interpretability and, in some cases, limited applicability to non-continuous or mixed data types. Poinsot *et al.* (2024) provided an overview of deep structural causal models and their use in counterfactual inference.

The TRAM-DAG framework proposed by Sick and Dürr (2025) builds a bridge between these classical and neural-network-based modeling approaches, by combining interpretable transformation models with the flexibility of neural networks. At its core, the structural equations are modeled using transformation models (Hothorn *et al.*, 2014), a flexible class of distributional regression methods. These models were subsequently extended to deep transformation models (Deep TRAMs) by Sick *et al.* (2021), enabling the use of neural networks to parameterize conditional distributions in a customizable way. In the TRAM-DAG framework, these deep TRAMs are applied according to a known causal graph, allowing the model to be fitted to observational data and used to answer causal queries across all three levels of Pearl's hierarchy. The framework is introduced in more detail in Section 2.1.3.

1.3 Goals and contributions

This thesis contributes to the further exploration of the TRAM-DAG framework and to adressing challenges in the estimation of personalized treatment effects.

The first part of this thesis focuses on a systematic analysis and extension of TRAM-DAGs. This includes applying the model across a variety of settings, such as different data types, model complexities, and neural network configurations (e.g., activation functions, batch normalization, dropout). Most analyses are conducted on simulated data to know the underlying data-generating process, but the model is also applied to real-world data to demonstrate its practical utility.

The second focus of the thesis is on the estimation of individualized treatment effects (ITEs). Recent work by Chen *et al.* (2025) showed that most causal machine learning models trained on RCT data failed to generalize when evaluated out of sample. In this thesis, we replicate some of their work by applying various models, including TRAM-DAGs, to the same data and analyzing whether we come to a similar conclusion. We further investigate why ITE estimation can fail in such settings, and under which conditions reliable estimates can be obtained. In addition, we demonstrate that TRAM-DAGs can be effectively used to estimate ITEs also in non-randomized observational settings, provided that the causal graph is known and fully observed. In doing so, we explore the potential of TRAM-DAGs as a framework for answering complex causal questions across different levels of Pearl's hierarchy.

Formally, we aim to answer following research questions in this thesis:

- How can TRAM-DAGs be applied under different scenarios such as ordinal predictors, scaled vs. raw variables or allowing for interactions between variables?

- Do we obtain similar results when estimating ITEs on a real-world RCT dataset, as reported by Chen *et al.* (2025)?

- What are possible reasons for the failure of ITE estimation in some cases when causal machine learning models are validated out of sample?
- How can TRAM-DAGs be used to estimate ITEs in both randomized controlled trials and observational settings involving confounding and mediating variables?

With this work, we aim to contribute to the important and evolving field of causal inference in observational settings and to the challenging task of estimating individualized treatment effects.

Chapter 2

Methods

This chapter introduces the methodological foundations and experimental designs used in this thesis. Section 2.1 presents the concept and functionality of TRAM-DAGs, along with the necessary theoretical background. Section 2.2 provides the framework for estimating individualized treatment effects. Finally, Sections 2.3–2.6 describe the four experiments conducted to address the research questions outlined in Section 1.3.

2.1 TRAM-DAGs

The goal of TRAM-DAGs is to estimate the structural equations of a given DAG in a flexible and, if desired, interpretable way. This enables the sampling of observational and interventional distributions, as well as to make counterfactual statements. The approach requires data and a known causal graph describing the underlying structure. It is assumed that there are no hidden confounders. TRAM-DAGs estimate, for each variable X_i, a transformation function $Z_i = h_i(X_i \mid \mathrm{pa}(X_i))$, where Z_i denotes the noise variable and $\mathrm{pa}(X_i)$ are the causal parents of X_i. Crucially, this relationship can be inverted to yield the structural equation $X_i = h_i^{-1}(Z_i \mid \mathrm{pa}(X_i))$. The monotonically increasing transformation functions h_i represent the conditional distribution $X_i \mid \mathrm{pa}(X_i)$ on a latent scale Z_i. They are based on the framework of transformation models as introduced by Hothorn *et al.* (2014) and were extended to deep TRAMs by Sick *et al.* (2021). The following summarize the key ideas of these models, which form the building blocks of TRAM-DAGs.

2.1.1 Transformation Models

Transformation models are a flexible class of distributional regression models applicable to various data types. They can be regarded as generalizations of standard models such as linear regression, logistic regression, or proportional odds models, while also allowing for the modeling of outcome distributions that do not belong to a known parametric family. This is achieved by modeling components of the transformation function in a flexible, semi-parametric way, thereby reducing the strength of assumptions required about the underlying data-generating process. 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 $\mathbf{x}$. The transformation function $h(y \mid \mathbf{x})$ maps the outcome variable y onto the latent scale of variable Z, and F_Z is the cumulative distribution function (CDF) of Z, the so-called inverse-link function that maps $h(y \mid \mathbf{x})$ to probabilities. In the basic version shown in Equation 2.1, the transformation function can be split into an intercept part $h_I(y)$ and a linear shift component $\mathbf{x}^\top \boldsymbol{\beta}$, where $\mathbf{x}$ are the predictors and $\boldsymbol{\beta}$ the corresponding coefficients.

If the latent distribution Z is chosen to be the standard logistic, each coefficient β_i can be interpreted as log-odds ratio: an increase of one unit in the predictor x_i, while holding other predictors unchanged, increases the log-odds (latent scale) of the outcome Y by β_i. After applying the inverse link function F_Z, this can induce a non-linear change to the conditional distribution of Y on the original scale. Further discussion on the choice of latent distribution and the interpretation of coefficients is provided in in Appendix 6.1.

For a continuous outcome Y, the intercept $h_I(y)$ 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 $\vartheta_k, k = 0, \ldots, M$ are the coefficients and $\mathrm{B}_{k,M}(y)$ are the corresponding Bernstein basis polynomials. The coefficients ϑ_k are constrained to be monotonically increasing to ensure that the transformation function $h_I(y)$ is strictly increasing. More details on the technical implementation of Bernstein polynomial in the context of deep-TRAMs are provided in the Appendix 6.2

For a discrete outcome Y, the intercept h_I is represented by a set of cut-points ϑ_k, which define the thresholds separating the different levels of the outcome. For example, a binary outcome requires one cut-point, while an ordinal outcome with K levels requires $K-1$ 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 shown in Figure 2.1.

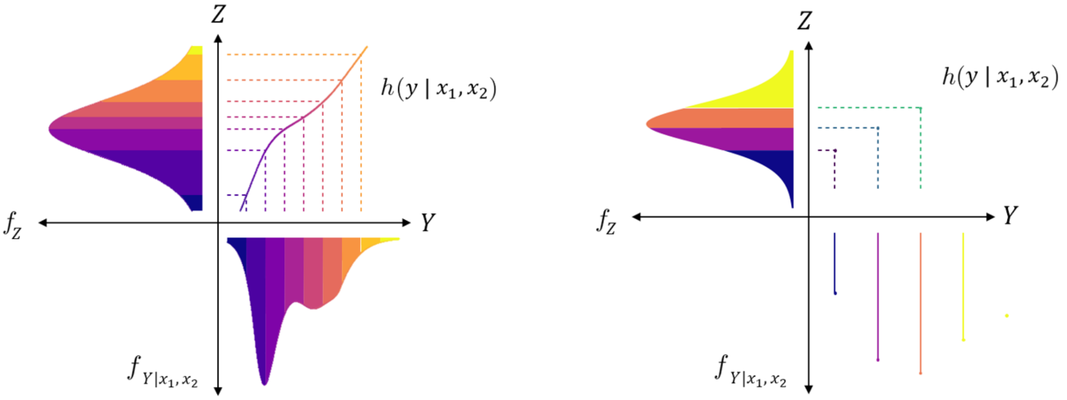


Figure 2.1: Left: Example of a transformation model for a continuous outcome Y with a smooth transformation function. Right: Example of a transformation model for an ordinal outcome Y with 4 levels. The transformation function consists of cut-points separating the levels of the outcome. In both cases the latent distribution Z is the standard logistic and the predictors $\mathbf{x}$ induce a linear (vertical) shift of the transformation function.

The parameters $\boldsymbol{\beta}$ and $\boldsymbol{\vartheta}$ are estimated by minimizing the negative log-likelihood (NLL), defined as:

$$\mathrm{NLL} = -\frac{1}{n} \sum_{i=1}^{n} \log \left(f_{Y|\mathbf{X}=\mathbf{x}}(y_i) \right), \tag{2.4}$$

where $f_{Y|\mathbf{X}=\mathbf{x}}(y_i)$ is the conditional density function of the outcome Y given the predictors $\mathbf{x}$ of the i-th observation under the current parameterization. A full derivation is provided in

Appendix 6.3.

Throughout this thesis, these transformation models form the foundation for estimating conditional distributions of variables within the TRAM-DAG framework. Unless stated otherwise, the latent distribution F_Z is chosen to be the standard logistic, resulting in a logistic transformation model.

2.1.2 Deep TRAMs

The transformation models introduced earlier were extended into deep TRAMs using a modular neural network architecture (Sick *et al.*, 2021). The goal is to obtain a parameterized transformation function of the form

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

where $h_I(y)$ is the intercept (simple or complex), $\mathbf{x}_L$ are predictors with a linear effect on the transformation function, and $\mathbf{x}_C$ are those with a potentially complex, non-linear influence. The parameters for each component of the model – intercept, linear shift, and complex shift – are obtained by a neural network module. The user can assign predictors to these components depending on the assumed structure of the data and the desired flexibility. Figure 2.2 illustrates an example of a transformation function with these tree components (SI-LS-CS). For simplicity, it shows the simple intercept (SI) case where the intercept does not depend on predictors.

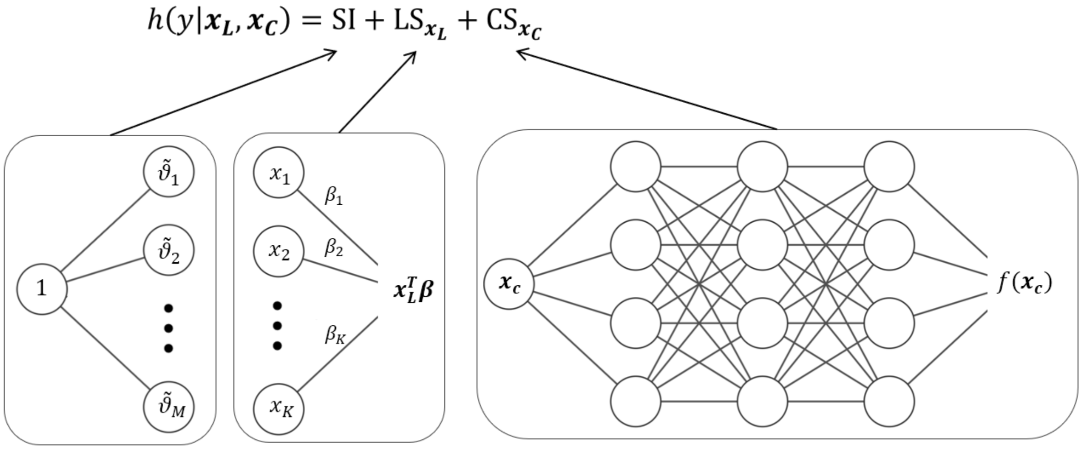

$h(y|\boldsymbol{x_L}, \boldsymbol{x_C}) = \mathrm{SI} + \mathrm{LS}_{\boldsymbol{x_L}} + \mathrm{CS}_{\boldsymbol{x_C}}$
$\tilde{\vartheta}_1$
$\tilde{\vartheta}_2$
$\tilde{\vartheta}_M$
1
x_1
x_2
x_K
β_1
β_2
β_K
$\boldsymbol{x}_L^T\boldsymbol{\beta}$
$\boldsymbol{x}_c$
$f(\boldsymbol{x}_c)$

Figure 2.2: Modular deep transformation model (deep TRAM). The transformation function $h(y \mid \mathbf{x})$ is constructed from the outputs of three neural networks: intercept (SI/CI), linear shift (LS), and complex shift (CS). The intercept module (left) shows the simple intercept (SI) case and could be extended to a complex intercept (CI) by feeding predictors into the module and adding hidden layers and nonlinear activation functions.

Intercept: The intercept defines the baseline shape of the transformation function when $\mathbf{x}_L^\top \boldsymbol{\beta} = 0$ and $f(\mathbf{x}_C) = 0$. For continuous outcomes, it is modeled using a smooth Bernstein polynomial (Equation 2.2), and for discrete outcomes via cut-points. The preliminary parameters $\tilde{\vartheta}_k$ are the output nodes of a neural network, and subsequently transformed to be monotonically increasing ϑ_k. A simple intercept (SI) assumes no dependency on predictors and uses only a constant input. A complex intercept (CI), by contrast, allows the parameters to vary as a function of selected predictors by feeding $\mathbf{x}$ into a neural network with hidden layers, producing predictor dependent parameters $\vartheta_k(\mathbf{x})$. This allows the baseline distribution to adapt flexibly to different covariate settings. Details on the computation of the Bernstein polynomial are provided in Appendix 6.2.

Linear shift: Predictors with an assumed linear influence on the transformation funciton are modeled via the linear shift (LS) component. Here, a neural network without hidden layers and without bias nodes receives $\mathbf{x}_L$ as input and returns a linear combination $\mathbf{x}_L^\top \boldsymbol{\beta}$ as output. This results in a vertical, linear shift of the transformation function. The parameters $\boldsymbol{\beta}$ are interpretable coefficients, and in the case of a logistic transformation model, represent log-odds ratios. See Appendix 6.1 for more details.

Complex shift: Non-linear dependencies between predictors and the transformation function can be modeled by the complex shift (CS) component. The corresponding predictors $\mathbf{x}_C$ are input into a deep neural network (with at least one hidden layer and non-linear activation functions), yielding a scalar output $f(\mathbf{x}_C)$. This allows modeling of non-linear predictor effects, including interactions between predictors if input multiple predictors into the neural network module. See Appendix 6.6 for an example of modeling an interaction with a complex shift for the purpose of estimating individualized treatment effects (ITEs).

Level of complexity: A key advantage of the deep TRAM architecture is that users can specify the role of each predictor: linear (LS), complex (CS), or influencing the shape of the transformation function (CI). For example, Herzog *et al.* (2023) predicted the ordinal functional outcome three months after stroke by combining structured tabular data with image features (via a CNN) as predictors. This flexible design allows the integration of different data modalities into a single model.

The resulting distribution function of deep TRAMs is invariant to the choice of the inverse-link function F_Z (latent distribution) in an unconditional (Hothorn *et al.*, 2018) or fully flexible (CI) setting. However, once restrictions are introduced on the influence of the predictors – such as LS or CS – the model assumes a fixed scale of dependence. The choice of latent distribution may depend on (i) assumptions about the data-generating process, (ii) the conventional and widely used interpretation scale for parameters (e.g., log-odds ratios for discrete outcomes or log-hazard ratios in survival analysis), and (iii) a data-driven criterion such as selecting the distribution that minimizes the negative log-likelihood (NLL) on the observed data.

Parameter estimation: The weights of the neural networks are learned by minimizing the NLL of the conditional model. Starting from random weight initialization, the networks are trained using the Adam optimizer (Kingma and Ba, 2015). Outputs of the modules are assembled to compute the NLL, and backpropagation (Rumelhart *et al.*, 1986) is used to adjust weights iteratively. Deep learning techniques such as dropout (Srivastava *et al.*, 2014), early stopping (Prechelt, 2012), and batch normalization (Ioffe and Szegedy, 2015) can be applied to prevent overfitting and improve generalization. Nonlinear activation functions (e.g., ReLU Glorot *et al.*, 2011 or sigmoid Rumelhart *et al.*, 1986) are used in hidden layers to capture complex relationships. Schmidhuber (2015) provides an extensive historical overview of deep learning methods and their development.

2.1.3 TRAM-DAGs: Deep TRAMS applied in a causal setting

In TRAM-DAGs, deep transformation models are applied within a causal setting. We assume a pre-specified DAG that represents the causal dependencies among variables. Figure 2.3 illustrates the basic idea using a simple DAG with three variables and no hidden confounders. Each node's conditional distribution is modeled using a deep TRAM given its parent variables in the DAG. The assumed influence of the parent variables must be specified as CI, LS or CS. In this example, X_1 is a continuous source node (i.e., without parents) and its transformation function consists only of a simple intercept (SI). X_2 is also continuous and depends on X_1 by a linear shift (LS). X_3 is an ordinal variable with four levels; its transformation function is influenced linearly by X_1 (LS) and non-linearly by X_2 (CS). The cut-points $h(x_{3,k} \mid x_1, x_2)$ represent the cumulative log-odds of levels $k = 1, 2, 3$ of X_3. The probability of the highest category ($k = 4$) is the complement of the cumulative probability of the first three.

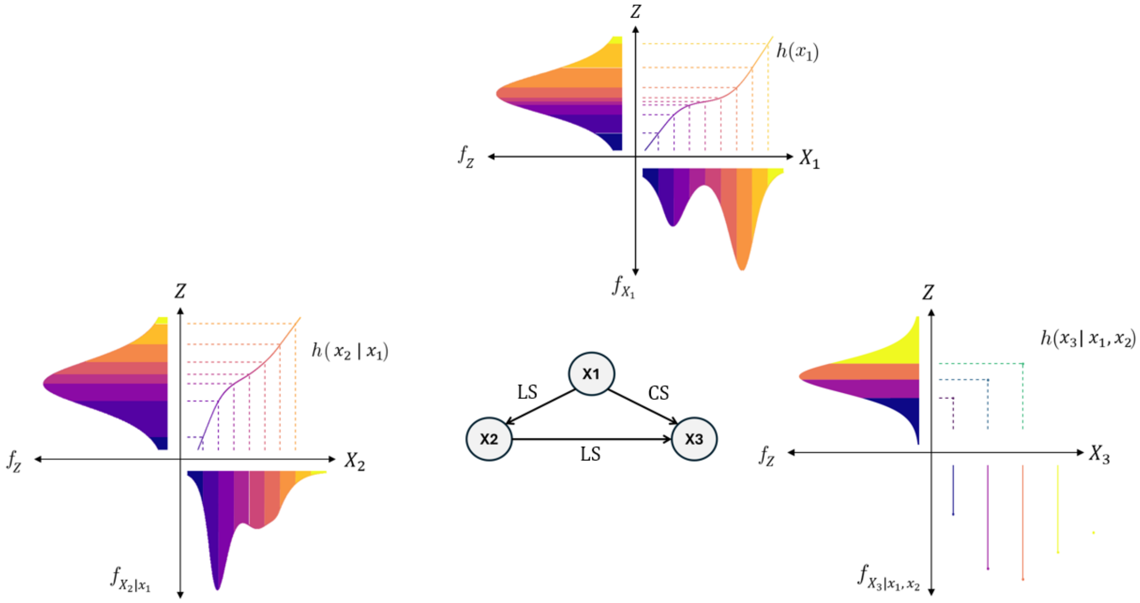


Figure 2.3: Example of a TRAM-DAG with three variables: X_1, X_2, and X_3. Each variable's distribution is modeled conditionally on its parents using deep TRAMs. Arrows indicate the causal dependencies.

This DAG, along with its assumed dependencies, can be represented by an adjacency matrix (Equation 2.6), where rows indicate sources and columns indicate targets of causal influence:

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

Based on this DAG, each variable's conditional distribution is modeled using a deep TRAM, enabling generative sampling and causal inference. The corresponding conditional distributions are:

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

Constructing the modular neural network: As described in Section 2.1.2, the transformation functions are constructed using a modular neural network. The inputs into the framework include the variables and the adjacency matrix 2.6, which governs the information flow and ensures that connections match to the specified causal structure. Discrete variables with few categories are dummy encoded, and continuous variables should be scaled prior to input. Further discussion of encoding and scaling is provided in Appendix 6.4 and 6.5.

Once the data are preprocessed and the structure is specified, the architecture of the neural networks for the complex shift or complex intercept must be defined – including choices such as depth, width, activation functions, and whether to use dropout or batch normalization. These decisions depend on the assumed complexity of the effects and the need to regularize against overfitting.

Each node's transformation function is assembled from the outputs of its three modular components: the intercept (SI or CI), linear shift (LS), and complex shift (CS). Model training is performed by minimizing the NLL, optimizing all parameters at the same time. The estimated parameters $\boldsymbol{\beta}$ in the linear shift are interpretable as log-odds ratios corresponding to a one-unit increase in the respective parent variable, holding all other parent variables unchanged.

2.1.4 Sampling from TRAM-DAGs

Once a TRAM-DAG is fitted on data, it can be used to sample from the observational or interventional distribution, or to perform counterfactual queries. The structural equations $X_i = f(Z_i, \mathrm{pa}(X_i))$ are represented by the inverse of the conditional transformation functions, i.e., $X_i = h^{-1}(Z_i \mid \mathrm{pa}(X_i))$, since the transformation function maps from observed values to the latent scale: $Z_i = h(X_i \mid \mathrm{pa}(X_i))$.

Observational sampling: The sampling process from the observational distribution is described in Algorithm 1 and illustrated in Figure 2.4. In each iteration, one complete sample of all variables in the DAG is generated – i.e. a sample from the joint observational distribution.

Algorithm 1 Generate a complete sample from the observational distribution

```
Given: A fitted TRAM-DAG with structural equations $X_i = h^{-1}(Z_i \mid \mathrm{pa}(X_i))$
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
        Solve $h(x_i \mid \mathrm{pa}(x_i)) - z_i = 0$ for $x_i$         ▷ numerical root-finding
    else if $X_i$ is discrete then
        Find category $x_i = \max\left(\{0\} \cup \{x : z_i > h(x \mid \mathrm{pa}(x_i))\}\right) + 1$
    end if
end for
```

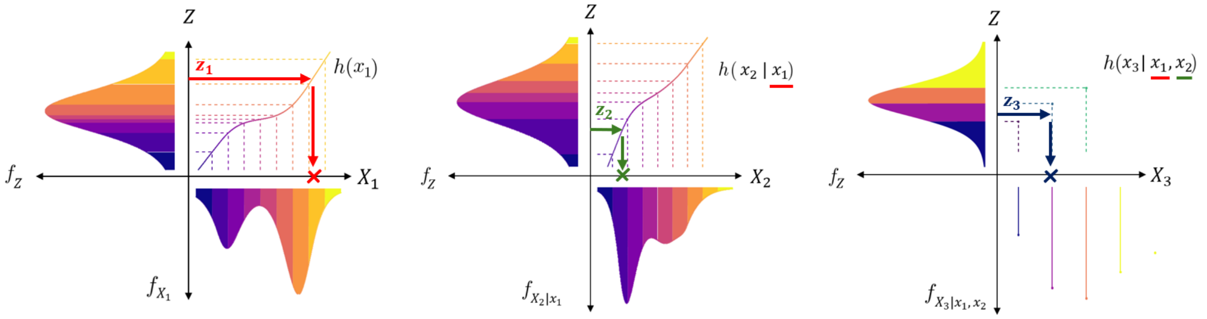


Figure 2.4: One sampling iteration for the three variables from the estimated transformation functions $h(x_i \mid \mathrm{pa}(x_i))$. The latent values z_i are sampled from the standard logistic distribution. The values x_i are determined by applying the inverse of the transformation function for continuous variables or by finding the corresponding category for the ordinal variable. See Algorithm 1 for details.

Interventional sampling: To sample from an interventional distribution, we apply the *do*-operator as described by Pearl (1995). The *do*-operator fixes a variable at a specific value, thereby removing its causal dependencies. This allows to simulate interventions by holding that variable constant while sampling the remaining variables as described earlier in Algorithm 1. For example, suppose we intervene on X_2 and set it to a specific value α, denoted as $\mathrm{do}(X_2 = \alpha)$. To sample from the resulting interventional distribution, we proceed as in observational sampling, with the only difference that X_2 is no longer sampled but fixed to the value α. X_1 would not be affected by the intervention, as it is not a descendant of X_2 in the DAG. Therefore, sampling X_3 under this intervention would follow:

$$x_3 = \max\left(\{0\} \cup \{x : z_3 > h(x \mid x_1, x_2 = \alpha)\}\right) + 1.$$

Counterfactual queries: In a counterfactual query, we are interested in what the value of a variable X_i would have been, had another variable X_j taken a different value than actually observed. Pearl (2009b) describes a three-step procedure to answer such queries. In short, let $\mathbf{x}$ denote the observed values (a sample) of all variables in a DAG and let $\mathbf{z}$ denote the corresponding latent variables inferred via the transformation functions: $z_k = h_k(x_k \mid \mathrm{pa}(x_k))$. Then, in a counterfactual query where X_j is set to a new value α (intervention), we retain the same latent values $\mathbf{z}$, but apply the transformation functions in a post-interventional DAG with $X_j := \alpha$ to compute the counterfactual outcomes for the remaining variables. The counterfactual estimation procedure is outlined in Algorithm 2 and visualized in Figure 2.5. Note that this visualization is simplified to illustrate only one parent variable (X_j); in practice, the set of parents $\mathrm{pa}(X_i)$ may include additional predictors.

Algorithm 2 Answer a single counterfactual query

Given: A TRAM-DAG model with estimated structural equations $X_k = h_k^{-1}(Z_k \mid \mathrm{pa}(X_k))$
Input: Observed sample $\mathbf{x}$, intervention $X_j := \alpha$

Step 1 (Abduction): For each observed variable x_k, compute the corresponding latent value $z_k = h_k(x_k \mid \mathrm{pa}(x_k))$

Step 2 (Action): Modify the DAG by setting $X_j := \alpha$ (intervention)

Step 3 (Prediction): Using the counterfactual DAG and the intervened value $X_j = \alpha$, go along the causal order and determine the counterfactual values for all descendants X_k of X_j. This is done by evaluating

$$x_k = h_k^{-1}(z_k \mid \mathrm{pa}^*(x_k)),$$

where z_k is the latent value obtained in Step 1, and $\mathrm{pa}^*(x_k)$ are the (possibly updated) parent values after the intervention.

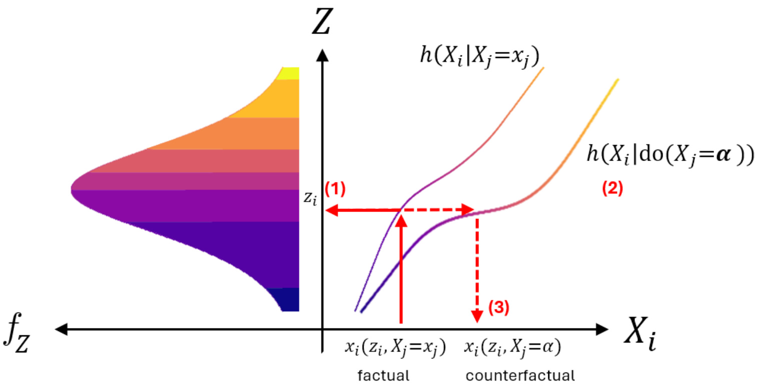


Figure 2.5: Illustration of a counterfactual query for variable X_i, had its parent X_j taken a different value, following the three-step procedure: (1) abduction of the latent value z_i from the observed outcome, (2) intervention by setting $X_j = \alpha$, (3) prediction of the counterfactual outcome using the same z_i and the modified transformation function. This simplified example assumes a single parent variable; in practice, $\mathrm{pa}(X_i)$ may include multiple predictors.

While counterfactual probabilities (e.g., $P(X_i = x \mid \mathrm{do}(X_j = \alpha), z_i)$) are well-defined in both continuous and discrete settings, determining the counterfactual realizations is generally only possible for continuous variables. In the discrete case, no unique latent value z_k can be recovered from an observed category, as multiple z_k values may map to the same outcome. Consequently, only the probabilities of counterfactual outcomes can be evaluated for discrete variables, not their exact values. This is a fundamental limitation of counterfactual reasoning in discrete settings with continuous latent variables.

2.2 Individualized Treatment Effect (ITE)

RCTs are considered the gold standard for estimating the causal effect of a treatment due to their ability to eliminate confounding through randomization (Little and Rubin, 2000). However, RCTs typically aim to estimate the average treatment effect (ATE), which summarizes the effect of the treatment across the entire study population. This approach overlooks individual-level variation in treatment response: depending on their unique characteristics, some individuals may benefit substantially, others not at all, or even be harmed. While the homogeneous treatment effect refers to the part of the effect that is equal for all patients, the heterogeneous treatment effect (HTE) describes the variation in treatment effects across individuals or subgroups. One reason for treatment effect heterogeneity is the presence of interaction variables (Hoogland *et al.*, 2021), also referred to as effect modifiers (Christensen *et al.*, 2021), which influence how the treatment effect varies depending on individual characteristics. However, even in the absence of explicit interaction variables, treatment effects can still exhibit heterogeneity due to the scale on which personalized effects are evaluated (Hoogland *et al.*, 2021). This phenomenon is further explored in Experiment 4 and will be discussed in detail in Section 4.4.

In personalized medicine and informed decision-making, individual-level treatment effects are more relevant than population averages. The individual treatment effect compares the potential outcomes for an individual i under two different treatment options: the potential outcome $Y_i(1)$ under treatment, and $Y_i(0)$ under control, as defined by Rubin's potential outcomes framework (Rubin, 2005). It is typically defined as the difference between the two potential outcomes:

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

However, only one of the two potential outcomes can be observed for each individual, which is known as the fundamental problem of causal inference (Holland, 1986). Instead, most methods aim to estimate the expected treatment effect conditional on a given set of covariates $\mathbf{X}$. This quantity is known as the conditional average treatment effect (CATE), which is valid under assumptions discussed later:

$$\begin{aligned}\mathrm{CATE}_i(\mathbf{x}_i) &= \mathbb{E}[Y_i(1) - Y_i(0) \mid \mathbf{X}_i = \mathbf{x}_i] \\ &= \mathbb{E}[Y_i \mid T = 1, \mathbf{X}_i = \mathbf{x}_i] - \mathbb{E}[Y_i \mid T = 0, \mathbf{X}_i = \mathbf{x}_i]\end{aligned} \tag{2.8}$$

It reflects the expected, or average, treatment effect for an individual i, conditional on that individual's specific covariates $\mathbf{X}_i = \mathbf{x}_i$.

In fields such as healthcare, the CATE is often referred to as the individualized treatment effect (ITE), as for example done by Hoogland *et al.* (2021). In this thesis, we use the term ITE to denote individualized treatment effects, i.e., CATE.

The ITE for an individual with covariates $\mathbf{X}_i = \mathbf{x}_i$ in the case of a binary outcome and binary treatment is defined as:

$$\mathrm{ITE}_i(\mathbf{x}_i) = P(Y_i = 1 \mid T = 1, \mathbf{X}_i = \mathbf{x}_i) - P(Y_i = 1 \mid T = 0, \mathbf{X}_i = \mathbf{x}_i). \tag{2.9}$$

Assuming $Y = 1$ indicates recovery, a positive ITE means the individual is expected to benefit from the treatment compared to control. A negative ITE suggests the outcome is expected to be worse under treatment, while an ITE of zero implies no expected difference between treatment and control for that individual.

Assumptions for Identifiability: Unlike standard predictive modeling, the estimation of the individualized treatment effect relies on untestable assumptions to ensure identifiability of causal effects from observational data. A central assumption is consistency, which requires that the observed outcome equals the potential outcome under the received treatment: $Y = Y(1)$ if $T = 1$, and $Y = Y(0)$ if $T = 0$. The Stable Unit Treatment Value Assumption (SUTVA) assumes no interference between units and that treatments are well-defined (Rubin, 1980). The most critical assumption is ignorability (or unconfoundedness), which states that, conditional on observed covariates X, the treatment assignment is independent of the potential outcomes (Rosenbaum and Rubin, 1983):

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

This implies that there are no unmeasured confounders affecting both treatment and outcome. Additionally, the positivity assumption requires that every individual has a non-zero probability of receiving each treatment level:

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

Models for ITE Estimation: A variety of methods exist to estimate ITEs. In this thesis, we focus on metalearners, specifically the T-learner and S-learner frameworks (Künzel *et al.*, 2019). These methods are model-agnostic, meaning that any predictive algorithm can be used as a base model. There exists a variety of other approaches to model ITEs. The T-learner fits two separate models: one on the treated group and on the control group. The ITE for an individual is then estimated as the difference between the predicted outcomes from these two models. In contrast, the S-learner fits a single model on all individuals, incorporating the treatment indicator as an additional covariate to distinguish between treatment conditions. Therefore, the S-learner requires a model that is complex enough, to capture interaction effects between treatment and covariates. The experiments in this thesis were conducted using S- and T-learners that were based on TRAM-DAGs, logistic regression, logistic lasso regression, and random forests. Applied as S-learners, TRAM-DAGs with complex shift or intercept terms allow for flexible modeling of treatment effect heterogeneity. Accurate ITE estimation requires models that generalize well. Since ITEs are differences between two outcome predictions, even small errors can accumulate. Hoogland *et al.* (2021) emphasize the need to balance model complexity with data availability to avoid overfitting. They also highlight that estimating risk differences is inherently more challenging than predicting single outcomes. Calibration is crucial when ITEs are derived from predicted probabilities. While neural networks may be accurate, they often produce poorly calibrated probabilities (Guo *et al.*, 2017), which can mislead treatment effect estimation. Similarly, conventional models like logistic regression can overfit in small samples or high-dimensional settings, leading to extreme predictions. Penalization methods such as lasso shrink coefficients to improve generalization (Riley *et al.*, 2021), though studies have shown that they can perform inconsistently in low-sample scenarios (Calster *et al.*, 2020). These factors are important to consider when ITEs. When using random forests as S-learners, care must be taken to ensure that the treatment variable is included in the splits. Proper tuning of hyperparameters like tree depth and variable selection (e.g., `mtry`) may be necessary to avoid overfitting. In summary, model choice for ITE estimation should consider the complexity of treatment-outcome relationships, sample size, and calibration needs. While complex models like TRAM-DAGs and random forests can capture complicated heterogeneity, simpler models may offer more robust performance in smaller datasets or where treatment-covariate interactions are straightforward.

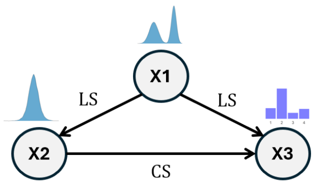
Validation of ITE Estimation: Hoogland *et al.* (2024) examined a variety of evaluation methods for assessing discrimination and calibration in ITE estimation. They emphasize that accurate prediction of potential outcomes should be considered a prerequisite for reliable ITE modeling. Furthermore, as in standard prediction modeling, proper validation should ideally be performed on independent (out-of-sample) data. In this thesis, although various quantitative evaluation metrics exist, we primarily relied on visual validation tools. One such approach is the ITE-ATE plot, where individuals are grouped based on their predicted ITEs. Within each ITE-subgroup, the empirical ATE is calculated using observed outcomes. If the ITE estimates are well-calibrated, the observed ATEs should align the predicted values, resulting in a calibration curve that lies along the identity line. Additionally, in simulation experiments where the true ITEs are known, estimation accuracy was assessed by directly comparing predicted and true effects. Furthermore, model calibration in terms of predictive performance $P(Y \mid \mathbf{X} = \mathbf{x})$ was evaluated using calibration plots. As an additional check, the average of the estimated ITEs across the population should equal the estimated ATE, i.e., $\mathbb{E}[\text{ITE}(X)] = \text{ATE}$. Large deviations from this condition may indicate systematic bias. However, satisfying this equality does not guarantee that the estimated ITEs are accurate at the individual level. All evaluations were conducted both on the training data and on a separate hold-out test set.

2.3 Experiment 1: TRAM-DAG (simulation)

This experiment demonstrates the application of TRAM-DAGs on a synthetic dataset, using the illustrative DAG previously shown in Figure 2.3. The objective is to show how TRAM-DAGs can learn structural equations from observational data by fitting a model to the joint distribution that resulted from the underlying causal structure.

We visualized the model fitting in terms of the training loss and subsequently showed and interpreted the learned components of the transformation functions, such as intercepts, linear and complex shifts. Finally, we drew samples from the estimated distributions to obtain observational and interventional distributions. We also conducted counterfactual queries on the learned model.

Data-generating process: We simulated a dataset with three variables, X_1, X_2, and X_3, following the structure of the DAG and its associated meta-adjacency matrix shown in Figure 2.6. The matrix describes the functional dependencies between variables, where LS indicates a linear shift and CS a complex shift. Rows represent the source of the effect, and columns the target.



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

Figure 2.6: Causal graph (left) and meta-adjacency matrix (right) for Experiment 1. The transformation function of X_2 depends on X_1 via a linear shift (LS). The transformation function of X_3 depends on X_1 via a linear shift (LS) and on X_2 via a complex shift (CS).

The variable X_1 is continuous and bimodally distributed, and acts as a source node in the DAG, i.e., it is not influenced by any other variable:

$$X_1 = \begin{cases} \mathcal{N}(0.25,\, 0.1^2) & \text{with probability } 0.5, \\ \mathcal{N}(0.73,\, 0.05^2) & \text{with probability } 0.5 \end{cases}$$

The second variable, X_2, is continuous and linearly dependent on X_1 on the log-odds scale, with a true coefficient of $\beta_{12} = 2$. Its transformation function is

$$h(X_2 \mid X_1) = h_I(X_2) + \beta_{12} X_1,$$

where the baseline transformation (i.e., intercept) of X_2 is $h_I(X_2) = 5X_2$.

The third variable, X_3, is ordinal and depends on both X_1 (LS) and X_2 (CS). We define the complex shift induced by X_2 as $f(X_2) = 0.5 \cdot \exp(X_2)$, and specify the linear shift parameter for X_1 as $\beta_{13} = 0.2$. The transformation function for category k of the ordinal variable X_3 with 4 levels (K) is thus defined by

$$h(X_{3,k} \mid X_1, X_2) = \vartheta_k + \beta_{13} X_1 + f(X_2),$$

with cut-points $\vartheta_k \in \{-2, 0.42, 1.02\}$ defining the thresholds of the ordinal variable. We generated samples for X_2 and X_3 as described in Section 2.1.4, by first sampling a latent value from the standard logistic distribution and then determining the corresponding observation using the transformation function.

This simulation allows us to assess whether the TRAM-DAG model can correctly recover the functional forms of the conditional dependencies and the associated parameters (linear and complex).

Model: Given the adjacency matrix and the simulated observations, we construct a modular neural network based on the TRAM-DAG framework. The complex shift from X_2 to X_3 is modeled using a neural network with 4 hidden layers and 2 nodes per layer, as illustrated in Figure 2.7. A total of 20,000 samples are generated according to the defined DGP to fit the model. The model is trained for 400 epochs using the Adam optimizer (Kingma and Ba, 2015) with a learning rate of 0.005.

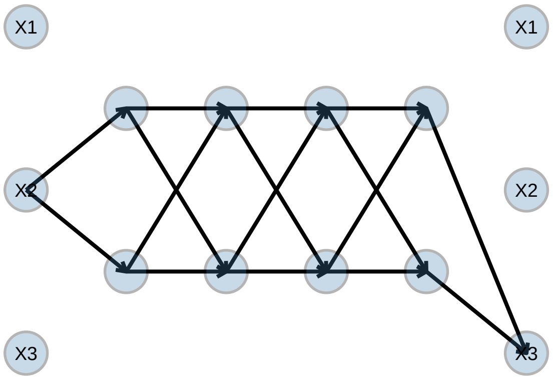


Figure 2.7: Neural network architecture for the complex shift on X_3 from X_2. The complex shift is modeled by a neural network with 4 hidden layers of shape (2, 2, 2, 2), using non-linear activation functions (sigmoid).

Model evaluation: We compare the estimated intercepts, learned coefficients, and the complex shift to the true values used in the DGP. We also compare the sampled observational and interventional distributions to the true distributions. For the counterfactual queries, we show the estimated counterfactual values for X_2 under an intervention on X_1 at a specific value, and compare these to the true counterfactual outcomes.

2.4 Experiment 2: ITE on International Stroke Trial (IST)

Chen *et al.* (2025) evaluated multiple causal ML methods on the International Stroke Trial (IST), to estimate the individualized treatment effects. They demonstrated that none of the applied ML methods generalized well, as performance on the test data differed significantly from the training data on the chosen evaluation metrics. In this experiment, we replicate the analysis on the same data by applying three causal ML methods for ITE estimation, to investigate whether we obtain similar results as the authors.

Data: The International Stroke Trial was a large, randomized controlled trial conducted in the 1990s to assess the efficacy and safety of early antithrombotic treatment in patients with acute ischemic stroke (International Stroke Trial Collaborative Group, 1997). Using a 2x2 factorial design, 19,435 patients across 36 countries were randomized within 48 hours of symptom onset to receive aspirin, subcutaneous heparin, both, or neither. Patients allocated to aspirin (300 mg daily for 14 days) had a 6-month death or dependency rate of 62.2%, compared to 63.5% in the control group not receiving aspirin, corresponding to a statistically significant absolute risk reduction after adjustment for baseline prognosis (1.4%, p = 0.03). The authors stated that there was no interaction between aspirin and heparin in the main outcomes. In this thesis, we focus exclusively on the aspirin vs. no aspirin comparison and the outcome of death or dependency at 6 months after stroke.

The dataset used in this experiment was made publicly available by Sandercock *et al.* (2011) and contains individual-level data, including baseline covariates assessed at randomization, treatment allocation, and 6-month outcomes, with a follow-up rate of 99%.

We used the same data pre-processing steps as Chen *et al.* (2025) to ensure comparability of results. 5.9% of individuals had incomplete data and were removed from the dataset. We used 2/3 of the data for fitting the models and 1/3 as a hold out test set. The final dataset included 21 baseline variables recorded at randomization: aspirin allocation (treatment), age, delay between stroke and randomization (in hours), systolic blood pressure, sex, CT performed before randomization, visible infarct on CT, atrial fibrillation, aspirin use within 3 days prior to randomization, and presence or absence of neurological deficits (including face, arm/hand, leg/foot deficits, dysphasia, hemianopia, visuospatial disorder, brainstem or cerebellar signs, and other neurological deficits), as well as consciousness level, stroke subtype, and geographical region. The outcome variable was death or dependence at 6 months.

Models for ITE estimation: The aim is to estimate the ITE based on baseline characteristics. As a benchmark, we apply a T-learner logistic regression (following Chen *et al.* (2025), using the `stats` package). As a more complex model, we apply a T-learner tuned random forest (using the `comets` package (Kook, 2024)), which tunes the number of variables considered for splitting at each node (`mtry`) and the maximum tree depth (`max.depth`) using out-of-bag error, with 500 trees. Additionally, we apply an S-learner TRAM-DAG. For the random forest and TRAM-DAG based methods, we additionally scale numerical and dummy encode categorical covariates prior to model training. The transformation function of the outcome is modelled by a complex intercept $h(Y \mid T, \mathbf{X}) = CI(T, \mathbf{X})$, with 4 hidden layers of shape (20, 10, 10, 2). This architecture allows for interaction between the treatment and covariates. Furthermore, batch normalization, ReLU activation, and dropout (0.1) are applied to prevent overfitting and stabilize learning. A validation set comprising 20% of the training data is used to select the model with the lowest out-of-sample negative log-likelihood, while the test set remains untouched for final evaluation. Since the IST stroke trial is a randomized controlled trial, the full potential of TRAM-DAGs (that lies in the observational setting) is not needed, as 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.

Model evaluation: 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 predict the potential outcomes with the trained models to estimate the ITE on the training and test set in terms of the risk difference $\text{ITE}_i = \text{P}(Y_i = 1|T = 1, \mathbf{X}_i) - \text{P}(Y_i = 1|T = 0, \mathbf{X}_i)$. For visual validation, we show the densities of the estimated ITEs on both datasets, and the ITE-ATE plots to assess whether the estimated ITEs align with the observed outcomes.

2.5 Experiment 3: ITE model robustness in RCTs (simulation)

In this section, we perform a simulation study to estimate the ITE using different models in an RCT setting under various scenarios. The aim is to identify conditions under which ITE estimation fails, and whether such failure is model-agnostic – i.e., driven by external factors such as unobserved covariates or the strength of the treatment effect, rather than by the model class itself. This may provide insight into why ITE estimation can fail in real-world applications, as demonstrated by Chen *et al.* (2025) on the IST dataset and replicated in our own work in Experiment 2 (Section 3.2). The simulation is based on a data-generating process (DGP) that resembles an RCT. We assume a binary outcome and a set of covariates that influence the outcome. There may also be treatment-covariate interactions that are responsible for heterogeneity in the treatment effect.

Data-generating process: Data is generated similarly to the approach proposed by Hoogland *et al.* (2021). The binary treatment (T) is sampled from a Bernoulli distribution with probability 0.5. The five covariates ($\mathbf{X}$), representing patient-specific characteristics at baseline, are drawn from a multivariate standard normal distribution with a compound symmetric covariance matrix ($\rho = 0.1$). The binary outcome (Y) is sampled from a Bernoulli distribution with probability $\text{P}(Y_i = 1 \mid \mathbf{X_i} = \mathbf{x_i}, T_i = t_i) = \text{logit}^{-1}\left(\beta_0 + \beta_T t_i + \boldsymbol{\beta}_X^\top \mathbf{x_i} + t_i \cdot \boldsymbol{\beta}_{TX}^\top \mathbf{x_{TX,i}}\right)$, where i denotes the patient index, and $\mathbf{x}_{TX,i}$ denotes the subset of covariates that interact with the treatment.

The simulated datasets are generated under three scenarios, where coefficients are set to different values or not all variables are observed. In Scenario 1, the coefficients are: $\beta_0 = 0.45$ (intercept), $\beta_T = -0.85$ (direct treatment effect), $\boldsymbol{\beta}_X = (-0.5, 0.8, 0.2, 0.6, -0.4)$ (direct covariate effects), and $\boldsymbol{\beta}_{TX} = (0.9, 0.1)$ (interaction effects between treatment and covariates X_1 and X_2 on the outcome). In Scenario 2, the same coefficients are used, but the covariate X_1, which is responsible for a large portion of the heterogeneity, is not observed in the final dataset. This is expected to cause difficulties in estimating the ITE. In Scenario 3, the coefficients for the direct treatment and interaction effects are set to $\beta_T = -0.05$ and $\boldsymbol{\beta}_{TX} = (-0.01, 0.03)$ to represent a weak treatment effect and low heterogeneity. All other coefficients remain unchanged, and all covariates are observed. The DAGs corresponding to the three scenarios are presented in Figure 2.8.

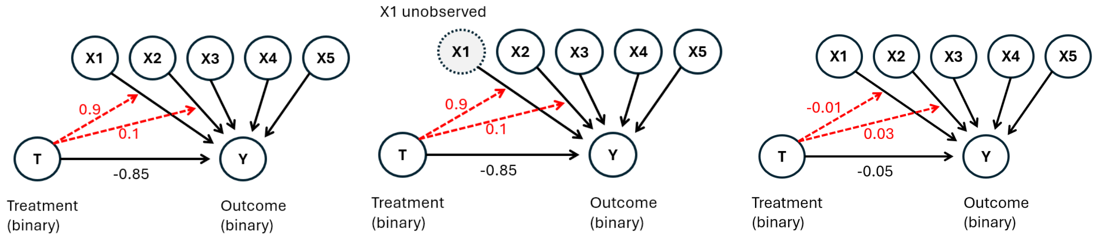


Figure 2.8: Data-generating process (DGP) for the three scenarios in the ITE simulation study (RCT). Interaction effects between treatment (T) and covariates (X_1 and X_2) on the outcome (Y) are shown in red. Left: Scenario 1, where all covariates are observed and both treatment effect and heterogeneity are strong; Middle: Scenario 2, with the same DGP as in Scenario 1, but where covariate X_1 is not observed; Right: Scenario 3, where the treatment effect and heterogeneity are weak, and all covariates are observed.

Models for ITE estimation: The datasets generated from the DGP under the three scenarios are used to estimate the ITE using different models. We applied the following models: T-learner logistic regression (`stats` package), T-learner logistic lasso regression (`glmnet` package (Friedman *et al.*, 2010), with the regularization parameter λ estimated via 10-fold cross-validation), S-learner logistic lasso regression (same as the T-learner), T-learner random forest (`randomForest` package (Breiman, 2001), 100 trees), and T-learner tuned random forest (`comets` package (Kook, 2024), which tunes the number of variables considered for splitting at each node (`mtry`) and the maximum tree depth (`max.depth`) using out-of-bag error, 500 trees).

While all models were applied, we present only the results of the T-learner logistic regression as a benchmark (same model as used in the data-generating process), and the tuned random forest as representation of a complex non-parametric model. In Appendix 6.8, we additionally present the results for a standard random forest evaluated for Scenario 1 to illustrate the importance of model tuning to prevent overfitting and ensure accurate calibration.

All models were trained on a training set and evaluated on a test set, each consisting of 10,000 samples generated from the same DGP. TRAM-DAGs would also be well suited for ITE estimation in this setting, but we chose not to apply them in this experiment, since the main objective is to assess behavioral differences between complex and simple models under different scenarios. TRAM-DAGs are applied in other experiments in this thesis.

Model evaluation: Model performance is evaluated visually on both the training and test datasets. For predictive performance, we present true vs. predicted probabilities $\mathrm{P}(Y = 1 \mid X, T)$ to assess how well each model is calibrated. Plots of true vs. predicted ITEs show how closely the model estimates match the true effects. Since the true probabilities and ITEs are known by design in this simulation, direct evaluation of calibration and prediction accuracy is possible, unlike in real-world applications.

To assess whether estimated ITEs correspond to actual observed outcomes, we use ITE-ATE plots. These show the observed average treatment effect (ATE), calculated as $\mathrm{P}(Y = 1 \mid T = 1) - \mathrm{P}(Y = 1 \mid T = 0)$, in the respective subgroups of estimated ITEs. Accurate models should produce ITE-ATE points that align with the identity line.

These simulation scenarios allow us to assess ITE estimation performance under challenging conditions such as omitted variables and weak treatment effects. The subsequent results reveal which models remain robust under such violations and provide insight into possible real-world estimation failures.

2.6 Experiment 4: ITE estimation with TRAM-DAGs (simulation)

We claim that the TRAM-DAG framework can be effectively used for ITE estimation on observational data with confounding and mediating variables, provided that the identifiability assumptions are satisfied and the DAG is fully known. Therefor, we apply TRAM-DAGs in both a confounded and a randomized setting, using data simulated according to the DAGs shown in Figure 2.9. The binary treatment (X_4) is the intervention variable, and the goal is to estimate the ITE for the continuous outcome Y.

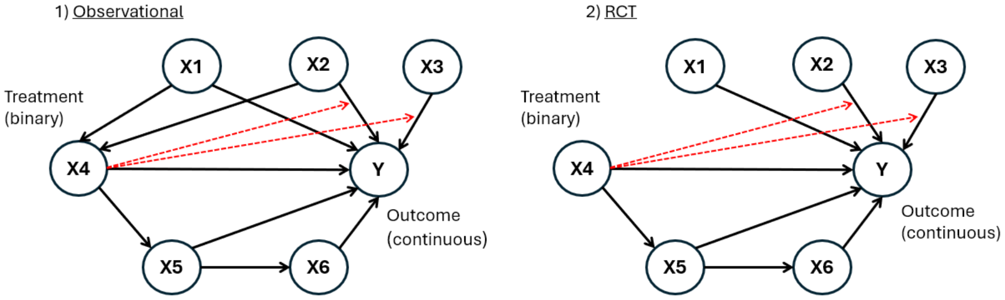


Figure 2.9: DAGs used for the simulation to estimate the ITE. Left: observational; Right: RCT setting. The source nodes X_1, X_2, and X_3 come from a multivariate standard normal distribution ($\rho = 0.1$). In the observational setting, the binary treatment X_4 depends on the parents X_1 and X_2. In the RCT setting, this dependency is omitted due to randomization. The outcome Y depends on all variables, with additional interaction effects between the treatment and the variables X_2 and X_3. All variables except the treatment X_4 are continuous.

Illustrative scenario: An possible real-world scenario that follows 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 (X_4) sent to customers. If the treatment is not randomized, it depends on prior total spend (X_1) and the customer engagement score (X_2). The outcome is the total spend in the 30 days following the email, denoted as Y. The prior total spend (X_1) and customer engagement score (X_2) act as confounders, influencing both the treatment and the outcome. The customer satisfaction score (X_3), obtained from a recent survey, is another predictor. The time spent on the website after receiving the email (X_5) is a mediator that affects the number of product pages viewed (X_6), which in turn influences the total spend (Y). Interaction effects exist between the treatment (X_4) and both X_2 and X_3, meaning the treatment effect differs based on the customer's engagement and satisfaction levels. The goal is to estimate the individualized treatment effect (ITE) of the marketing email (X_4) on the total spend (Y), in order to personalize customer targeting.

Data-generating process: The standard logistic distribution was chosen as the noise distribution to align with other examples in this thesis. Any other noise distribution could also be used here, as we are not interested in coefficient interpretability in this experiment. All variables except the binary treatment X_4 are continuous. The source nodes X_1, X_2, and X_3 are generated from a multivariate standard normal distribution with a compound symmetric covariance matrix ($\rho = 0.1$). These variables represent baseline patient characteristics.

In the observational setting, X_1 and X_2 act as confounders by influencing both the treatment assignment X_4 and the outcome Y. In the RCT setting, these dependencies are removed due to randomization. The mediator X_5 depends on treatment X_4, and X_6 depends on X_5. The log-odds of the continuous outcome Y depend linearly on all covariates, including additional interaction terms between the treatment and X_2 and X_3. Equation 2.12 defines the outcome on the log-odds scale:

$$h(y \mid \mathbf{X}) = h_I(y) + \boldsymbol{\beta}_X^\top \mathbf{X} + X_4 \cdot (\boldsymbol{\beta}_{TX}^\top \mathbf{X}_{\text{TX}}) \tag{2.12}$$

Here, $h_I(y)$ is the intercept function, $\mathbf{X}$ is the full covariate vector, and $\mathbf{X}_{\text{TX}} = \{X_2, X_3\}$ denotes the interaction covariates that affect the outcome only when treatment is applied ($X_4 = 1$). The intercept function $h_I(y)$ must be smooth and monotonically increasing. We define it as $h_I(y) = \tan(y/2)/0.2$ for $y \in [-2, 2]$, and extrapolate linearly at the boundaries.

The coefficients are set as $\boldsymbol{\beta}_X = (-0.5,\ 0.5,\ 0.2,\ 1.5,\ -0.6,\ 0.4)$, where the value 1.5 represents the direct effect of treatment X_4 on the outcome. The interaction coefficients are set to $\boldsymbol{\beta}_{TX} = (-0.9,\ 0.7)$.

Three scenarios: The experiment is conducted under three different scenarios regarding the effect of the treatment on the outcome Y in the DGP: (1) both direct and interaction effects, (2) only a direct effect, and (3) only interaction effects. Depending on the scenario, the corresponding coefficients in $\boldsymbol{\beta}_X$ and $\boldsymbol{\beta}_{TX}$ are set to zero.

TRAM-DAG estimation: In both the observational and RCT settings, the TRAM-DAG is fitted as an S-learner (i.e., a single model including the treatment variable). To allow for full flexibility, all nodes with parents are modeled using complex intercepts with three hidden layers of size (10, 10, 10), without batch normalization or dropout, unsing ReLU activation. The model is trained on a dataset of 20,000 samples. To prevent overfitting, an additional validation set of 10,000 samples is used, and the final model is selected using early stopping based on validation loss.

ITE estimation procedure: In contrast to most of the research we analyzed, where the ITEs are typically defined in terms of expected values of the potential outcomes, here we estimate the quantile treatment effect (QTE), specifically at the median. For each individual, we calculate the difference between the 0.5-quantiles of the potential outcome distributions under treatment and control. Chernozhukov and Hansen (2005), for example, highlighted the ability of quantile regression models in heterogeneous treatment effect estimation. QTEs are particularly relevant when the distributional behavior of outcomes beyond the mean is of interest. The median QTE is defined as

$$\text{QTE}^{(0.5)}(\mathbf{x}) = Q_{Y(1)|\mathbf{X}=\mathbf{x}}(0.5) - Q_{Y(0)|\mathbf{X}=\mathbf{x}}(0.5), \tag{2.13}$$

where $Q_{Y(t)|\mathbf{X}=\mathbf{x}}(q)$ denotes the q-th quantile of the potential outcome distribution under treatment t.

Once the TRAM-DAG model is fitted on observed data, we can access the inverse transformation functions $X_i = h^{-1}(Z_i \mid \text{pa}(X_i))$, which represent the structural equations of the DAG. ITE estimation proceeds in three steps, according to Algorithm 3. First, the latent values z_{ij} for the explanatory variables $X_i \in \{X_1, X_2, X_3, X_5, X_6\}$ are computed for each sample j using the transformation functions conditioned on their observed parents. Second, the treatment variable X_4 is intervened on using the do-operator for both $X_4 = 0$ and $X_4 = 1$. For each of these two treatment states, X_5, X_6, and the potential outcome (distribution) Y are sampled sequentially using the latent encodings and inverse transformations. This means that the counterfactuals for X_5 and X_6 are determined. This results in two potential outcome distributions per individual, as illustrated in Figure 2.10. Finally, for each individual, the median is determined for both potential outcome distributions and the QTE is calculated as the difference between the two medians. Note that estimating the potential outcomes in terms of expected values would also be possible – either by repeatedly sampling from each outcome distribution or, potentially, by numerical integration. However, for this experiment, we chose to estimate the QTE. For simplicity, we will refer to these as ITEs throughout the remainder of the experiment.

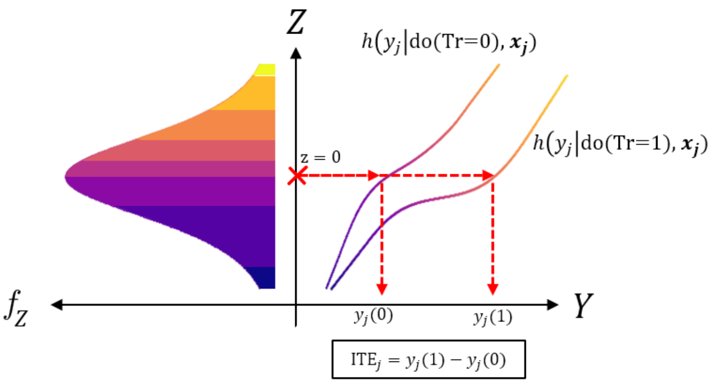


Figure 2.10: ITE estimation in terms of quantile treatment effect (QTE) at the median with TRAM-DAGs. The two transformation functions represent the distributions of the potential outcomes under both treatments. For the QTE(0.5), the median of the latent distribution (0 for the standard logistic) is evaluated on both transformation functions to determine the median potential outcomes. We define the ITE for an individual as the difference of the median potential outcomes.

Algorithm 3 ITE estimation (QTE) using TRAM-DAGs

```
Input: Fitted TRAM-DAG, dataset of $n$ individuals
for each individual $j = 1$ to $n$ do
    Step 1: Determine latent values
    for each explanatory node $X_i \in \{X_1, X_2, X_3, X_5, X_6\}$ do
        Compute latent value: $z_{ij} = h_i(x_{ij} \mid \mathrm{pa}(x_{ij}))$
    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 $z_{7,j} = 0$ (median of the potential outcome
distribution)
    end for
    Step 3: Compute ITE (QTE) for individual $j$
    $\mathrm{ITE}_j = y_j(1) - y_j(0)$
end for
Output: ITE estimates $\{\mathrm{ITE}_j\}_{j=1}^{n}$
```

Model evaluation: Validation is conducted on the training dataset and on an independent test dataset of same size. During the data-generating process, the true potential outcomes under both treatment states were recorded for each individual, which allows for exact computation of the true ITE. The estimated ITEs are evaluated against the true values using several visual and numerical metrics. These include density plots of the estimated ITEs, scatter plots of true vs. estimated ITEs, and ITE-ATE plots where the observed ATE per ITE subgroup is computed as the difference in medians. In addition, the average of the estimated ITEs is compared to the true average ITE and to the empirical ATE from the RCT setting. The scatter plot of true versus estimated ITEs is the most informative validation, as it directly reflects how accurately the model estimated ITEs.

2.7 Software

All code used in this thesis is available on GitHub: `https://github.com/mikekr97/MA_Mike`.

All analyses were conducted in R (R Core Team, 2024) (version 4.4.2) using RStudio. The packages `keras` (Chollet *et al.*, 2017) (version 2.15.0), `tensorflow` (Allaire and Tang, 2025) (version 2.16.0), and `reticulate` (Ushey *et al.*, 2025) (version 1.40.0) were used to build and train neural networks through Python's TensorFlow backend. These tools allowed for the use of deep learning methods directly within the R environment.

Chapter 3

Results

3.1 Experiment 1: TRAM-DAG (simulation)

In this section, we present the results of the simulation study to evaluate the performance of a TRAM-DAG in a simple scenario as illustrated by the DAG in Figure 2.6. The model was fitted on synthetic data. Figure 3.1 shows the loss and the estimated parameters for the linear shifts over epochs during training. The loss was minimized during training and the estimated parameters β_{12} and β_{13} converged to the true values used in the DGP. The linear shift parameters are the interpretable part of the model (log-odds ratios). From the fitted model, we generated samples from the observational distribution, as shown in Figure 3.2. The TRAM-DAG can recover the observational distribution as its samples align with the data that was used to fit the model. Then we drew samples from the interventional distribution, where $X_2 = 1$ is fixed, as shown in Figure 3.3. Fixing X_2 leads to a distributional change in X_3, which was also captured by the model. The TRAM-DAG learns the linear shifts (β_{12}, β_{13}) and the complex shift $f(X_2)$, which are shown in Figure 3.4. Figure 3.5 presents the intercepts learned for each of the nodes. For comparison, we added the estimated intercept functions from the Continuous Outcome Logistic Regression (Colr() function from the `tram` package (Hothorn *et al.*, 2018)) for X_1 and X_2, and the true values used in the DGP for the ordinal variable X_3 (three cut-points for the four levels). Since the transformation functions for X_1 and X_2 contain no complex terms, they match the default form used in Colr(). Finally, Figure 3.6 shows the counterfactuals for X_2 estimated by the TRAM-DAG for varying values of X_1. The counterfactuals are the predicted values of X_2 had X_1 taken other values instead of the initially observed one.

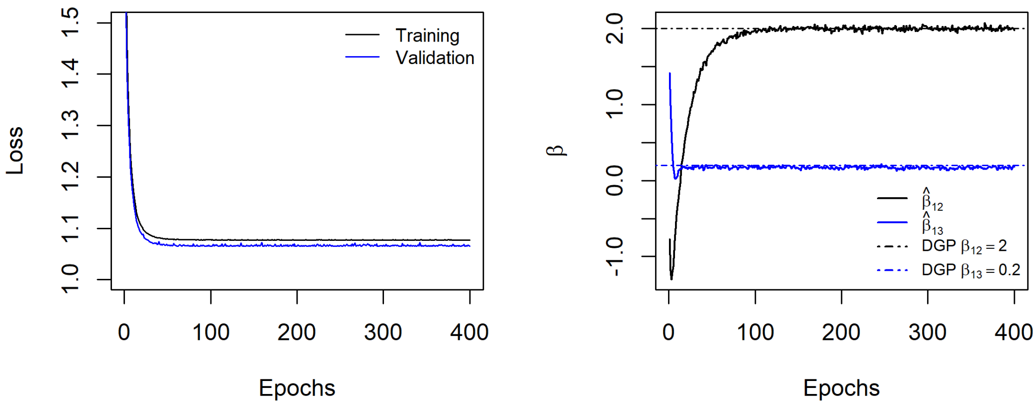


Figure 3.1: TRAM-DAG model fitting over 400 epochs for Experiment 1. Left: loss functions on the training and validation sets; Right: estimated parameters (betas) for the linear shift components over epochs. The estimates converge to the true values used in the DGP.

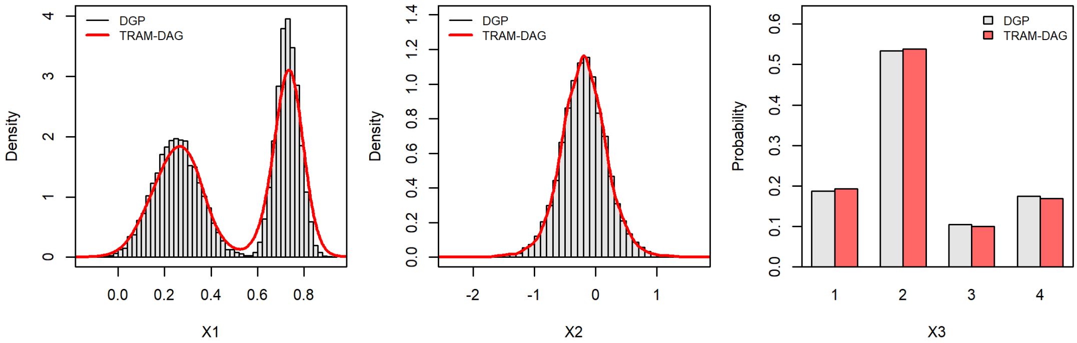


Figure 3.2: Samples generated by the TRAM-DAG from the learned observational distribution, compared to the true observations from the DGP.

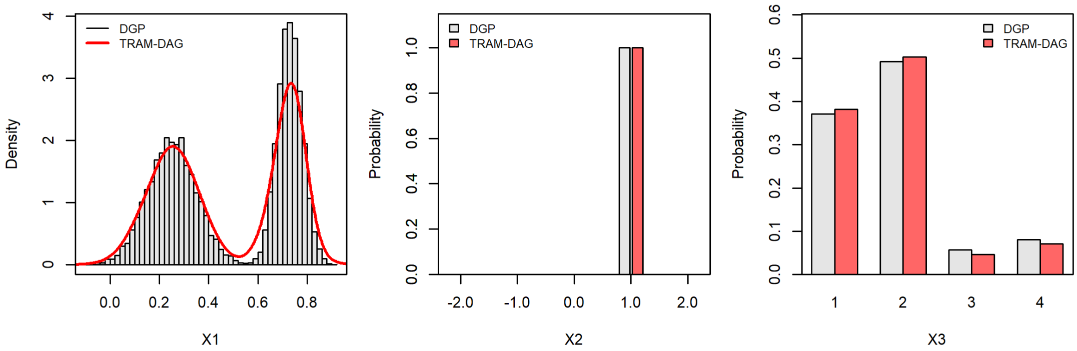


Figure 3.3: Samples generated by the TRAM-DAG compared to the true observations from the interventional distribution of the DGP, where $X_2 = 1$ is fixed. According to the DAG, this intervention induces a distributional change in X_3.

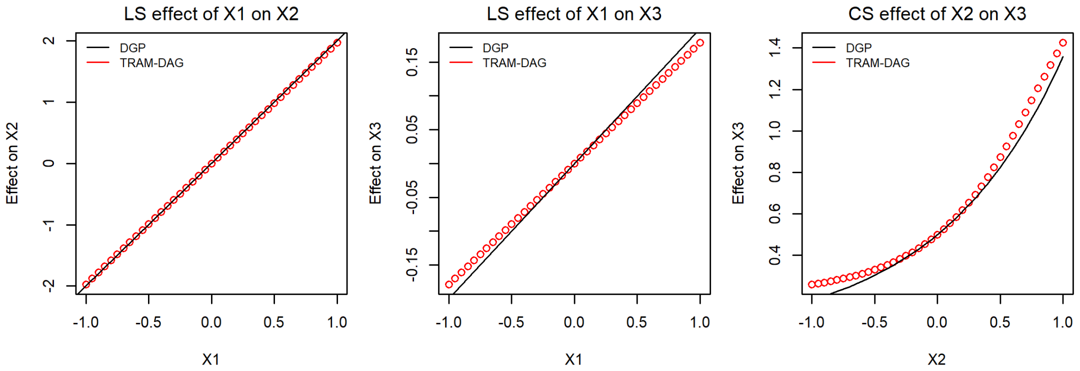


Figure 3.4: Linear and complex shifts learned by the TRAM-DAG. Left: LS(X_1) on X_2; Middle: LS(X_1) on X_3; Right: CS(X_2) on X_3. For visualization, we subtracted $\delta_0 = \text{CS}(0) - f(0)$ from the estimated complex shift CS(X_2) to align it with the true shift function $f(X_2)$ from the DGP.

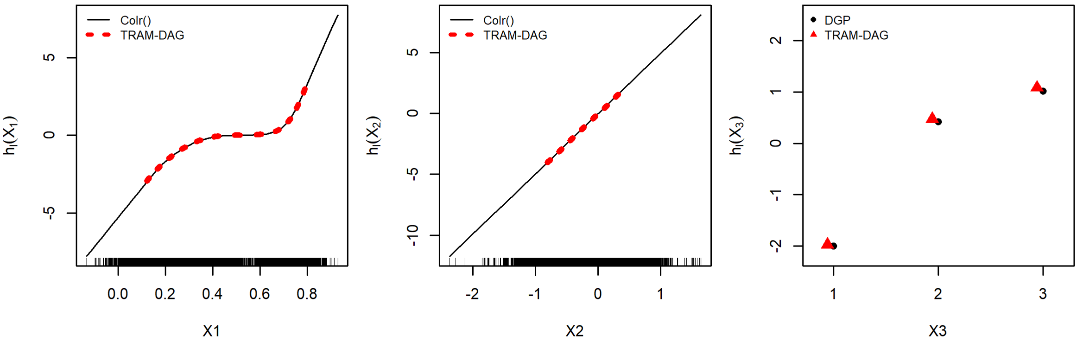


Figure 3.5: Intercepts learned for each of the nodes, along with the estimates from the `Colr()` function for the continuous variables and the true values from the DGP for the ordinal variable X_3. Left: Smooth baseline transformation function for continuous X_1; Middle: Smooth baseline transformation function for continuous X_2; Right: Cutpoints as the baseline transformation function for ordinal X_3. For the last plot, we added $\delta_0 = \text{CS}(0) - f(0)$ to the estimated cut-offs to make them comparable to the true parameters from the DGP.

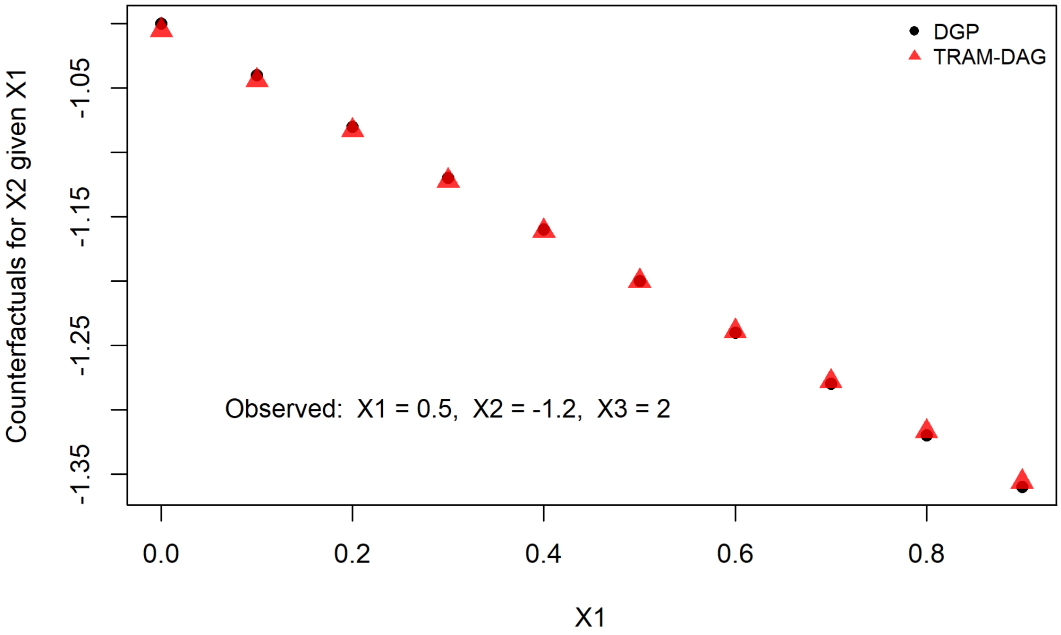


Figure 3.6: Counterfactuals for X_2 estimated with the TRAM-DAG for varying values of X_1. We assumed the observed values $X_1 = 0.5$, $X_2 = -1.2$, and $X_3 = 2$, and determined the counterfactual values of X_2 had X_1 taken different values instead of the actually observed one. This illustrates how the model estimates alternative outcomes under hypothetical interventions on X_1.

3.2 Experiment 2: ITE on International Stroke Trial (IST)

In this section, we present the results of the ITE estimation on the International Stroke Trial (IST) dataset. The observed treatment effect (ATE), defined as $\mathrm{P}(Y = 1|T = 1) - \mathrm{P}(Y = 1|T = 0)$, was -2.4% absolute risk reduction on the training set, with a 95% confidence interval from -4.1% to -0.6%. On the test set, the observed treatment effect was -0.1%, with a 95% confidence interval from -2.6% to 2.3%. The ITEs were estimated using three different models: T-learner logistic regression, T-learner tuned random forest, and S-learner TRAM-DAG. The estimated average treatment effect on the test set, calculated as $\mathrm{ATE}_{\mathrm{pred}} = \mathrm{mean}(\mathrm{ITE}_{\mathrm{pred}})$, was -2.5% for the T-learner logistic regression, -2.2% for the T-learner tuned random forest, and -3.1% for the S-learner TRAM-DAG. The density of predicted ITEs and the ITE-ATE plots for risk difference per estimated ITE subgroup, including 95% confidence intervals, are presented in Figures 3.7 - 3.9. Calibration plots are provided in Appendix 6.7, Figures 6.5 - 6.7.

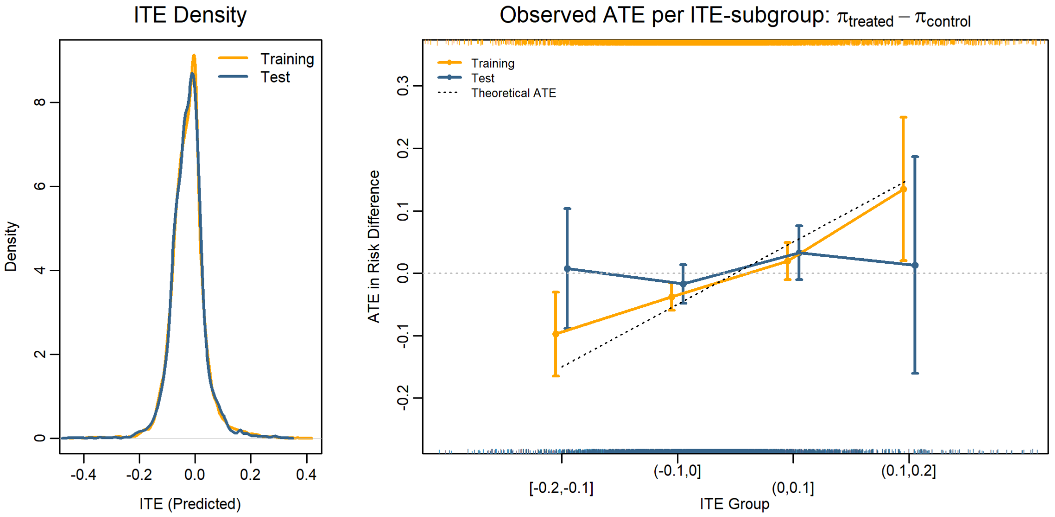


Figure 3.7: Results for the International Stroke Trial (IST) using the T-learner logistic regression. Left: density of predicted ITEs in the training and test sets; Right: observed ATE in terms of risk difference per estimated ITE subgroup.

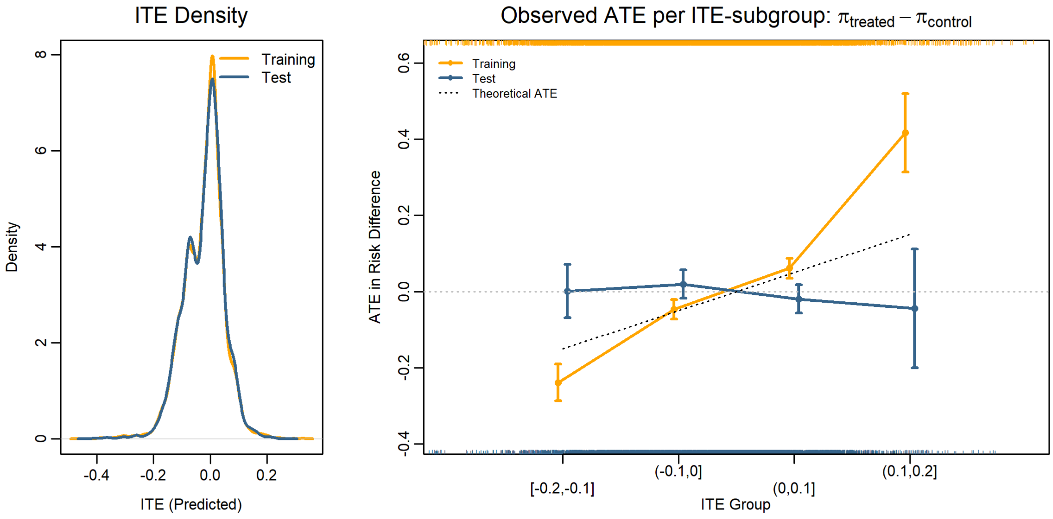


Figure 3.8: Results for the International Stroke Trial (IST) using the T-learner tuned random forest. Left: density of predicted ITEs in the training and test sets; Right: observed ATE in terms of risk difference per estimated ITE subgroup.

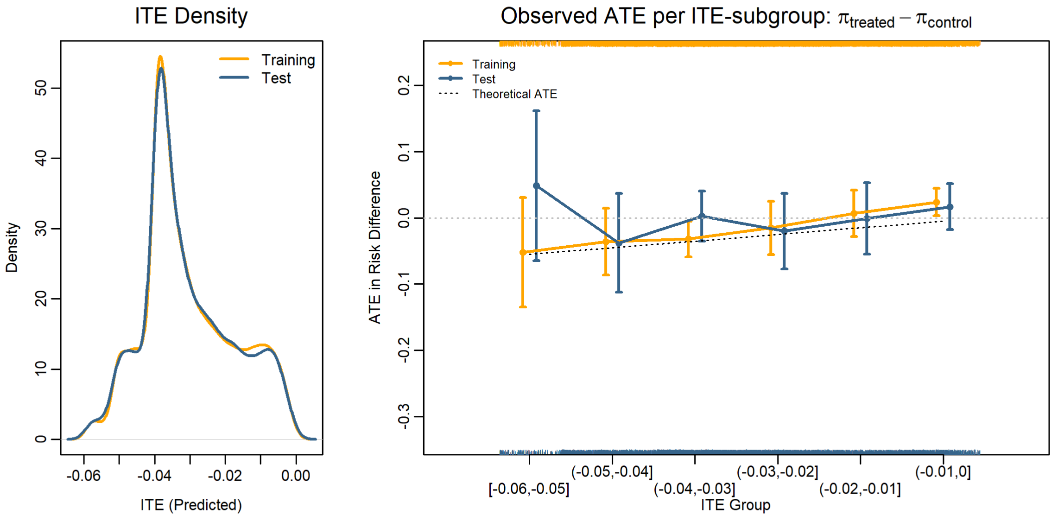


Figure 3.9: Results for the International Stroke Trial (IST) using the S-learner TRAM-DAG. Left: density of predicted ITEs in the training and test sets; Right: observed ATE in terms of risk difference per estimated ITE subgroup.

3.3 Experiment 3: ITE model robustness in RCTs (simulation)

In this section, we present the performance of two causal ML models for estimating the ITE under different simulated scenarios. Scenario 1 represents the ideal case where all variables are observed and treatment effects and heterogeneity are large. Scenario 2 uses the same DGP as in Scenario 1 but removes the covariate X_1, which has a strong interaction effect with the treatment, from the dataset and treats it as unobserved. Finally, in Scenario 3, the coefficients for the direct and interaction treatment effects are reduced, resulting in low heterogeneity. All variables are observed again in this last scenario.

In each scenario, we applied the T-learner logistic regression and the T-learner tuned random forest. The results of the models for the three scenarios are presented in Figures 3.11 to 3.18.

3.3.1 Scenario (1): Fully observed, large effects

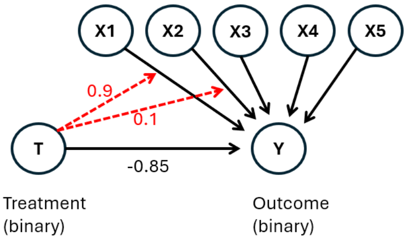


Figure 3.10: DAG for Scenario (1), where all variables are observed and both treatment and interaction effects are strong. The numbers indicate the coefficients on the log-odds scale. Red arrows represent interaction effects between treatment (T) and covariates (X_1 and X_2) on the outcome (Y).

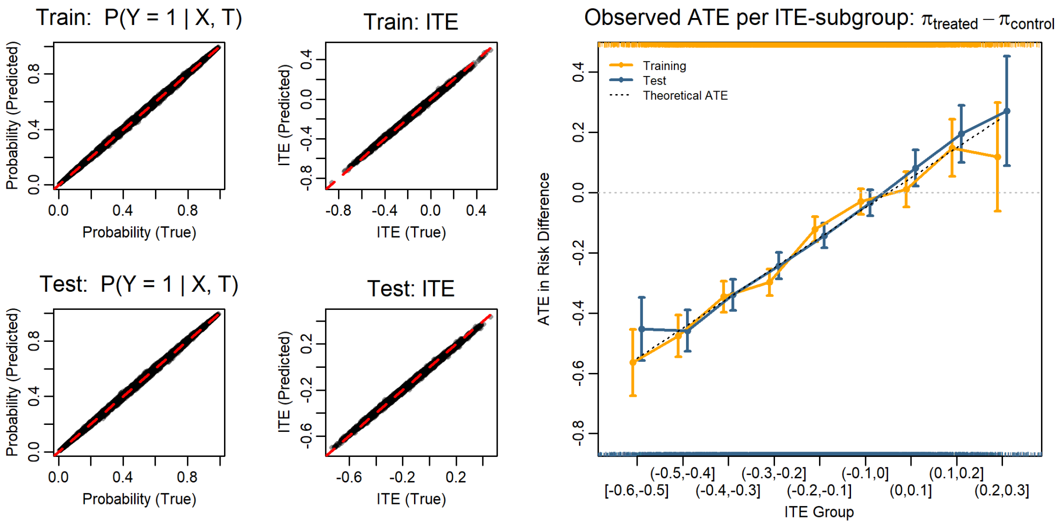


Figure 3.11: Results of the T-learner logistic regression in Scenario (1), where the DAG is fully observed and both treatment and interaction effects are strong. Left: true vs. predicted probabilities for $\mathrm{P}(Y = 1 \mid X, T)$; Middle: true vs. predicted ITEs; Right: observed ATE in terms of risk difference per estimated ITE subgroup.

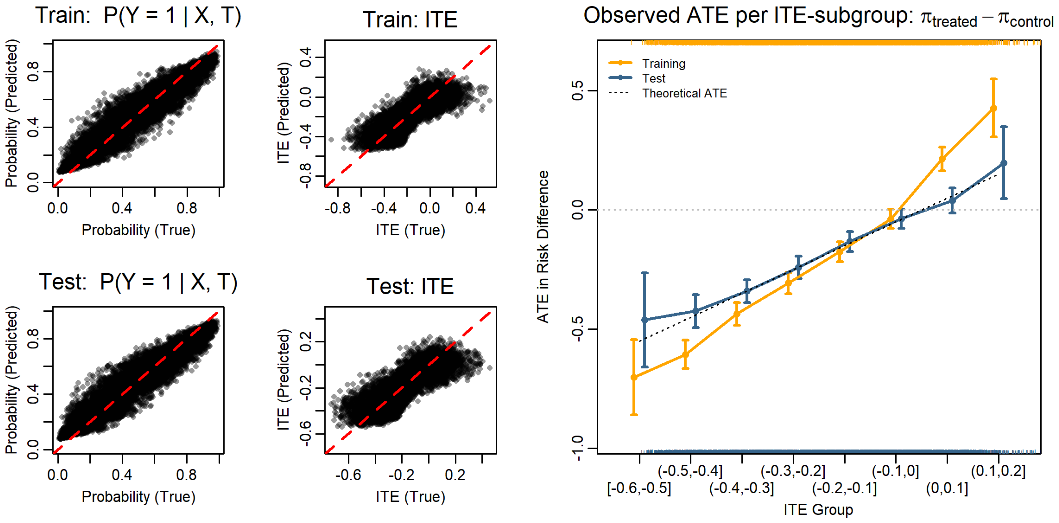


Figure 3.12: Results of the T-learner tuned random forest in Scenario (1), where the DAG is fully observed and both treatment and interaction effects are strong. Left: true vs. predicted probabilities for $\text{P}(Y = 1 \mid X, T)$; Middle: true vs. predicted ITEs; Right: observed ATE in terms of risk difference per estimated ITE subgroup.

3.3.2 Scenario (2): Unobserved interaction

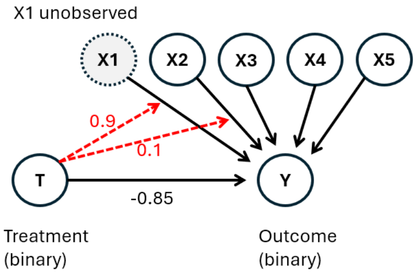


Figure 3.13: DAG for Scenario (2), where there are strong treatment and interaction effects, but variable X_1 is not observed. The numbers indicate the coefficients on the log-odds scale. Red arrows represent interaction effects between treatment (T) and covariates (X_1 and X_2) on the outcome (Y).

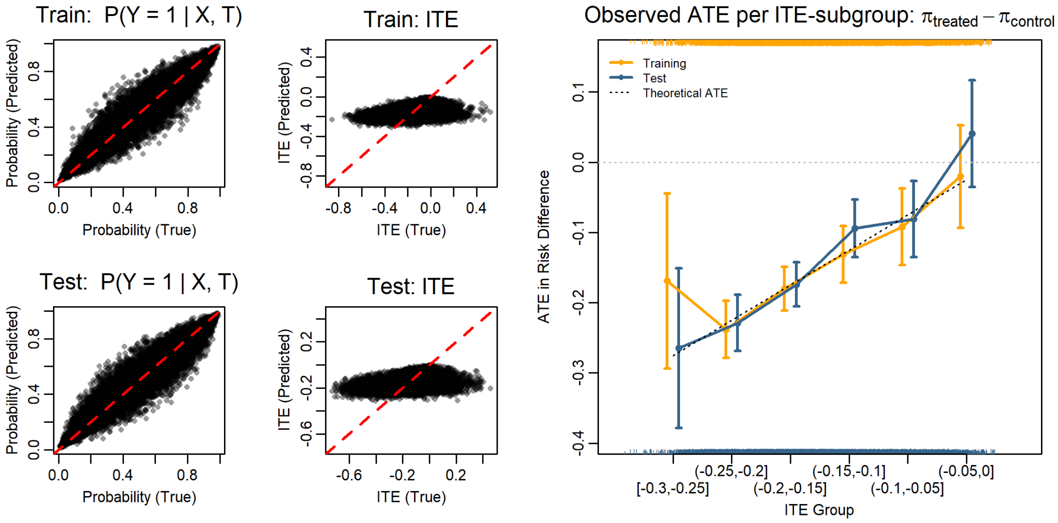


Figure 3.14: Results of the T-learner logistic regression in Scenario (2), where there are strong treatment and interaction effects, but variable X_1 is not observed. Left: true vs. predicted probabilities for $\mathrm{P}(Y = 1 \mid X, T)$; Middle: true vs. predicted ITEs; Right: observed ATE in terms of risk difference per estimated ITE subgroup.

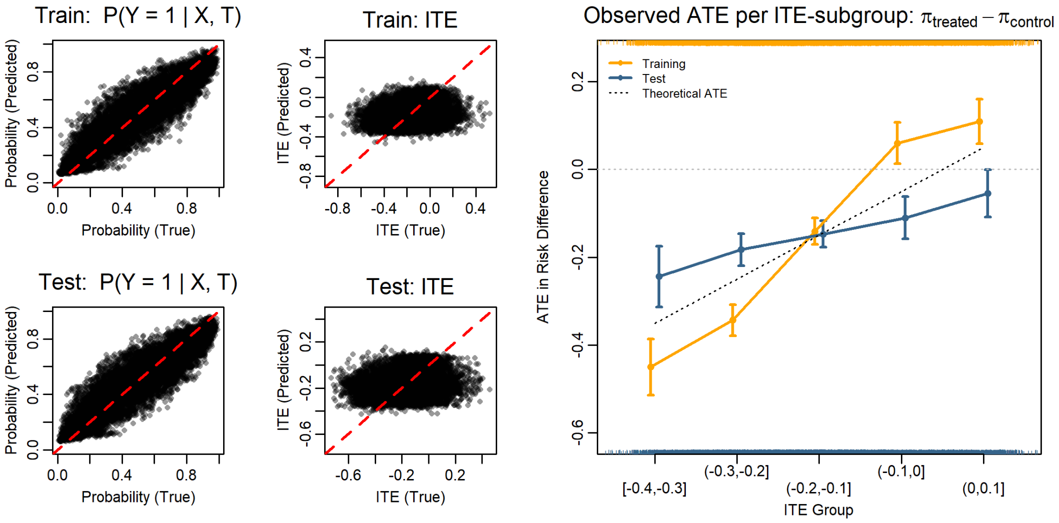


Figure 3.15: Results of the T-learner tuned random forest in Scenario (2), where there are strong treatment and interaction effects, but variable X_1 is not observed. Left: true vs. predicted probabilities for $\mathrm{P}(Y = 1 \mid X, T)$; Middle: true vs. predicted ITEs; Right: observed ATE in terms of risk difference per estimated ITE subgroup.

3.3.3 Scenario (3): Fully observed, small effects

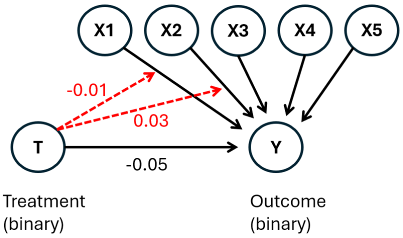


Figure 3.16: DAG for Scenario (3), where all variables are observed and both treatment and interaction effects are weak. The numbers indicate the coefficients on the log-odds scale. Red arrows represent interaction effects between treatment (T) and covariates (X_1 and X_2) on the outcome (Y).

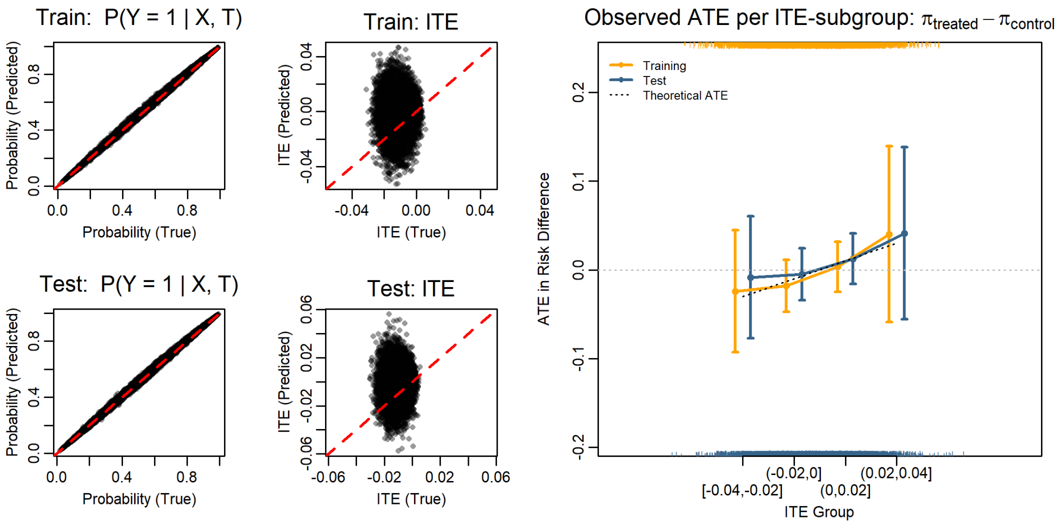


Figure 3.17: Results of the T-learner logistic regression in Scenario (3), where the DAG is fully observed and both treatment and interaction effects are weak. Left: true vs. predicted probabilities for P($Y = 1 \mid X, T$); Middle: true vs. predicted ITEs; Right: observed ATE in terms of risk difference per estimated ITE subgroup.

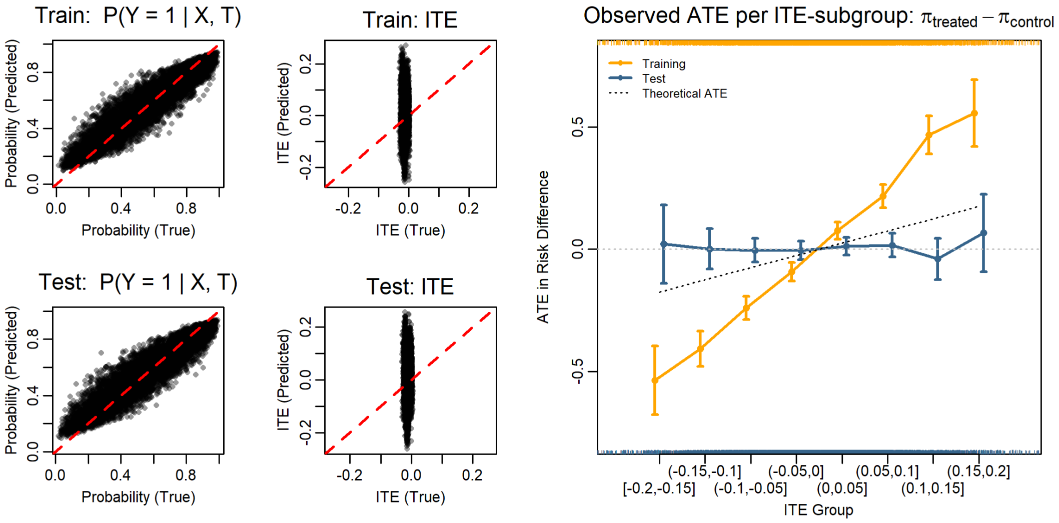


Figure 3.18: Results of the T-learner tuned random forest in Scenario (3), where the DAG is fully observed and both treatment and interaction effects are weak. Left: true vs. predicted probabilities for $\mathrm{P}(Y = 1 \mid X, T)$; Middle: true vs. predicted ITEs; Right: observed ATE in terms of risk difference per estimated ITE subgroup.

3.4 Experiment 4: ITE estimation with TRAM-DAGs (simulation)

First, we present the results for Scenario (1), which includes both direct and interaction effects of treatment. Then, we show the results for Scenario (2), which has a direct effect but no interaction effects, and finally Scenario (3), which includes interaction effects but no direct treatment effect. For each scenario, we compare the results in an observational setting with confounded treatment allocation and in a randomized controlled trial (RCT) setting without confounding. We also compare the average treatment effect (ATE), which can be directly calculated in the RCT setting on observed outcomes, with the ATE derived from the estimated ITEs. 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 to them as ITEs. The aim is to investigate how the TRAM-DAG performs in the presence or absence of direct and interaction effects of the treatment, both in confounded and in randomized settings, for the purpose of ITE estimation.

3.4.1 Scenario (1): Direct and interaction effects

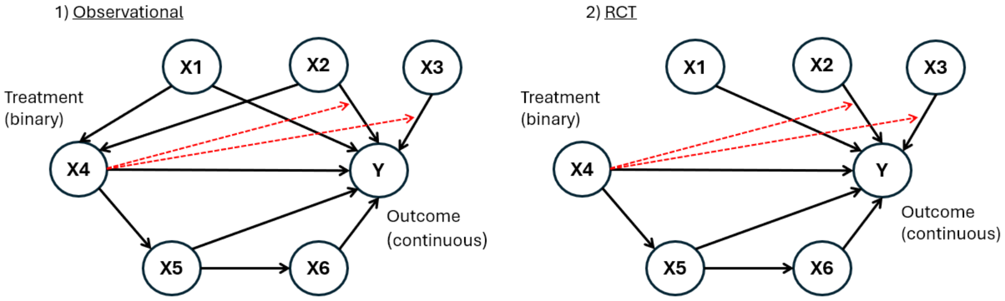


Figure 3.19: DAGs for Scenario (1), which includes a direct effect of the treatment on the outcome and additional interaction effects with covariates X_2 and X_3. Left: observational setting; Right: RCT setting.

Scenario (1) includes a direct effect of the treatment on the outcome, and interaction effects with X_2 and X_3, as shown in Figure 3.19. Train and test sets with 20,000 samples each were generated.

In the observational setting, treatment was confounded by X_1 and X_2. In the train set, 38.6% of individuals were in the control group and 61.4% in the treatment group. The test set had a similar distribution.

In the RCT setting, treatment was randomly assigned. In the train set, 49.8% were in control and 50.2% in treatment; in the test set, the shares were 50.2% and 49.8%, respectively.

Figure 3.20 shows the true ITE distribution from the DGP, which displays some heterogeneity due to interaction effects. Figure 3.21 shows the marginal distributions of all variables in the DGP and as estimated by the TRAM-DAG. The distribution of the outcome Y under do$(X_4 = 0)$ and do$(X_4 = 1)$ is shown in Figure 3.22. The ITEs were estimated as the difference in medians of the potential outcomes. Figure 3.23 compares the densities of the estimated and true ITEs. In both observational and RCT settings, the estimated ITEs are close to the true ones in both train and test sets. Figure 3.24 shows the scatterplots of estimated vs. true ITEs. Figures 3.25 and 3.26 show the ITE-ATE plots, where the ATE is calculated as the median difference in observed outcomes within ITE subgroups. The trends are similar across train and test sets.

The ATEs calculated based on different measures in both the training and test sets are shown in Table 3.1. In the RCT setting (training set), the difference in means of the outcomes between the two treatment groups was -0.563, with a confidence interval of -0.582 to -0.543.

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. $\mathrm{Y}^{(\mathrm{Tr})}_{\mathrm{observed}}$ denotes the observed outcome under the treatment (Tr) actually received. The estimated ATE from mean($\mathrm{ITE}_{\mathrm{estimated}}$) can be directly compared to the true mean($\mathrm{ITE}_{\mathrm{true}}$), whereas comparisons to the empirical ATEs based on outcome differences should be interpreted with caution.

Measure	**Observational**		**RCT**	
	Train	**Test**	**Train**	**Test**
ATE as mean($\mathrm{Y}^{(1)}_{\mathrm{observed}}$) − mean($\mathrm{Y}^{(0)}_{\mathrm{observed}}$)	NA	NA	-0.563	-0.563
ATE as median($\mathrm{Y}^{(1)}_{\mathrm{observed}}$) − median($\mathrm{Y}^{(0)}_{\mathrm{observed}}$)	NA	NA	-0.626	-0.638
ATE as mean($\mathrm{ITE}_{\mathrm{true}}$)	-0.620	-0.622	-0.620	-0.622
ATE as mean($\mathrm{ITE}_{\mathrm{estimated}}$)	-0.617	-0.620	-0.619	-0.622

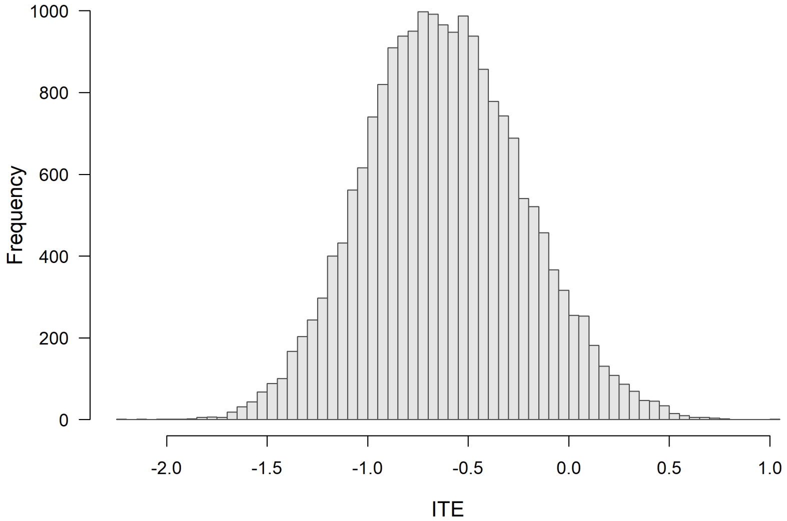


Figure 3.20: True ITE distribution resulting from the DGP for Scenario (1), which includes both direct and interaction effects. The true ITEs are identical for each individual in the observational and RCT settings, as they are based on the potential outcomes under both treatment conditions.

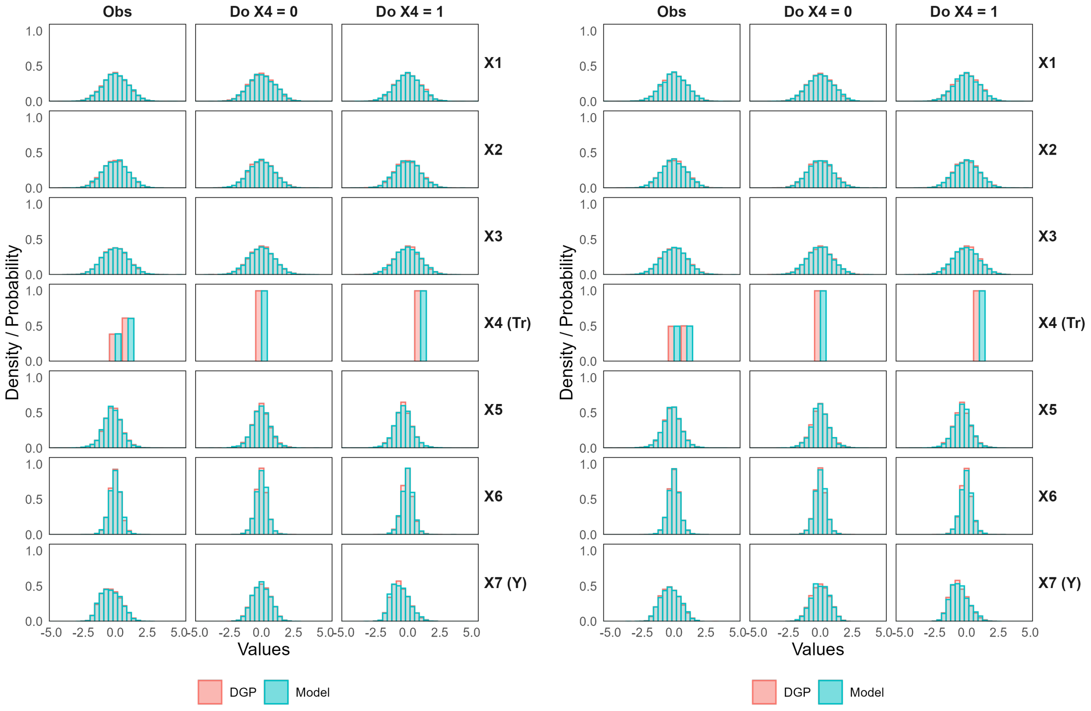


Figure 3.21: Marginal distributions of variables from the DGP and from samples generated by the fitted TRAM-DAG for Scenario (1) with direct and interaction effects. Distributions are shown as observed (Obs), under the control intervention (do(X_4 = 0)), and under the treatment intervention (do($X_4 = 1$)). Left: Observational; Right: RCT setting.

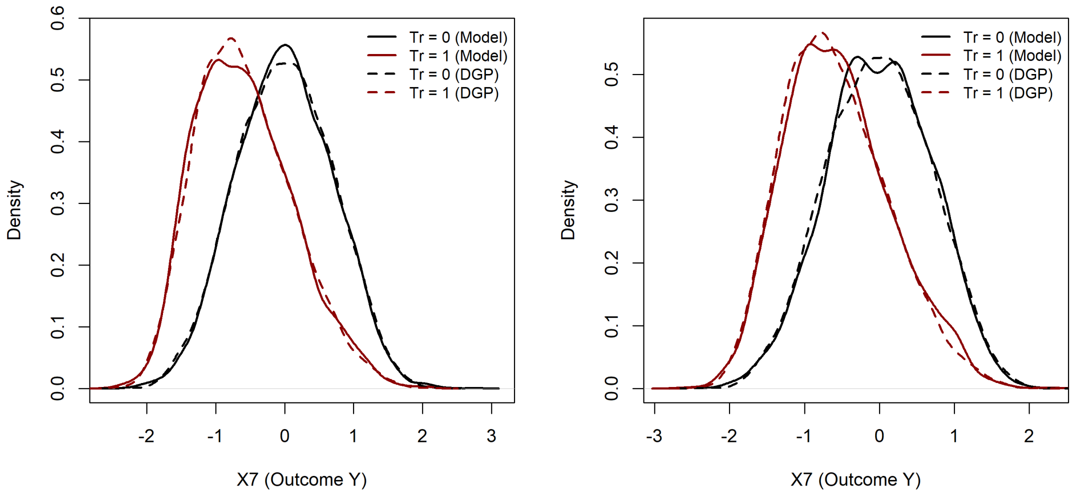


Figure 3.22: Distributions of the outcome variable (X_7) under control and treatment interventions for Scenario (1), which includes both direct and interaction effects. This plot provides a more detailed view of the X_7 panels shown under do(X_4 = 0) and do($X_4 = 1$) in Figure 3.21. Left: Observational; Right: RCT setting.

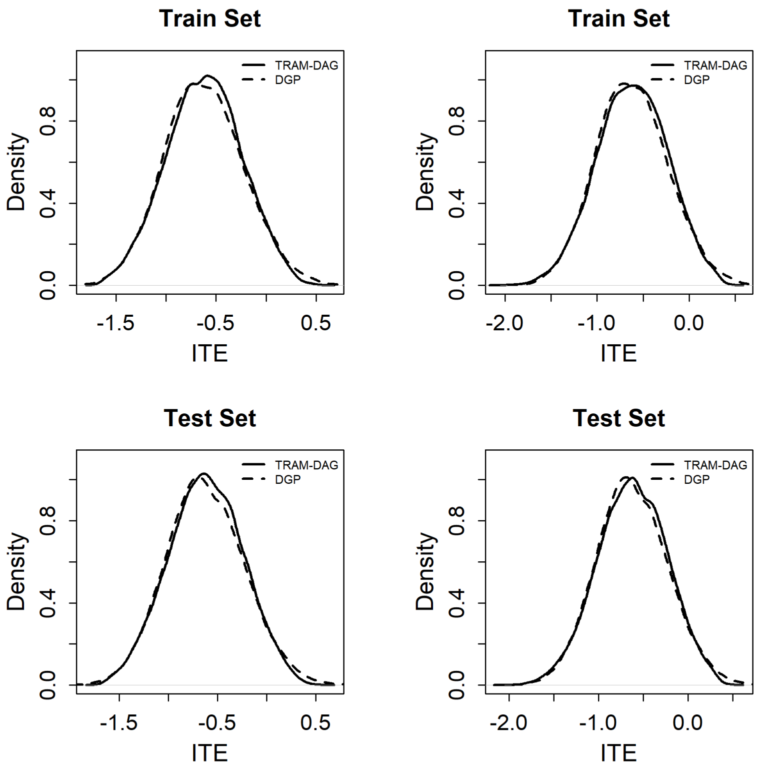


Figure 3.23: Densities of the estimated ITEs compared to the true ITEs in the training and test datasets for Scenario (1), which includes direct and interaction effects. Left: Observational; Right: RCT setting.

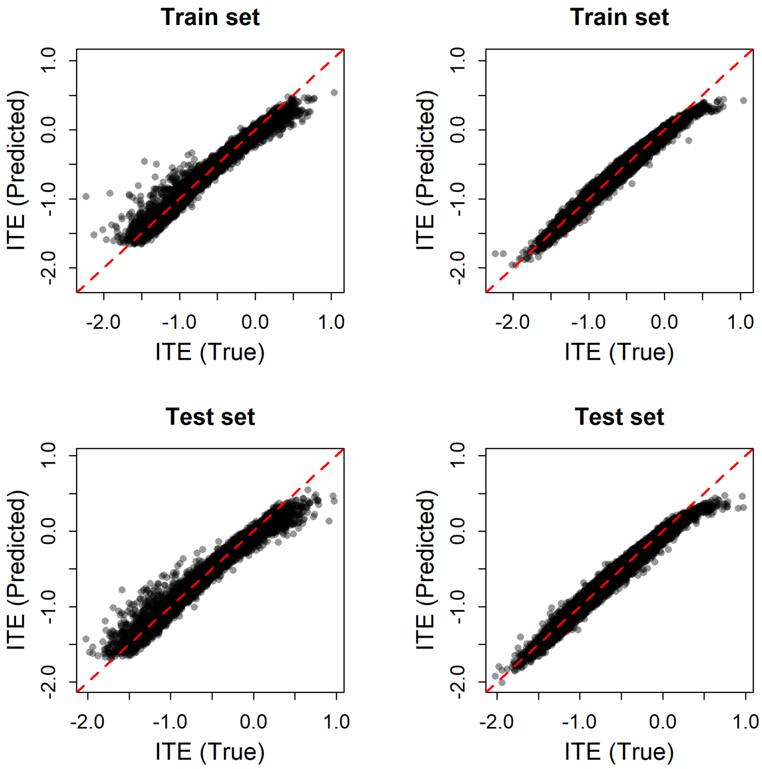


Figure 3.24: Scatterplots of estimated ITEs vs. true ITEs in the training and test datasets for Scenario (1), which includes direct and interaction effects. Left: Observational; Right: RCT setting.

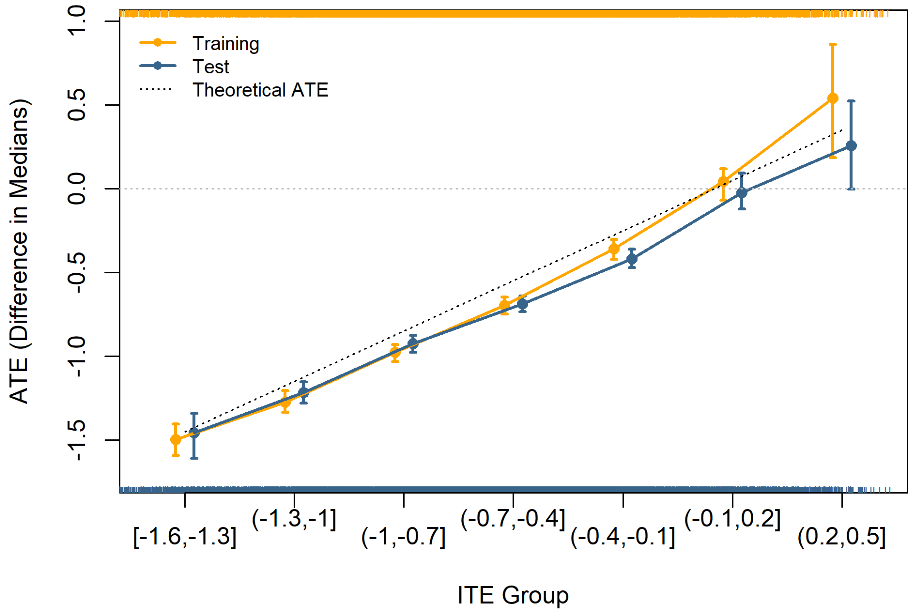


Figure 3.25: ITE-ATE plot for Scenario (1) in the observatinal setting, which includes direct and interaction effects. Individuals are grouped into bins based on their estimated ITEs, and within each bin, the ATE is calculated as the difference in medians of the observed outcomes under the two treatments. 95% bootstrap confidence intervals indicate the uncertainty.

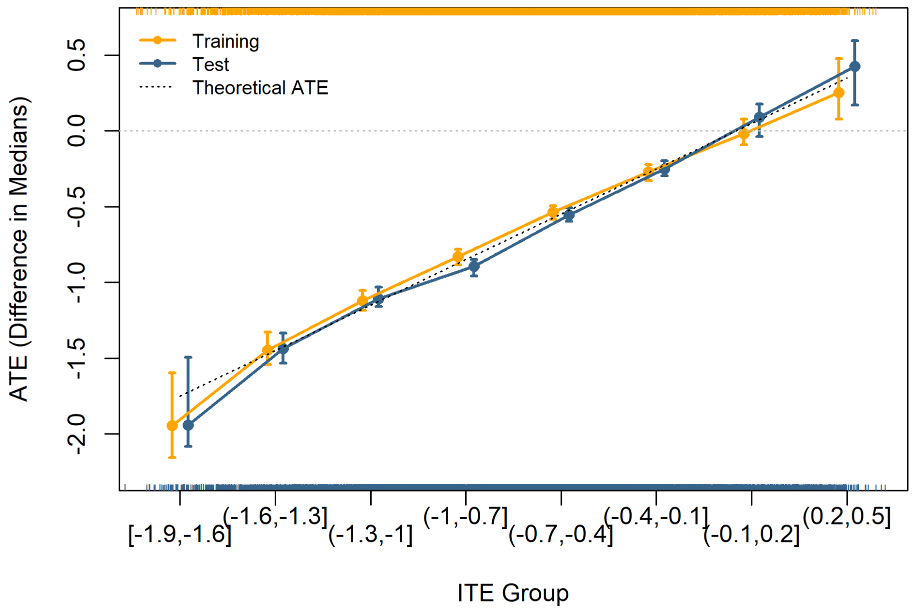


Figure 3.26: ITE-ATE plot for Scenario (1) in the RCT setting, which includes direct and interaction effects. Individuals are grouped into bins based on their estimated ITEs, and within each bin, the ATE is calculated as the difference in medians of the observed outcomes under the two treatments. 95% bootstrap confidence intervals indicate the uncertainty.

3.4.2 Scenario (2): With direct effect, but no interaction effects

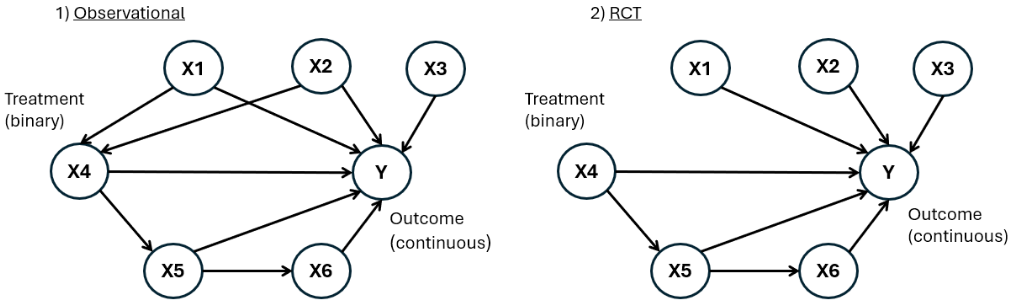


Figure 3.27: DAGs for Scenario (2), which includes a direct effect of the treatment on the outcome, but no interaction effects that would induce treatment effect heterogeneity. Left: Observational setting; Right: RCT setting.

Scenario (2) includes a direct effect of the treatment on the outcome, while the coefficients for the interaction terms are set to zero. This results in less heterogeneity in the ITE distribution compared to Scenario (1), as shown in Figure 3.28. Why there is still some heterogeneity despite the absence of interactions is discussed in Section 4.4. The observational and interventional densities generated by the fitted TRAM-DAG closely match the true densities defined by the DGP, as illustrated in Figures 3.29 and 3.30. However, there is a notable difference in variance between the estimated and true ITE distributions, visible in Figures 3.31 and 3.32. The ITE-ATE plots in Figures 3.33 and 3.34 are less informative than in Scenario (1), as expected given the reduced heterogeneity. Table 3.2 presents the ATE measures for Scenario (2). In the test set of the RCT setting, the ATE based on the true ITEs was -0.633, while the ATE based on the estimated ITEs was -0.586.

Table 3.2: Scenario (2), including a direct treatment effect but no interaction effects: Comparison of ATE measures across train and test sets for the observational and RCT setting. $\mathrm{Y}^{(\mathrm{Tr})}_{\mathrm{observed}}$ denotes the observed outcome under the treatment (Tr) actually received. The estimated ATE from mean($\mathrm{ITE}_{\mathrm{estimated}}$) can be directly compared to the true mean($\mathrm{ITE}_{\mathrm{true}}$), whereas comparisons to the empirical ATEs based on outcome differences should be interpreted with caution.

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

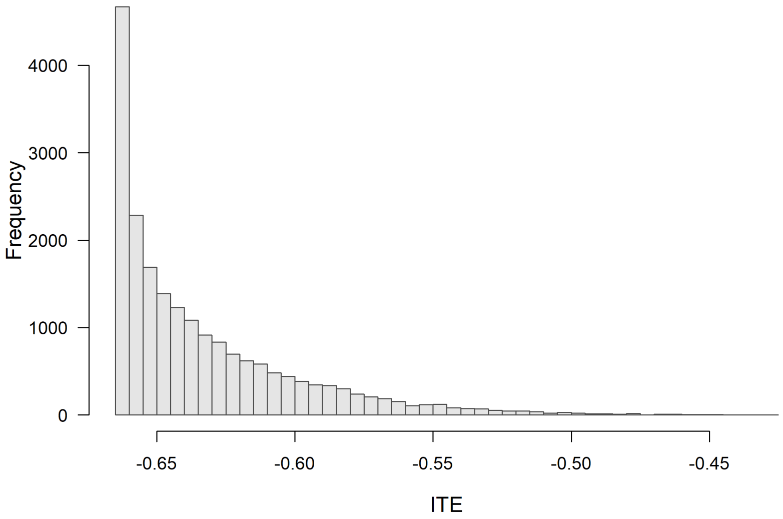


Figure 3.28: True ITE distribution resulting from the DGP for Scenario (2), which includes a direct treatment effect but no interaction effects. The true ITEs are identical in the observational and RCT settings, as they depend only on the potential outcomes under both treatment allocations.

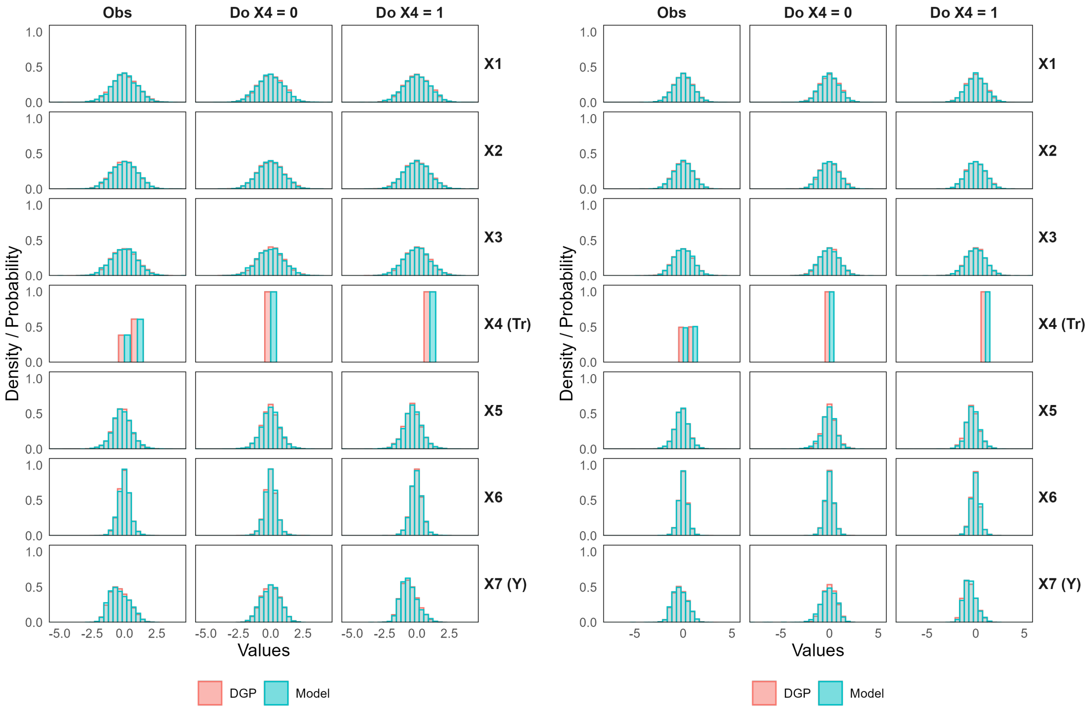


Figure 3.29: Marginal distributions of variables from the DGP and samples generated by the fitted TRAM-DAG for Scenario (2), which includes a direct treatment effect but no interaction effects. The distributions are shown as observed (Obs), under control intervention ($\text{do}(X_4 = 0)$), and under treatment intervention ($\text{do}(X_4 = 1)$). Left: Observational; Right: RCT setting.

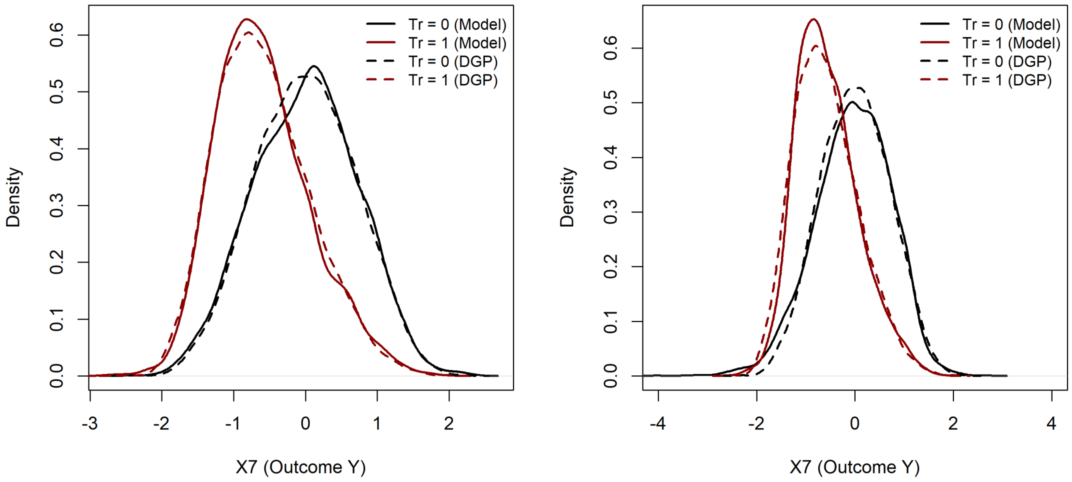


Figure 3.30: Distributions of the outcome variable (X_7) under treatment and control interventions for Scenario (2), which includes a direct treatment effect but no interaction effects. This plot provides a higher-resolution view of the X_7 panels under do($X_4 = 0$) and do($X_4 = 1$) from Figure 3.29. Left: Observational; Right: RCT setting.

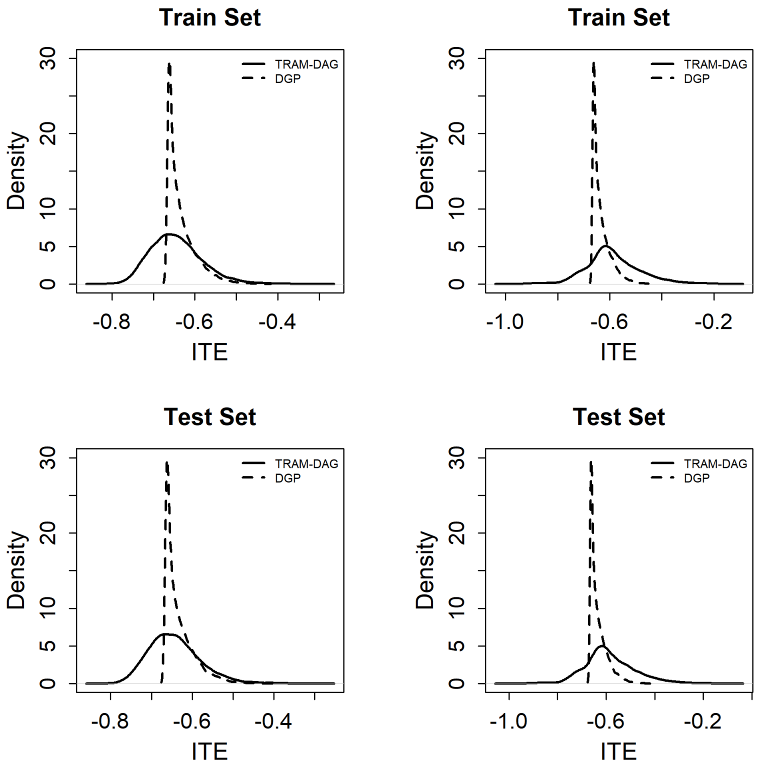


Figure 3.31: Densities of estimated ITEs compared to the true ITEs in the training and test datasets for Scenario (2), which includes a direct treatment effect but no interaction effects. Left: Observational; Right: RCT setting.

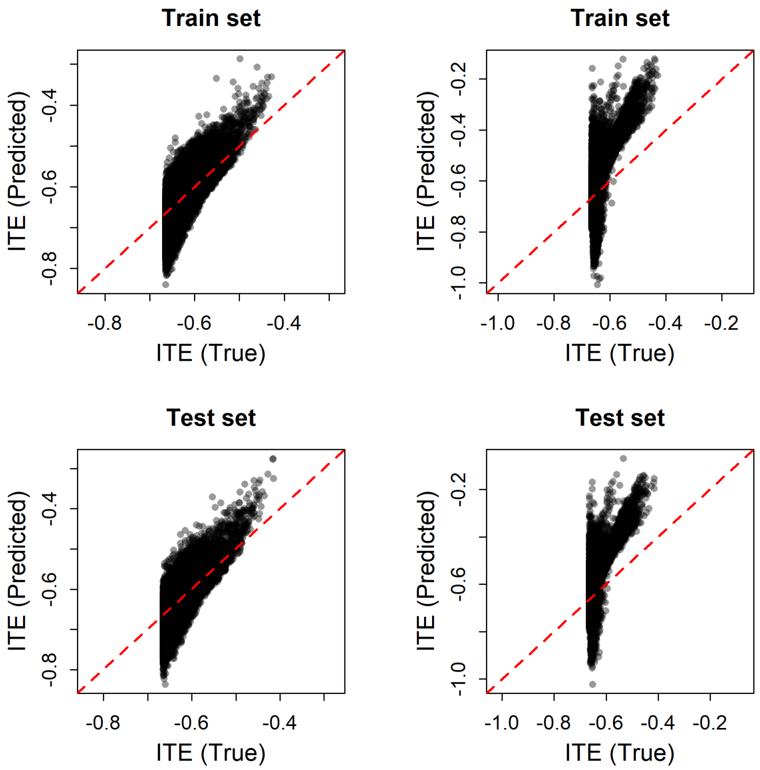


Figure 3.32: Scatterplots of estimated ITEs compared to the true ITEs in the training and test datasets for Scenario (2), which includes a direct treatment effect but no interaction effects. Left: Observational; Right: RCT setting.

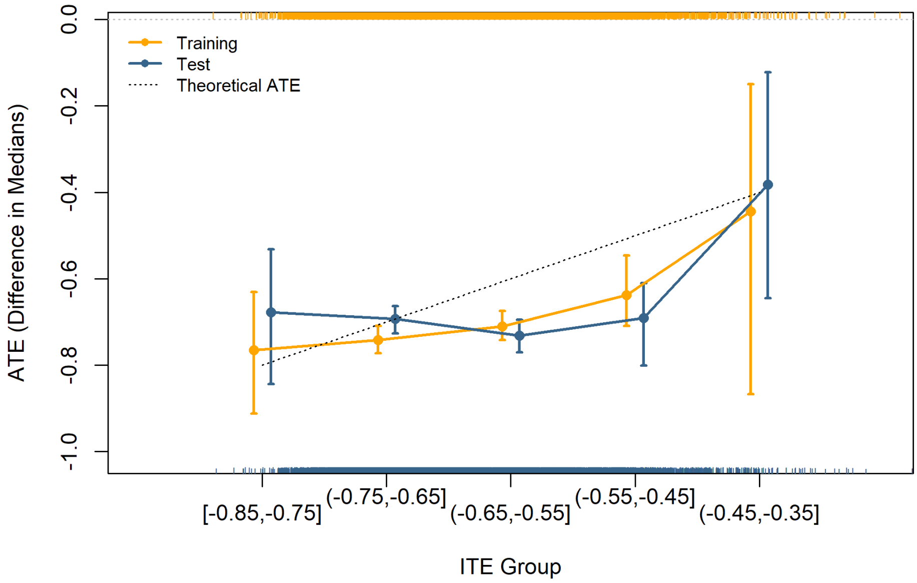


Figure 3.33: ITE-ATE plot for Scenario (2) in the observational setting, which includes a direct treatment effect but no interaction effects. Individuals are grouped into bins based on their estimated ITEs, and in each bin, the ATE is computed as the difference in medians of the observed outcomes under treatment and control. The 95% bootstrap confidence intervals indicate uncertainty.

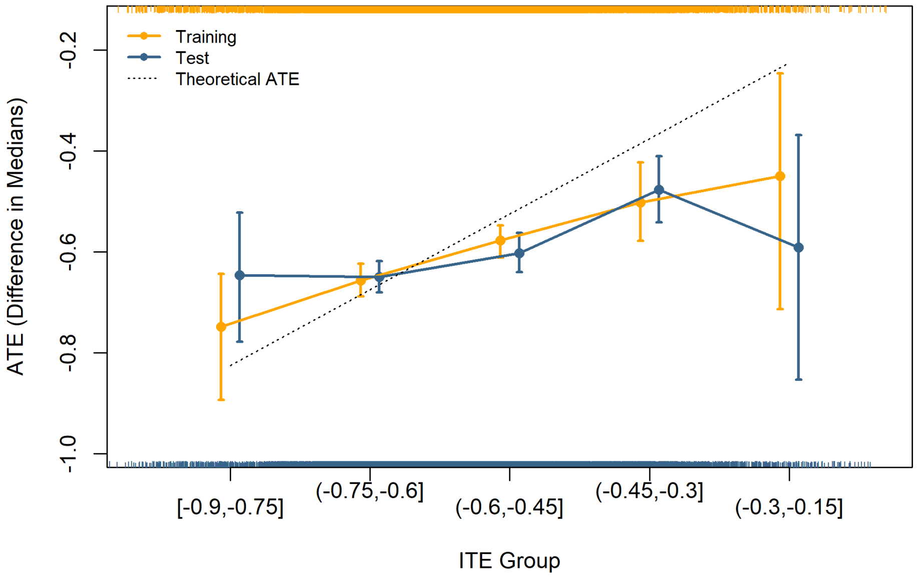


Figure 3.34: ITE-ATE plot for Scenario (2) in the RCT setting, which includes a direct treatment effect but no interaction effects. Individuals are grouped into bins based on their estimated ITEs, and in each bin, the ATE is computed as the difference in medians of the observed outcomes under treatment and control. The 95% bootstrap confidence intervals indicate uncertainty.

3.4.3 Scenario (3): No direct effect, but with interaction effects

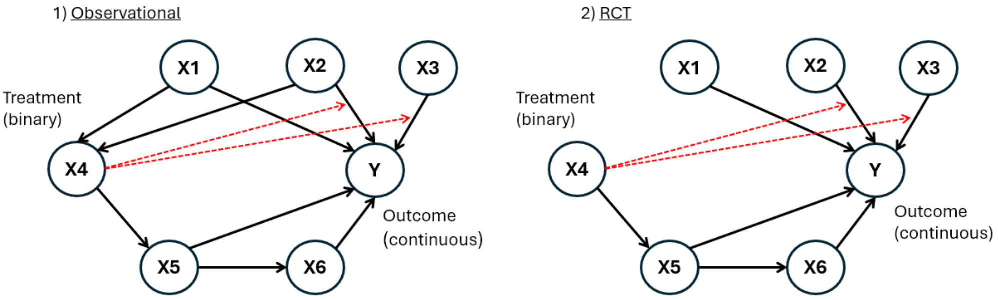


Figure 3.35: DAGs for Scenario (3), which includes no direct effect of the treatment on the outcome, but interaction effects with covariates X_2 and X_3. Left: Observational setting; Right: RCT setting.

Scenario (3) includes no direct effect of the treatment on the outcome, but does include interaction effects between the treatment and covariates X_2 and X_3. Compared to Scenario (1), removing the direct effect results in a more centered ITE distribution, as shown in Figure 3.36. In the test set of the RCT setting, the ATE measured as the difference in means was -0.048, with a 95% confidence interval from -0.068 to -0.028.

Table 3.3: Scenario (3), without a direct treatment effect but including interaction effects: Comparison of ATE measures across train and test sets for the observational and RCT setting. $\mathrm{Y}^{(\mathrm{Tr})}_{\mathrm{observed}}$ denotes the observed outcome under the treatment (Tr) actually received. The estimated ATE from mean($\mathrm{ITE}_{\mathrm{estimated}}$) can be directly compared to the true mean($\mathrm{ITE}_{\mathrm{true}}$), whereas comparisons to the empirical ATEs based on outcome differences should be interpreted with caution.

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

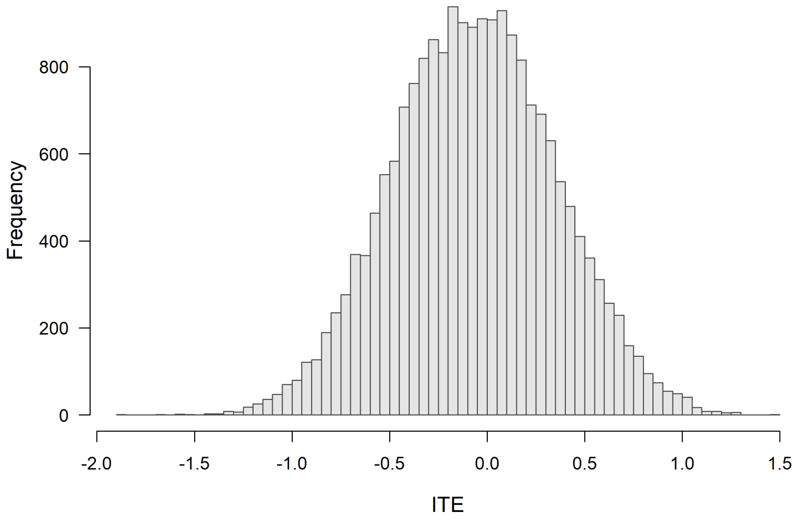


Figure 3.36: True ITE distribution resulting from the DGP for Scenario (3), which includes interaction effects but no direct treatment effect. The true ITEs are identical in the observational and RCT settings, since they are based on the potential outcomes under both treatment allocations. Left: Observational; Right: RCT setting.

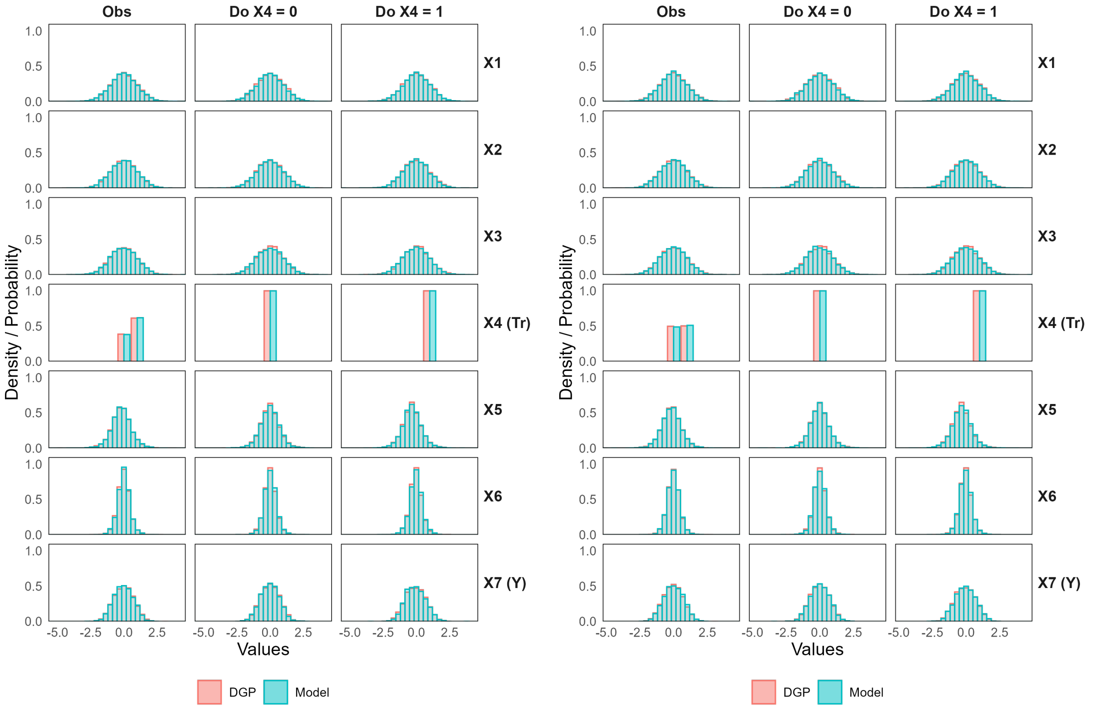


Figure 3.37: Marginal distributions of DGP variables and samples generated by the fitted TRAM-DAG for Scenario (3), which includes interaction effects but no direct treatment effect. The distributions are shown as observed (Obs), under control intervention ($\text{do}(X_4 = 0)$), and under treatment intervention ($\text{do}(X_4 = 1)$). Left: Observational; Right: RCT setting.

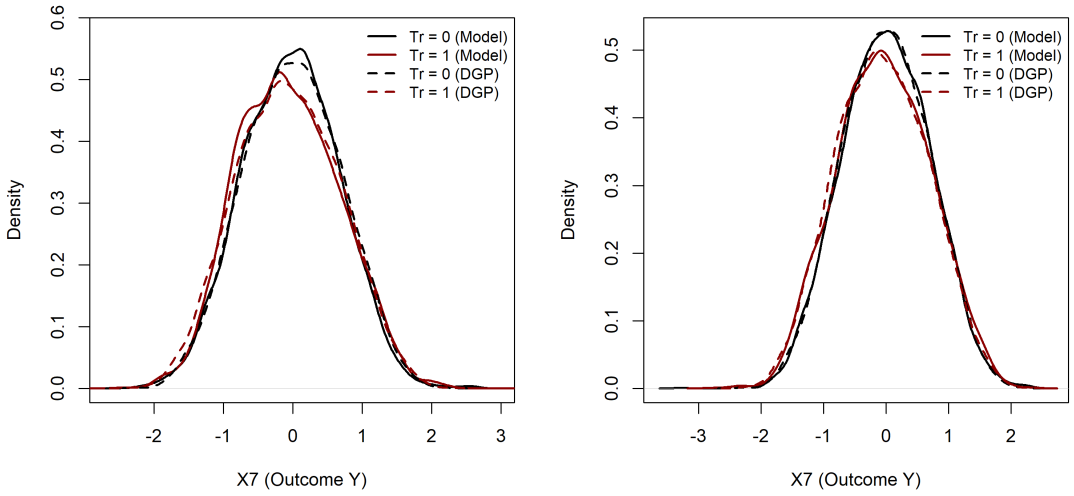


Figure 3.38: Distributions of the outcome variable (X_7) under treatment and control interventions for Scenario (3), which includes interaction effects but no direct treatment effect. This plot provides a higher resolution view of the X_7 panels under $\mathrm{do}(X_4 = 0)$ and $\mathrm{do}(X_4 = 1)$ from Figure 3.37. Left: Observational; Right: RCT setting.

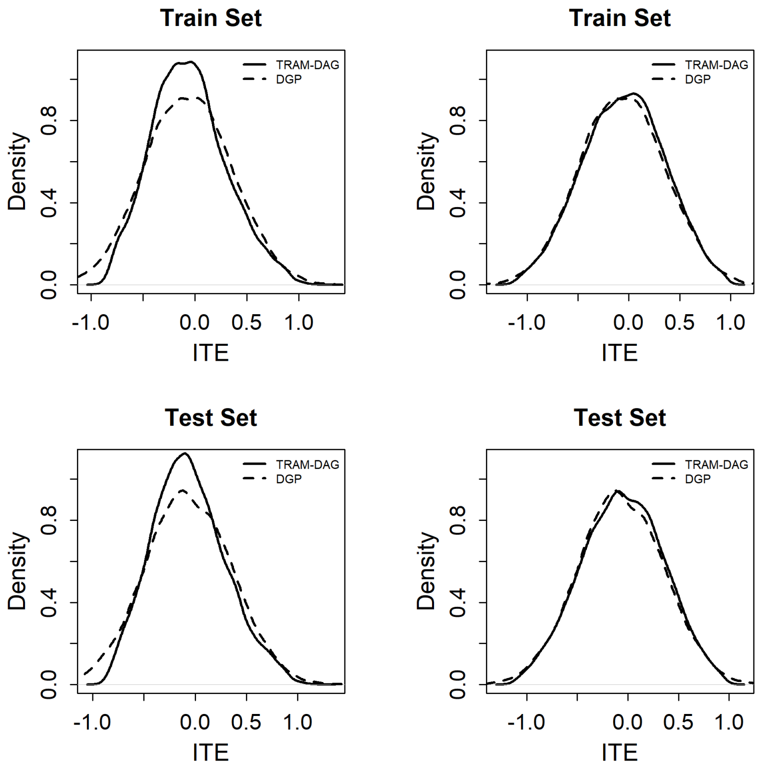


Figure 3.39: Densities of estimated ITEs compared to the true ITEs in the training and test datasets for Scenario (3), which includes interaction effects but no direct treatment effect. Left: Observational; Right: RCT setting.

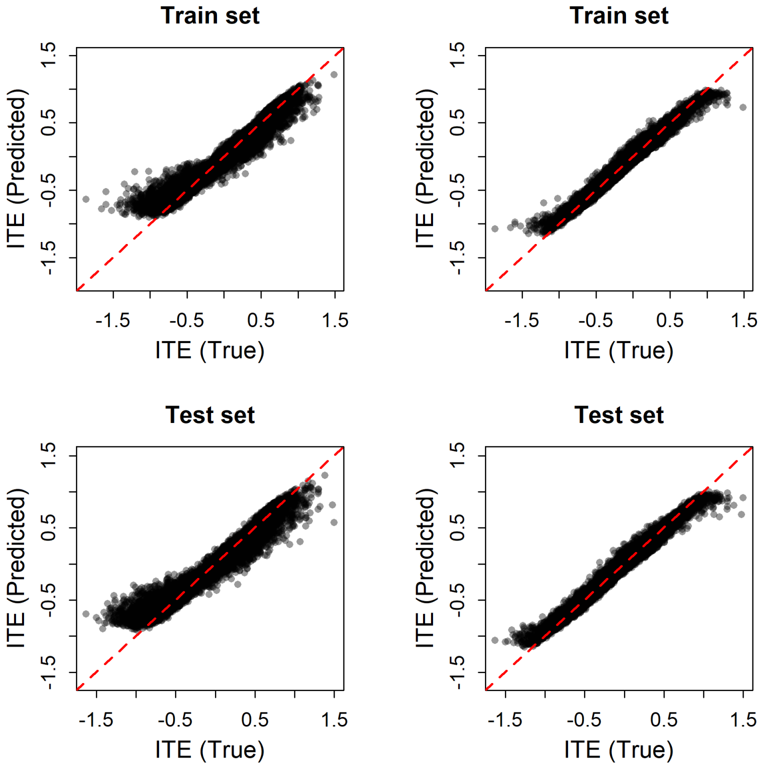


Figure 3.40: Scatterplots of estimated ITEs compared to the true ITEs in the training and test datasets for Scenario (3), which includes interaction effects but no direct treatment effect. Left: Observational; Right: RCT setting.

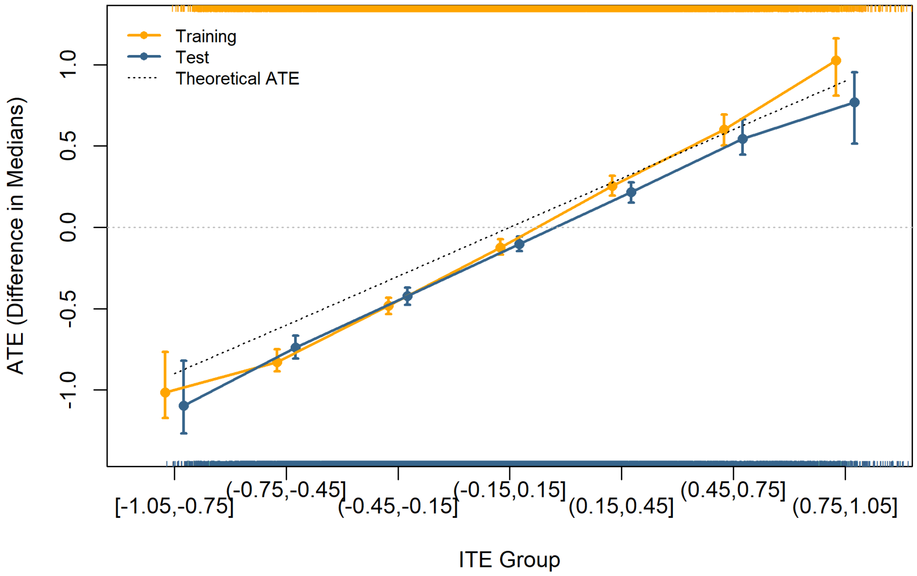


Figure 3.41: ITE-ATE plot for Scenario (3) in the observational setting, which includes interaction effects but no direct treatment effect. Individuals are grouped into bins based on the estimated ITE, and within each bin the ATE is computed as the difference in medians of the observed outcomes under treatment and control. 95% bootstrap confidence intervals reflect the uncertainty.

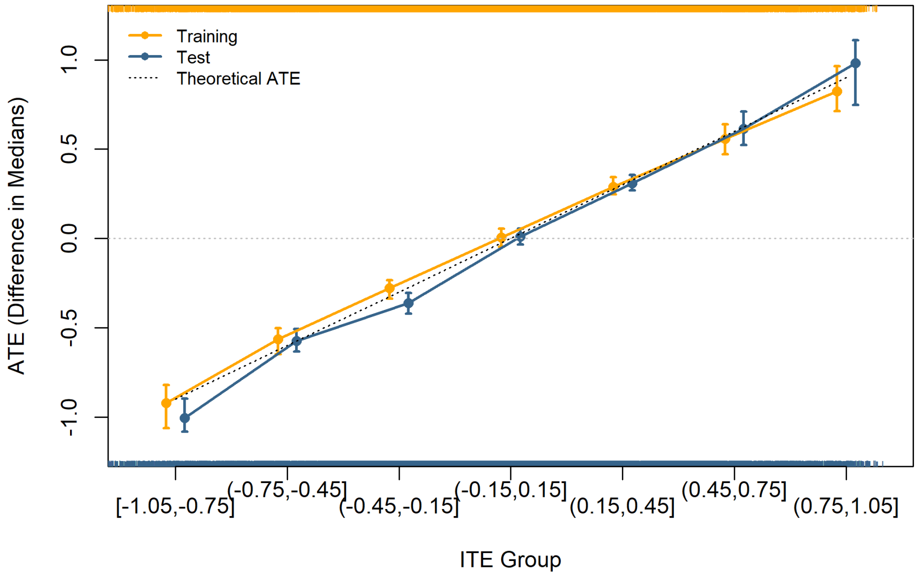


Figure 3.42: ITE-ATE plot for Scenario (3) in the RCT setting, which includes interaction effects but no direct treatment effect. Individuals are grouped into bins based on the estimated ITE, and within each bin the ATE is computed as the difference in medians of the observed outcomes under treatment and control. 95% bootstrap confidence intervals reflect the uncertainty.

Chapter 4

Discussion

4.1 Experiment 1: TRAM-DAG (simulation)

The results demonstrate that the TRAM-DAG framework can learn the true parameters and both linear and complex shifts from the data, enabling it to act as a generative model for predicting interventions and counterfactuals. It successfully reproduced observational and interventional distributions and predicted correct counterfactual outcomes.

This experiment serves as a small proof of concept that TRAM-DAGs can be specified flexibly, with both interpretable and complex components, to capture causal relationships of varying complexity when the true DAG is known and the data is generated accordingly.

4.2 Experiment 2: ITE on International Stroke Trial (IST)

We observed similar results to those reported by Chen *et al.* (2025) when estimating ITEs on the International Stroke Trial dataset across all three models: the T-learner logistic regression, the T-learner tuned random forest, and the S-learner TRAM-DAG. The logistic model showed moderate discrimination in the training set, which did not generalize to the test set, as illustrated by the ITE-ATE plot in Figure 3.7. The tuned random forest model showed stronger discrimination in the training set but similarly failed to generalize to the test set (Figure 3.8). In contrast, the S-learner TRAM-DAG estimated less heterogeneity than the other two models, as shown in the density plot in Figure 3.9, resulting in weak discrimination in both the training and test sets. For all three models, the confidence intervals in the ITE-ATE plots on the test set included the zero line, suggesting no significant effect in any of the estimated ITE subgroups.

Poor calibration does not appear to explain the limited ITE performance, as calibration on the test set was good, as shown in Appendix 6.7, Figures 6.5-6.7. However, since the ground truth is unknown, it remains unclear whether the models fail to capture true treatment effect heterogeneity, or whether the underlying heterogeneity is too small, or driven by unobserved effect modifiers. We explore this further in Experiment 3 (ITE Simulation Study; see Sections 2.5, 3.3) and 4.3).

4.3 Experiment 3: ITE model robustness in RCTs (simulation)

In Scenario 1, where treatment effect heterogeneity was large and all covariates were observed, the T-learner logistic regression accurately estimated the ITE. The observed ATE, conditional on the respective ITE subgroup, was well calibrated in both the training and test datasets, as shown in the ITE-ATE plot in Figure 3.11. This is as expected, since the data were generated with the same model class (logistic regression), and applying logistic regression as a T-learner for ITE estimation can accurately capture the interaction effects.

The tuned random forest model also performed well. As illustrated in Figure 3.12, choosing a different model class than that used in the DGP may lead to worse prediction accuracy in terms of $\mathrm{P}(Y = 1 \mid X, T)$ and ITE. This difference between the two models arises because the logistic regression model has only a small number of parameters, and with sufficient data, these parameters can converge to their true values as used in the logistic DGP, allowing near-perfect recovery of the true probabilities and thus ITEs. In contrast, the non-parametric random forest must infer the underlying probabilities from the observed binary outcomes (0 or 1), which are themselves realizations of a Bernoulli process. This introduces inherent noise, making it harder for the model to estimate the true risk accurately – even with large sample sizes. Nonetheless, the tuned random forest also captured the general trend of the ITEs, as reflected in the ITE-ATE plot. Both models were able to capture treatment effect heterogeneity well under full observability of covariates.

In Scenario 2, where treatment effect heterogeneity remained large but one important interaction covariate (X_1) was not observed, prediction accuracy decreased for both models, and the estimated heterogeneity in terms of the ITE was smaller than the true heterogeneity. Although not all heterogeneity could be recovered, the T-learner logistic regression still estimated the ITEs in the correct direction. As shown in Figure 3.14, the confidence intervals for the ATE per ITE subgroup covered the calibration line. This indicates that individuals estimated to have a smaller ITE indeed experienced worse outcomes under treatment compared to untreated individuals in the same subgroup. Although a considerable number of individuals had a true ITE that was positive, the T-learner logistic regression did not predict a single positive ITE. This shows that the missing covariate X_1 prevents detection of individuals who would actually benefit from the treatment.

In contrast, the T-learner tuned random forest estimated larger treatment effect heterogeneity than the logistic model, but still could not accurately estimate the ITE and also failed to detect patients who would benefit from the treatment. The ITE-ATE plot in Figure 3.15 illustrates that the model discriminates too strongly in the training set and does not generalize well to the test set.

In Scenario 3, where the true treatment effect heterogeneity was small and all covariates were observed, the T-learner logistic regression estimated a larger heterogeneity than actually present. In the ITE-ATE plot in Figure 3.17, the confidence intervals of all ITE subgroups overlap and include the zero line, indicating that the treatment effect is not significantly different from zero. This matches expectations given the small true effect sizes.

However, the T-learner tuned random forest model incorrectly estimated even larger treatment effect heterogeneity than the logistic regression model. As shown in Figure 3.18, the model exhibited strong discrimination in the training set but did not to replicate this pattern in the test set, where – regardless of the estimated ITE – the observed outcomes in the subgroups were similar.

Tuning more flexible models like random forests using cross-validation improved generalization to the test set but led to poor calibration in terms of predicted probabilities vs. empirically observed outcomes in the training set. An illustrative case is shown in Appendix 6.9 for the T-learner tuned random forest in Scenario 3 (with weak effects), where calibration was poor in the training set but aligned well with the identity line in the test set. We repeatedly observed this pattern in the tuned random forest when, in the ITE-ATE plot, results from the training set did not generalize to the test set. This highlights the importance of evaluating models on an independent test set, when tuning a model to prevent overfitting. Although, evaluation on a test set should be done in any case.

In this experiment, we showed that even when causal ML models for ITE estimation are well calibrated in terms of prediction accuracy $\mathrm{P}(Y = 1 \mid \mathbf{X}, T)$, they can still fail to estimate the ITE accurately under less favorable scenarios. In cases of full observability of covariates but low interaction effects, models may estimate too high heterogeneity that is not present in the data. However, this can become visible in the ITE-ATE plot on the test set, which reveals that the apparent heterogeneity does not generalize. But we also observed that when important effect-modifying covariates are missing, the models may fail to detect treatment effect heterogeneity altogether, as shown in Scenario 2. In such cases, the estimated ITEs may be too small or even negative, suggesting that the model does not capture the true treatment effect heterogeneity. This makes it difficult to distinguish between a true lack of heterogeneity and the failure to capture it due to unobserved effect modifiers.

Vegetabile (2021) also analyzed the effect of unobserved interaction variables. He pointed out that as long as all confounding variables $\mathbf{X}$ are observed and conditioned on, the ignorability assumption required for ITE estimation is satisfied – even in the presence of an unobserved interaction variable Z. However, if such a variable Z exists, the estimated ITEs would be biased, and this issue could arise even in an RCT setting where confounding is removed through randomization.

Nichols (2007) discusses various methods for estimating causal effects from observational data, including in the presence of unobserved variables. One of these methods, instrumental variables (IV), can help reduce bias from unobserved confounding. Whether IV methods can also address unobserved effect modifiers in the context of ITE estimation is not something we explored, and remains beyond the scope of this thesis.

4.4 Experiment 4: ITE estimation with TRAM-DAGs (simulation)

We analyzed ITE estimation under an observational setting (confounded) and under an RCT setting (randomized treatment allocation) in three different scenarios: direct and interaction treatment effect, only direct but no interaction effect, and no direct but with interaction effect. The TRAM-DAG could successfully estimate the ITE in Scenario 1 and Scenario 3 where interaction effects were present. There was no notable difference between the observational and RCT settings. Scatterplots of estimated ITEs vs. true ITEs showed good prediction accuracy (see Figures 3.24 and 3.40). Also the ATE based on the mean of estimated ITEs was close to the ATE based on true ITEs in both scenarios (see Tables 3.1 and 3.3). These results highlight TRAM-DAG's ability to compute counterfactuals for mediators and to estimate individualized treatment effects even in relatively complex DAG structures.

In Scenario 2, where no interaction effects were present, ITE estimation was poor. This aligns with our discovery in Experiment 3 that when true heterogeneity is weak, models tended to estimate too large heterogeneity, as e.g. shown in Figure 3.18 with the T-learner tuned random forest.

What might be surprising in Scenario 2 is the presence of heterogeneity (true ITEs), despite the absence of explicitly specified interaction terms in the data-generating process. As shown in Figure 3.28, one might have expected the ITEs to be constant across individuals – equal to the ATE – given the model's additivity on the log-odds scale. However, as described by Hoogland *et al.* (2021), such heterogeneity arises because a constant treatment effect on the log-odds scale does not translate into a constant effect on a different scale, such as the probability scale. This phenomenon results from the nonlinearity of the inverse-link function (e.g., logit^{-1}), which transforms additive effects in the linear predictor into non-additive effects on the outcome scale. As the authors point out, the same shift induced by the treatment on the log-odds scale leads to different absolute risk reductions depending on the outcome risk under the control treatment. In other words, even with a homogeneous effect on the linear predictor, variation in covariates $\mathbf{X}$ leads to different treatment effects on the probability scale.

This would not have occurred under a linear model where the transformation function h is the identity. In that case, the ITE would simplify as follows:

$$\text{ITE} = \mathbb{E}[Y(1)] - \mathbb{E}[Y(0)] = (\beta_0 + \beta_t + \boldsymbol{\beta}_x^\top \mathbf{X} + \epsilon) - (\beta_0 + \boldsymbol{\beta}_x^\top \mathbf{X} + \epsilon) = \beta_t \tag{4.1}$$

Here, the ITE is constant and equal to the treatment coefficient, independent of the covariates or the noise term, which cancels out.

In contrast, under a nonlinear model, such as the logistic transformation model with a nonlinear intercept function used in this experiment, the ITE becomes:

$$\text{ITE} = \mathbb{E}[h^{-1}(Z + \beta_t + \boldsymbol{\beta}_x^\top \mathbf{X})] - \mathbb{E}[h^{-1}(Z + \boldsymbol{\beta}_x^\top \mathbf{X})] \tag{4.2}$$

Since h^{-1} is nonlinear, the difference depends on the covariate profile $\mathbf{X}$ and on the noise term Z, even though the treatment effect β_t is additive in the linear predictor, i.e., on the log-odds scale. It may therefore be worth thinking about whether analyzing the ITE on a scale where the effect is constant offers any advantages.

Maybe also related to this phenomenon, although not directly in the context of ITEs, is the concept of noncollapsibility, as discussed by Dandl and Hothorn (2025). Noncollapsibility refers to the case when the treatment effect estimated from a marginal model (i.e., without covariates) does not correspond to the marginal effect that is obtained by averaging conditional treatment effects (i.e., adjusted for prognostic covariates) over the covariate distribution Aalen *et al.* (2015). Hence, treatment effects from two conditional models that use different sets of covariates for adjustment are not directly comparable if the model is noncollapsible. Dandl and Hothorn (2025) proposed a solution based on nonparanormal models (Liu *et al.*, 2009; Klein *et al.*, 2022) to estimate a marginal treatment effect, while maintaining comparability (unaffected by covariates) and gaining from increased precision by adjusting for prognostic factors. Whether and how such an approach could be applied to ITE estimation is not explored further in this thesis.

Chapter 5

Conclusions

In this thesis, we further investigated the application of TRAM-DAGs as a flexible approach to estimate structural equations in a known DAG. We explained how to incorporate ordinal predictors, how to model interactions, and what the scaling of variables implies for interpretability. Furthermore, we explored the estimation of individualized treatment effects (ITEs), showing that TRAM-DAGs can also be applied to estimate ITEs in relatively complex DAG structures. In simulation experiments, we examined potential limitations and challenges in ITE estimation.

Our findings included that TRAM-DAGs were able to successfully recover structural equations when the DAG was fully known and all variables were observed. They also worked well for ITE estimation in simulation settings. The simulation experiments further revealed limitations in ITE estimation, especially in the presence of unobserved effect modifiers. We concluded that unmeasured effect-modifying variables pose a significant challenge and that the ignorability assumption alone may not be enough to ensure unbiased estimates. This or weak treatment effect heterogeneity might explain why ITE estimation failed in the real-world application on the International Stroke Trial. We also found that proper calibration of causal machine learning models is important to achieve accurate ITE estimates but that calibration alone may not be sufficient for valid predictions.

TRAM-DAGs offer several advantages. The model inherently knows when to adjust for covariates based on the DAG structure and the learned functions. Structural causal models, in contrast to classical regression approaches, account for all known relationships and can consistently address confounding. Classical regression models risk adjusting for the wrong covariates, which may lead to biased estimates. TRAM-DAGs are generative causal models that, once fitted to a correct DAG, allow sampling from observational, interventional, and counterfactual distributions. Their ability to combine flexible components with interpretable structure makes them well suited for practical use cases where both predictive power and transparency matter.

However, there are also some limitations. While we aimed to make the simulation scenarios as realistic as possible while still retaining some interpretability, they may not fully reflect the complexity of real-world data. However, applying and evaluating models like TRAM-DAGs on real data for causal questions such as ITE estimation is inherently difficult, as the true effects are usually unknown. TRAM-DAGs also rely on neural networks, which require time to train, depending on network complexity, sample size, and computational resources. And although TRAM-DAGs offer flexibility, we still need to make assumptions – for example, about the scale on which conditional effects occur – if we want to retain some level of interpretability.

Future work could apply TRAM-DAGs to other real-world datasets, potentially also including semi-structured data, to fully exploit the potential of their modular neural network structure. It would also be valuable to further investigate ITE estimation in the presence of unmeasured interaction variables.

Overall, this thesis contributes to the growing field of causal inference, especially in observational data and personalized interventions. We hope to have provided some insights into the capabilities of neural causal models and the challenges of ITE estimation.

Bibliography

Aalen, O., Cook, R., and Røysland, K. (2015). Does Cox analysis of a randomized survival study yield a causal treatment effect? *Lifetime Data Analysis*, **21**, 579–593. 56

Allaire, J. and Tang, Y. (2025). *tensorflow: R Interface to 'TensorFlow'*. R package version 2.16.9. 24

Breiman, L. (2001). Random Forests. *Machine Learning*, **45**, 5–32. 20

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. 15

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. arXiv preprint 2501.04061. iii, v, 4, 18, 19, 53

Chernozhukov, V. and Hansen, C. (2005). An IV model of quantile treatment effects. *Econometrica*, **73**, 245–261. 22

Chollet, F., Allaire, J.,*et al.* (2017). R Interface to 'Keras'. R package version 2.15.0. 24

Christensen, R., Bours, M. J., and Nielsen, S. M. (2021). Effect modifiers and statistical tests for interaction in randomized trials. *Journal of Clinical Epidemiology*, **134**, 174–177. 14

Dandl, S. and Hothorn, T. (2025). Nonparanormal adjusted marginal inference. arXiv preprint 2503.01657. 56

Dawid, A. P. (2000). Causal inference without counterfactuals. *Journal of the American Statistical Association*, **95**, 407–424. 2

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

Friedman, J. H., Hastie, T., and Tibshirani, R. (2010). Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software*, **33**, 1–22. 20

Glorot, X., Bordes, A., and Bengio, Y. (2011). Deep sparse rectifier neural networks. In *Proceedings of the 14th International Conference on Artificial Intelligence and Statistics*, volume 15, 315–323. Proceedings of Machine Learning Research. 10

Guo, C., Pleiss, G., Sun, Y., and Weinberger, K. Q. (2017). On calibration of modern neural networks. In *Proceedings of the 34th International Conference on Machine Learning (ICML)*, volume 70, 1321–1330. Proceedings of Machine Learning Research. 15

Gutierrez, P. and Gérardy, J.-Y. (2017). Causal inference and uplift modelling: A review of the literature. In *Proceedings of The 3rd International Conference on Predictive Applications and APIs*, volume 67, 1–13. Proceedings of Machine Learning Research. 2

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

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. 10

Holland, P. W. (1986). Statistics and causal inference. *Journal of the American Statistical Association*, **81**, 945–960. 2, 14

Hoogland, J., Efthimiou, O., Nguyen, T. L., and Debray, T. P. A. (2024). Evaluating individualized treatment effect predictions: A model-based perspective on discrimination and calibration assessment. *Statistics in Medicine*, **43**, 4481–4498. 16

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. 14, 15, 19, 55

Hothorn, T., Kneib, T., and Bühlmann, P. (2014). Conditional transformation models. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, **76**, 3–27. 4, 7

Hothorn, T., Möst, L., and Bühlmann, P. (2018). Most likely transformations. *Scandinavian Journal of Statistics*, **45**, 110–134. 10, 25

International Stroke Trial Collaborative Group (1997). The International Stroke Trial (IST): a randomised trial of aspirin, subcutaneous heparin, both, or neither among 19,435 patients with acute ischaemic stroke. *The Lancet*, **349**, 1569–1581. 18

Ioffe, S. and Szegedy, C. (2015). Batch Normalization: Accelerating deep network training by reducing internal covariate shift. In *Proceedings of the 32nd International Conference on Machine Learning (ICML)*, volume 37, 448–456. Proceedings of Machine Learning Research. 10

Kingma, D. P. and Ba, J. (2015). Adam: A method for stochastic optimization. In *Proceedings of the 3rd International Conference on Learning Representations (ICLR)*. 10, 17

Klein, N., Hothorn, T., Barbanti, L., and Kneib, T. (2022). Multivariate conditional transformation models. *Scandinavian Journal of Statistics*, **49**, 116–142. 56

Kook, L. (2024). *comets: Covariance Measure Tests for Conditional Independence*. R package version 0.1-1. 18, 20

Künzel, S. R., Sekhon, J. S., Bickel, P. J., and Yu, B. (2019). Metalearners for estimating heterogeneous treatment effects using machine learning. *Proceedings of the National Academy of Sciences*, **116**, 4156–4165. 15

Little, R. J. and Rubin, D. B. (2000). Causal effects in clinical and epidemiological studies via potential outcomes: Concepts and analytical approaches. *Annual Review of Public Health*, **21**, 121–145. 14

Liu, H., Lafferty, J., and Wasserman, L. (2009). The nonparanormal: Semiparametric estimation of high dimensional undirected graphs. *Journal of Machine Learning Research*, **10**, 2295–2328. 56

Nichols, A. (2007). Causal inference with observational data. *The Stata Journal*, **7**, 507 – 541. 1, 55

Pearl, J. (1995). Causal diagrams for empirical research. *Biometrika*, **82**, 669–688. 12

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, 13

Poinsot, A., Leite, A., Chesneau, N., Sébag, M., and Schoenauer, M. (2024). Learning structural causal models through deep generative models: Methods, guarantees, and challenges. In *Proceedings of the 33rd International Joint Conference on Artificial Intelligence (IJCAI).* 4

Prechelt, L. (2012). Early stopping — but when? In Orr, G. B. and Müller, K.-R., editors, *Neural Networks: Tricks of the Trade: Second Edition*, 53–67. Springer, Berlin, Heidelberg. 10

R Core Team (2024). *R: A Language and Environment for Statistical Computing.* R Foundation for Statistical Computing, Vienna, Austria. 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 Conference on Neural Information Processing Systems (NeurIPS).* 67

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. 15

Rosenbaum, P. R. and Rubin, D. B. (1983). The central role of the propensity score in observational studies for causal effects. *Biometrika*, **70**, 41–55. 15

Rubin, D. B. (1980). Randomization analysis of experimental data: The Fisher randomization test comment. *Journal of the American Statistical Association*, **75**, 591–593. 15

Rubin, D. B. (2005). Causal inference using potential outcomes: Design, modeling, decisions. *Journal of the American Statistical Association*, **100**, 322–331. 14

Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1986). Learning representations by back-propagating errors. *Nature*, **323**, 533–536. 10

Sandercock, P. A., Niewada, M., Członkowska, A., and the International Stroke Trial Collaborative Group (2011). The International Stroke Trial database. *Trials*, **12**, 101. 18

Schmidhuber, J. (2015). Deep learning in neural networks: An overview. *Neural Networks*, **61**, 85–117. 10

Sick, B. and Dürr, O. (2025). Interpretable neural causal models with TRAM-DAGs. arXiv preprint 2503.16206, accepted at the CLeaR 2025 Conference. iii, v, 1, 4

Sick, B., Hothorn, 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. 4, 7, 9

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, **15**, 1929–1958. 10

Ushey, K., Allaire, J., and Tang, Y. (2025). *reticulate: Interface to 'Python'*. R package version 1.42.0. 24

Vegetabile, B. G. (2021). On the distinction between "conditional average treatment effects" (CATE) and "individual treatment effects" (ITE) under ignorability assumptions. arXiv preprint 2108.04939, presented at the Workshop on the Neglected Assumptions in Causal Inference (NACI), 38th International Conference on Machine Learning, 2021. v, 55

Zhao, Z. and Harinen, T. (2020). Uplift modeling for multiple treatments with cost optimization. arXiv preprint 1908.05372. 2

Zheng, X., Aragam, B., Ravikumar, P., and Xing, E. P. (2018). DAGs with no tears: Continuous optimization for structure learning. arXiv preprint 1803.01422, accepted at NeurIPS 2018. 3

Chapter 6

Appendix

6.1 Interpretation of linear coefficients

The transformation model framework allows for interpretation of the coefficients in the linear shift component. The choice of the inverse-link function F_Z determines the interpretation of the coefficients. For example, if the standard logistic distribution is chosen as the latent scale, i.e. $F_Z(z) = \text{expit}(z)$, the coefficients can be interpreted as log-odds ratios. For $F_Z(z) = 1 - \exp(-\exp(z))$, the interpretation of the coefficients changes to log-hazard ratios.

Consider the conditional transformation model:

$$F_{X_2|X_1}(x_2) = \text{expit}(h(x_2) + \beta_{12}x_1), \tag{6.1}$$

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

Applying the logit link (expit^{-1}) yields:

$$\log\left(\frac{F_{X_2|X_1}(x_2)}{1 - F_{X_2|X_1}(x_2)}\right) = h(x_2) + \beta_{12}x_1. \tag{6.2}$$

This shows that on the log-odds scale, X_1 has an additive linear effect on X_2. The corresponding odds ratio when increasing x_1 by one unit is:

$$\text{OR}_{x_1 \to x_1+1} = \frac{\exp(h(x_2) + \beta_{12}(x_1 + 1))}{\exp(h(x_2) + \beta_{12}x_1)} = \exp(\beta_{12}). \tag{6.3}$$

Therefore, $\exp(\beta_{12})$ represents the multiplicative change in odds of $X_2 \leq x_2$ for a one-unit increase in X_1, holding all else constant. This means that β_{12} can be interpreted as a log-odds ratio.

6.2 Bernstein polynomial for continuous outcomes

In deep TRAMs, the intercept for a continuous outcome y is modeled as a smooth and monotonically increasing function using a Bernstein polynomial of order M. Here, we focus on the more general case, where the intercept depends on the predictors $\mathbf{x}$. The same logic also applies to the simple intercept case without dependence on covariates.

The intercept of the transformation function can be written as:

$$h_I(y \mid \mathbf{x}) = \sum_{k=0}^{M} \vartheta_k(\mathbf{x}) \cdot B_{k,M}(s(y)), \tag{6.4}$$

where $s(y) \in [0, 1]$ is a scaled version of the outcome y, and $B_{k,M}(s(y))$ denotes the k-th Bernstein basis polynomial of degree M, defined as:

$$B_{k,M}(s(y)) = \binom{M}{k} s(y)^k (1-s(y))^{M-k}.$$

The parameters $\vartheta_k(\mathbf{x})$ depend on the predictors and determine the shape of the intercept function. To ensure that $h_I(y \mid \mathbf{x})$ is monotonically increasing in y, the coefficients must satisfy:

$$\vartheta_0(\mathbf{x}) \leq \vartheta_1(\mathbf{x}) \leq \cdots \leq \vartheta_M(\mathbf{x}).$$

To ensure monotonicity, the unbounded parameters $\tilde{\vartheta}_k(\mathbf{x}) \in \mathbb{R}$ are first predicted (by the neural network) and then transformed using a cumulative sum of softplus-transformed values. This guarantees that the resulting coefficients are non-decreasing, and therefore that the transformation function $h_I(y \mid \mathbf{x})$ is smooth and strictly monotonically increasing in y. Bernstein polynomials can approximate a wide range of smooth functions, provided the degree M is sufficiently large.

6.2.1 Scaling and extrapolation of the Bernstein polynomial

Because the Bernstein polynomial is only defined on the range $[0,1]$, the outcome variable y must be scaled to the unit interval. For parameter estimation alone, this scaling would be sufficient. However, the transformation function $h(y \mid \mathbf{x})$ must also be evaluable at arbitrary values of y, particularly those outside the range of the training data. This is essential, for instance, when performing generative sampling, where predicted outcomes may lie beyond the originally observed range of y.

To address this, we extend the Bernstein polynomial with a linear extrapolation in the tails. Specifically, we construct the core transformation within the 5% and 95% quantiles of the training outcome y using the smooth Bernstein polynomial in Equation 6.4, and extrapolate beyond this range linearly using the boundary derivatives. This results in a piecewise-defined transformation that is smooth, differentiable, strictly monotonic, and defined for all real-valued outcomes.

Let $q_{0.05}$ and $q_{0.95}$ denote the 5th and 95th empirical quantiles of y, computed on the training set. The scaled outcome is defined as:

$$s(y) = \frac{y - q_{0.05}}{q_{0.95} - q_{0.05}}. \tag{6.5}$$

This transformation maps the central interval $[q_{0.05}, q_{0.95}]$ onto the unit interval $[0,1]$, the domain of the Bernstein basis polynomials. Let $h_I(s(y) \mid \mathbf{x})$ be the core transformation function as defined in Equation (6.4). The extrapolated transformation $\tilde{h}_I(y \mid \mathbf{x})$ is then defined as:

$$\tilde{h}_I(y \mid \mathbf{x}) = \begin{cases} h_I(0 \mid \mathbf{x}) + h_I'(0 \mid \mathbf{x}) \cdot (s(y) - 0), & \text{if } s(y) < 0 \\ h_I(s(y) \mid \mathbf{x}), & \text{if } 0 \leq s(y) \leq 1 \\ h_I(1 \mid \mathbf{x}) + h_I'(1 \mid \mathbf{x}) \cdot (s(y) - 1), & \text{if } s(y) > 1 \end{cases} \tag{6.6}$$

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 $\vartheta_k(\mathbf{x})$, ensuring smooth and differentiable transitions at the boundaries (see Section 6.2.2).

This construction ensures several desirable properties. First, the transformation function $\tilde{h}_I(y \mid \mathbf{x})$ is globally defined on $\mathbb{R}$, avoiding undefined regions or discontinuities. Second, monotonicity is guaranteed due to the softplus parameterization of the coefficients $\vartheta_k(\mathbf{x})$.

6.2.2 Analytical derivative of the Bernstein polynomial

To compute gradients and ensure differentiability at the extrapolation boundaries, we derive the analytical form of the derivative of the intercept of the transformation function.

Recall the general form of the intercept of the transformation function:

$$h_I(y \mid \mathbf{x}) = \sum_{k=0}^{M} \vartheta_k(\mathbf{x}) \cdot B_{k,M}(s(y)), \tag{6.7}$$

where $s(y) \in [0,1]$ is the scaled outcome, $\vartheta_k(\mathbf{x})$ are the (monotonized) coefficients, and $B_{k,M}(s(y))$ are the Bernstein basis polynomials of degree M.

To compute the derivative with respect to y, we apply the chain rule:

$$\frac{d}{dy} h_I(y \mid \mathbf{x}) = \sum_{k=0}^{M} \vartheta_k(\mathbf{x}) \cdot \frac{d}{dy} B_{k,M}(s(y)) = \sum_{k=0}^{M} \vartheta_k(\mathbf{x}) \cdot \frac{dB_{k,M}(s)}{ds} \cdot \frac{ds}{dy}. \tag{6.8}$$

Since $s(y) = \frac{y - q_{0.05}}{q_{0.95} - q_{0.05}}$, its derivative is:

$$\frac{ds}{dy} = \frac{1}{q_{0.95} - q_{0.05}}. \tag{6.9}$$

The derivative of the Bernstein basis polynomial is:

$$\frac{d}{ds} B_{k,M}(s) = M \left[B_{k-1,M-1}(s) - B_{k,M-1}(s) \right]. \tag{6.10}$$

Therefore, the full derivative is:

$$\frac{d}{dy} h_I(y \mid \mathbf{x}) = \frac{M}{q_{0.95} - q_{0.05}} \sum_{k=0}^{M} \vartheta_k(\mathbf{x}) \left[B_{k-1,M-1}(s(y)) - B_{k,M-1}(s(y)) \right]. \tag{6.11}$$

This expression is used to evaluate the slope of the transformation function at the borders and is also critical when computing the likelihood.

6.3 Negative log-likelihood

6.3.1 Continuous Outcome

For a continuous outcome Y, the conditional cumulative distribution function (CDF) is given by:

$$F_{Y \mid \mathbf{X}=\mathbf{x}}(y) = F_Z(h(s(y) \mid \mathbf{x})), \tag{6.12}$$

where F_Z is the CDF of the standard logistic distribution:

$$F_Z(z) = \frac{1}{1 + e^{-z}}, \quad z \in \mathbb{R}, \tag{6.13}$$

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

The outcome y must be scaled to the unit interval $[0,1]$ because the Bernstein polynomial is defined on this range:

$$s(y) = \frac{y - \min(y)}{\max(y) - \min(y)}. \tag{6.14}$$

To compute the conditional density, we apply the change-of-variables formula:

$$f_{Y|\mathbf{X}=\mathbf{x}}(y) = f_Z(h(s(y) \mid \mathbf{x})) \cdot h'(s(y) \mid \mathbf{x}) \cdot s'(y), \tag{6.15}$$

where f_Z is the PDF of the standard logistic distribution:

$$f_Z(z) = \frac{e^z}{(1+e^z)^2}, \quad z \in \mathbb{R}. \tag{6.16}$$

The negative log-likelihood (NLL) contribution for a single observation is then given by:

$$\text{NLL} = -\log f_{Y|\mathbf{X}=\mathbf{x}}(y). \tag{6.17}$$

Plugging in the expressions yields:

$$\begin{aligned} \text{NLL} &= -\log f_{Y|\mathbf{X}=\mathbf{x}}(y) \\ &= -h(s(y) \mid \mathbf{x}) - 2\log(1 + \exp(-h(s(y) \mid \mathbf{x}))) \\ &\quad + \log h'(s(y) \mid \mathbf{x}) - \log(\max(y) - \min(y)). \end{aligned} \tag{6.18}$$

6.3.2 Discrete Outcome

For a discrete outcome (binary, ordinal, categorical) with ordered categories y_k, $k = 1, \ldots, K$, the transformation model defines the conditional CDF as:

$$F(Y \leq y_k \mid \mathbf{X} = \mathbf{x}) = F_Z(h(y_k \mid \mathbf{x})). \tag{6.19}$$

The likelihood contribution for an observation in class y_k is:

$$f_{Y|\mathbf{X}=\mathbf{x}}(y_k) = \begin{cases} F_Z(h(y_1 \mid \mathbf{x})), & k = 1, \\ F_Z(h(y_k \mid \mathbf{x})) - F_Z(h(y_{k-1} \mid \mathbf{x})), & k = 2, \ldots, K-1, \\ 1 - F_Z(h(y_{K-1} \mid \mathbf{x})), & k = K. \end{cases} \tag{6.20}$$

The corresponding NLL contribution is then:

$$\text{NLL} = -\log f_{Y|\mathbf{X}=\mathbf{x}}(y_k). \tag{6.21}$$

6.4 Encoding of discrete variables

In TRAM-DAGs, a variable X_i can act either as a predictor variable for a child node or as an outcome variable that depends on parent nodes.

When X_i is the outcome variable, and it is discrete with K ordered categories (e.g., ordinal), its conditional distribution is modeled via a transformation function h that defines $K-1$ cutpoints. The modeling differences between continuous and discrete outcomes have already been discussed.

However, when a discrete variable X_i with K categories is used as a predictor variable, it should be dummy encoded. Dummy encoding creates $K-1$ binary (0/1) indicator variables. Each binary variable corresponds to one of the non-reference categories, with the first category serving as the reference level that is not explicitly represented.

Example: Let X be an ordinal variable with three levels: 1, 2, 3. Dummy encoding results in two binary variables:

- X_1: 1 , if $X = 2$, 0 otherwise
- X_2: 1 , if $X = 3$, 0 otherwise

Now assume a continuous outcome Y that depends on X. The transformation model is:

$$F(Y \mid X) = F_Z(h_I(y) + x_1\beta_1 + x_2\beta_2)$$

This gives us the following cases:

- If $X = 1$ (reference level): $x_1 = 0$, $x_2 = 0$, so
  $F(Y \mid X = 1) = F_Z(h_I(y))$
- If $X = 2$: $x_1 = 1$, $x_2 = 0$, so
  $F(Y \mid X = 2) = F_Z(h_I(y) + \beta_1)$
- If $X = 3$: $x_1 = 0$, $x_2 = 1$, so
  $F(Y \mid X = 3) = F_Z(h_I(y) + \beta_2)$

The coefficients β_1 and β_2 represent the additive shift on the latent scale (e.g., log-odds) when moving from the reference category (1) to categories 2 and 3, respectively.

Dummy encoding ensures that discrete predictors can be incorporated into the deep TRAM framework and maintain interpretability.

6.5 Scaling of continuous variables

Neural networks work best when the input variables are standardized. While scaling (i.e., zero-mean and unit-variance normalization) removes the marginal variance pattern, Reisach *et al.* (2021) showed that many structure learning algorithms rely heavily on this pattern and may suffer a drop in performance when it is eliminated. However, since our approach assumes a known DAG structure, this is not a concern in our setting. Therefore, we scale continuous input variables in TRAM-DAGs.

A linear, monotonic, and invertible transformation of a predictor variable changes the interpretation of the coefficient. Scaling a predictor variable X as $X_{\text{std}} = (X - \mu_X)/\sigma_X$ implies 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 differs from the interpretation of the original coefficient β, which represents the change in log-odds for a one-unit increase in the raw predictor variable.

In contrast, the standardization of the outcome variable does not affect the interpretation of the model, due to the scale invariance of the log-odds. Suppose 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, the probability of being below a particular quantile remains the same after standardization. Consequently, the interpretation of coefficients in models with a continuous outcome remains unchanged. Specifically, 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 per one-unit increase in the (standardized) predictor, regardless of whether the outcome is standardized or not.

The general form of the transformation model is:

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

but now consider the case where this model is fitted using 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 $\tilde{\beta}$ are the estimated transformation function and coefficients after standardizing the outcome and predictors.

Example: To evaluate the probability $P(Y < 20 \mid X = 3)$ in the standardized setting, we use:

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

In summary, standardizing predictors changes coefficient interpretation, whereas outcome standardization does not affect interpretability or model validity.

6.6 Modeling interactions with complex shift (CS)

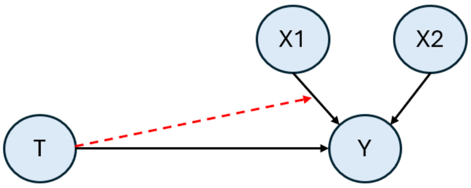


Figure 6.1: DAG for the complex shift example. The treatment T interacts with the covariate X_1 to influence the outcome Y. The covariate X_2 has a direct effect on the outcome.

The aim of this section is to show how TRAM-DAGs can effectively model interactions between multiple variables in a customizable way, using either a complex shift (CS) or a complex intercept (CI).

Here, we provide an example based on a simple DAG, as visualized in Figure 6.1. Assume that the binary treatment T is sampled from a Bernoulli distribution with probability 0.5. The covariates X_1 and X_2 represent baseline characteristics and are sampled from a standard normal distribution with a compound symmetric covariance matrix ($\rho = 0.1$). The outcome Y is binary and sampled from a Bernoulli distribution with a probability that depends on the treatment T and the covariates X_1 and X_2. The treatment T interacts with the covariate X_1 to influence the outcome, while the covariate X_2 has a direct effect on the outcome. The formula for the outcome (on the log-odds scale) is given by:

$$\operatorname{logit}(P(Y=1 \mid T, X_1, X_2)) = \beta_0 + X_1\beta_1 + X_2\beta_2 + T\beta_3 + TX_1\beta_4$$

where $\beta_0 = 0.45$, $\beta_1 = -0.5$, $\beta_2 = 0.1$, $\beta_3 = -0.85$, and $\beta_4 = 0.7$. The interaction term TX_1 allows the effect of X_1 on the outcome to vary depending on the treatment group. This results in a heterogeneous treatment effect. We sample 20,000 training observations from this logistic data-generating process, and model it using a TRAM-DAG following the assumed structure in the DAG in Figure 6.1. The TRAM-DAG is specified with a complex shift (CS) of the treatment T and the covariate X_1, which allows it to capture the interaction effect.

$$h(Y_k \mid T, X_1, X_2) = \vartheta_k + \mathrm{CS}(T, X_1) + \mathrm{LS}(X_2)$$

where $K = 1$ because the outcome is binary. The neural network for the CS is defined as shown in Figure 6.2, taking the treatment T and the covariate X_1 as inputs, which allows them to interact. After training the TRAM-DAG, we can visualize the resulting learned complex shift for each of the treatment groups, as illustrated in Figure 6.3. Due to the interaction, the effect of the covariate X_1 on the outcome Y differs between the treatment groups. The estimated effect aligns with the one specified in the data-generating process.

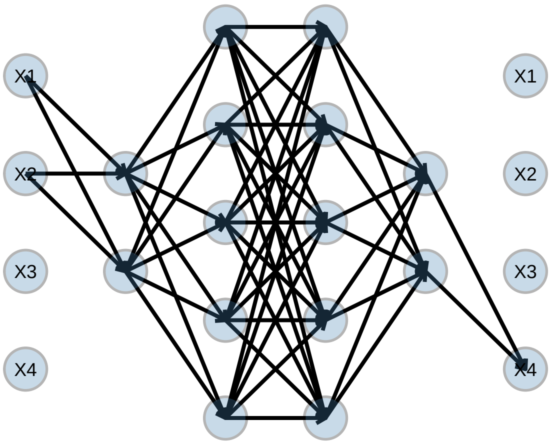


Figure 6.2: Neural network architecture for the complex shift (CS) effect of the treatment T and the covariate X_1 on the outcome Y (X_4). ReLU activation and batch normalization were used in the hidden layers. The deep neural network enables interactions between the input variables.

The fitted TRAM-DAG was then used to estimate the individualized treatment effect (ITE) for each sample i, defined as $\mathrm{ITE}_i(\mathbf{x}_i) = P(Y_i = 1 \mid T = 1, \mathbf{X}_i = \mathbf{x}_i) - P(Y_i = 1 \mid T = 0, \mathbf{X}_i = \mathbf{x}_i)$. We estimated the ITEs on the test set using the trained TRAM-DAG. In the uncalibrated model, the predicted ITEs did not perfectly match the true ITEs – although the difference was only a small deviation. Therefore, we recalibrated the model by fitting a generalized additive model (GAM) on a separate calibration dataset using the predicted probabilities from the TRAM-DAG. The recalibrated ITEs aligned well with the true ITEs. The results are shown in Figure 6.4.

This example demonstrates how TRAM-DAGs can be used to model interactions, which are crucial for accurate ITE estimation when applying the model as an S-learner.

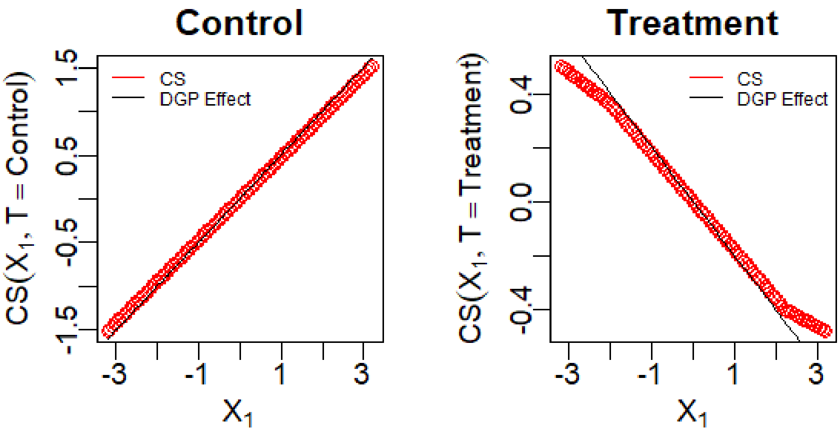


Figure 6.3: Learned complex shift (CS) effect of the treatment T and the covariate X_1 on the outcome (log-odds scale). Left: effect of X_1 on the outcome Y in the control group ($T = 0$); Right: effect of X_1 on the outcome Y in the treatment group ($T = 1$).

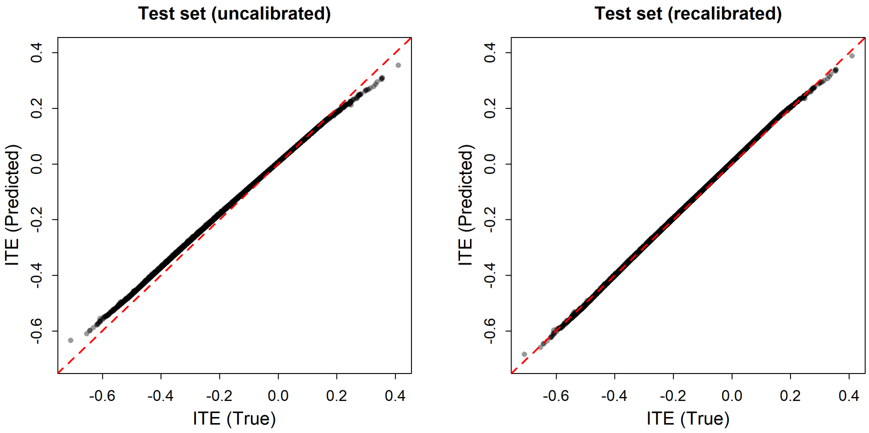


Figure 6.4: Estimated individualized treatment effects (ITE) using the fitted TRAM-DAG with a complex shift (CS). Left: estimated vs. true ITEs using the uncalibrated model. Right: estimated vs. true ITEs after recalibration. Recalibration was performed using an independent calibration dataset, where a generalized additive model (GAM) was fitted to recalibrate the probability estimates produced by the TRAM-DAG.

6.7 Experiment 2: Calibration plots

Figures 6.5-6.7 show the calibration plots of predicted risks versus observed outcome proportions for the models applied in Experiment 2 (International Stroke Trial, IST).

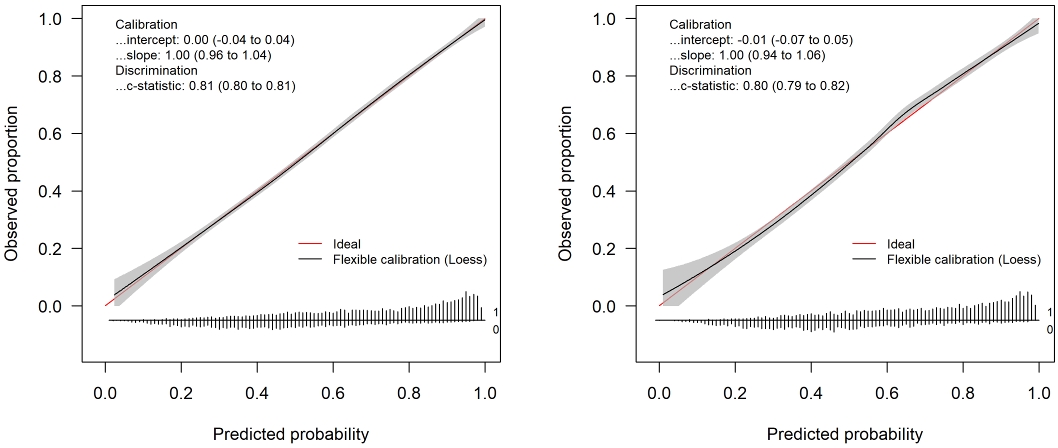


Figure 6.5: Calibration plot for the T-learner logistic regression applied to the International Stroke Trial (IST) in Experiment 2. The plot shows predicted risks versus observed event proportions. Left: training dataset; Right: test dataset.

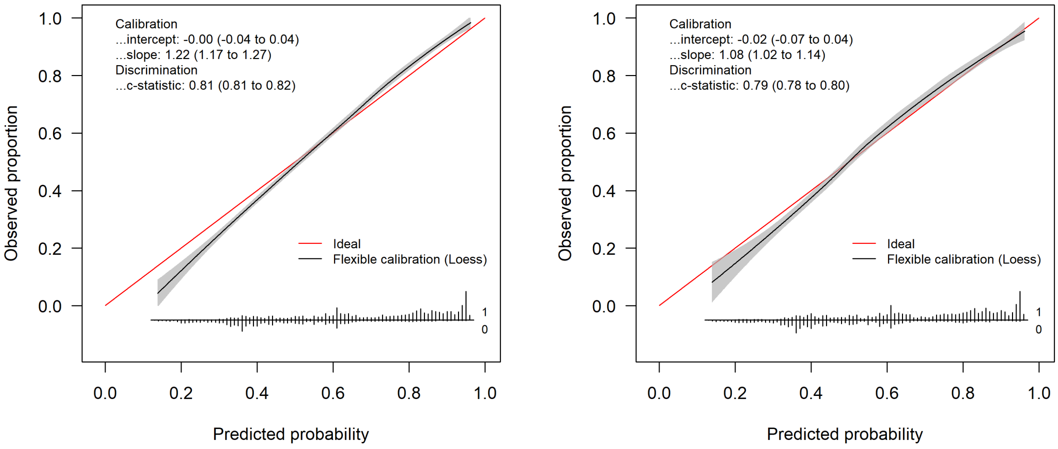


Figure 6.6: Calibration plot for the T-learner tuned random forest applied to the International Stroke Trial (IST) in Experiment 2. The plot shows predicted risks versus observed event proportions. Left: training dataset; Right: test dataset.

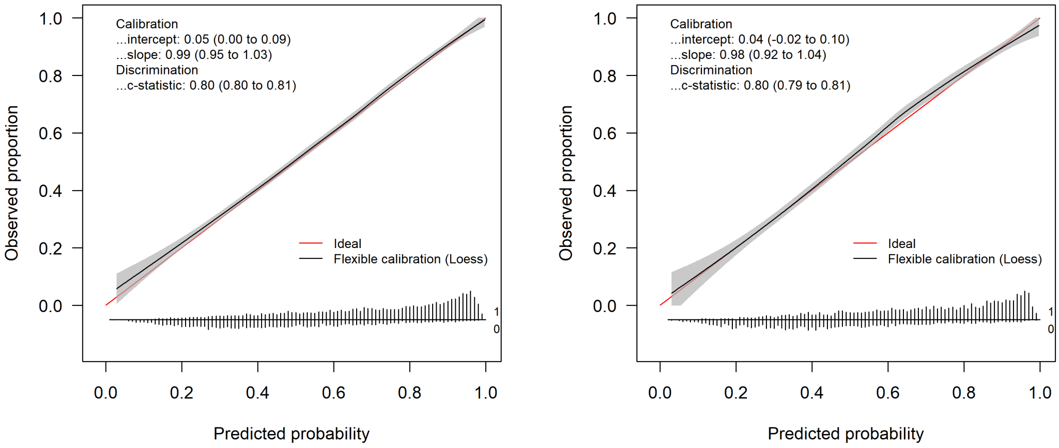


Figure 6.7: Calibration plot for the S-learner TRAM-DAG applied to the International Stroke Trial (IST) in Experiment 2. The plot shows predicted risks versus observed event proportions. Left: training dataset; Right: test dataset.

6.8 Experiment 3: Standard random forest for ITE estimation

In Section 2.2, we emphasized the importance of model calibration when estimating individualized treatment effects. Here, we present results from a default (untuned) random forest model in Scenario (1), where all variables are observed and both treatment and interaction effects are strong (see Figure 6.8). The corresponding model results are shown in Figure 6.9.

In the scatterplot of true vs. predicted probabilities for $\mathrm{P}(Y_i = 1 \mid \mathbf{X}_i = \mathbf{x}_i, T_i = t_i)$ in the training set, it is visible that the model does not predict probabilities accurately and is therefore poorly calibrated. This poor calibration also affects the estimated ITEs.

By comparison, the results of the tuned random forest (Figure 3.12) show that the model is better calibrated and the estimated ITEs are close to the true ITEs. These results highlight the importance of model tuning for ITE estimation, as poor calibration can lead to biased estimates of individualized treatment effects.

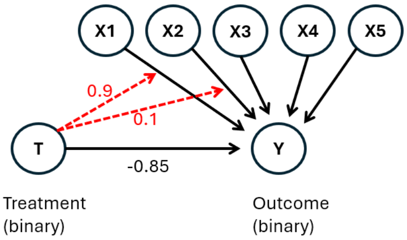


Figure 6.8: DAG for Scenario (1), where all variables are observed and both treatment and interaction effects are strong. The numbers indicate the coefficients on the log-odds scale. Red arrows represent interaction effects between treatment (T) and covariates (X_1 and X_2) on the outcome (Y).

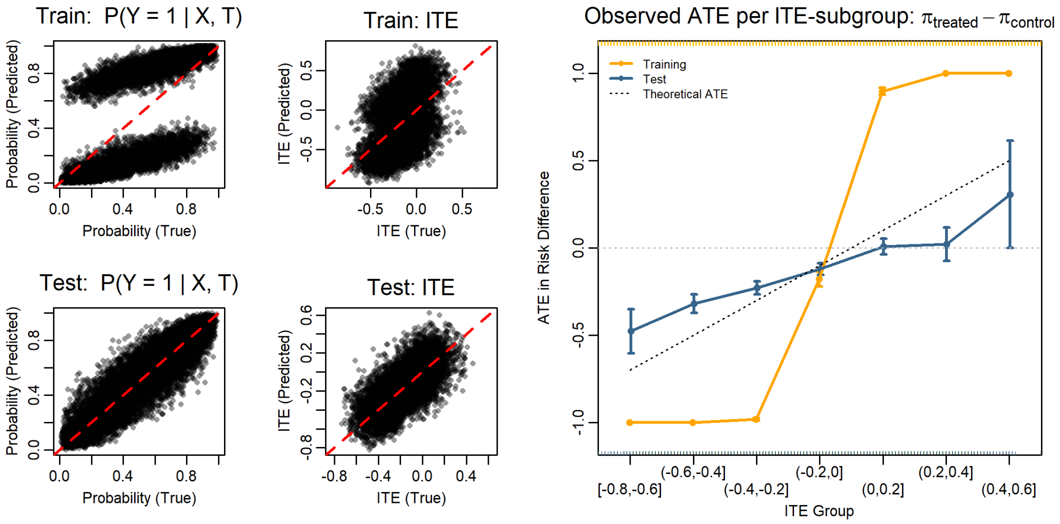


Figure 6.9: Results of the default random forest in Scenario (1), where the DAG is fully observed and both treatment and interaction effects are strong. Left: true vs. predicted probabilities for $\mathrm{P}(Y = 1 \mid X, T)$; Middle: true vs. predicted ITEs; Right: observed ATE in terms of risk difference per estimated ITE subgroup.

6.9 Experiment 3: Calibration plots

Figure 6.10 shows the calibration plots in terms of the predicted risks against the the observed proportions of the event for the T-learner tuned random forest in Scenario (3) with weak direct and interaction effects. This is in contrast to the prediction plots presented in Section 3.3 where we presented the true probabilities of the event $P(Y = 1 \mid X, T)$ against the predicted probabilities. It becomes apparent, that tuning the random forest model out-of-bag leads to a poor calibration on the training set, but due to improved generalization it leads to a better calibration on the test set.

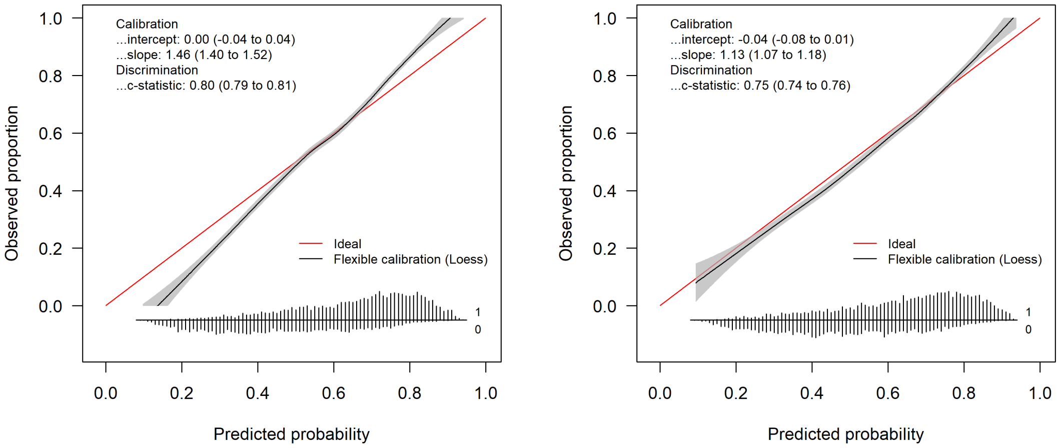


Figure 6.10: Calibration plot for the T-learner tuned random forest for Scenario (3) with weak direct and interaction treatmetn effects. It shows the predicted risks against the the observed proportions of the event. Left: training dataset; Right: test dataset.

6.10 Declaration of tools and services used

During the preparation of this thesis, I used ChatGPT, Google's Gemini and Github Copilot in order to to support language refinement, such as checking grammar, spelling and clarity of expression as well as to assist in plotting and resolving some R coding errors. After using theses tools/services, I reviewed and edited the content as needed and I take full responsibility for the content of the report.