Deep Learning for Semiparametric Difference-in-Differences Estimation

Master Thesis Presented to the

Department of Economics at the

Rheinische Friedrich-Wilhelms-Universität Bonn

In Partial Fulfillment of the Requirements for the Degree of

Master of Science (M.Sc.)

Supervisor: Prof. Dr. Christoph Breunig

Submitted in July 26, 2024 by:

Norman Lothar Metzinger

Matriculation Number: 3501090

Abstract

This thesis explores the implementation of deep feedforward neural networks into semiparametric Difference-in-Differences estimation (DiD), highlighting its potential under conditional parallel trends assumption. It reviews current classical and deep learning estimation methods for first-step DiD estimation and conducts a Monte Carlo Simulation to test their validity for second-step inference. The results demonstrate that deep learning performs nearly as well as the best classical approaches and outperforms those in scenarios with incorrectly specified outcomes. To further investigate deep learning, multiple deep learning architectures are tested, showing sensitivity towards their hyperparameters. Finally, DiD deep learning estimators show promise in real-world applications, effectively handling heterogeneous treatment effects.

Contents

1 Introduction

The evaluation of policy changes is crucial in economics and other social sciences, as it determines the effectiveness of governmental interventions. To deliver accurate evaluations in these quasi-experimental settings, researchers use the widely applied **Difference-in-Differences** (DiD) method. DiD is an econometric method to estimate the effect of a policy change on a group of individuals, known as the treatment group. To achieve this, the method requires comparing treatment group and a control group before and after the policy change. The critical underlying assumption is the **parallel trends assumption** (PTA), which states that the treatment and control group would have developed similarly without the policy change. This assumption is essential to identify the causal effect on the treatment group, referred to as the **average treatment effect on the treated** (ATT).

In practice, it is impossible to verify if the PTA holds, as it is by design untestable. If individuals are selected for treatment based on characteristics that influence the outcome, the PTA is violated. To address this issue, researchers condition on these characteristics, assuming conditional PTA (see Manfe and Nunziata, 2023; Sant'Anna and Zhao, 2020)

This thesis explores how researchers can use more flexible semiparametric approaches to achieve robust DiD estimation under the conditional PTA. A variety of machine and deep learning models are considered to replace the first-step estimation in the DiD framework. Especially the use of deep learning marks a novelty and I follow Farrell, T. Liang, and Misra (2021b) to contribute to the young but growing literature.

This thesis aims to contribute to the literature in four ways. First, I review the current state of classical and machine learning techniques used for DiD estimation. This includes revisiting the classic DiD estimation with **Two-Way Fixed Effects** (TWFE) and semi-parametric approaches such as **Outcome Regression** (OR) (see Heckman, Ichimura, and Todd, 1998), **Inverse Probability Weighting** (IPW) (see Abadie, 2005), and **Doubly-Robust DiD** (DRDiD) (see Sant'Anna and Zhao, 2020).

Second, I introduce a new approach using deep learning for first-step DiD estimation. As the literature is rather new, I aim to review how deep neural networks work and why they are a valid approach for inference, following the results of Farrell, T. Liang, and Misra (2021b).

Third, I conduct a **Monte Carlo Simulation** (MCS) to compare the performance of traditional techniques with deep learning variations of the IPW and DRDiD approaches. The simulation design follows Sant'Anna and Zhao (2020), using a **data generating process** (DGP) for panel data with four variations. The variations differ whether the outcome was generated correctly by OR or IPW, both or none. The results show that deep learning approaches perform nearly as well as the best traditional estimators in the first three variations and outperform others when the outcome is not correctly specified by any estimation strategy. Additionally, I introduce heterogeneity in treatment for the incorrectly specified case and demonstrate that the DRDiD deep learning case is the best-performing technique. These results suggest that deep learning is a valid approach for first-step estimation, especially when imposing few restrictions on the DiD model.

Lastly, I want to apply these techniques to a real-world dataset of Meyer, Viscusi, and Durbin (1995) to show the potential of the DRDiD deep learning estimator. The estimator shows promising results, robust to heterogeneous treatment effects assuming conditional PTA.

Finally, this thesis cannot cover all aspects relevant to semiparametric DiD estimation. It is crucial to emphasize the sensitivity of semiparametric estimators to the selection of the DGP, especially in low-dimensional examples such as viewed here (Zimmert, 2018). More research is needed to apply deep learning to other DGPs, such as repeated cross-sectional data. Additionally, this thesis focuses on the simple 2x2 DiD setting, excluding more complex scenarios like multiple treatment groups or multiple periods (see Callaway and Sant'Anna, 2021; Goodman-Bacon, 2021). Further research is required to develop more computationally efficient deep learning models, making these methods accessible to econometric practitioners.

The remainder of the thesis is structured as follows: Section I introduces the methodology of all techniques used in this thesis. Section II introduces deep learning in general and its use for inference. Section III introduces the MCS and discusses its results. Section IV applies the techniques to a real-world dataset. Section V discusses further extensions and research. Section VI concludes the thesis.

The code of this thesis can be found on GitHub for replication purposes: https://github.com/NormProgr/Deep-Learning-for-Semiparametric-DiD-Estimation.git.

2 Methodology

2.1 2x2 Difference in Differences

To introduce a common ground for the rest of the thesis, I want to introduce the notation for the basic 2x2 DiD model. The model has two time periods given by T, where $t \in 0, 1$. These define the pre- and post-treatment period of the policy change. Throughout the thesis, I use panel data, with i denoting the individual observed over time. The two groups are defined by D, where $d \in 0, 1$, such that $d = 1$ is the treatment group and $d = 0$ is the control group. The outcome variable is given by Y and the variable of interest, the ATT, is given by $\tau^{fe} = \mathbb{E}(Y_{1,1} - Y_{0,1} \mid X, D = 1)$. Therefore the ATT describes the expected difference in outcomes between the treated group (when they receive the treatment) and what their outcomes would have been if they had not received the treatment.

For my MCS I use the following common TWFE model notation to display the 2x2 DiD as in Sant'Anna and Zhao (2020):

$$Y_{it} = \alpha_1 + \alpha_2 T_i + \alpha_3 D_i + \tau^{fe}(T_i \cdot D_i) + \theta' X_i + \epsilon_{it} \tag{1}$$

Equation 1 implicates two assumptions that are of main focus in this thesis. First, it assumes homogeneity in treatment effects, such that τ^{fe} is constant over all individuals. Second, it assumes that the PTA holds, such that the treatment and control group would have developed similarly in the absence of the policy change such that $\mathbb{E}[Y_1 - Y_0 \mid X, D = d] = \mathbb{E}[Y_1 - Y_0 \mid D = d]$. If one or both of these assumptions are violated the TWFE estimator in Eq. 1 is inconsistent and biased.

To control for heterogeneous treatment effects and to account for conditional PTA, we can extend the model in Eq. 1 by adding interactions of X, T, and D (see Hansen, 2022; Manfe and Nunziata, 2023). Therefore, one can rewrite the Eq. 1 the following:

$$(2)\ Y_{it} = \alpha + \gamma T_{it} + \beta D_i + \tau^{corr}(T_{it} \cdot D_i) + X'_{it}\theta + (T_{it} \cdot X'_{it})\omega + (D_i \cdot X'_{it})\nu + (T_{it} \cdot D_i \cdot X'_{it})\rho + \epsilon_{it}$$

Note that $T_{it} \cdot D_i \cdot X'_{it}$ is the change of the treatment depending on X, thus the conditional PTA holds (Manfe and Nunziata, 2023). Eq. 2 is therefore, in a correctly specified case, neither biased nor inconsistent. The issue is that the econometric practitioner needs good reasoning and understanding to add the correct interactions. In the following sections, I introduce more flexible techniques circumventing this issue.

2.2 Outcome Regression

In this part, I revise OR as it is an important technique used in DRDiD, rather than using OR itself for estimating. OR is a generalized DiD estimation approach that estimates the outcomes as a function of covariates, given by $Y_i = g_i(X) + \epsilon_i$, where $i \in 0, 1$. The basic idea is to predict the control group outcomes based on their covariates and then compare these predicted outcomes to the actual outcomes observed for the treated group. The prediction can be computed through a linear regression or other non-linear models like p-nearest neighbor matching (Heckman, Ichimura, and Todd, 1998).

In this thesis, I fit a regression to estimate OR within the DRDiD framework, allowing to formulate the following model:

$$\hat{\tau}^{or} = \bar{Y}_{1,1} - \bar{Y}_{1,0} - \left[\frac{1}{n_{treat}} \sum_{i|D_i=1} (\hat{\mu}_{0,1}(X_i) - \hat{\mu}_{0,0}(X_i)) \right], \tag{3}$$

where $\bar{Y}_{1,1} - \bar{Y}_{1,0}$ is the average outcome among treated units between pre- and post-treatment period. The part in brackets of eq. 3 is the difference between the predicted control outcomes in the post- and the predicted control outcomes in the pre-treatment period. The key expression is $\hat{\mu}_{d,t}(X)$ which estimates the true, unknown $m_{d,t}(x) \equiv \mathbb{E}[Y_t \mid D = d, X = x]$. Intuitively, it estimates what the outcome would be for a person with specific traits if they were either treated or not treated. Note that if $\hat{\mu}_{d,t}(X)$ is linear, it would be close to the correct TWFE estimator τ^{corr}. Therefore, it is crucial that $\hat{\mu}_{d,t}(X)$ is correctly specified; otherwise, the ATT is biased and inconsistent.

2.3 Inverse Probability Weighting

The IPW estimator is another common approach to estimate ATT, which relaxes the conditional PTA as considered in this thesis. Contrary to the OR approach, IPW does not directly model the change in the outcome (Sant'Anna and Zhao, 2020). Instead, the idea is to only control for covariates that affect the probability of the treatment. If the probability of an individual receiving the treatment or being in the control group is the same, then the only difference between control and treatment is chance. Thus, there are no biases through confounding variables.

Therefore, it is crucial to correctly estimate the probability of being treated (Angrist and Pischke, 2009). The true probability is estimated by the so-called propensity score, given by $p(x) = P(D = 1 \mid X)$, which is not directly observable. Therefore it is estimated by $\hat{\pi}(X)$. Note that there are several ways to estimate the propensity score, such as logistic regression, probit regression, or machine learning techniques. These techniques are used in

the first step to estimate the propensity score, and in the second step, the outcome model is estimated parametrically (Abadie, 2005). In this thesis, I use logistic regression and deep neural networks to estimate the propensity scores.

The IPW estimator is given by:

$$\hat{\tau}^{\text{ipw}} = \frac{1}{\mathbb{E}_n[D]}\mathbb{E}_n\left[\frac{D-\hat{\pi}(X)}{1-\hat{\pi}(X)}(Y_1 - Y_0)\right], \tag{4}$$

where $\mathbb{E}_n$ is the sample average of the treatment D. The term $\frac{D-\hat{\pi}(X)}{1-\hat{\pi}(X)}$ reweights the treatment and control to account for the probability of receiving the treatment. $Y_1 - Y_0$ captures the change in the outcome for each individual.

Lastly, there are two remarks regarding the IPW estimator. First, the IPW estimator is consistent and unbiased if the propensity scores are correctly specified. Second, it is crucial to consider all relevant covariates in the propensity score estimation (Angrist and Pischke, 2009), than to improve the prediction of propensity scores (Chernozhukov, Whitney K. Newey, and Singh, 2022b). The reason is that including irrelevant covariates to improve the prediction of the propensity scores also increases the variance of the estimator, without adding any more information (Hernan and Robins, 2024).

2.4 Double Robust Difference in Differences

The DRDiD of Sant'Anna and Zhao (2020) is a combination of the two approaches discussed before; OR and IPW. The DRDiD identifies the ATT correctly if either the OR or IPW is correctly specified. In this case, the aforementioned weaknesses of either approach are avoided which makes the DRDiD double robust.

Recall from before that $\hat{\pi}(X)$ estimates $p(X)$ the true, unknown propensity score model. $\mu_{d,t}(X)$ is a model for the true, unknown outcome regression $m_{d,t}(x) \equiv \mathbb{E}[Y_t|D=d, X=x]$, $d, t = 0, 1$. In this thesis, I only view panel data such that I can write $\Delta Y = Y_1 - Y_0$ for the change in the outcome. The expression $\mu_{d,\Delta}(X) \equiv \mu_{d,1}(X) - \mu_{d,0}(X)$ represents the

difference in the expected outcomes before and after treatment, adjusted for covariates X, for the group with treatment status $D = d$.

Thus, one can se how OR and IPW are constructed within the DRDiD estimator, given by:

$$\tau^{dr} = \mathbb{E}\left[\left(w_1(D) - w_0(D, X; \hat{\pi})\right)\left(\Delta Y - \mu_{0,\Delta}(X)\right)\right], \tag{5}$$

where $w_1(D) - w_0(D, X; \hat{\pi})$ directly corresponds to the IPW estimator and $\Delta Y - \mu_{0,\Delta}(X)$ corresponds to the OR estimator from eq. 3 and eq. 4 respectively. Note that $w_1(D)$ is a weighting assigned to the treatment group and $w_0(D, X; \hat{\pi})$ is a weighting assigned to the control group, are given by:

$$w_1(D) = \frac{D}{\mathbb{E}[D]}, \quad \textit{and} \quad w_0(D, X; \hat{\pi}) = \frac{\hat{\pi}(X)(1-D)}{1-\hat{\pi}(X)} \Bigg/ \mathbb{E}\left[\frac{\hat{\pi}(X)(1-D)}{1-\hat{\pi}(X)}\right]. \tag{6}$$

The DRDiD estimator is consistent and unbiased if both OR and IPW are correctly specified but it is less obvious if only one of the two is correctly specified. To clarify this, assume the IPW is incorrectly specified and the OR is correctly specified. The incorrect specification of IPW is reflected in $w_0(D, X; \hat{\pi})$ in eq. 6 because $\hat{\pi}$ is biased. Meaning the weight for the control group is misspecified for IPW. This effect is nullified by the correct specification of OR in $\Delta Y - \mu_{0,\Delta}(X)$ because the change in the outcome evolution is zero in expectation. Intuitively, the OR correctly identifies that the change in the outcome of control should not change over time, as it is not treated, therefore any multiplication of it becomes zero as well. A similar argument can be made for the OR being misspecified and the IPW being correctly specified.

3 Deep Learning

3.1 Revision of Deep Learning

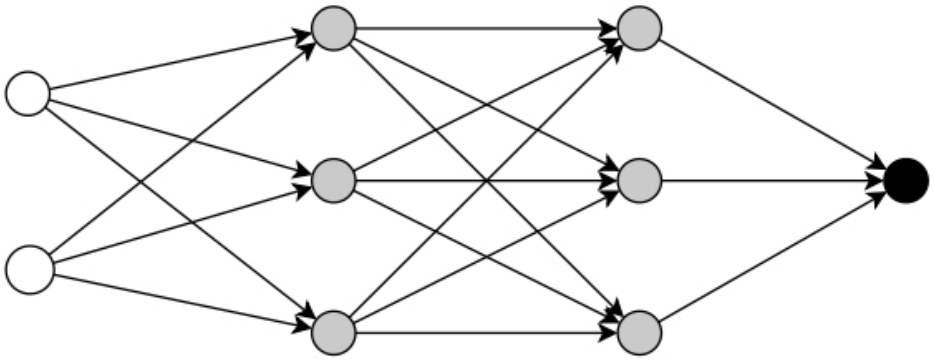
Deep learning is a rapidly developing field within machine learning that has recently received significant attention in economics. The idea is to transform complex data into a series of simpler representations, each of which is expressed in terms of the previous one (Goodfellow, Bengio, and Courville, 2016). A common example is the *feedforward neural network* architecture, which consists of a series of layers of neurons, each of which is connected to the next layer. The first layer is the input layer, the last layer is the output layer, and the layers in between are called hidden layers. The input layer corresponds to the covariates X, the output layer corresponds to the outcome Y. Figure 1 illustrates the layer and node structure of a **multilayer perceptron** (MLP), which is a special class of feedforward networks and is commonly used in empirical applications (Farrell, T. Liang, and Misra, 2021b). In this thesis, I use the wordings of MLP, feedforward neural network, and deep learning interchangeably as these are the approaches used here.

The actual computation within the neural networks is done by the *activation function* $\sigma : \mathbb{R} \to \mathbb{R}$, which is applied to the output of each hidden neuron. The most common activation function is the **rectified linear unit** (ReLU) function, defined as $\sigma(x) = \max(0, x)$, which is used in this thesis. The advantage of the linear ReLU is its computational efficiency and its ability to circumvent the vanishing gradient problem, which is a common problem in deep learning (Schmidt-Hieber, 2020).[1] The ReLU takes any linear combination given by $\tilde{x}'w + b$ and transforms it to $\sigma(\tilde{x}'w + b)$, where w is the weight vector, b is the constant term[2] , and $\tilde{x}$ is the input vector. Thus, the ReLU function sets all negative values

[1] The vanishing gradient issue arises especially by activation functions like *sigmoid* and *tanh*. When the neural network model is trained, all the weights of the model are updated through a process called *backpropagation*. Backpropagation is the algorithm used to compute the gradient of the loss function with respect to each parameter, which is then used to update the parameters such that they minimize the loss. The issue that can arise is that updating of parameters is hindered or training is completely stopped (Abuqaddom, Mahafzah, and Faris, 2021).

[2] The actual term in computer science is *bias* but to avoid confusion with the econometric term I follow Farrell, T. Liang, and Misra (2021b) and use the term *constant*.

Figure 1: Illustration of a feedforward neural network (Farrell, T. Liang, and Misra, 2021b)



Note: This figure illustrates the basic structure of a MLP $\mathcal{F}_{\text{MLP}}$, showing the input layer with $d = 2$ neurons in white. The two ($H = 2$) hidden layers in grey with $U = 6$ neurons, and one output layer in black ($L = 1$). The total amount of weights is $W = 25$.

from the linear combination $\tilde{x}'w + b$ to zero, while keeping all positive values unchanged. An example of a ReLU function is shown in the Appendix Figure 4.

The main problem the neural network aims to solve is estimating the unknown function $f^*(x)$. More precisely, f^* is a function that maps the input $\tilde{x}$ to the output $\tilde{y}$. As f^* is unknown, the neural network tries to estimate it by minimizing the expected loss function $\mathbb{E}[\ell(f, Z)]$, which can be written as:

$$f^* = \arg\min_f \mathbb{E}[\ell(f, Z)], \tag{7}$$

where $\ell(f, Z)$ is the loss function, and the function f predicts the outcome $\hat{Y} = f(X)$ and $Z = (Y, X')' \in \mathbb{R}^{d+1}$ is the set of random variables. The loss function can take various forms, such as least squares or logistic regression, with the latter used for propensity score estimation in this thesis. The logistic loss function is therefore defined as follows:

$$f^*(x) := \log\left(\frac{\mathbb{E}[Y|X = x]}{1 - \mathbb{E}[Y|X = x]}\right) \quad \text{and} \quad \ell(f, z) = -yf(x) + \log(1 + e^{f(x)}). \tag{8}$$

The logistic regression in eq. 8 is a method used for binary classification and estimates the probability that the covariates X take on values 0 or 1. The loss function measures the difference between the estimated probabilities and the actual class values. Unlike simple logistic regression applications, deep learning minimizes the loss function $\ell(f, z)$ by updating iteratively the weights and constants. For this Z is passed multiple times through the neural network until the loss is minimized. The amount of iterations is called *epochs*. Each epoch refers to one complete pass of the training dataset through the neural network. In practice, multiple epochs are required for the network to minimize the loss function.

Combining all the elements described above, the full neural network can be formalized through recursion:

$$\hat{f}_{\text{MLP}}(x) = W_L\sigma\left(\cdots\sigma\left(W_3\sigma\left(W_2\sigma\left(W_1\sigma\left(W_0x + b_0\right) + b_1\right) + b_2\right) + b_3\right) + \cdots\right) + b_L, \tag{9}$$

where $\hat{f}$ can be interpreted as a series of nested functions, where each function is a linear combination of the previous function. Note that W_l is the weight matrix for layer l. Equation 9 shows how the ReLU activation function *sigma* is applied to each hidden layer, starting with the covariates x. The transformed output of each hidden layer becomes the input for the next layer, and this process continues iteratively until the loss function is minimized. During this iterative process, the weights W and biases b are adjusted using the gradients computed from the loss function, typically through gradient descent. This ensures that the network parameters are optimized to improve prediction accuracy.

Equation 10 shows an example of a neural network with L hidden layers and the final output layer:

$$\begin{aligned} h_1 &= \sigma(W_1X + b_1), \\ h_2 &= \sigma(W_2h_1 + b_2), \\ &\vdots \\ h_L &= \sigma(W_Lh_{L-1} + b_L), \end{aligned} \tag{10}$$

Finally, the output of the network is:

$$\hat{Y} = W_L h_{L-1} + b_L,$$

Some final remarks on neural networks in general. First, if deep learning practitioners mention tuning parameters then they refer to adjusting the width and depth of the network, e.g. the amount of hidden layers (Farrell, T. Liang, and Misra, 2021b). These parameters are also the ones on which the neural network is repeatedly trained. Secondly, unlike inference, there is no understanding within the deep learning literature on how to select the optimal architecture or tuning parameters (see Schmidt-Hieber, 2020; Telgarsky, 2016). Consequently, the chosen neural network architecture may not be the optimal one, and selecting the right architecture is often arbitrary.

3.2 Deep Learning for Inference

The recent applications of deep learning for inferences mostly focus on prediction problems, such as outcome or propensity score prediction. The idea is to embed the neural network within a semiparametric framework, where the neural network is used to estimate the unknown function f^*. In this framework, the neural network is used as a first-step estimator, where the fitted values estimated by the neural network are used as the input for the second-step inference. Other techniques such as tree-based methods, logistic regression, or hybrid models are also used as first-step estimators. The advantage of these machine learning approaches is their robustness towards heterogeneous treatment effects, conditional controls, or many covariates (Belloni et al., 2017). Belloni et al. (2017) shows that neural networks perform as well as other machine learning approaches in terms of recovering the true treatment effect as a first-step estimator. Chernozhukov et al. (2018) approves those results in a low-dimensional setting but reports issues with the neural network if n is small. Common across literature are the benefits of deep learning regarding

handling heterogeneous treatment effects (see Belloni et al., 2017; Chernozhukov et al., 2018; Farrell, T. Liang, and Misra, 2021a)

Another advantage of Deep learning is its effectiveness in high covariate settings (Chernozhukov, Whitney K Newey, and Singh, 2022a).[3] This advantage arises from deep learning's capability to perform variable selection. Especially employing some form of regularization, deep learning can reduce the variance induced by high covariates, albeit at the cost of increased bias (Chernozhukov et al., 2018). Many machine learning techniques (e.g., lasso or certain tree-based methods) have similar properties. As such this thesis does not try to promote deep learning as an optimal technique but aims to investigate its utility as a valid first-step estimator in a semiparametric DiD framework.

To evaluate deep learning performance there are multiple important criteria to consider. First, deep learning should have good approximation power (Belloni et al., 2017), meaning it can closely approximate the true underlying function of the data. Second, deep learning should avoid overfitting the data (Belloni et al., 2017), leading to poor generalization of new data. Third, Belloni et al. (2017) emphasize that doubly robust estimation methods ensure valid inference for many machine learning frameworks, including neural networks.[4]

4 Monte Carlo Simulations

4.1 Data Generating Process

In this section, I introduce the DGP for the Monte Carlo simulations. The DGP is based on the simulation study by Kang et al. (2007) and Sant'Anna and Zhao (2020). The advantage of this setup is to ensures comparability with previous studies and allow to validate novel

[3] Belloni et al. (2017) demonstrate this for *moderately high* and *very high* amount of covariates compared to the sample size.

[4] Belloni et al. (2017) also point out that some form of orthogonal moment condition can also lead to valid inference in this setting. See Farrell, T. Liang, and Misra (2021a) for a discussion that with even weaker conditions than doubly robust or orthogonality valid semiparametric inference is achievable, although these are not of focus here.

approaches as the use of deep neural networks for DiD estimation. For all simulations, the DGP has a total sample size of $n = 1000$. There are two time periods $t = 0, 1$ and two groups $i = 0, 1$, such that it allows to apply the classical 2x2 DiD estimator. Since individuals are tracked over time, the data is panel data. Kang et al. (2007) created the DGP to include covariate specific trends and homogenous treatment effects. The true ATT is $\tau = 0$. In the first simulation shown in Table 1, I adhere to this specification. In the second simulation in Table 3, I extend the DGP to allow for heterogeneous treatment effects.

To introduce the outcome generation of the DGP, consider the arbitrary input vector $M = (M_1, M_2, M_3, M_4)'$ and let the true OR and propensity score-based IPW model be defined as follows:

$$f_{\text{or}}(M) = 210 + 27.4 \cdot M_1 + 13.7 \cdot (M_2 + M_3 + M_4), \tag{11}$$

$$f_{\text{ps}}(M) = 0.75 \cdot (-M_1 + 0.5 \cdot M_2 - 0.25 \cdot M_3 - 0.1 \cdot M_4). \tag{12}$$

Note a selection bias is constructed within this data (Kang et al., 2007) such that naive estimators are likely to be biased. As M is arbitrary, Kang et al. (2007) introduces two variations of covariates Z and X that are in use in the simulations. Z is a set of observable variables, while X is a set of unobservable variables. In this simulation study, $f_{\text{or}}(M)$ and $f_{\text{ps}}(M)$ are constructed only by Z, only by X, or a combination of both. Thus there are four different DGP setups labeled as DGP1, DGP2, DGP3, and DGP4. These four setups differ because Z is a non-linear transformation of X.

Consider $\mathbf{X} = (X_1, X_2, X_3, X_4)'$ distributed as $N(0, I_4)$, where I_4 is the 4×4 identity matrix. For $j = 1, 2, 3, 4$, Kang et al. (2007) define the following variations of $Z_j = \frac{\tilde{Z}_j - \mathbb{E}[\tilde{Z}_j]}{\sqrt{\text{Var}(\tilde{Z}_j)}}$

where:

$$
\begin{aligned}
\tilde{Z}_1 &= \exp(0.5X_1), \\
\tilde{Z}_2 &= 10 + \frac{X_2}{1+\exp(X_1)}, \\
\tilde{Z}_3 &= (0.6 + \frac{X_1 X_3}{25})^3, \quad \text{and} \\
\tilde{Z}_4 &= (20 + X_2 + X_4)^2.
\end{aligned} \tag{13}
$$

Each variation of Z differs by their functional form as they are quadratic, exponential, and cubic. They also include variations of interactions of X. This complexity in the functional form of Z is added to invoke potential biases when estimating the ATT. For example, when the true DGP is based on X but the model estimates are based on Z, then the estimates are likely to be biased. As we can construct either the OR or IPW model based on Z, X, or a combination of both, we have four different setups. The setup for each DGP is presented below, indicating which model is correctly specified and which is not.

DGP1

(IPW and OR models correct)

$$
\begin{aligned}
Y_0(0) &= f_{\text{or}}(Z) + \nu(Z, D) + \epsilon_0, \\
Y_1(d) &= 2 \cdot f_{\text{or}}(Z) + \nu(Z, D) + \epsilon_1(d) \\
p(Z) &= \frac{\exp\left(f_{\text{ps}}(Z)\right)}{1+\exp\left(f_{\text{ps}}(Z)\right)}, \\
D &= 1\{p(Z) \geq U\};
\end{aligned}
$$

DGP2

(IPW model incorrect, OR correct)

$$
\begin{aligned}
Y_0(0) &= f_{\text{or}}(Z) + \nu(Z, D) + \epsilon_0, \\
Y_1(d) &= 2 \cdot f_{\text{or}}(Z) + \nu(Z, D) + \epsilon_1(d) \\
p(X) &= \frac{\exp\left(f_{\text{ps}}(X)\right)}{1+\exp\left(f_{\text{ps}}(X)\right)}, \\
D &= 1\{p(X) \geq U\};
\end{aligned}
$$

DGP3

(IPW model correct, OR incorrect)

$$
\begin{aligned}
Y_0(0) &= f_{\text{or}}(X) + \nu(X, D) + \epsilon_0, \\
Y_1(d) &= 2 \cdot f_{\text{or}}(X) + \nu(X, D) + \epsilon_1(d) \\
p(Z) &= \frac{\exp\left(f_{\text{ps}}(Z)\right)}{1 + \exp\left(f_{\text{ps}}(Z)\right)}, \\
D &= 1\{p(X) \geq U\};
\end{aligned}
$$

DGP4

(IPW and OR models incorrect)

$$
\begin{aligned}
Y_0(0) &= f_{\text{or}}(X) + \nu(X, D) + \epsilon_0, \\
Y_1(d) &= 2 \cdot f_{\text{or}}(X) + \nu(X, D) + \epsilon_1(d) \\
p(X) &= \frac{\exp\left(f_{\text{ps}}(X)\right)}{1 + \exp\left(f_{\text{ps}}(X)\right)}, \\
D &= 1\{p(X) \geq U\};
\end{aligned}
$$

4.2 Homogenous Treatment Effects Simulation

In this section, I present the results of the Monte Carlo simulations for the homogenous treatment effects case. Table 1 and Table 3 report the average bias, median bias, root mean squared error, variance, and coverage of the estimators. $\hat{\tau}^{corr}$ are the results of the correctly specified TWFE estimators from equation 2, which can be interpreted as a baseline for the other estimators. $\hat{\tau}^{fe}$ is the naive TWFE estimator from Equation 1, as argued, the estimator is highly biased because the naive selection of controls does not reflect the underlying function of the data. The bias can bee seen as the coverage probabilities of the $\hat{\tau}^{fe}$ estimator across DGPs are almost zero. $\hat{\tau}^{ipw}$ and $\hat{\tau}^{dr}$ are the results of the IPW and DRDiD estimators, respectively. In both cases are the propensity scores estimated with a logistic regression. The results of $\hat{\tau}^{fe}$, $\hat{\tau}^{ipw}$, and $\hat{\tau}^{dr}$ are directly comparable to the results of Sant'Anna and Zhao (2020) panel data case. $\hat{\tau}^{ipw,dl}$ and $\hat{\tau}^{dr,dl}$ are the results of the IPW and DRDiD estimators, respectively, where the propensity scores are estimated with a neural network. Note that the OR part of $\hat{\tau}^{dr,dl}$ and $\hat{\tau}^{dr}$ is estimated as a linear model.

In DGP1, all estimators exhibit relatively small biases, except for the naive $\hat{\tau}^{fe}$ estimator. Additionally, the coverage is quite high, with the deep learning applications even reaching 1. This is likely due to inflated confidence interval length and the applied regularization (Farrell, T. Liang, and Misra, 2021b). A more thorough discussion is presented in

Table 1: Monte Carlo Simulation with Homogenous Treatment Effects

Estimator	Reference	Av. Bias	Med. Bias	RMSE	Variance	Cover
DGP1						
$\hat{\tau}^{fe}$	Regression, Eq. (1)	-20.963	-20.816	21.277	13.247	0.000
$\hat{\tau}^{corr}$	Regression, Eq. (2)	-0.002	-0.001	0.196	0.038	0.840
$\hat{\tau}^{ipw}$	Abadie (2005)	-0.376	-0.469	9.396	45.704	0.840
$\hat{\tau}^{ipw,dl}$	Abadie (2005) + DL	-3.819	-3.697	3.841	36.065	1.000
$\hat{\tau}^{dr}$	Sant'Anna and Zhao (2020)	0.003	0.008	0.218	0.022	0.834
$\hat{\tau}^{dr,dl}$	Sant'Anna and Zhao (2020) + DL	-0.121	-0.120	0.121	0.020	1.000
DGP2						
$\hat{\tau}^{fe}$	Regression, Eq. (1)	-19.261	-19.040	19.606	13.403	0.000
$\hat{\tau}^{corr}$	Regression, Eq. (2)	-0.004	-0.001	0.195	0.038	0.832
$\hat{\tau}^{ipw}$	Abadie (2005)	-0.498	-0.472	9.660	47.106	0.839
$\hat{\tau}^{ipw,dl}$	Abadie (2005) + DL	-21.983	-22.114	22.335	43.872	0.000
$\hat{\tau}^{dr}$	Sant'Anna and Zhao (2020)	0.005	0.002	0.207	0.021	0.802
$\hat{\tau}^{dr,dl}$	Sant'Anna and Zhao (2020) + DL	-0.148	-0.150	0.148	0.020	1.000
DGP3						
$\hat{\tau}^{fe}$	Regression, Eq. (1)	13.122	12.899	14.028	24.575	0.109
$\hat{\tau}^{corr}$	Regression, Eq. (2)	0.142	-0.114	4.869	23.685	0.782
$\hat{\tau}^{ipw}$	Abadie (2005)	0.109	0.219	9.630	43.498	0.817
$\hat{\tau}^{ipw,dl}$	Abadie (2005) + DL	-0.810	-0.794	0.824	40.227	1.000
$\hat{\tau}^{dr}$	Sant'Anna and Zhao (2020)	-0.104	0.052	4.599	11.165	0.840
$\hat{\tau}^{dr,dl}$	Sant'Anna and Zhao (2020) + DL	0.228	0.219	0.293	11.240	1.000
DGP4						
$\hat{\tau}^{fe}$	Regression, Eq. (1)	-16.434	-16.283	17.226	26.633	0.033
$\hat{\tau}^{corr}$	Regression, Eq. (2)	-3.063	-3.165	6.162	28.588	0.654
$\hat{\tau}^{ipw}$	Abadie (2005)	-3.881	-4.063	10.576	47.230	0.798
$\hat{\tau}^{ipw,dl}$	Abadie (2005) + DL	-4.992	-4.962	5.005	43.947	1.000
$\hat{\tau}^{dr}$	Sant'Anna and Zhao (2020)	-3.177	-3.162	5.899	12.259	0.752
$\hat{\tau}^{dr,dl}$	Sant'Anna and Zhao (2020) + DL	1.630	1.593	1.652	15.876	1.000

Notes: Simulations based on panel data with sample size $n = 1000$ and 1000 Monte Carlo repetitions. The average bias "Av. Bias", median bias "Med. Bias", root mean squared error "RMSE", and average variance "Variance" of the estimators are reported. The "Cover" describes the coverage probability of how often the estimated treatment coefficient falls within the confidence intervall of the true treatment effect. The methods that predict propensity scores with deep learning are marked by "DL". The true treatment effect is $\tau = 0$ in all cases and homogenous.

Section 4.4. The results are consistent with the theory, as DGP1 marks the case where the IPW and OR models are correctly specified. In DGP2, the propensity score approach is misspecified such that $\hat{\tau}^{ipw}$ and $\hat{\tau}^{ipw,dl}$ are biased but the bias for the $\hat{\tau}^{ipw,dl}$ is substantial. Possible reasons could be overfitting or the prediction of extreme propensity scores. The IPW model approach generally produces high variance, which is consistent with Sant'Anna and Zhao (2020). This high variance also appears in other data structures, such as repeated cross-sections (Manfe and Nunziata, 2023; Sant'Anna and Zhao, 2020). On the other hand, in DGP3 one can see that all estimators are relatively unbiased, except for the naive TWFE estimator as before. These results are consistent as DGP3 marks the case where the IPW model is correctly specified. In DGP1-3 are both DRDiD estimators $\hat{\tau}^{dr}$ and $\hat{\tau}^{dr,dl}$ relatively unbiased and produce low variance. Notably, the classic $\hat{\tau}^{dr}$ of Sant'Anna and Zhao (2020) does perform slightly better in terms of variance and bias.

DGP4 is the most challenging but probably most realistic case as both the IPW and OR models are misspecified. One can see clearly that all estimators are now more biased and have higher variance. Surprisingly, the $\hat{\tau}^{dr,dl}$ estimator reports the smallest bias and relatively low variance compared to the other estimators. This result is consistent with the findings of Belloni et al. (2017), Chernozhukov et al. (2018), and Farrell, T. Liang, and Misra (2021b) that deep learning is useful to recover the true treatment effect if there is a nuisance in the data.

Overall, the results seem to be consistent with the findings of the literature on deep learning and DiD estimation. It should be noted that the biases of the deep learning estimators are relatively similar distributed within each DGP. This suggests that the deep learning model results across Monte Carlo runs are consistent and not heavily driven by outliers. The results also mirror a structural aspect of deep learning that especially when using regularization methods, they are prone to produce symmetrically distributed errors around zero (Koh and P. Liang, 2017).

To evaluate if the deep learning model is robust, I report the minimum loss of the training and validation set in Table 2. Note that I implemented one model and applied

it on all DGP setups, such that the results are comparable. Across all DGP setups the deep learning model reports similar losses, indicating that the model is robust across the different DGPs. Importantly, across all setups is the validation loss smaller than the training loss, indicating that the model is not overfitting (see Farrell, T. Liang, and Misra, 2021b; Goodfellow, Bengio, and Courville, 2016).

Table 2: Performance of the Neural Network across DGPs

Minimum Loss	DGP1	DGP2	DGP3	DGP4
Training	0.634	0.631	0.634	0.632
Validation	0.617	0.626	0.617	0.625

Notes: The neural networks width is set to 32. The depth is set to 3 and the learning rate is set to 0.01. The number of epochs is set to 50. These specifications are set across all DGPs for the same neural network.

4.3 Heterogeneous Treatment Effects Simulation

In the previous sections, I outlined the advantage of deep learning, or machine learning in general, when dealing with heterogeneous treatment effects. The problem of heterogeneous treatment effect arises when the treatment effect $\theta(X)$ varies across groups (Hansen, 2022). To validate how the aforementioned estimators perform under heterogeneous treatment effects, I introduce heterogeneity to the DGP4. DGP4 is the most general and possibly the most realistic case of the observed DGPs. The main difference to the DGP4 with homogeneous treatment effects is the introduction of $\theta(X)$, which directly influences the outcome $Y_1(d)$ depending on the value of X. DGP4 can therefore be rewritten as follows:

DGP4 with Heterogeneous Treatment Effects

$$
\begin{aligned}
Y_0(0) &= f_{\text{or}}(X) + \nu(X) + \epsilon_0, \\
Y_1(d) &= 2 \cdot f_{\text{or}}(X) + \nu(X) + \theta(X) \cdot d + \epsilon_1(d), \\
p(X) &= \frac{\exp\left(f_{\text{ps}}(X)\right)}{1 + \exp\left(f_{\text{ps}}(X)\right)}, \\
D &= 1\{p(X) \geq U\},
\end{aligned}
$$

where: $\theta(X) = 10 \cdot (Z_1 + Z_2 - Z_3 + Z_4)$.

The results of the Monte Carlo simulations with heterogeneous treatment effects are presented in Table 3. The estimators $\hat{\tau}^{fe}$,$\hat{\tau}^{corr}$,$\hat{\tau}^{ipw}$, and $\hat{\tau}^{dr}$ are now more biased and have higher variance compared to the homogenous treatment effects case. These results are consistent with the literature as these methods do not account for heterogeneity (Hansen, 2022). Manfe and Nunziata (2023) reports similar more biased results for the IPW and DRDiD estimator in the case of repeated cross-sectional data. The $\hat{\tau}^{ipw,dl}$ also reports higher bias and variance compared to the homogenous treatment effects case. But the $\hat{\tau}^{ipw,dl}$ is now overall less biased and has lower variance than the comparable $\hat{\tau}^{ipw}$. The same applies to the $\hat{\tau}^{dr,dl}$, which reports the smallest bias and variance across all estimators. These results are interesting as they indicate that neural networks are more robust towards covariate-specific trends and heterogeneous treatment effects than comparable estimators viewed in this thesis. This is consistent with the findings of Farrell, T. Liang, and Misra (2021b) and Chernozhukov et al. (2018).

4.4 Comparison of Deep Learning Architectures

The choice of the correct neural network seems generally arbitrary as discussed in Section 3. This is due to the chosen activation function, neural network class, or hyperparameters. Table 4 shows the influence of different hyperparameters on the DGP used in Section 4.3.

Table 3: Monte Carlo Simulation with Heterogenous Treatment Effects in DGP4

Estimator	Reference	Av. Bias	Med. Bias	RMSE	Variance	Cover
$\hat{\tau}^{fe}$	Regression, Eq. (1)	-20.418	-20.245	21.242	34.360	0.015
$\hat{\tau}^{corr}$	Regression, Eq. (2)	-9.429	-9.355	10.602	23.483	0.225
$\hat{\tau}^{ipw}$	Abadie (2005)	-7.866	-7.899	13.231	55.196	0.712
$\hat{\tau}^{ipw,dl}$	Abadie (2005) + DL	-8.424	-8.408	8.432	51.123	1.000
$\hat{\tau}^{dr}$	Sant'Anna and Zhao (2020)	-7.238	-7.128	10.115	24.060	0.619
$\hat{\tau}^{dr,dl}$	Sant'Anna and Zhao (2020) + DL	-1.596	-1.607	1.633	16.570	1.000

Notes: Simulations based on panel data with sample size $n = 1000$ and 1000 Monte Carlo repetitions. The average bias "Av. Bias", median bias "Med. Bias", root mean squared error "RMSE", and average variance "Variance" of the estimators are reported. The methods that predict propensity scores with deep learning are marked by "DL". The true treatment effect is $\tau = 0$ and heterogenous in all cases.

Note that the activation function is ReLU and the neural network is a feedforward neural network across all architectures.

In table 4 one can see, that there is no strictly better alternative architecture when comparing these neural network classes, except variation 6. Every change in the hyperparameters comes to the cost of either higher bias or higher variance, which is a common trade-off in machine learning. For example, the architecture used throughout the thesis has a higher bias compared to the first variation, which has more hidden layers and units. This comes at the cost of higher variance. Interestingly, there is no clear sign that deeper neural networks (with more units and layers) impose better results than shallower networks. Generally, deeper neural networks come with higher computational costs, which can be extensive, especially in the case of complex and large data (Thompson et al., 2020).

Additionally, one can see that the first variation and the last variation have a bigger validation loss than training loss. The neural networks 4 and 5 report very small differences between the losses. As argued, a higher validation loss compared to the training loss indicates that the model is overfitting. For practitioners, it might be useful to use the results of the loss to evaluate which neural networks to select.

Observing the coverage probabilities in Table 4, one can see again the extreme coverage of 1 in all but two cases. The exemptions are variations 4 and 6 which report a coverage

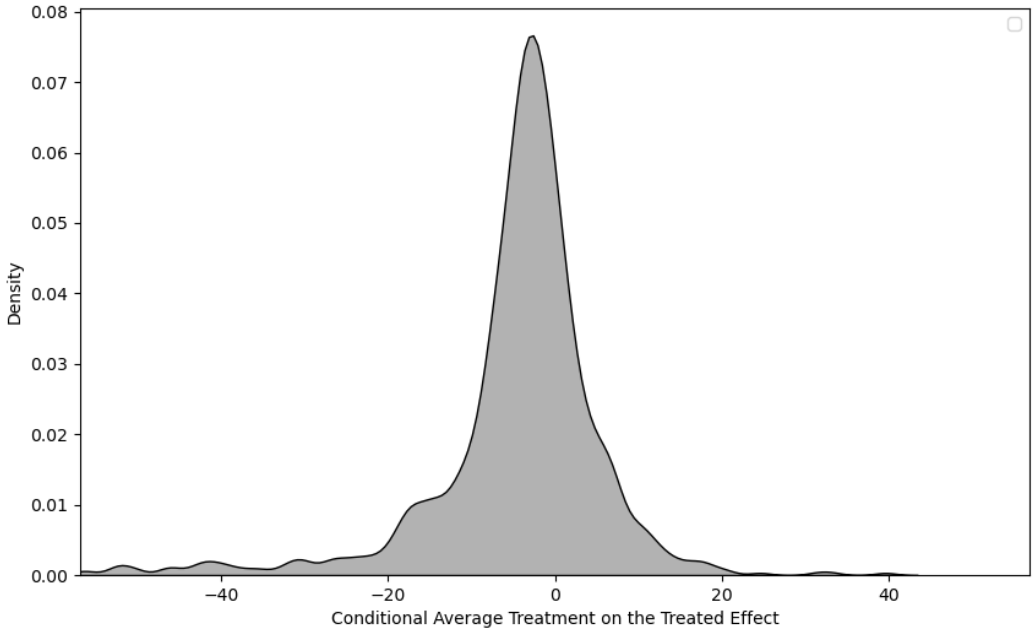
Table 4: Simulation Results across Neural Network Architectures

Metric\Architecture	Thesis	1	2	3	4	5	6
Avg Bias	-1.596	-0.409	-2.731	-6.011	-5.501	-4.369	4.328
Med Bias	-1.607	-0.557	-2.669	-6.002	-5.524	-4.483	-3.052
RMSE	1.633	1.192	2.7739	6.012	5.573	4.389	12.204
Variance	16.570	18.920	15.793	11.059	11.592	11.811	45.494
Training Loss	0.632	0.608	0.627	0.658	0.723	0.657	0.510
Validation Loss	0.625	0.612	0.618	0.656	0.722	0.651	0.566
Cover	1.000	1.000	1.000	1.000	0.840	1.000	0.874
Depth	3	5	2	3	3	6	3
Units	32	64	16	128	16	128	32
Learning Rate	0.01	0.001	0.01	0.001	0.0001	0.001	0.01
L2 Regularization	0.01	0.001	0.01	0.1	0.01	0.01	0.00

Notes: All networks use the ReLU activation function. The classic architecture is the one used throughout the thesis. The other architectures are variations of the classic architecture with different hyperparameters. For example shows architecture 1 the following: the depth is 5 such there are 5 hidden layers, each hidden layer has 64 units (or *neurons*). Each hidden layer applies a L2 regularization with value 0.001.

probability of 0.84 and 0.874, respectively. Variation 4 reports the lowest learning rate and variation 6 is equal to the architecture used throughout the thesis but with no regularization imposed. The learning rate of variation 4 is comparable to Farrell, T. Liang, and Misra (2021b) which also report very high coverage probabilities but not as extreme as 1. Farrell, T. Liang, and Misra (2021b) argue that regularization can lead to extreme coverage probabilities but it is not clear how large the impact is. The results of the thesis application and variation 6 support this argument that imposing regularization might lead to extreme coverage probabilities. Variation 4 has a similar regularization as the other variations but a smaller learning rate, which could hint that learning rate and regularization affect the distribution of estimates and thus the coverage probabilities. The results emphasize the effect of hyperparameter selection on the model outcome and imply caution when imposing regularization within neural networks as Farrell, T. Liang, and Misra (2021b) suggest.

Figure 2: Conditional Average Treatment on the Treated Effekt without Regularization



To better understand the extreme coverage probabilities reported in this thesis, one can observe Figure 2. It shows for 1000 runs of the MCS the estimated conditional ATT of variation 6 without regularization. The highest density of the distribution is around 0, which is the "true" effect. In Appendix Figure 3 one can see the distribution of the neural network implementation used throughout the thesis. In this case, the estimated conditional ATT never reaches the average upper or lower bound, thus indicating the extreme coverage probabilities of 1. The distribution of the estimates is very dense and narrow as comparable to the results of Farrell, T. Liang, and Misra (2021b). Note that the application used throughout the thesis does not recover the true treatment effect shown in Appendix Figure 3 even though reporting much lower bias and variance than variation 6.

These results show that there is still no clear understanding of how to select optimal deep learning models for inference. The comparison of the neural network architectures even implies a strong sensitivity towards selected hyperparameters. Especially the choice

of regularization has a substantial effect on the simulation results. For selecting suitable hyperparameters frameworks like TensorFlow® offer grid search approaches that can help selecting hyperparameters for simulations but for observational data, the choice of optimal neural networks remains unclear.

5 Application

An early application of DiD is the paper of Meyer, Viscusi, and Durbin (1995) who investigate the effect of a workers' compensation reform on their time out of work. In 1980, the states of Kentucky and Michigan substantially increased the compensations in case of work-induced disability or injury. As the policy affected high-earning workers, Meyer, Viscusi, and Durbin (1995) took low-earning workers as a control group. Their idea was that low- and high-earning workers are comparable except that high-earning workers are treated with the compensation policy. The distribution of the pre-treatment out-of-work duration for low- and high-earning workers can be seen in the Appendix Figure 5. In their original study, they report a significant increase in time out of work for Kentucky but not for Michigan.

Meyer, Viscusi, and Durbin (1995) implemented a classical 2x2 DiD design, which makes it suitable to the methods discussed in this thesis. Due to the low sample size of the Michigan data, the analysis is solely focused on Kentucky. Therefore, the DiD identification strategy can be formulated as follows:

$$(14)\quad \text{Duration}_{it} = \alpha + \beta_1 \text{Post}_t + \beta_2 \text{HighEarnings}_i + \beta_3 (\text{Post}_t \times \text{HighEarnings}_i) + \gamma X_{it} + \epsilon_{it},$$

where the interaction $(\text{Post}_t \times \text{HighEarnings}_i)$ is the DiD estimator. X_{it} is a vector of control variables such as injury type, age, or gender. As Meyer, Viscusi, and Durbin (1995) use many of these pre-treatment covariates like age or gender it implicates that they assume conditional PTA. By design, it is not possible to test for PTA but the conditional PTA

seems to be a more robust assumption in an observational study (Sant'Anna and Zhao, 2020). A second remark is towards heterogeneity in the treatment group, which is given in almost all contexts (Farrell, T. Liang, and Misra, 2021a). The Appendix Table 6 shows the difference in duration of out-of-work time across injury types before and after the treatment. The magnitude of the differences is quite large, hinting towards heterogeneity in the treatment group. Based on that, I conducted a regression exclusion test following Hansen (2022) where the results can be seen in Table 7 in the Appendix. The regression exclusion test is significant, which implies that homogeneous treatment effects cannot be assumed.

These results are reason to apply the DRDiD + DL[5] estimator to the data of Meyer, Viscusi, and Durbin (1995). To compare different designs, I also estimate a saturated dummy design without controls and the regression equation 14 with controls and interactions. The estimates are reported in Table 5. Note that the results of the regression equation 14 are slightly different from the results of Meyer, Viscusi, and Durbin (1995). This should be due to different handling of the data-cleaning process than to the different estimation methods.

In Table 5, one can see that the results for all parametric models are quite similar. Note that the results of the saturated design are similar to the regression equation 14 implying that the controls might not have a large impact on the results. Assuming a homogenous treatment effect this result would normally imply that the parametric model is robust and sufficient for estimation. Due to the potential heterogeneity in the treatment group, the DRDiD + DL estimator is arguably the most robust model. One can see that the DRDiD + DL estimator is still able to retrieve significant results as the parametric models, even though the standard errors are slightly larger. The magnitude of the ATT is also larger than in the other models, implying an even bigger effect of the compensation policy in Kentucky than previously assumed.

[5]Here I use the prebuilt software implementation of Bach, Kurz, Chernozhukov, Spindler, and Klaassen (2024) to be able to report summary statistics for the DRDiD + DL estimation.

Table 5: Regression Results

Model	Coef.	Std.Err.	t	P>|t|	[0.025	0.975]
Saturated design	0.191	0.069	2.782	0.005	0.056	0.325
Regression Eq. (14)	0.172	0.064	2.694	0.007	0.047	0.297
DRDiD	0.338	0.247	1.368	0.171	-0.146	0.823
DRDiD + DL	0.250	0.075	3.34	0.000	0.103	0.396
Authors' model	0.162	0.059	2.745	0.006	0.046	0.278

Notes: In this table are reported the results of a saturated dummy design without controls, the regression equation (14) with controls and interactions. The DRDiD and DRDiD + DL estimation including controls and the results from Meyer, Viscusi, and Durbin (1995) of Table 6 Column (ii) are shown. Reported are the treatment coefficients, the standard errors, the t-values, P>|t| is the p-value, and the lower- and upper bound of the 95 percent confidence interval. The dependent variable is the log of the duration of work leave. The dataset is taken from the online resources of Wooldridge (2019). The sample size is $n = 5347$.

Even though the DRDiD + DL is more robust to unspecified models and heterogeneity in the treatment group, there is the potential issue of overfitting. Note that the neural network used in Table 5 has the same architecture as in the simulation study. Contrary to the simulation study, the neural network reports in this application a higher validation loss (0.467) than training loss (0.445). Even though the size of the difference is small, it is a hint towards overfitting. Arguably the issue of heterogenous treatment effects is more severe than the issue of overfitting.

Finally, the results of the DRDiD + DL estimator are quite promising and emphasize more applications of deep learning in observational studies. In settings with large n, conditional PTA, and potential heterogeneity in the treatment group, the DRDiD + DL estimator seems to be a robust and efficient alternative.

6 Further Research

This thesis cannot do justice to all aspects of semiparametric estimation with deep learning, such that there are multiple ways to extend the results in further research. One possible extension is to apply the aforementioned methods to repeated cross-sectional data. Although panel data is the preferred data structure for causal inference due to its generally lower variance, cross-sectional data is often the only available option (Wooldridge, 2010). Nonetheless, Sant'Anna and Zhao (2020) and Manfe and Nunziata (2023) demonstrate the usefulness of implementing DiD on repeated cross-sectional data. Applying deep learning to this data structure could be a promising approach to estimating causal effects.

Another extension is to move away from the classical 2x2 DiD setting and apply the deep learning approach to multiple periods or groups. A natural extension would be the use of the work of Callaway and Sant'Anna (2021), which extend the results of Sant'Anna and Zhao (2020) that form the basis of this thesis. The main contributions of Callaway and Sant'Anna (2021) are the application of the OR, IPW, and DRDiD methods to multiple periods and groups. Accounting for conditional PTA and heterogenous treatment effect in this setting would make the application of deep learning particularly interesting.

The work of De Chaisemartin and d'Haultfoeuille (2024) would be another interesting extension to apply deep learning. Similar to Callaway and Sant'Anna (2021), they extend the DiD estimator to multiple periods and groups but focus on non-binary, non-absorbing treatments with lags.

A major criticism of the deep learning approach is the arbitrary choice of architecture and hyperparameters. There is still a lack of understanding of the inherent structure of deep learning and how to choose the right architecture. It is unclear how the number of hidden layers, the number of neurons, or the choice of activation function impacts the results. Farrell, T. Liang, and Misra (2021b) discuss the unclear effect of l2 regularization on deep learning in inference, even though often used in practice. In Section 4.4, I discuss that the choice of hyperparameters can influence the magnitude of the results. More research needs

to be done to give guidance on how to choose the right architecture and hyperparameters for deep learning in causal inference.

Even though it is unclear how the size of the architecture influences the outcome of the neural network, the more hidden layers and neurons a neural network has, the more computationally costly it is (Thompson et al., 2020). For economists, this issue has been a minor concern in the past, but with the advent of deep- and machine learning, it has become more important. Especially for large and complex data sets, the computational cost of deep learning can be demanding and computation time extensive. Further research is needed to understand how to make deep learning computationally more efficient as suggested by Farrell, T. Liang, and Misra (2021b).

Finally, a relatively new approach using deep learning for inference is the direct estimation of treatment effects. Instead of incorporating deep learning within a semiparametric framework, it is used directly to recover parameter functions, as suggested by the work of Farrell, T. Liang, and Misra (2021a). This approach allows for second-stage inference, such as estimating how treatment impacts evolve over time or across different subgroups. Incorporating direct estimation of ATT with deep learning, rather than using it solely for first-stage estimation, could provide an interesting extension for estimating causal effects.

7 Conclusion

In this thesis, I have revised common semiparametric DiD estimators and implemented deep learning as a novel approach to estimate the ATT. In the context of conditional PTA and panel data, I have compared the OR, IPW, and DRDiD first-step estimation approaches. The results indicate that the DRDiD estimator outperforms the other estimators in terms of bias and variance under homogenous treatment effect. The deep learning implementation of the DRDiD estimator achieves nearly as good results as the traditional DRDiD estimator. Under heterogeneous treatment effects, the DRDiD with deep learning outperforms all other comparable estimators. To illustrate these methods, I conducted a simulation study

and applied the methods to an empirical data set. The results of Kentucky's worker's compensation data suggest that the DRDiD estimator with deep learning is a promising approach to estimate the ATT under heterogenous treatment effects.

The results underline the potential of deep learning in causal inference, especially in cases of complex and large data. Particularly under strong heterogeneous treatment effects, deep learning seems to be an advisable approach. However, the shortcoming is the arbitrary choice of the deep learning architecture and suitable settings. Further research is needed to apply deep learning for DiD estimation on repeated cross-sectional data or on multiple time and group period cases.

Appendix

Figure 3: Conditional Average Treatment on the Treated Effekt with Regularization

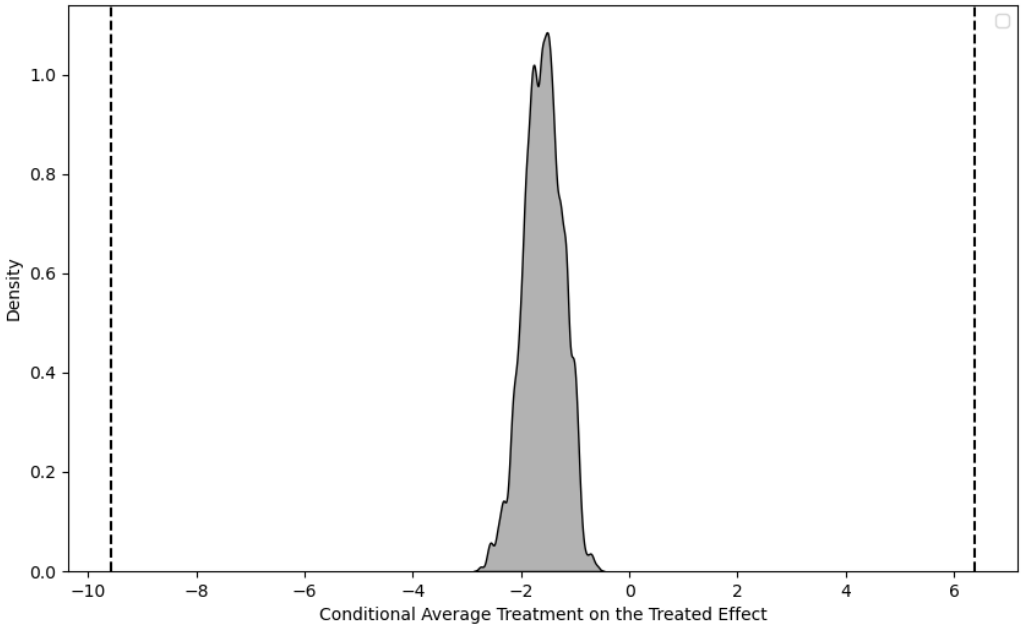


Figure 4: Example of a ReLU Activation Function

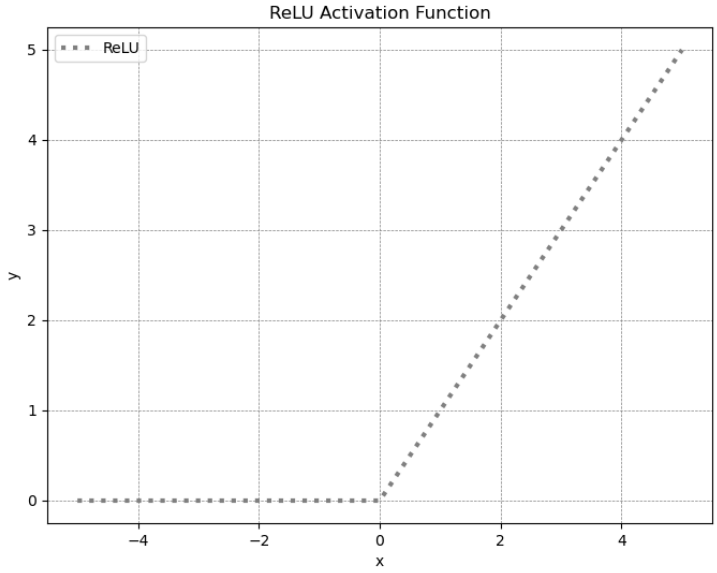


Figure 5: Distribution of Log Leave Duration by Earnings Category

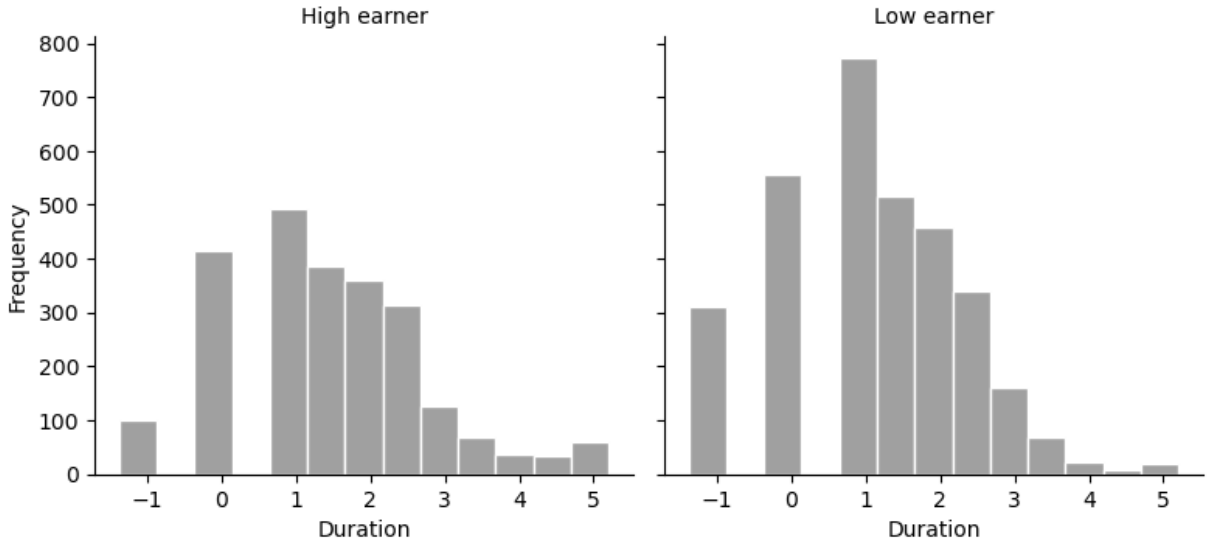


Table 6: Comparison of Duration Across Injury Types Before and After 1980

Injury Type	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
after_1980 = 0	778.25	282.00	1993.75	1663.25	5806.0	2649.75	194.25	413.5
after_1980 = 1	771.50	815.25	2341.75	1997.75	5362.5	2816.25	305.00	559.5
Difference	-6.75	533.25	348.00	334.50	-443.5	166.50	110.75	146.0

Notes:
Injury Type: Categories of injuries.
after_1980 = 0: Duration of work leave before 1980.
after_1980 = 1: Duration of work leave after 1980.
Difference: Difference in duration values between after_1980 = 1 and after_1980 = 0.

Table 7: Regression Exclusion Test

Residual Degrees of Freedom	Sum of Squared Residuals	Degrees of Freedom Difference	Sum of Squares Difference	F	Pr(>(F))
2385.0	1.927912e+06	7.0	32812.658	5.798	0.000

Notes: The table reports the results of the regression exclusion test. The test compares the full model with the restricted model, excluding the variable of interest.

References

Abadie, Alberto (2005). "Semiparametric Difference-in-Differences Estimators". In: *The Review of Economic Studies* 72.1, pp. 1–19.

Abuqaddom, Inas, Basel A Mahafzah, and Hossam Faris (2021). "Oriented stochastic loss descent algorithm to train very deep multi-layer neural networks without vanishing gradients". In: *Knowledge-Based Systems* 230, p. 107391.

Angrist, Joshua D and Jörn-Steffen Pischke (2009). "Mostly harmless econometrics: An empiricist's companion". In: *Princeton university press*.

Bach, Philipp, Malte S. Kurz, Victor Chernozhukov, Martin Spindler, and Sven Klaassen (2024). "DoubleML: An Object-Oriented Implementation of Double Machine Learning in R". In: *Journal of Statistical Software* 108.3, pp. 1–56.

Belloni, Alexandre, Victor Chernozhukov, Ivan Fernandez-Val, and Christian Hansen (2017). "Program evaluation and causal inference with high-dimensional data". In: *Econometrica* 85.1, pp. 233–298.

Callaway, Brantly and Pedro H.C. Sant'Anna (2021). "Difference-in-Differences with Multiple Time Periods". In: *Journal of Econometrics* 225.2, pp. 200–230.

Chernozhukov, Victor et al. (2018). "Double/Debiased Machine Learning for Treatment and Structural Parameters". In: *The Econometrics Journal* 21.1, pp. C1–C68.

Chernozhukov, Victor, Whitney K Newey, and Rahul Singh (2022a). "Automatic debiased machine learning of causal and structural effects". In: *Econometrica* 90.3, pp. 967–1027.

— (2022b). "Automatic Debiased Machine Learning of Causal and Structural Effects". In: *Econometrica* 90.3, pp. 967–1027.

De Chaisemartin, Clément and Xavier d'Haultfoeuille (2024). "Difference-in-differences estimators of intertemporal treatment effects". In: *Review of Economics and Statistics*, pp. 1–45.

Farrell, Max H., Tengyuan Liang, and Sanjog Misra (2021a). "Deep Learning for Individual Heterogeneity: An Automatic Inference Framework". In: *Cemmap working paper*.

— (2021b). "Deep Neural Networks for Estimation and Inference". In: *Econometrica* 89.1, pp. 181–213.

Goodfellow, Ian, Yoshua Bengio, and Aaron Courville (2016). "Deep Learning". In: *MIT Press*.

Goodman-Bacon, Andrew (2021). "Difference-in-Differences with Variation in Treatment Timing". In: *Journal of Econometrics* 225.2, pp. 254–277.

Hansen, B. (2022). "Econometrics". In: *Princeton University Press*.

Heckman, James J., Hidehiko Ichimura, and Petra Todd (1998). "Matching As An Econometric Evaluation Estimator". In: *Review of Economic Studies* 65.2, pp. 261–294.

Hernan, Miguel A and James M Robins (2024). "Causal Inference: What If". In: *CRC Boca Raton, FL*.

Kang, JD et al. (2007). "Demystifying double robustness: A comparison of alternative strategies for estimating a population mean from incomplete data". In: *Statistical Science* 22, pp. 569–573.

Koh, Pang Wei and Percy Liang (2017). "Understanding black-box predictions via influence functions". In: *International conference on machine learning*. PMLR, pp. 1885–1894.

Manfe, Tommaso and Luca Nunziata (2023). "Difference-In-Difference Design With Repeated Cross-Sections Under Compositional Changes: A Monte-Carlo Evaluation of Alternative Approaches". In: *Working Paper*.

Meyer, Bruce D, W Kip Viscusi, and David L Durbin (1995). "Workers' Compensation and Injury Duration: Evidence from a Natural Experiment". In: *The American Economic Review* 85.3, p. 322.

Sant'Anna, Pedro H.C. and Jun Zhao (2020). "Doubly Robust Difference-in-Differences Estimators". In: *Journal of Econometrics* 219.1, pp. 101–122.

Schmidt-Hieber, Johannes (2020). "Nonparametric regression using deep neural networks with ReLU activation function". In: *The Annals of Statistics* 48.4, pp. 1875–1897.

Telgarsky, Matus (2016). "Benefits of depth in neural networks". In: *Conference on learning theory*. PMLR, pp. 1517–1539.

Thompson, Neil C, Kristjan Greenewald, Keeheon Lee, and Gabriel F Manso (2020). "The computational limits of deep learning". In: *arXiv preprint arXiv:2007.05558* 10.

Wooldridge, Jeffrey M (2010). "Econometric analysis of cross section and panel data". In: *MIT press*.

— (2019). "Introductory Econometrics: A Modern Approach 7e". In: *Cengage AU*.

Zimmert, Michael (2018). "Efficient difference-in-differences estimation with high-dimensional common trend confounding". In: *arXiv preprint arXiv:1809.01643*.

"I hereby confirm that the work presented has been performed and interpreted solely by myself except for where I explicitly identified the contrary. I assure that this work has not been presented in any other form for the fulfillment of any other degree or qualification. Ideas taken from other works in letter and in spirit are identified in every single case."

July 26, 2024

Norman Metzinger