Double Machine Learning for Difference in Difference: Fundamentals and Application

Martín Gabriel Cargnel

Nov, 2025

Table of contents

Summary

Machine Learning (ML) models have traditionally been associated with prediction tasks due to their flexibility, while social scientists have typically relied on simpler, often linear, regressions for assessing causality. However, a novel framework named Double Machine Learning (DML) has emerged, providing a way to leverage the predictive performance of these complex methods for robust causal estimation. This thesis examines the fundamentals and applications of double machine learning for two distinct popular Difference-in-Differences (DiD) settings.

Keywords: Double Machine Learning, Causal Inference, Difference-in-Differences

1 Introduction

As discussed in (Shmueli, 2010), when working in applied statistics there is a clear distinction between "predicting" and "explaining." Prediction is usually associated with achieving the best performance on a selected goodness-of-fit metric, while explaining is focused on understanding the effects or relationships between variables. This distinction makes it clear why flexible, and often considered "black-box," Machine Learning (ML) algorithms are widely used when the goal is to achieve the best prediction performance. However, when the goal is to interpret results or to draw causal relationships, other, usually simpler approaches like linear regressions are preferred. This distinction is not new, as it was highlighted by (Breiman, 2001), who clearly differentiated between statistics and machine learning when the latter field started to gain popularity.

The focus on prediction and not explanation generated criticism among economists and other social scientists who needed tools for understanding and studying causal relationships, which is not possible with black-box models. However, the new models were appealing, so in recent years, many authors have explored how to bridge this gap. Early work, such as (Varian, 2014), highlighted how ML techniques, especially tree-based models, could complement traditional econometric methods in settings with non-linearities and complex interactions. Similarly, (Mullainathan & Spiess, 2017) explored the practical applications of machine learning in econometrics, particularly emphasizing prediction and cautioning against drawing causal conclusions about the effects of independent variables without careful consideration.

Although the prevailing view cautioned against using machine learning for explaining, these algorithms gained popularity among researchers over the years. As shown in (Desai, 2023), which reviews how ML algorithms are being integrated into economic analysis, this popularity has grown significantly.

An attempt to bridge the gap between black-box, prediction-focused machine learning methods and simpler, more interpretable methods was the emergence of interpretable machine learning. There are now several techniques that aim to combine the predictive power of complex, "black box" machine learning models with methods to interpret them. A comprehensive overview of these techniques can be found in (Molnar, 2025). However, it is important to note that many of these methods focus on describing the behavior of the model itself, rather than uncovering the underlying data generating process (DGP) as is common in classical statistics.

This work focuses its attention on a particularly influential framework from (Chernozhukov et al., 2018): Double Machine Learning (DML). The DML framework provides a general, robust method that formally combines the predictive power of machine learning with the theoretical

rigor of causal inference to estimate a specific parameter of interest. The DML framework can be adapted to multiple familiar settings for economists, such as Instrumental Variables (IV) and, central to this thesis, Difference-in-Differences (DiD). This remains a flourishing area of research. Therefore, this document aims to provide an accessible introduction to this powerful technique for practitioners.

The rest of this document is structured as follows: First, we introduce the classic Difference-in-Differences framework and its econometric tools. Second, we introduce the Double Machine Learning (DML) framework and detail its specific application for DiD setups. Finally, we provide two real-world applications of this algorithm to demonstrate its practical utility.

2 Difference in Differences

Difference in Differences (DiD) is a widely used econometric technique for estimating causal effects when randomized experiments are not feasible. It is particularly useful in policy analysis, economics, and social sciences to evaluate the impact of a treatment or intervention over time. Essentially, the DiD approach compares the changes in outcomes over time between a group that is exposed to a treatment (the treatment group) and a group that is not (the control group). The key idea is to control for unobserved factors that are constant over time and for common trends affecting both groups. An excellent introduction to the method can be found in (Cunningham, 2021).

2.1 The DiD Estimator

Suppose we observe two groups over two periods: before and after a treatment is implemented. The DiD estimator is calculated as:

$$\hat{\delta}^{t,c} = (Y^t_{post} - Y^t_{pre}) - (Y^c_{post} - Y^c_{pre}) \tag{2.1}$$

where:

- Y^t_{post}: Average outcome for the treatment group after the intervention
- Y^t_{pre}: Average outcome for the treatment group before the intervention
- Y^c_{post}: Average outcome for the control group after the intervention
- Y^c_{pre}: Average outcome for the control group before the intervention

This double differencing removes biases from permanent differences between the groups and from trends that affect both groups equally and can be seen as the average treatment effect , defined as

$$ATT = E[Y^{(1)} - Y^{(0)}|D = 1] \tag{2.2}$$

where $Y^{(1)}$ would be the potential outcome if treated, $Y^{(0)}$ the potential outcome if not treated, and $D \in \{0, 1\}$ the treatment indicator, with $D = 1$ if treated and $D = 0$ if not. So it's the expected treatment effect for the units that actually received the treatment.

2.1.1 Estimation

DiD models are often estimated using regression analysis, typically with a specification like:

$$Y_{it} = \alpha + \beta \text{Post}_t + \gamma \text{Treat}_i + \delta(\text{Post}_t \times \text{Treat}_i) + \epsilon_{it}$$

where:

- Y_{it}: Outcome for unit i at time t
- Post_t: Indicator for the post-treatment period
- Treat_i: Indicator for the treatment group
- δ: The DiD estimator (treatment effect)

2.2 Assumptions

The main identifying assumption of DiD is the parallel trends assumption: in the absence of treatment, the average change in the outcome would have been the same for both groups. If this assumption holds, the DiD estimator provides an unbiased estimate of the treatment effect.

A nice way to see this is by working with (Equation 2.1), expanding it to

$$\hat{\delta}^{t,c} = (E[Y^t|post] - E[Y^t|pre]) - (E[Y^c|post] - E[Y^c|pre])$$

After some algebra, we can end up with this expression:

$$\hat{\delta}^{t,c} = (E[Y^{t,1}|post] - E[Y^{t,0}|post]) + [E[Y^{t,0}|post] - E[Y^{t,0}|pre]] - [E[Y^{c,0}|post] - E[Y^{c,0}|pre]]$$

So, in this decomposition, we can see that the first term corresponds to the ATT estimator (Equation 2.2). Please note that the superscripts denote whether the group corresponds to the treated (t) or control (c), and whether it was treated (1) or not (0).

But the second and third terms get cancelled out if the parallel trends assumption holds, basically because it's saying that if the group that received the treatment and the group that didn't wouldn't receive the treatment, then both would be equal before and after the treatment. So the terms would be cancelled out, and we would only have the ATT.

A popular way to validate this assumption is to use a parallel trend plot. This visualization allows us to evaluate how the dependent variable evolves for the control and treatment groups before and after the treatment. An example with simulated data can be found in Figure 2.1, where we can see that both control and treatment units behave similarly before the treatment

(denoted by a vertical red dotted line) but differ after it. On the other hand, Figure 2.2 is an example of a plot where the assumption does not hold, because the trends for the two groups are not parallel before the treatment. Meaning that the groups are not comparable.

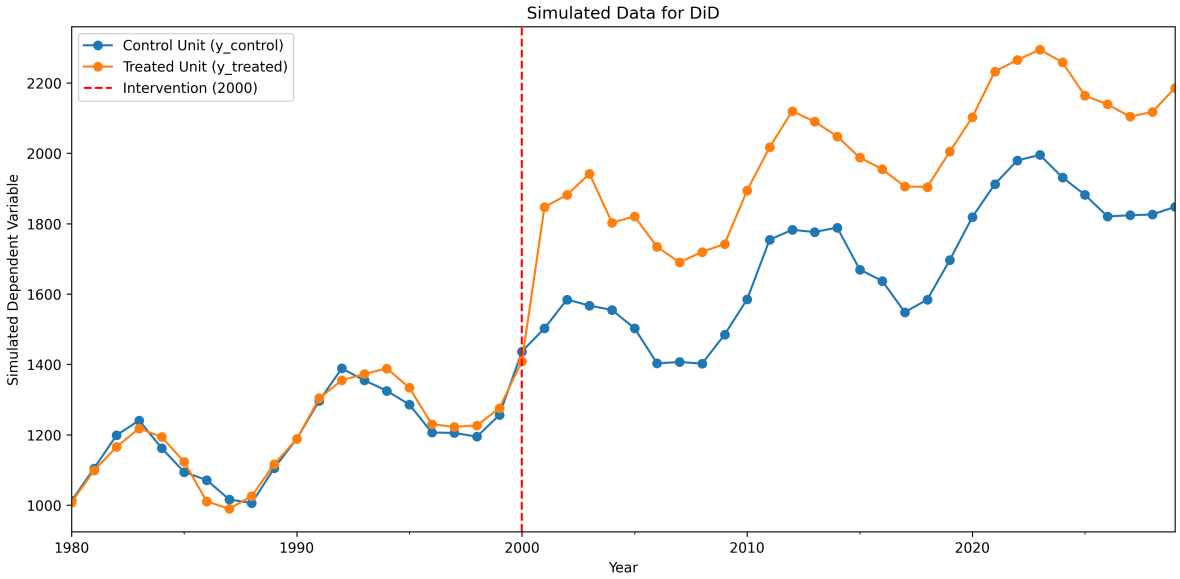

Figure 2.1: Example Did

2.3 Extension: Staggered DID

In many empirical applications, treatments are not implemented at the same time for all treated units. Instead, different units receive the treatment at different points in time—a situation known as staggered adoption. The standard two-period DiD framework does not account for this complexity, so extensions are needed.

2.3.1 Estimation

A common approach is to use a two-way fixed effects (TWFE) regression:

$$Y_{it} = \alpha_i + \lambda_t + \delta D_{it} + \epsilon_{it}$$

where:

- Y_{it}: Outcome for unit i at time t
- α_i: Unit fixed effects

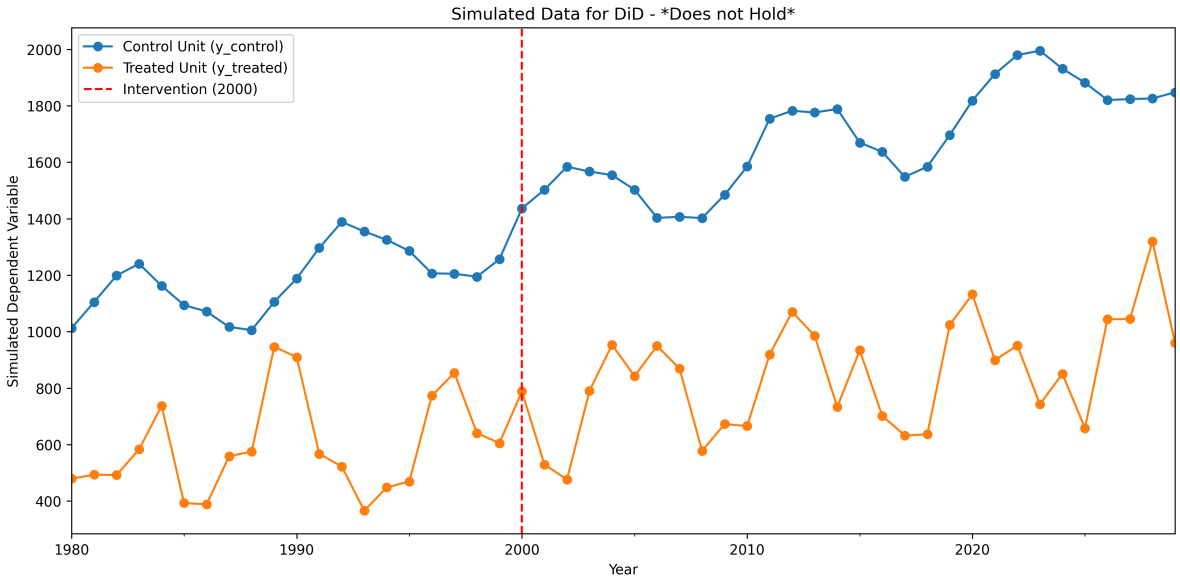

Figure 2.2: Example Did not val

- λ_t: Time fixed effects
- D_{it}: Indicator for whether unit i is treated at time t
- δ: Average treatment effect

2.3.2 Limitations and Recent Advances

Recent research, mainly pioneered by the decomposition demonstrated in (Goodman-Bacon, 2021), has shown that the TWFE estimator can be seen as a weighted average of all potential 2x2 DD estimates, where weights are based on both group sizes and variance in treatment. However, this decomposition revealed that TWFE can produce biased estimates when treatment effects are heterogeneous across groups or over time in a staggered design. This is because the estimator may compare already-treated units to newly-treated units, contaminating the control group. Also, it assumes that groups in the middle of the panel should be weighted more than those at the end.

To address these issues, alternative estimators have been developed by different authors. However, in this paper, we will be focusing on the proposal from (Callaway & Santa'Anna, 2021), who propose a reliable way to estimate staggered DiD.

2.4 Extensions to covariates

The standard parallel trends assumption can be restrictive. In many settings, it may be more plausible to assume conditional parallel trends: the trends between the treated and control groups would be parallel, *conditional on* a set of covariates X.

Including covariates can thus strengthen the validity of the DiD design. In a traditional regression framework, this is done by simply adding the covariates X_{it} to the estimation equation:

$$Y_{it} = \alpha + \beta \text{Post}_t + \gamma \text{Treat}_i + \delta(\text{Post}_t \times \text{Treat}_i) + \theta' X_{it} + \epsilon_{it}$$

This model is often estimated as a fixed-effects model (similar to the TWFE specification) to control for time-invariant unobservables:

$$Y_{it} = \alpha_i + \lambda_t + \delta D_{it} + \theta' X_{it} + \epsilon_{it}$$

A limitation of this approach is that it assumes the covariates X_{it} have a linear and additive effect on the outcome Y_{it}. If the true relationship is non-linear or involves complex interactions, this model is misspecified, and the estimate of δ can be biased.

This limitation provides a key motivation for using machine learning. The Double Machine Learning (DML) framework, as we will discuss in the next chapter, is designed to overcome this exact problem. It allows us to control for a rich set of covariates X_{it} in a flexible, non-parametric way, thereby avoiding the biases associated with model misspecification.

However, the practitioner must be careful when including covariates as might introduce confounding or collider bias.

3 Double Machine Learning

Over the years, Machine Learning approaches were relegated to prediction tasks. Mainly because of their flexibility, they achieve great predictive performance for high dimensional datasets. But when it comes to interpretation, which is what usually economists and social scientists look for, then is not that helpful because interpretation and finding causal relationships is the key. In that regard, interpretable machine learning could be considered an intermediate point, whith a battery of methods for interpreting the complex machine learning algorithms results. A great review of these methods can be found in (Molnar, 2025), but in reality they are very much tied to correlations and not to understand causal relationships.

That's why a new framework was introduced by (Chernozhukov et al., 2018) where using Frish Waigh Lovell theorem end up using the flexibility of machine learning models for estimating causal relationships. In this chapter, I'll introduce the framework more formally and then present how to apply it for differences in differences including the extension to staggered.

3.1 Framework

3.1.1 The Goal: Estimating the Average Treatment Effect (ATE)

The primary goal is to estimate the causal effect of a treatment D on an outcome Y, controlling for a set of covariates X. We assume a constant treatment effect, θ, which represents the Average Treatment Effect (ATE).

$$ATE = E[Y_i(1) - Y_i(0)]$$

Where $Y_i(1)$ is the potential outcome for unit i if treated, and $Y_i(0)$ is the potential outcome if untreated. The fundamental problem of causal inference is that we can only observe one of these potential outcomes for each unit.

To model this, we use a Partially Linear Model (PLM), which is a common setup for DML:

$$\begin{aligned} Y_i &= \theta D_i + g(X_i) + \epsilon_i \quad \text{(Outcome Model)} \\ D_i &= m(X_i) + u_i \quad \text{(Treatment Model)} \end{aligned}$$

Where:

- Y_i is the observed outcome.
- D_i is the observed treatment status (e.g., 1 if treated, 0 if not).
- X_i is a vector of covariates.
- θ is the causal parameter of interest (the ATE, assuming a constant effect).
- $g(X_i)$ and $m(X_i)$ are unknown, potentially complex functions, known as "nuisance functions", that represent how the covariates X affect the outcome and the treatment, respectively.
- ϵ_i and u_i are error terms, which we assume are exogenous (i.e., $E[\epsilon_i|X_i, D_i] = 0$ and $E[u_i|X_i] = 0$).

3.1.2 The Problem: Confounding Bias

We cannot simply estimate θ by regressing Y on D. The covariates X introduce confounding bias (a form of omitted variable bias) because they affect both the treatment D (via $m(X)$) and the outcome Y (via $g(X)$).

We can visualize this confounding path using a Directed Acyclic Graph (DAG):

To get an unbiased estimate of θ, we must "control for" or "partial out" the influence of X.

3.1.3 The Theoretical Solution: Orthogonalization

The DML framework builds on the Frisch-Waugh-Lovell (FWL) theorem. The theorem shows how to estimate a parameter in a multivariate regression by first residualizing all variables. We can apply this logic to our PLM.

The goal is to find an estimating equation for θ that is no longer dependent on the nuisance functions $g(X)$ and $m(X)$. We can derive this by "partialling out" X from Y and D.

Start with the outcome model: $Y_i = \theta D_i + g(X_i) + \epsilon_i$ and take the conditional expectation of Y_i given X_i:

$$E[Y_i|X_i] = E[\theta D_i + g(X_i) + \epsilon_i|X_i]$$

Assuming $E[\epsilon_i|X_i] = 0$ and since $g(X_i)$ is a function of X_i, $E[g(X_i)|X_i] = g(X_i)$:

$$E[Y_i|X_i] = \theta E[D_i|X_i] + g(X_i)$$

This gives us an expression for the confounder $g(X_i)$:

$$g(X_i) = E[Y_i|X_i] - \theta E[D_i|X_i]$$

Now, substitute this expression for $g(X_i)$ back into the original outcome model:

$$Y_i = \theta D_i + (E[Y_i|X_i] - \theta E[D_i|X_i]) + \epsilon_i$$

Finally, rearrange the terms to isolate Y and D from their conditional expectations:

$$Y_i - E[Y_i|X_i] = \theta(D_i - E[D_i|X_i]) + \epsilon_i$$

Let's define our residuals: $\tilde{Y}_i = Y_i - E[Y_i|X_i]$ (The "residualized" outcome) and $\tilde{D}_i = D_i - E[D_i|X_i]$ (The "residualized" treatment). Then our equation becomes:

$$\tilde{Y}_i = \theta \tilde{D}_i + \epsilon_i$$

This is the key insight. We have transformed the complex PLM into a simple linear regression. If we could get the true residuals $\tilde{Y}_i$ and $\tilde{D}_i$, we could estimate θ without bias using a simple regression of $\tilde{Y}$ on $\tilde{D}$.

3.1.4 The Practical Implementation: Double Machine Learning

In reality, we do not know the true conditional expectation functions $E[Y|X]$ and $E[D|X]$. The innovation of Double Machine Learning is to use flexible, high-performance machine learning models to estimate them.

So, let $\hat{l}(X_i)$ be an ML-based estimate of $E[Y_i|X_i]$ that we can estimate as a standard regression task (since Y is often continuous) and let $\hat{m}(X_i)$ be an ML-based estimate of $E[D_i|X_i]$ that is usually estimated as a classification task given that D is binary. In this case $\hat{m}(X_i)$ is an estimate of the propensity score, $P(D_i = 1|X_i)$.

This is the "double" in DML: we use machine learning to estimate the nuisance functions for both the outcome and the treatment models. We can use any suitable ML model, such as Random Forests, Gradient Boosting Machines, or Neural Networks.

We then compute the estimated residuals:

$$\hat{Y}_i = Y_i - \hat{l}(X_i) \quad \text{and} \quad \hat{D}_i = D_i - \hat{m}(X_i)$$

And finally, we estimate θ using the simple linear regression:

$$\hat{Y}_i = \theta \hat{D}_i + \hat{\epsilon}_i$$

3.1.5 Addressing Machine Learning Biases for Valid Inference

Using flexible ML models to estimate nuisance functions introduces two main statistical challenges that could invalidate our final estimate of θ: overfitting bias and estimation bias (e.g., from regularization). The DML framework employs two crucial techniques to solve these problems and ensure our final estimate is statistically valid.

3.1.5.1 Cross-Fitting: Solving Overfitting Bias

If we use the same data observations to train the ML models ($\hat{l}$ and $\hat{m}$) and to estimate the final parameter θ, our estimate will be biased. This is a form of overfitting, where the generated residuals ($\hat{Y}_i, \hat{D}_i$) would have a spurious correlation simply because the model was optimized using those same Y_i and D_i values. The solution is cross-fitting (or sample splitting). This procedure ensures that the residuals for any given observation are generated by a model that was not trained on that same observation. This "breaks" the overfitting link.

While K-fold cross-fitting is standard, the process is easiest to understand with a 2-fold split:

1. Split: Randomly partition the dataset into two equal halves (e.g., Fold 1 and Fold 2).
2. Train on Fold 1, Predict on Fold 2
    - Train the ML models $\hat{l}_1$ and $\hat{m}_1$ using only the data in Fold 1.
    - Use these trained models to generate residuals ($\hat{Y}_i = Y_i - \hat{l}_1(X_i)$, $\hat{D}_i = D_i - \hat{m}_1(X_i)$) for the data in Fold 2.
3. Train on Fold 2, Predict on Fold 1:
    - Train new models $\hat{l}_2$ and $\hat{m}_2$ using only the data in Fold 2.
    - Use these models to generate the residuals for the data in Fold 1.
4. Estimate: Combine the residuals generated in step 2 (for Fold 2) and step 3 (for Fold 1) into one complete dataset.
5. Run the final, simple OLS regression $\hat{Y}_i = \theta \hat{D}_i + \hat{\epsilon}_i$ on this combined set of residuals to get the single, unbiased estimate of θ.

3.1.5.2 Neyman Orthogonality: Solving Estimation Bias

ML models (like Random Forest or Lasso) are designed for optimal prediction, not for unbiasedly estimating the true functions $l(X)$ and $m(X)$. Their estimates, $\hat{l}$ and $\hat{m}$, will inevitably contain some "estimation bias" (e.g., from regularization). We must ensure that this bias in our nuisance function estimates does not "contaminate" or "leak into" our final estimate of θ.

The solution lies in the structure of our final estimating equation: $\tilde{Y}_i = \theta \tilde{D}_i + \epsilon_i$. This specific equation, derived from the FWL theorem, possesses a critical property known as Neyman Orthogonality.

This property means that the final estimate of θ is first-order insensitive to small errors or biases in the estimation of the nuisance functions $l(X)$ and $m(X)$. Because we have residualized both Y (which depends on $l(X)$) and D (which depends on $m(X)$), the estimation errors in $\hat{l}$ and $\hat{m}$ effectively cancel each other out, leaving our estimate of θ asymptotically unbiased.

This orthogonality is the key theoretical property that allows DML to work: it permits us to use "imperfect" but powerful ML models for the complex prediction tasks, while still achieving a statistically valid (unbiased and asymptotically normal) estimate for our single causal parameter of interest, θ.

3.2 DML for DID

This framework can be adapted for Difference-in-Differences (DID) settings, where we want to estimate the Average Treatment Effect on the Treated (ATT) in a panel data context with treatment and control groups over time. In this section we will show how can be used when treatments occur at the same time and in the next one we will be extending that case for the staggered case (e.g. when treatments can occur at different times). For the more simple scenario, we will be following (Chang (2020)), where it proposed an adjustment to create a score function that is Neyman-Orthogonal. In essence, the paper proposed:

Given Y_{i0} the pre-treatment outcome, Y_{i1} the post-treatment outcome, D_i the treatment indicator and a vector of covariates X_i we need to calculate ψ_i using the following Neyman Orthogonal formula:

$$\psi_i = \frac{D_i - E[D=1|X]}{E[D](1-(E[D=1|X]))}[(Y_{i1} - Y_{i0}) - E[Y_{i1} - Y_{i0}|D=0, X]]$$

Once we've have this score, we simply need to take the average across observations for the ψ_i

$$\hat{\psi} = \frac{1}{n}\sum_{i=1}^{n} \psi_i$$

Now, if we want to analize the calculation of ψ_i: the first key part is the "Residualized Outcome Change": $(Y_{i1} - Y_{i0}) - E[Y_{i1} - Y_{i0}|D=0, X]$. Here, $(Y_{i1} - Y_{i0})$ is the observed change in the outcome for unit i. The second term, $E[Y_{i1} - Y_{i0}|D=0, X]$, is the nuisance function for the outcome. It's our best machine-learning-based prediction of the outcome change that a unit with covariates X would have experienced if it were in the control group ($D=0$). The entire term thus represents the "unexplained" change in Y. For a treated unit, this is their observed

change minus the change we would have expected based on "parallel trends" derived from the control group with similar X.

The second key part is the "Weighting Term": $(D_i - E[D = 1|X])/(E[D] * (1 - E[D = 1|X]))$. This is the "doubly-robust" weight. It relies on the other two nuisance functions: $E[D = 1|X]$, which is the propensity score (the probability of unit i receiving treatment, given its covariates X), and $E[D]$, which is the unconditional probability of treatment (estimated as the sample average of D_i). This term re-weights the control group to look like the treated group and ensures the entire score is Neyman-orthogonal.

By multiplying these two terms together and averaging them across all observations, we get a high-quality, doubly-robust estimate of the ATT.

To ensure this works properly and to prevent overfitting, the "nuisance functions" must be estimated using cross-fitting. These functions are: $E[D = 1|X]$, $E[D]$, and $E[Y_{i1} - Y_{i0}|D = 0, X]$. Cross-fitting involves splitting the data into several "folds," using some folds to train the machine learning models and other folds to calculate the psi_i scores for the observations that were not used in training.

3.3 DML for Staggered DID

As mentioned before, this framework can be extended to settings when the treatment occurs in different periods for different groups. For example groups that a adopt a policy in different years. For solving the issues commented in the previous chapter, we will be following (Callaway & Santa'Anna (2021)) where they proposed a doubly robust estimator $ATT_{g,t}$ with g the first period when an unit was treated and t the any post-treatment period. With that, they basically compare the treated units against non treated (including not-yet treated) for each g. Because we will have multiple g, we will ended up with multiple estimations for $ATT_{g,t}$ that we will need to aggregate afterwards for getting the $\hat{ATT}$.

More formally we can define the $\hat{ATT}$ as

$$\hat{ATT} = \sum_{g,t} w_{g,t} ATT(g,t)$$

with $w_{g,t} = \frac{N_{g,t}}{\sum_{g',t'} N_{g',t'}}$ being the weights that reflects the groups sized. For estimating $ATT(g,t)$ the authors proposed

$$\begin{aligned} \text{ATT}(g,t) = \frac{1}{n_g} \sum_{i:G_g=1} (Y_{it} - E[Y_t - Y_1|D = 0, X]) \\ - \sum_{i:C=1} \frac{E[D = 1|X]}{1 - E[D = 1|X]} (Y_{it} - E[Y_t - Y_1|D = 0, X]) \end{aligned}$$

4 Application

In this chapter, we present two comparative applications to illustrate how DML estimators compare against traditional estimators for Difference-in-Differences (DiD). We examine two distinct settings: first, a canonical case where the treatment was implemented at a single point in time, and second, a staggered adoption setting where treatment was adopted in different periods. In both cases, we will replicate the empirical results from existing studies, using the statsmodels Python package (Seabold & Perktold (2010)) for the classic estimators and DoubleML (Bach, Chernozhukov, Kurz, & Spindler (2022)) for the Double Machine Learning implementation.

4.1 Difference in Differences with treatment in one period

We begin by reproducing the results from (GonzÃ¡lez (2025)), specifically the initial sections where the author uses the Difference-in-Differences (DiD) method for estimating the effect of fracking on environmental regulatory activities. The dataset contains 143,275 observations from fracking wells at the zip-code-year level. The data focuses on states where the fracking boom was more pronounced: Arkansas, Louisiana, North Dakota, Oklahoma, Pennsylvania, Texas, and Virginia. In the original paper, three different dependent variables are used, all measured at the zip-code-year level: (1) Actions, the total number of environmental activities; (2) Facilities, the total number of facilities that received at least one regulatory action; and (3) Formal, the total number of formal environmental activities.

4.1.1 Replication of Gonzales

The author also included county-level employment and total establishments as control variables. These variables serve as proxies for local economic activity and help isolate the impact of fracking on regulation. It is important to note that while the original paper presented estimations both with and without controls, we focus only on reproducing the estimations that include controls.

More formally and following the structure from the paper, we can express the Two-Way Fixed Effects (TWFE) estimation equation as:

$$\begin{aligned}\log(1 + y_{it}) = \alpha + \delta \text{fracked}_i + \gamma \text{Post } 2005_t \\ + \theta(\text{fracked}_i * \text{Post } 2005_t) + vX_{it} + \mu_i + \nu_t + \epsilon_{it}\end{aligned}$$

where i represents zip codes and t represents years. The dependent variable y it is the log-transformed outcome, using the $log(1 + y_{it})$ transformation to accommodate zero values for Actions, Facilities, and Formal. Following the notation from previous chapters, our coefficient of interest is θ, which represents the DiD estimate. While the model estimates other parameters (e.g., δ, γ, ν), our focus is on θ, as the other terms (including the main effects $fracked_i$ and $Post_{2005t}$) serve to isolate the causal effect. X it is a vector of control variables (employment and establishments). μ_i and ν_t represent zip-code and year fixed effects, respectively. Finally, ϵ it is the error term. Standard errors are clustered at the zip-code level, the unit at which treatment was assigned.

As the specification shows, the treatment $Post_{2005t}$ occurred in the same year (2005) for all treated zip codes. This date was chosen as it marks when technological advancements made fracking broadly profitable. This scenario represents a classic Difference-in-Differences (DiD) setup, for which the Two-Way Fixed Effects (TWFE) specification above is appropriate.

As mentioned in previous sections, this methodology relies on the parallel trends assumption, which assumes that both groups (treated and non-treated) would have followed similar trends in the outcome variable in the absence of the treatment. To visually inspect this assumption, we refer to {Figure 4.1}. This figure shows a replication of the parallel trends plot from (GonzÃ¡lez (2025)) for the 'Actions' outcome, along with corresponding plots for the other two dependent variables. We observe that prior to the technological advancement (marked by the vertical line at 2005), both groups exhibited similar trends for all three outcomes. These trends clearly diverge after 2005, providing visual support for the validity of the parallel trends assumption.

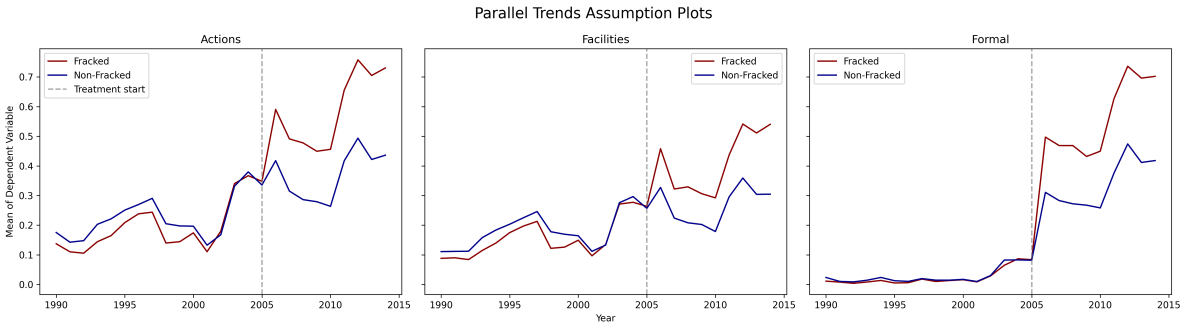

Figure 4.1: Visual check for the parallel trends assumption. Each panel plots the mean trend of a different environmental regulation outcome (lnactionnonoil, lnone_non_oil, and lnstate_formal_nonoil) for the treatment (Fracked) and control (Non-Fracked) groups. The vertical line represents the pre-treatment/post-treatment cutoff in 2005. Parallel pre-treatment trends support the validity of the Difference-in-Differences design.

4.1.2 Double Machine Learning

With this setup, we now turn to the Double Machine Learning approach. We apply the DML-DiD estimator, as formally introduced in a previous chapter, to estimate the causal effect of fracking on regulatory activities in the non-energy sector. This DML-DiD approach, following (Chang (2020)), is also a fixed-effects estimator, just like the PanelOLS model. It accounts for the time-invariant, unit-specific unobserved heterogeneity (i) by operating on the differences in outcomes before and after the treatment period (e.g., $Y_{i,post} - Y_{i,pre}$).

Therefore, the critical difference between the PanelOLS (TWFE) model and the DML-DiD model is not in their handling of fixed effects, but in how they control for the covariates (X):

1. PanelOLS (TWFE): Assumes the controls have a linear and additive effect on the outcome.
2. DML-DiD: Makes no such assumption. It uses flexible machine learning to model the complex, non-linear relationships between the controls and the outcome.

As described in the previous chapter, to estimate the ATT, this model requires the estimation of several nuisance functions using cross-fitting. Based on the score function for the ATT, the key functions to be estimated by our LightGBM models are:

The outcome model: $g_\theta(X) = E[Y_{i1} - Yi0 \mid D = 0, X]$. This function predicts the outcome change that a unit would have experienced had it not been treated, based on its covariates X. This flexibly models the "parallel trends" conditional on X.

The treatment model (Propensity Score): $m(X) = E[D = 1 \mid X]$. This function predicts the probability of a unit being in the treatment group, given its covariates X.

By using machine learning (LightGBM) to estimate these functions, the DML-DiD estimator is robust to potential model misspecification bias that can arise from the rigid linearity assumption of the classic PanelOLS model.

Given the different nature of these functions, we will use a regression model for g (since Y is numeric) and a classification model for m (since D is binary). In this exercise, we have chosen the LightGBM algorithm (Ke et al. (2017)) for both tasks, as this model can handle both regression and classification.

LightGBM is a highly efficient implementation of the popular Gradient Boosting Machine (GBM) model (Friedman (2001)). It is a tree-based, non-parametric method that builds decision trees sequentially. Each new tree is trained to correct the errors (residuals) of the previous one, allowing the model's predictive performance to be gradually improved. Given the flexibility of this setup, overfitting can be a significant issue. To mitigate this, a shrinkage hyperparameter (often called a 'learning rate') is used to regulate the contribution of each new tree.

This algorithm achieves excellent performance in many applications and is well-known as a go-to model for predictive tasks. We use the LightGBM implementation, which is an efficient,

open-source framework from Microsoft. It is particularly well-suited for large datasets due to efficiency improvements like Gradient-based One-Side Sampling (GOSS) and Exclusive Feature Bundling (EFB). In this implementation, we used the default hyperparameters from the lightgbm library; robustness checks with different hyperparameters did not yield significantly different results. For reference, the entire DML estimation, which involves training two LightGBM models with cross-fitting, took approximately 14 seconds to run on a MacBook Air (M3) with 24GB of RAM.

4.1.3 Comparing results

The results of both estimations are compared in Figure 4.2. We observe two key findings:

1. The DML coefficient estimates for all three outcomes are consistently higher than their PanelOLS (TWFE) counterparts.
2. The confidence intervals for the DML estimates are noticeably smaller, suggesting a higher degree of precision.

This difference in the point estimates is a significant finding and directly supports the arguments presented in (Chang (2020)). Both the PanelOLS (TWFE) model and the DML-DiD model are fixed-effects estimators that account for time-invariant unit-level heterogeneity (μi). The critical difference lies in how they handle the control variables (X_{it}).

The classic TWFE model assumes that the controls have a linear and additive effect on the outcome. If this assumption is violated—if the true relationship is complex and non-linear—the TWFE estimate for θ will suffer from model misspecification bias.

The DML-DiD model, by its construction, is robust to this misspecification. It uses a flexible, non-parametric (LightGBM) model to estimate the nuisance functions, effectively "partialling out" the complex, non-linear effects of the controls. Therefore, the higher estimates from the DML model can indicate that the classic TWFE model was biased downwards due to its rigid linearity assumption. The fact that DML also produces smaller confidence intervals suggests that, in this application, the non-parametric approach is not only more robust but also more efficient.

4.2 Staggered DID

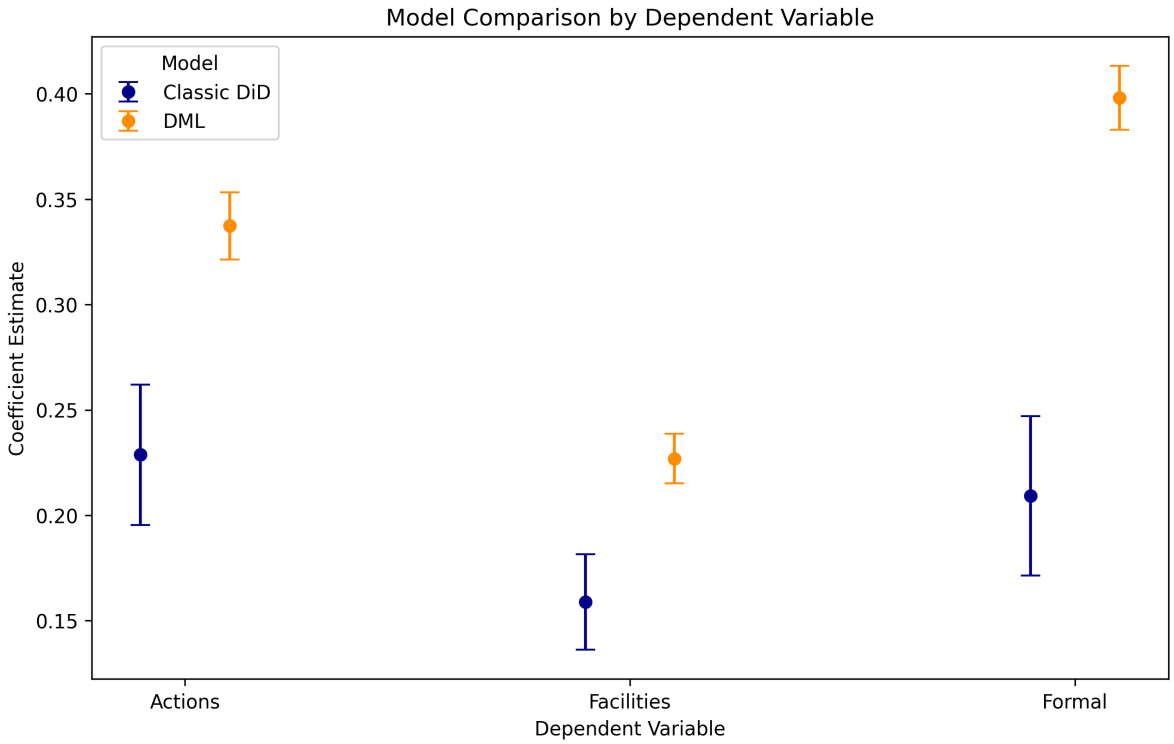

Figure 4.2: Comparing coefficients and standard errors for double machine learning and classic estimators.

References

Bach, P., Chernozhukov, V., Kurz, M. S., & Spindler, M. (2022). DoubleML – An object-oriented implementation of double machine learning in Python. *Journal of Machine Learning Research*, *23*(53), 1–6. Retrieved from http://jmlr.org/papers/v23/21-0862.html

Breiman, L. (2001). Statistical Modeling: The Two Cultures (with comments and a rejoinder by the author). *Statistical Science*, *16*(3), 199–231. https://doi.org/10.1214/ss/1009213726

Callaway, B., & Santa'Anna, P. H. C. (2021). Difference-in-differences with multiple time periods. *Journal of Econometrics*, *225*(2), 200–230. https://doi.org/10.1016/j.jeconom.2020.12.001

Chang, N.-C. (2020). Double/debiased machine learning for difference-in-differences models. *The Econometrics Journal*, *23*(2), 177–191. https://doi.org/10.1093/ectj/utaa001

Chernozhukov, V., Chetverikov, D., Demirer, M., Duflo, E., Hansen, C., Newey, W., & Robins, J. (2018). Double/debiased machine learning for treatment and structural parameters. *The Econometrics Journal*, *21*(1), C1–C68. https://doi.org/10.1111/ectj.12097

Cunningham, S. (2021). *Causal inference: The mixtape*. Yale University Press. Retrieved from http://www.jstor.org/stable/j.ctv1c29t27

Desai, A. (2023). *Machine Learning for Economics Research: When What and How?* (Papers No. 2304.00086). arXiv.org. Retrieved from arXiv.org website: https://ideas.repec.org/p/arx/papers/2304.00086.html

Friedman, J. H. (2001). Greedy function approximation: A gradient boosting machine. *The Annals of Statistics*, *29*(5), 1189–1232. https://doi.org/10.1214/aos/1013203451

GonzÃ¡lez, J. P. (2025). Environmental regulation, regulatory spillovers and rent-seeking. *Public Choice*, *202*(1), 217–250. https://doi.org/10.1007/s11127-024-01189-7

Goodman-Bacon, A. (2021). Difference-in-differences with variation in treatment timing. *Journal of Econometrics*, *225*(2), 254–277. https://doi.org/10.1016/j.jeconom.2021.03.014

Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., … Liu, T.-Y. (2017). LightGBM: A highly efficient gradient boosting decision tree. *Proceedings of the 31st International Conference on Neural Information Processing Systems*, 3149–3157. Red Hook, NY, USA: Curran Associates Inc.

Molnar, C. (2025). *Interpretable machine learning: A guide for making black box models explainable* (3rd ed.). Retrieved from https://christophm.github.io/interpretable-ml-book

Mullainathan, S., & Spiess, J. (2017). Machine learning: An applied econometric approach. *Journal of Economic Perspectives*, *31*(2), 87–106. https://doi.org/10.1257/jep.31.2.87

Seabold, S., & Perktold, J. (2010). Statsmodels: Econometric and statistical modeling with python. *9th Python in Science Conference*.

Shmueli, G. (2010). To Explain or to Predict? *Statistical Science*, *25*(3), 289–310. https://doi.org/10.1214/10-STS330

Varian, H. R. (2014). Big data: New tricks for econometrics. *Journal of Economic Perspectives*, *28*(2), 3–28. https://doi.org/10.1257/jep.28.2.3