Causal Inference II: Tutorial Group 7

Elena Petridou, Valérie Valckx & Victoria Zillo

December 2025

1 Abstract

In this tutorial we will be exploring the application of the doubly-robust Targeted Maximum Likelihood Estimator (TMLE) on (a subset of) the data collected as part of a retrospective cohort study on twin pregnancies in which some patients experienced selective fetal growth restriction. (Noll et al., 2025) The aim of this tutorial is to introduce the reader to TMLE, and apply it step-by-step to the dataset in order to explore its strenghts, limitations and things to look out for when using the method. TMLE is a method for computing various causal effects (in this tutorial the Average Treatment Effect) in a two-step fashion: First, we can calculate the estimand of interest – as we did for example with outcome modelling or propensity score methods –, and afterwards apply a "targeting step" such that our causal estimand of interest is as unbiased as possible.

2 Introduction

Selective Intrauterine Growth Restriction (sIUGR) is a condition which affects approximately 10% of monochorionic twin pregnancies (Children's Hospital of Philadelphia, n.d.), wherein the smaller fetus grows at a much slower rate than the larger, and often does not make it to term as a result. Stillbirth, miscarriage or other complications suffered by the smaller twin can affect the viability of the larger twin, sometimes even leading to death for both. As such, in severe cases, medical practitioners may recommend that the mother undergo selective reduction, a surgical procedure by which blood-flow to the smaller, ailing twin is restricted in such a way as to minimise complications for the larger twin.

2.1 Research Question

In this paper, we are investigating the following question: *Does selective reduction of a smaller twin in pregnancies with selective fetal growth restriction reduce the chances of a bad outcome for the larger twin?*. Bad outcome Y = 1 for the larger twin is defined as a composite outcome of: death, birth before 32 weeks or major neonatal morbidity.

Essentially, we are studying the average treatment effect (ATE) on the outcome for the larger twin between a selective reduction strategy Y(1) and conservative management Y(0). The estimand of interest is thus the Average Treatement Effect, which in potential outcome notation can be expressed as:

$$\mathbb{E}[Y(1)] - \mathbb{E}[Y(0)]$$

Throughout this tutorial, we illustrate how we can calculate the ATE and with the help of TMLE ensure that this estimate is unbiased as possible.

2.2 Data

The dataset we are working with contains observations from 264 twin pregnancies. Among these, 12 observations contain missing values. Since the proportion of missing data (12/264)= 0.045 is below the 5% threshold, we decided to remove the rows with missing values from the dataset. The resulting dataset includes 252 observations, of which 70% are assigned to the Conservative Management treatment group and 30% to the Selective Reduction group.

The dataset includes four covariates: three binary variables and one continuous variable defined on the interval [0,1]. Further details on the covariates are summarized in Table 1. These covariates influence both the choice of management strategy and the risk of a bad outcome for the larger twin, and are therefore treated as confounders throughout the analysis, which we denote by W.

Covariate	Description	Scale / Values
`Gratacos`	Disease type based on blood flow between placenta and the smaller twin assessed with ultrasound.	Binary (2 or 3)
`DVabnormal_small`	Blood flow in the ductus venosus of the smaller twin assessed with ultrasound.	Binary (0 = no, 1 = yes)
`Oligo_small`	Oligohydramnios (low amniotic fluid volume) in the smaller twin.	Binary (0 = no, 1 = yes)
`EFWdisc`	Estimated Fetal Weight Discordance: difference in estimated weight, expressed as proportion of the weight of the larger twin.	Continuous (0–1)

Table 1: Covariates included in the analysis: names, clinical meaning, and measurement scales.

3 Assumptions

Before applying any causal inference method, it is important to verify that the core assumptions needed for identifying causal effects are plausible.

3.1 Consistency

Consistency – also referred to as the Stable Unit Treatment Value Assumption, SUTVA – requires two key conditions:

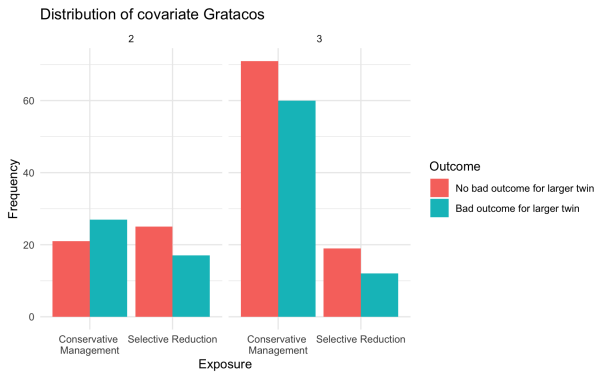
- Non-interference: the treatment received by one individual does not affect the outcome of another. In our case, whether one mother undergoes selective reduction cannot influence the growth or survival of twins from another patient.
- Stable exposure levels: treatment must be applied in a consistent way across all individuals, without different versions or "doses." For selective reduction, the procedure (selective cord occlusion) follows a standardized clinical protocol, so there are no meaningful variations of the intervention.

Note that consistency cannot be checked empirically in the data. It must be justified based on clinical understanding of how the treatment is delivered. As outlined above, we can assume that consistency holds in our case.

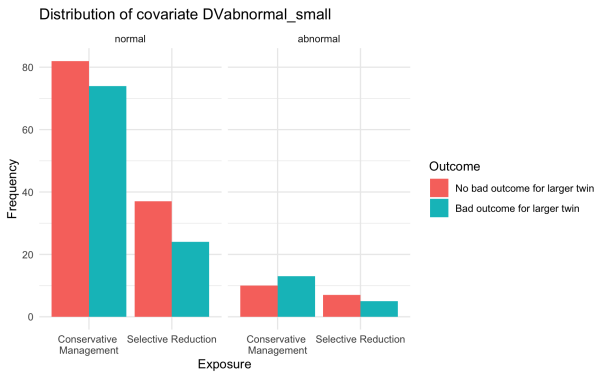
3.2 Positivity

Positivity requires that every individual has a non-zero probability of receiving each treatment level for all covariate levels, i.e. $0 < P(A = 1 \mid W = w) < 1$ whenever $f_W(w) > 0$. In our study this means that for every combination of relevant clinical covariates (such as the Gratacós classification and other baseline characteristics), both management strategies (conservative management and selective reduction) must occur at least once. This ensures sufficient overlap between the treatment groups. Without such overlap, the causal contrast cannot be identified because there would be no comparable control group within certain strata of the confounders. If, for example, all pregnancies with a specific Gratacós type were managed exclusively with one strategy, positivity would be violated and we would be unable to estimate the causal effect for that subgroup. Unlike consistency, this assumption can be examined directly in the data.

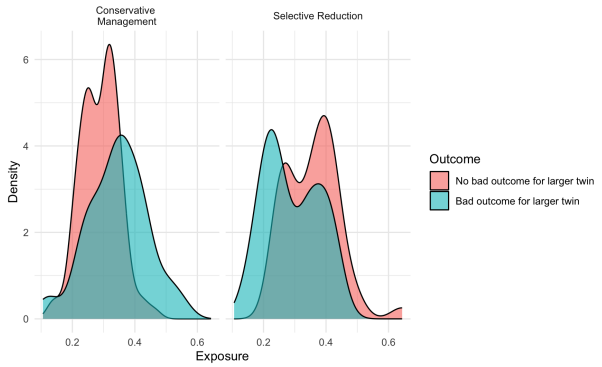
To evaluate the positivity assumption, we examined the distribution of the two management strategies across all covariates. For binary and categorical covariates, bar plots of management type within each category showed that both conservative management and selective reduction occurred at least once in every stratum, meaning no covariate–treatment combination had zero counts (Figure 1). For continuous covariates, density plots demonstrated substantial overlap in the distributions of the two treatment groups (Figure 1d). Together, these results indicate that all covariate patterns have support in both treatment arms, providing empirical evidence that the positivity assumption is satisfied in this dataset.



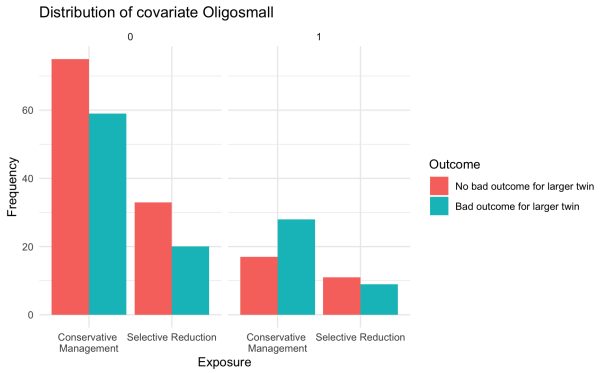
(a) Covariate Gratacos



(b) Covariate DVabnormal_small



(c) Covariate EFWdisc



(d) Covariate Oligosmall

Figure 1: Histograms and Density plot graphically checking the assumptions of positivity and exchangeability in our dataset. For positivity: we can see that for every level of each covariate, each level of the exposure occurs. Thus, positivity holds. For exchangeability: we see that it does not hold without adjustment, as the exposure (and the outcome) varies across the levels of the covariates.

3.3 Conditional exchangeability

In order to draw causal conclusions from data we need to establish exchangeability, meaning that A must be independent of $Y(A = a)$ for all a, or in words: that the treatment assignment (whether one received selective reduction or not) is independent of the outcomes (whether the larger twin suffered a bad outcome or not). However, as we can see from the distribution of our exposure and outcome levels with respect to the covariates in Figure 1, this does not appear to be the case. Informed by these exploratory findings and the study by Noll et al. (2025) which identified these covariates as confounders, our posited Directed Acylical Graph (DAG) in Figure 2 further illustrates why exchangeability does not hold: the covariates all play a role in determining both the exposure and the outcome, making them confounders.

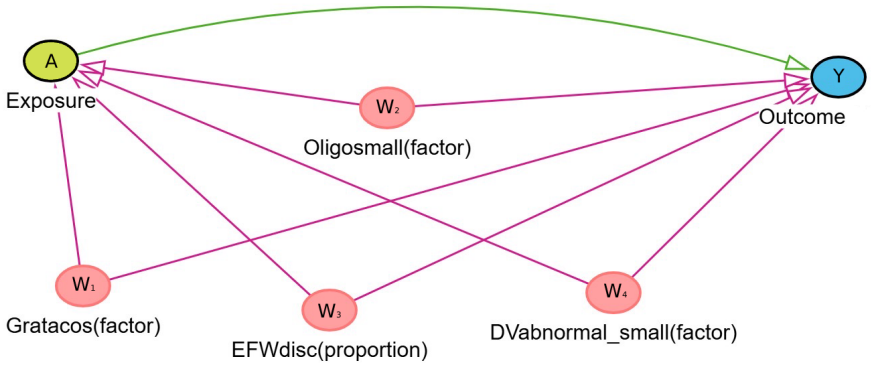


Figure 2: Directed Acyclical Graph (DAG) of the causal path between exposure A (selective reduction or conservative management) and outcome Y (bad outcome or absence of bad outcome for larger twin), as well as how these are both affected by the confounders W.

Therefore, we aim to meet a "constrained" form of exchangeability: Conditional exchangeability, which requires that, after adjusting for all relevant confounders, the treatment assignment is independent of the potential outcomes. Formally, this is written as $Y(x) \perp A \mid W$, meaning that within levels of the covariates W, patients who receive different management strategies are comparable with respect to their underlying prognosis. In a randomized trial, exchangeability is guaranteed because treatment is assigned independently of patient characteristics. In observational data, however, treatment decisions may depend on clinical factors that also influence the outcome, and these must be adequately controlled for to approximate the balance achieved by randomization.

To evaluate whether conditional exchangeability is plausible in our observational dataset, we assessed covariate balance before and after propensity score weighting. Figure 3 shows the standardized mean differences of all covariates in the two treatment groups. Before weighting, substantial imbalances were present, indicating that treatment assignment was related to several clinical characteristics. After weighting, the standardized mean differences moved much closer to zero for all covariates, demonstrating improved balance between the groups. This supports the assumption that, conditional on the included covariates, treatment assignment can be considered approximately independent of the potential outcomes.

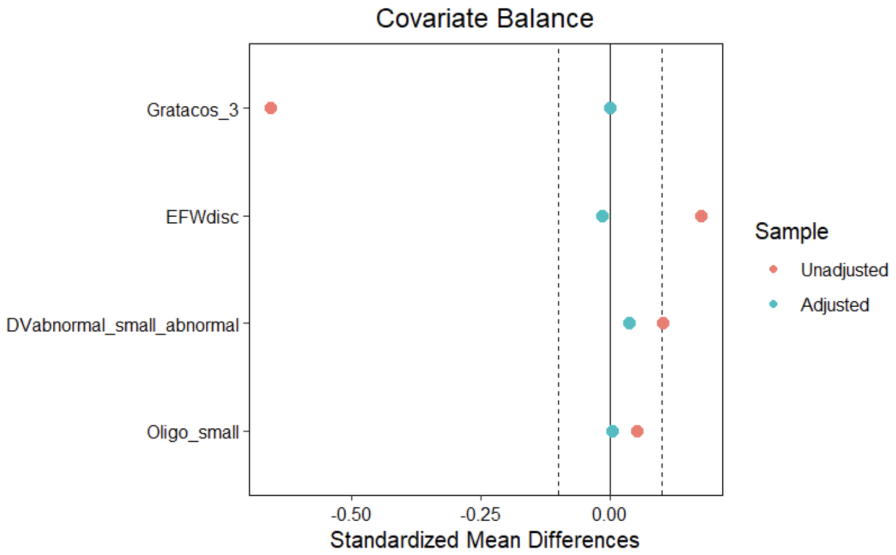


Figure 3: Covariate balance before and after propensity score weighting

3.4 Correct model specification

Correct model specification refers to the requirement that the statistical models used to estimate treatment and/or outcome mechanisms adequately represent the true data-generating processes. In many traditional causal inference approaches, this means that the parametric assumptions of the model must be correct. Often, in order to estimate the ATE, we try to model either (1) the outcome generating mechanism (Q), as with outcome regression, or (2) the exposure generating mechanism (g), as with Inverse Probability Weighting. For (1), Q is essentially modeling how the outcome (Y) is determined via the exposure (A) and the confounders (W). For (2), g is modeling how the exposure (A) is determined via the confounders(W).

Since we are usually modeling just one of these mechanisms, it is important that the parametric assumptions of the model we are using to estimate this mechanism with are met. Since linear models are often used, this is unfortunately not always the case; real-world phenomena rarely display neat linear relationships.

In light of this, TMLE is particularly attractive because it is *doubly robust*: the estimator remains consistent as long as *either* the model estimating outcome model (Q) or the model estimating the treatment mechanism (g) is correctly specified (Luque-Fernandez et al., 2018). This essentially "softens" the assumption of correct model specification.

Moreover, TMLE does not require either of these components to be modeled via simple parametric models. Both can be estimated flexibly using machine-learning methods, including non-parametric and ensemble algorithms, which substantially reduces the burden of specifying the correct functional form, as these models often make no parametric assumptions. This flexibility increases the likelihood that at least one of the two required components is well-estimated, thereby strengthening the robustness of the causal inference procedure.

4 Targeted Maximum Likelihood Estimation

4.1 What does TMLE do in a nutshell?

As we have seen in the preceding sub-section, TMLE allows us to be more flexible about the assumption of correct model specification which is the property that makes it *doubly robust*. This makes TMLE already very useful, as the underlying assumptions made by parametric models are often too simplistic for real-world data.

Another great feature of TMLE is that it allows us to carry out targeted estimation of our parameter (estimand) of interest. Recall that in our case, the estimand of interest is the average treatment effect (ATE) of receiving selective reduction surgery versus not receiving it:

$$\mathbb{E}[Y(1)] - \mathbb{E}[Y(0)]$$

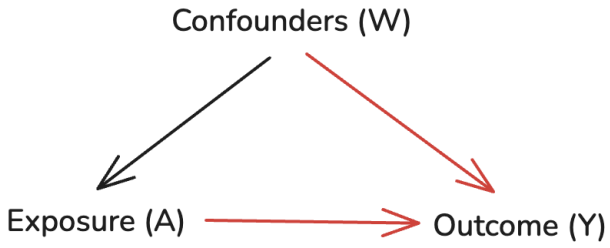


Figure 4: Estimating the outcome mechanism Q. The arrows in red show the paths we are estimating when trying to model $E(Y|A, W)$. This is essentially the expectation of the outcome conditional on the exposure and the covariates.

In order to calculate this effect, we need to estimate $E[Y(1)]$, the expected outcome had everyone received the treatment and $Y(0)$, the expected outcome had nobody received the treatment. Then, we can subtract these estimates from each other to obtain the ATE.

Recall that we have learned a few techniques through which we can estimate $Y(1)$ and $Y(0)$ in Causal Inference I.

(1) Our first option was to estimate *the outcome mechanism* Q. This models the expected value of the outcome given the exposure and confounders $E(Y|A, W)$ (the path visualised in Figure 4), which is what we do via G-computation.

(2) Our second option was estimating the probability of the exposure (treatment) given the confounders $P(A|W)$, which we can also call the propensity score of treatment/no treatment, and is the path shown in Figure 5. This is also called *the treatment mechanism* g.

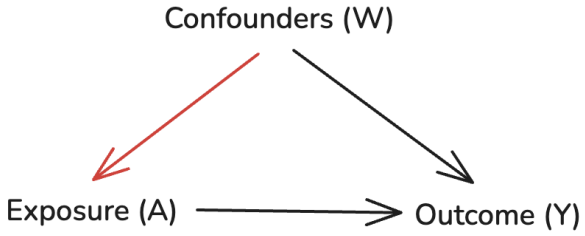


Figure 5: Estimating the treatment mechanism g. The arrow in red shows the path we are estimating when trying to model $P(A|W)$. This is essentially the probability of receiving or not receiving treatment given a patient's values of the other covariates.

This is where the intuition for targeting as well as double-robustness comes in: whereas with other methods we either estimate pathway (1) the outcome mechanism or pathway (2) the treatment mechanism and need the model to be correctly specified, with TMLE we estimate both paths. Then, we use the information from pathway (2) to update the estimates from pathway (1), such that they are optimal for our estimand of interest: in our case, the ATE. As such, as long as one of the models is correctly specified, we can have a valid estimator for the ATE.

4.2 Computing the TMLE manually, to build intuition

We explain how the TMLE algorithm works step-by-step, with inspiration drawn from Luque-Fernandez et al. (2018) and Hoffman (2020), in order to build intuition by showing the purpose and workings of each step. However, Gruber and van der Laan (2012)'s tmle r-package has the ability to compute all these steps using one line of code, which we illustrate after the step-by-step section. We recommend using the library, and go through the steps only to help conceptual understanding.

4.2.1 Step 1: Estimate Outcome Mechanism Q/the Expected Value of the Outcome $E(Y|A,W)$

The first step to applying the TMLE algorithm is to estimate the expected value of the outcome conditional on the exposure and the confounders, or the relationship visualised in Figure 4. For illustrative purposes, we will use a logistic regression to more closely approximate how we implemented G-computation in Causal Inference I. However, bear in mind that any statistical model, including machine learning models can be used in this step, and that is what we do at the end of this tutorial.

```
outcome_regression = glm(outcome ~ ManagementType + Gratacos + EFWdisc+
DVabnormal_small+ Oligo_small, family = binomial, data = data_clean)
```

Using this model, we construct three new vectors. The first one contains the predicted outcome for each patient, had they all received selective reduction. We do this by setting the ManagementType (Exposure) column to all 1's.

```
Q1W = predict(outcome_regression, newdata = data.frame(ManagementType = 1, data_clean
[,c( "Gratacos" , "EFWdisc", "DVabnormal_small","Oligo_small")]), type = "response")
```

The second one contains the predicted outcome for each patient, had no one received selective reduction:

```
Q0W = predict(outcome_regression, newdata = data.frame(ManagementType = 0, data_clean
[,c( "Gratacos" , "EFWdisc", "DVabnormal_small","Oligo_small")]), type = "response")
```

The third is a vector containing what our model predicts the outcome would be at every patient's current exposure level.

```
QAW = predict(outcome_regression, type = "response")
```

At this stage, we could subtract vector Q0W from vector Q1W, and take the mean of the resulting vector to obtain the ATE as we did during G-computation. However, since our outcome regression was optimising the bias-variance trade-off for calculating $E(Y|A,W)$ and not $E(Y(1)) - E(Y(0))$, we can proceed with the following steps to target our estimate to the ATE.

4.2.2 Step 2: Estimate Treatment Mechanism g

For step 2, we will estimate the path shown in Figure 5 in order to use this information to improve our estimates from step 1.

Once again for illustrative purposes, we use a GLM to approximate IPW as presented in Causal Inference I, but we could have used any statistical model here. This time, the outcome of the regression will be a patient's exposure ("Management Type"), and the predictors are the confounders. The predictions we yield from this model are the propensity score values for each patient, here denoted gW.

```
PS1.model <- glm(ManagementType ~  Gratacos + EFWdisc +  DVabnormal_small  +
Oligo_small,
             family = binomial,
             data = data_clean)
gW = predict(PS1.model, type = "response")
```

Then, we use this model to produce three vectors – the so-called "clever covariates" – which we will use in the targeting step to "tailor" our estimate to the ATE.

First, we compute the inverse probability of receiving treatment, for the treated: $H(A = 1, W) = \frac{1}{P(A=1|W)}$:

```
H1W = (data_clean$ManagementType / gW)
```

Then, we compute the negative inverse probability of not receiving treatment, for those who did not receive it: $H(A = 0, W) = -\frac{1}{P(A=0|W)}$:

```
H0W = (1 - data_clean$ManagementType) / (1 - gW)
```

4.2.3 Step 3: Calculate how much adjustment is required to target our estimate to the ATE

This is where TMLE crucially differs from other methods. The steps from hereon out allow us to make use of the information computed in steps 1 and 2 in order to "calibrate" our estimate to be as accurate as possible for our estimand of interest (the ATE). To do this, we can use an *efficient influence function*.

What is an Efficient Influence Function? Since delving deeply into efficiency theory is beyond the scope of this tutorial, we could simply think of an efficient influence function as a way of determining how far our estimate $\hat{E}(Y|A,W)$ in Step 1 differs from the real $E(Y|A,W)$. According to Schuler and Rose (2017), the following efficient influence function can be used for quantifying this difference when estimating the ATE (Schuler & Rose, 2017, p. 68):

$$logit(\mathbb{E}[Y|A,W]) = logit(\hat{\mathbb{E}}[Y|A,W]) + \epsilon H(A,W) \tag{1}$$

The scale by which we should adjust our estimates from Step 1 ($\hat{\mathbb{E}}[Y|A,W]$)), is given by ϵ, which is also called the *fluctuation parameter*. This fluctuation parameter scales our "clever covariates" from Step 2 ($\epsilon H(A,W)$), which are then added to the initial Step 1 estimate. Since we are adjusting the estimate for H0W and H1W, our ϵ is a vector of length 2.

Estimating the Fluctuation Parameter How do we estimate ϵ? Note how similar the above formula looks to a standard, multivariate linear regression:

$$logit(Y) = \beta_0 + \beta_i X \tag{2}$$

where β_0 is the coefficient for the intercept and β is a vector of coefficients for each covariate in some design matrix X.

If we compare each term in Equation 1 with each term in Equation 2, we can treat $logit(\hat{\mathbb{E}}[Y|A,W]$ as a fixed-value coefficient β_0 and $H(A,W)$ as the vectors making up the input matrix to a logistic regression (which in our case are split into H0W and H1W), and use such a logistic regression to estimate the vector of coefficients β representing the estimated fluctuation parameter $\hat{\epsilon}$.

```
epsilon <- coef(glm(data_clean$outcome ~ -1 + H0W + H1W + offset(qlogis(QAW)), family
 = binomial))
epsilon
```

4.2.4 Step 4: Update the Initial Estimates of the Expected Outcome

This step is where the actual *targeting* happens. Using Equation 1, we can update the quantities calculated in Step 1, such that they are now given by:

$$\hat{\mathbb{E}}^*[Y|A=1,W] = expit(logit(\hat{\mathbb{E}}[Y|A=1,W] + \hat{\epsilon_1} * H(1,A)) \tag{3}$$

$$\hat{\mathbb{E}}^*[Y|A=0,W] = expit(logit(\hat{\mathbb{E}}[Y|A=0,W] + \hat{\epsilon_2} * H(0,A)) \tag{4}$$

```
Q0W_1 <- plogis(qlogis(Q0W) + epsilon[1] / (1 - gW))
Q1W_1 <- plogis(qlogis(Q1W) + epsilon[2] / gW)
```

Essentially, we have "shifted" our original estimates for $E(Y(0))$ and $E(Y(1))$, in the direction of the true ATE by using information we gained from the treatment mechanism g in Step 2. We have thus reduced the bias in our original estimate.

4.2.5 Step 5: Calculate the Updated ATE

Using these updated quantities, we can finally calculate our estimand of interest, the ATE, using the plug-in estimator as we always do:

```
ATEtmle1 <- mean(Q1W_1 - Q0W_1)
```

4.3 TMLE using the tmle R-package

As we mentioned at the outset of this section, we walked through the steps of performing TMLE from scratch in order to build intuition about what each step does by showing it in code. However, there is an easier and even more efficient way of computing an estimand of interest using TMLE, the tmle package in R (Gruber & van der Laan, 2012).

The package essentially does all the steps explained in the previous section under the hood with a single line of code:

```
TMLE_obj= tmle(Y= data_clean$outcome, A= data_clean$ManagementType, W= data_clean[, c
("Gratacos", "EFWdisc", "DVabnormal_small", "Oligo_small")], family = "binomial")
```

The required arguments are:

- Y: The outcome vector
- A: The exposure vector
- W: The matrix of covariates

If we wanted to implement `tmle()` as we did analytically we could use the `tmle()` function from tmle package and specify in the `Q.SL.library` and `g.SL.library` arguments that only GLMs should be used during estimation, as follows:

```
TMLE_same_as_manual = tmle(Y= data_clean$event, A= data_clean$ManagementType, W=
 data_clean[, c("Gratacos", "EFWdisc", "DVabnormal_small", "Oligo_small")], family = "
 binomial", Q.SL.library = c("SL.glm"), g.SL.library = c("SL.glm"))
```

The reason why we would need to specify an extra argument in order to reproduce the above analysis is that by default the TMLE package relies on an algorithm called SuperLearner to choose the best estimator for steps 1 (estimating outcome mechanism Q) and 2 (estimating treatment mechanism g) by comparing a wide range of candidate models using cross-validation. The resulting estimator is a weighted ensemble of the different candidate models, with the weight given to each predictor determined through cross-validation (Luque-Fernandez et al., 2018).

Without specifying the aforementioned Library arguments, the default 3 models for estimating Q (outcome mechanism) would be: "SL.glm" (generalised linear model with binary outcome); "tmle.SL.dbarts2" (Bayesian Additive Regression Tree); and "SL.glmnet" (a generalised linear model with elastic net penalisation). For estimating g (treatment mechanism), the default choices are: "SL.glm", "tmle.SL.dbarts.k.5" (Bayesian Additive Regression tree, but with k = 5), and "SL.gam" (generalised additive model). (Gruber, 2025)

In order to use other models than the default ones, it is possible to specify the SL.library arguments for each of the outcome and treatment mechanisms via the arguments `Q.SL.library` and `g.SL.library` and select among many choices of estimators, all of which can be found in the package's documentation. (Gruber, 2025)

5 Results and Discussion

This section presents the estimated average treatment effect (ATE) of selective reduction ($A = 1$) versus conservative management ($A = 0$) on the probability of a bad outcome for the larger twin, using different approaches. A negative ATE indicates that selective reduction is associated with a reduced risk of adverse outcome. We report all numbers rounded to 3 decimal places.

5.1 Outcome regression via G-computation (Estimating the Outcome Mechanism Q only)

Using a logistic regression model for the outcome conditional on treatment and baseline covariates, we estimated the counterfactual risks under universal selective reduction and universal conservative management.

```
Q1W = predict(outcome_regression, newdata = data.frame(ManagementType = 1, data_clean
[,c( "Gratacos" , "EFWdisc", "DVabnormal_small","Oligo_small")]), type = "response")

Q0W = predict(outcome_regression, newdata = data.frame(ManagementType = 0, data_clean
[,c( "Gratacos" , "EFWdisc", "DVabnormal_small","Oligo_small")]), type = "response")
```

Then, plugging into the plug-in estimator for ATE yields:

$$\widehat{\mathbf{ATE}}_{\mathbf{OR}} \approx \mathbf{-0.115.}$$

Which is the equivalent of:

```
mean(Q1W - Q0W)
```

This would indicate that selective reduction is associated with an estimated 11.5 percentage lower probability of a bad outcome for the larger twin. However, as this estimate has no uncertainty quantification, we cannot say whether the result is statistically significant at a particular alpha. Moreover, as this method relies entirely on the correct specification of the outcome model, the estimate is sensitive to model misspecification.

5.2 TMLE using only linear methods

Applying the TMLE procedure step-by-step as in Section 4.2, is equivalent to using the TMLE method and specifying only linear methods via the `Q.SL.library` and `g.SL.library` arguments. The step-by-step calculation yielded an estimated ATE of approximately -0.086 (in 3 decimals), while the estimate using the tmle library as shown in the Code listing in sub-section 4.3 yielded:

$$\widehat{\mathbf{ATE}}_{\mathbf{TMLE,GLM}} = \mathbf{-0.086},$$

with a 95% confidence interval $\mathbf{(-0.233,\ 0.061)}$, standard error $\boldsymbol{\sigma} = \mathbf{0.078}$ and p-value $\boldsymbol{p} = \mathbf{0.266}$

To obtain these estimates (ATE, CI's, standard error, p-value) from the object resulting from using the `tmle()` function, we just need to access the estimates attribute as follows:

```
tmle_same_as_manual$estimates$ATE
```

We observe that the estimate for the ATE is the same between the step-by-step and tmle with only GLMs for both mechanisms, as the steps are essentially what tmle runs behind the scenes.

Just as when using only outcome regression, the ATE remains negative, though now it is slightly closer to zero compared to the outcome regression estimator. The confidence interval is wide and includes zero, and the corresponding p-value is very large. This shows that our estimate is not statistically significant for any reasonable alpha, which would imply that Selective Reduction has no significant effect on the probability of a bad outcome for the larger twin.

Given the findings in the following section, we suspect that this result comes from the fact that linear models do not approximate the two mechanisms well.

5.3 TMLE with default methods

Next, we ran the `tmle()` function using only the default learners described in sub-section 4.3. This yielded the following results:

$$\widehat{\mathbf{ATE}}_{\mathbf{TMLE,default}} = \mathbf{-0.237},$$

with a 95% confidence interval $\mathbf{(-0.345,\ -0.130)}$, standard error $\boldsymbol{\sigma} = \mathbf{0.054}$ and p-value $\boldsymbol{p} < \mathbf{0.001}$

The small p-value indicates that our result is now statistically significant even for alpha level of $\alpha = 0.01$, which can be interpreted as the ATE of selective reduction being a 23.7% decrease in the probability of a bad outcome for the larger twin, when selective reduction is opted for versus when it is not.

For this run, only the default learners for the `tmle()` method are used. Recall that the SuperLearner creates an ensemble of weighted learners based on which learners performed best during cross-validation. To inspect which models SuperLearner chose to include in the ensemble, we can call summary() on the tmle object:

```
TMLE_default= tmle(Y= data_clean$event, A= data_clean$ManagementType, W= data_clean[,
 c("Gratacos", "EFWdisc", "DVabnormal_small", "Oligo_small")], family = "binomial")
summary(TMLE_default)
```

The summary contains tables of all the methods tried for estimating each of the mechanisms, along with their corresponding weights, as shown in Tables 2 and 3.

Coefficients	
GLM	0
BART with k = 2	1
GLM with Elastic Net Regularisation	0

Table 2: Models and corresponding weights used by TMLE_default to estimate the outcome mechanism Q

Coefficients	
GLM	0
BART with k = 5	1
General Additive Model (GAM)	0

Table 3: Models and corresponding weights used by TMLE_default to estimate the outcome mechanism g

From the available methods, the SuperLearner chose to use only a Bayesian Additive Regression tree (BART) model for estimating the outcome mechanism Q, and also a Bayesian Additive Regreession tree model for estimating the treatment mechanism g.

The fact that for both mechanisms the SuperLearner chooses flexible tree methods indicates once again that the assumption of a correctly-specified model when relying on GLMs for the manual estimation may have been violated, leading to the confidence intervals containing zero, and the higher standard error for the TMLE with linear methods only ($\sigma_{\text{TMLE,GLM}} = 0.078$, versus $\sigma_{\text{TMLE,default}} = 0.054$.

5.3.1 Why are the estimates between the two methods so different?

It is important here to ask whether the change in the estimate when using the TMLE with default methods is due to an improvement in the way the model fits the data or if something has gone wrong.

One promising observation that the new model represents a better fit is that the standard error has dropped from $\sigma_{\text{TMLE,GLM}} = 0.078$ to $\sigma_{\text{TMLE,default}} = 0.054$, indicating that the new model is able to capture more of the variation in the data. This can be interpreted as a gain in precision, which is also evident from the narrower confidence intervals $\text{CI}_{\text{TMLE,default}} = (-0.345,\ -0.130)$ versus $\text{CI}_{\text{TMLE,GLM}} = (-0.233,\ 0.061)$.

Moreover, the fact that the SuperLearner tried linear models during the cross-validation procedure also, but chose to assign them 0 weights is another indicator that the new estimate using BART better approximates the underlying structure of (either) the outcome or the treatment mechanism. Had the linear models improved the fit of the model in any of the cross-validation runs, we can assume the SuperLearner would have assigned at least some weight to them.

The fact that the SuperLearner selects more flexible methods for estimating both mechanisms indicates some non-linearity in at least one of them. This can be caused by treatment-covariate interactions in the case of the outcome mechanism (Q), or covariate-covariate interactions in the case of the treatment mechanism (g).

With regards to the treatment mechanism (Q), we observe a non-linear relationship between Exposure and Outcome, modulated by at least one of the confounders (EFWdisc), which is plotted in Figure 6

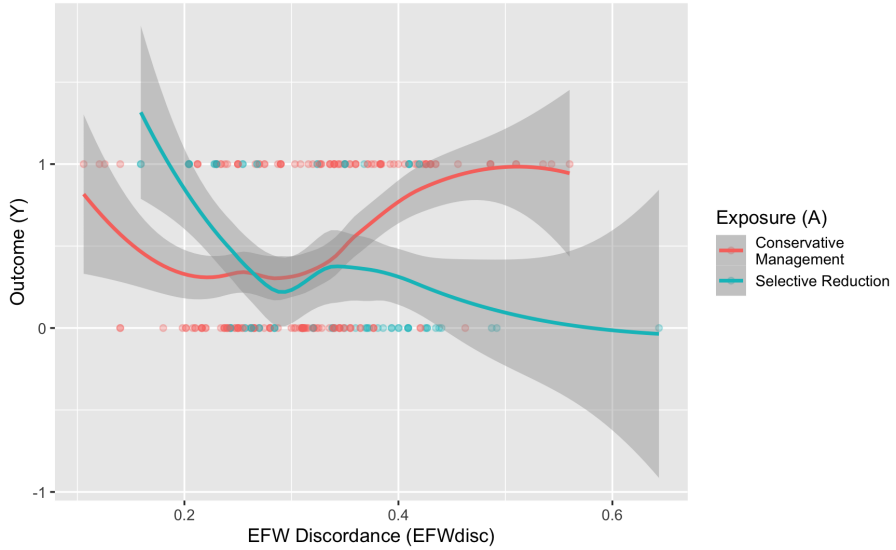


Figure 6: Non-linearity in the outcome mechanism Q, displayed in the way the relationship between the exposure and outcome behaves differently according to the level of the EFW Discordance confounder

Covariate-covariate interactions can cause the treatment mechanism g to be non-linear, which we investigated by comparing the distribution of the Exposure (A) at different combinations of the levels of the confounders. In Figure 7 we plot the distribution of the Exposure, given different combinations in the levels of the confounders Oligo_small and DVabnormal_small. g would be perfectly linear if the were more or less proportional for different combinations of the factors of the covariates. The slight "curve" visible on the graph shows that g is also not strictly linear.

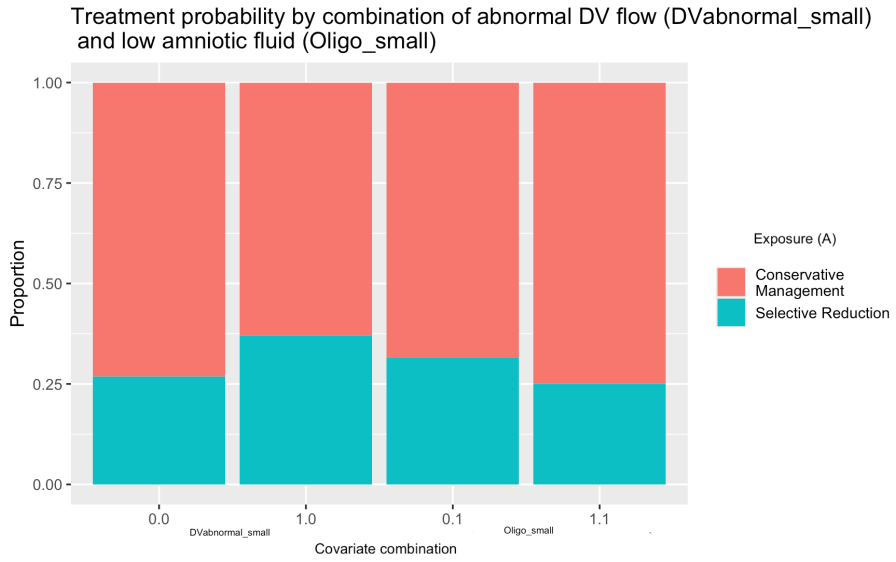


Figure 7: Non-linearity in the treatment mechanism g, displayed in the way the exposure (A) behaves according to different combinations of the levels of confounders DVabnormal_small and Oligo_small

Thus, a graphical exploration of the two treatment mechanism seems to support our theory that the improvement in fit was the result of the linear models not correctly approximating the two mechanisms.

Finally, recall also that the $\hat{\epsilon}$ coefficients we estimated in subsection 4.2.3 represented the extent to which we need to adjust our estimates from step 1 to make ATE more unbiased. Thus, larger ϵ means more adjustment took place during targeting.

As such, we can check the magnitude of the epsilons for the two methods and compare them. We can do this by running:

```
TMLE_GLM$epsilon
TMLE_default$epsilon
```

Which results in the following table of estimated coefficients:

Method	ϵ_1	ϵ_2
TMLE_{GLM}	0.00903994	0.032192209
$\text{TMLE}_{\text{default}}$	0.05028729	-0.08851482

Table 4: Epsilon coefficients estimated from TMLE with only linear and TMLE with default methods

The magnitude of the epsilons estimated using the default methods are greater, showing that the estimates from step 1 in this method were adjusted more than the estimates from the model using only linear methods.

Overall, these results seem indicative of a poorer fit for the model with only linear methods. More flexible and non-parametric methods have the ability to adapt to the underlying mechanisms in the data better in the presence of non-linearities. Thus, for our dataset the SuperLearner seems to prefer more flexible methods for estimating both the outcome and treatment mechanisms.

5.4 TMLE with Additional methods

For our final model, we applied the `tmle()` function using an extended library provided to the Super-Learner, which included: generalized linear models (and those with interactions among all variables), stepwise regression, general additive model (GAM), recursive partitioning and regression trees (RPART) random forest, BARTs and gradient boosted tree methods (XG Boost). We do this by specifying a vector of methods via the arguments `Q.SL.library` and `g.SL.library`:

```
Q_extras = c("SL.glm", "SL.glm.interaction", "SL.step","SL.step.interaction", "
tmle.SL.dbarts2", "SL.glm.interaction","SL.gam","SL.randomForest", "SL.rpart", "
SL.xgboost")
g_extras = c("SL.glm", "SL.glm.interaction", "SL.step","SL.step.interaction", "
tmle.SL.dbarts.k.5", "SL.glm.interaction","SL.gam","SL.randomForest", "SL.rpart", "
SL.xgboost")
```

```
TMLE_extended= tmle(Y= data_clean$event, A= data_clean$ManagementType, W= data_clean[
, c("Gratacos", "EFWdisc", "DVabnormal_small", "Oligo_small")], family = "binomial",
Q.SL.library = Q_extras, g.SL.library = g_extras)
```

This allows both the outcome regression and treatment mechanism to be estimated using flexible machine-learning approaches. We chose to specify more flexible methods seeing as among the default learners only the most flexible (BART) was chosen for both initial steps and also seeing that our graphical exploration of some of the relationships in the two mechanisms are non-linear.

The resulting ATE estimate is:

$$\widehat{\mathbf{ATE}}_{\mathbf{TMLE,SL_{extended}}} \approx \mathbf{-0.265},$$

with a 95% confidence interval $\mathbf{(-0.354,\ -0.175)}$, a standard error of $\sigma = 0.046$, and p-value $p < 0.001$.

The estimate is now more negative, and still has a very small p-value, and even lower standard error than before. This estimate indicates that selective reduction leads to a 26.5% lower probability for a bad outcome for the larger twin, if selective reduction is used versus if a conservative management approach is followed.

The more flexible learners appear to capture additional signal from the data, yielding a larger estimated reduction in risk under selective reduction, and better precision as the standard error fell to 0.046 from the previous model's 0.054. The summary of selected models once again confirms that more flexible models better approximate both outcome and treatment mechanisms, as seen in Tables 5 and 6. This further bolsters our claim that using only linear methods possibly led to the non-significant result encountered before, as the model was not able to account for enough of the variance in the data using only those models – as indicated by the lower standard error when more flexible methods are allowed. It also shows that using only linear models, the assumption of correct model specification was probably not met.

Coefficients	
GLM	0
GLM with interactions	0
Stepwise Regression	0
Stepwise Regression with interaction	0.1132083
BART with k = 2	0
Stepwise GLM with interaction	0
Stepwise GAM	0
Random Forest	0.2008734
RPART	0.09112411
XG Boost	0.5947942

Table 5: Models and corresponding weights used by TMLE_extended to estimate the outcome mechanism Q

Coefficients	
GLM	0
GLM with interactions	0
Stepwise GLM	0
Stepwise GLM with interactions	0
BART with k = 5	0
GAM	0
Random Forest	0.6745099
RPART	0
XG Boost	0.3254901

Table 6: Models and corresponding weights used by TMLE_extended to estimate the outcome mechanism g

Inspecting Table 5, we see that the selected learners for estimating the outcome mechanism are either linear with interactions, or flexible tree and ensemble methods. Table 6 shows that for the treatment mechanism, only bagging and boosting tree methods are used.

5.5 Comparison of the estimators

Across all methods, the estimated ATE is consistently negative, suggesting that selective reduction appears to be associated with a decreased risk of a bad outcome for the larger twin.

Method	**ATE**	**C.I.**	**standard error**	**p-value**
`tmle()` with only GLM	-0.086	(-0.233, 0.061)	0.078	0.266
`tmle()` with default learners	-0.237	(-0.345, -0.130)	0.054	<0.001
`tmle()` extended	-0.249	(-0.342, -0.155)	0.046	<0.001

Table 7: Comparison of ATE estimates and their uncertainty measures, across different learner combinations

- The *outcome regression estimator* estimated a protective effect for selective reduction of (≈ -0.115), but relies entirely on the assumption of correct model specification for the generalised linear model.

- The *manual TMLE* yielded an estimate (≈ -0.08), with wide confidence intervals including zero showing that the effect was negligible or could not be picked up by the model.
- The *TMLE with extended SuperLearner* produced a substantially larger estimated effect (≈ -0.249) and a confidence interval entirely below zero, suggesting that the true protective impact of the treatment may be stronger, as when flexible machine-learning methods are used to model complex relationships in the data, we are able to detect it.

While all methods point in the same direction, the magnitude of the estimated effect, as well as how confident we are about the produced estimates, depends on the modeling approach. The use of a richer SuperLearner library leads to a larger estimated benefit of selective reduction, possibly because it better captures nonlinearities and interactions that simpler models miss. However, given the limited sample size, these findings should still be interpreted with caution.

Overall, what this exploration of TMLE shows is that it is a very useful method for ensuring that our resulting estimate is as unbiased as possible for our estimand of interest (in this case the ATE). The fact that our result went from non-significant using only linear estimators to very significant and with a much greater absolute magnitude and lower standard error after specifying more flexible methods, also shows the benefit of using ensembling via the SuperLearner, as it allows us to try many different models and assume the most suitable one given the underlying patterns in each particular dataset.

We saw via graphical exploration that the relationship between confounders, outcome and exposure in the outcome mechanism Q were not exactly linear, and that a similar pattern was observed for the relationships between sets of confounders and the exposure for the treatment mechanism g. This further highlighted the benefit of being able to not only try out parametric methods which carry some assumptions about the underlying relationships in the data, but also more flexible methods.

Finally, the fact that the two mechanisms show some complex patterns also further highlights the importance of TMLE being *doubly robust*. Even if one of the mechanisms (outcome or treatment) cannot be fully captured by the ensemble of models that the SuperLearner ends up choosing, as long as the other mechanism is correctly modelled, our estimates remain valid.

6 Conclusions and Limitations

In conclusion, TMLE is a promising method for causal inference as it "softens" the assumption of correct model specification and provides robust and accurate estimates for many of the most popular causal estimands of interest. It does this by first estimating the outcome mechanism Q as we do in outcome regression/G-computation. Then, it estimates the treatment mechanism g as we do with propensity score methods. Using information from g, TMLE estimates how "off" our initial estimate of Q is, and uses this information to "shift" or "target" the estimate to the correct distribution of our estimand of interest. This results in a more unbiased estimate *tailored* to our estimand of interest.

TMLE is also easy to apply thanks to the `tmle` package available on CRAN. Furthermore, the package makes use of the SuperLearner algorithm to allow the use of any statistical learning model for estimating either of the mechanisms of interest. Through the cross-validation procedure used to create the final ensemble learner, we can be more certain that a suitable ensemble of learners will be selected for the estimation of each step.

While not explored in the tutorial, other estimands such as the Average Treatment Effect on the Treated (ATT), the Relative Risk (RR), the Odds Ratio (OR) and Controlled Direct Effect (CDE) can also be estimated and targeted by TMLE using the package on CRAN.

Finally, the theory of why the targeting step works was lightly discussed in section 4.2.3 in a very limited way due to the applied nature of the tutorial and time limitations. We treat the efficient influence function almost like a plug-in estimator as we saw it referenced in Schuler and Rose (2017) in order to explain how targeting is estimable via logistic regression. Deeper understanding of efficiency theory and why the efficient influence function provides the direction in which the estimates should be adjusted to become more unbiased for our estimand of interest is a good next step towards deeply understanding how TMLE works. Fisher and Kennedy (2021) provide an explanation of why influence function based estimators can achieve "asymptotically optimal mean-squared error" (Fisher & Kennedy, 2021, p.162) based on some of their properties. However, given the complexity of the topic, dissecting this is beyond the scope of this tutorial and also – we believe – not strictly necessary when looking to apply TMLE.

References

Children's Hospital of Philadelphia. (n.d.). *Selective intrauterine growth re- striction (siugr).* https://tinyurl.com/bddynt7m

Fisher, A., & Kennedy, E. H. (2021). Visually communicating and teaching intuition for influence functions. *The American Statistician*, *75*(2), 162–172.

Gruber, S. (2025). *Cran: Help for package tmle* [R package version 2.0-29]. https://cran.r-project.org/web/packages/tmle/refman/tmle.html#summary.tmle

Gruber, S., & van der Laan, M. J. (2012). tmle: An R package for targeted maximum likelihood estimation [doi:10.18637/jss.v051.i13]. *Journal of Statistical Software*, *51*(13), 1–35. https://www.jstatsoft.org/v51/i13/

Hoffman, K. (2020). *Illustrated guide to tmle, part ii: The algorithm.* KHStats. https://www.khstats.com/blog/tmle/tutorial-pt2

Luque-Fernandez, M. A., Schomaker, M., Rachet, B., & Schnitzer, M. E. (2018). Targeted maximum likelihood estimation for a binary treatment: A tutorial. *Statistics in medicine*, *37*(16), 2530–2546.

Noll, A. T. R., van Hoogstraten, A., Nulens, K., van Geloven, N., Van Mieghem, T., Shinar, S., Zuazagoitia, P. A., Bennasar, M., Russo, F. M., Tiblad, E., Herling, L., Lewi, L., (Joanne) Verweij, E. J. T., & the ALIGN Study Group. (2025). Outcome of monochorionic diamniotic twin pregnancy with selective fetal growth restriction and continuous or intermittent absent or reversed end-diastolic umbilical artery flow: International multicenter cohort study. *Ultrasound in Obstetrics & Gynecology*, *66*(1), 41–50. https://doi.org/https://doi.org/10.1002/uog.29241

Noll, A., Van Hoogstraten, A., Nulens, K., Van Geloven, N., Van Mieghem, T., Shinar, S., Zuazagoitia, P., Bennasar, M., Russo, F., Tiblad, E., et al. (2025). Outcome of monochorionic diamniotic twin pregnancy with selective fetal growth restriction and continuous or intermittent absent or reversed end-diastolic umbilical artery flow: International multicenter cohort study. *Ultrasound in Obstetrics & Gynecology*.

Schuler, M. S., & Rose, S. (2017). Targeted maximum likelihood estimation for causal inference in observational studies. *American journal of epidemiology*, *185*(1).