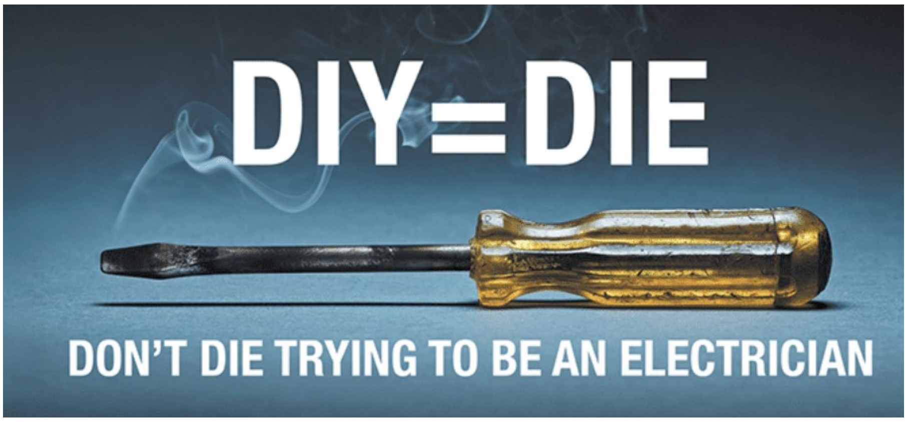
When observational data includes a treatment indicator and some possible confounders, it is very tempting to simply regress the outcome on all features (confounders and treatment alike), extract the coefficients associated with the treatment indicator, and proudly proclaim that “we have controlled for confounders and estimated the treatment effect”.

This approach is wrong. Very wrong. At least as wrong as that DIY electrical job you did last week: it looks all good and neat but you’ve made a critical mistake and there’s no way you can find out without killing yourself.



Or worse, by thinking you’ve controlled for confounders when you haven’t

I can’t explain *why* this is wrong (I’m not sure I understand it myself) but I can *show* you some examples proving that this approach is wrong. We’ll work through a few examples, where we compare the results with a traditional regression with a couple of legit causal inference libraries. Since we use simulated data, we’ll also be able to compare with the “true” treatment effect.

In all the following examples, X will be a $N \times 10$ matrix of random covariates; W will be a random treatment binary indicator (which may or may not depend on X); T will be the treatment effect; E will be the main effects; and Y will be the outcome variable ($Y^{(0)}$ if untreated, $Y^{(1)}$ if treated), so that $Y \sim \mathcal{N}(T W + E, 1)$. Our task is to estimate $\mathbb{E}(Y^{(1)} – Y^{(0)})$, the conditional average treatment effect (CATE) and $\mathbb{E}(Y^{(1)} – Y^{(0)} \mid W = 1)$, the conditional average treatment effect on the treated (CATT).

For each example we’ll estimate the treatment effects using:

- the causal forests from the grf package
- the double machine learning approach from the dmlmt package
- a random forest, using the ranger package, trained on the entire dataset
- two random forests (again from ranger) trained separately on the treated units and on the untreated units.

We’ll begin with the examples given in the `grf` package’s documentation, but first we load some required packages and set the size of the problem:

```
library(grf)
devtools::install_github(repo = "MCKnaus/dmlmt")
library(dmlmt)
library(ranger)
library(purrr)  # for rbernoulli
N <- 20000  # number of observations
```

```
P <- 10  # number of covariates
set.seed(1984)
```

Case 1: example from `causal_forest`’s help page

This first example is the one given in `causal_forest`’s help page, where the treatment assignment $W$ is completely randomized and the outcome only depends on $X_1$, $X_2$, and $X_3$:

```
X <- matrix(
  rnorm(N * P),
  nrow = N,
  ncol = P,
  dimnames = list(NULL, paste0('X', 1:P))
)
W <- rbernoulli(N, p = 0.5)
T <- pmax(X[, 1], 0)
E <- X[, 2] + pmin(X[, 3], 0)
Y <- T * W + E + rnorm(N)
```

The theoretical CATE is identical to the theoretical CATT and is given by the average positive part of the normal distribution, which is just $1 / \sqrt{2\pi}$:

```
1 / (sqrt(2 * pi))
```

```
## [1] 0.3989423
```

The empirical CATE and CATT agree very well with the theoretical value:

```
mean(T)  # empirical CATE
```

```
## [1] 0.4012137
```

```
mean(T[W])  # empirical CATT
```

```
## [1] 0.3995248
```

Let’s see now how well the causal models recover the treatment effects. First the causal forest:

```
c.forest <- causal_forest(X, Y, W)
average_treatment_effect(c.forest, target.sample = 'all')
```

```
##   estimate    std.err
## 0.39445852 0.01440154
```

```
average_treatment_effect(c.forest, target.sample = 'treated')
```

```
##   estimate    std.err
## 0.39432334 0.01440661
```

Pretty good. Next the Double Machine Learning approach:

```
invisible(dmlmt(X, W, Y))
```

```
##
##  Binary treatment
##
##
##
##  Potential outcomes:
##                      PO     SE
## Treatment 0 -0.3999623 0.0131
## Treatment 1 -0.0077989 0.0139
##
## Average effects
##              TE       SE      t         p
## T1 - T0 0.392163 0.015554 25.214 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## # of obs on / off support: 19997  /  3
```

Also very good. Let’s see now how a traditional regressor performs. There are two approaches. In the first one, we train a single model $\hat{f}()$ on the entire dataset such that $\hat{f}(X, W)$ estimates the outcome for covariates $X$ given treatment assignment $W$; the CATE is then given by $\hat{f}(X, 1) – \hat{f}(X, 0)$.

```
ranger.model <- ranger(Y ~ ., data = data.frame(X, W, Y))
```

```
ranger_cate <- function(ranger.model, X) {
  data_untreated <- data.frame(X, W = 0)
  data_treated <- data.frame(X, W = 1)
  mean(predict(ranger.model, data_treated)$predictions -
predict(ranger.model, data_untreated)$predictions)
}

ranger_cate(ranger.model, X)
```

```
## [1] 0.3480178
```

Really bad. Let’s see if another approach might work better. In that second approach, we train two models: $f_{(1)}(X[W])$ on the treated units, $f_{(0)}(X[\neg W])$ on the untreated units. The CATE is then given by $f_{(1)}(X) – f_{(0)}(X)$:

```
model_treated <- ranger(Y ~ ., data = data.frame(X, W, Y)[W, ])
model_untreated <- ranger(Y ~ ., data = data.frame(X, W, Y)[!W, ])
ranger_cate_two_models <- function(model_treated, model_untreated, X) {
  data_untreated <- data.frame(X, W = 0)
  data_treated <- data.frame(X, W = 1)
  mean(predict(model_treated, data_treated)$predictions -
predict(model_untreated, data_untreated)$predictions)
}

ranger_cate_two_models(model_treated, model_untreated, X)
```

```
## [1] 0.3945442
```

Much better. So with this first dataset with no confounding we see that “proper” causal models dominate a traditional regressor, unless we train two separate regressors on the treated and untreated units. Let’s see now the other examples.

Case 2: example with confounding from `causal_forest`’s home page

The home page for the `grf` package has an example slightly different from the example above, in which the treatment assignment $W$ depends on $X_1$:

```
X <- matrix(
  rnorm(N * P),
  nrow = N,
  ncol = P,
  dimnames = list(NULL, paste0('X', 1:P))
)
W <- rbernoulli(N, 0.4 + 0.2 * (X[, 1] > 0))
T <- pmax(X[, 1], 0)
E <- X[, 2] + pmin(X[, 3], 0)
Y <- T * W + E + rnorm(N)
```

The theoretical CATE is the same as above, but the theoretical CATT is slightly higher since being treated raises the expected treatment effect:

```
# Theoretical CATE
1 / sqrt(2*pi)
```

```
## [1] 0.3989423
```

```
# Empirical CATE
mean(T)
```

```
## [1] 0.3933349
```

```
# Theoretical CATT
1 / sqrt(2*pi) * 6 / 5
```

```
## [1] 0.4787307
```

```
# Empirical CATT
mean(T[W])
```

```
## [1] 0.4700207
```

As above, let's run the four models. First the causal forest:

```
c.forest <- causal_forest(X, Y, W)
average_treatment_effect(c.forest, target.sample = 'all')
```

```
##   estimate    std.err 
## 0.38286619 0.01456181
```

```
average_treatment_effect(c.forest, target.sample = 'treated')
```

```
##   estimate    std.err 
## 0.45825280 0.01486095
```

Excellent agreement with the ground truth. Next the `dmlmt`:

```
invisible(dmlmt(X, W, Y))
```

```
## 
##  Binary treatment
## 
## 
## 
##  Potential outcomes:
##                     PO     SE
## Treatment 0 -0.400063 0.0132
## Treatment 1 -0.012876 0.0139
## 
## Average effects
##               TE       SE      t         p    
## T1 - T0 0.387187 0.015584 24.846 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## # of obs on / off support: 19998  /  2
```

Also a very good agreement; note however that the `dmlmt` can only estimate the CATE, not the CATT. Let's see next the one-model regressor:

```
ranger.model <- ranger(Y ~ ., data = data.frame(X, W, Y))
ranger_cate(ranger.model, X)
```

```
## [1] 0.3420549
```

Just as in the previous case, this is not too good. Let's see finally the two-model regressors:

```
model_treated <- ranger(Y ~ ., data = data.frame(X, W, Y)[W, ])
model_untreated <- ranger(Y ~ ., data = data.frame(X, W, Y)[!W, ])
ranger_cate_two_models(model_treated, model_untreated, X)
```

```
## [1] 0.3885038
```

This, again, is not too bad, especially compared with the one-model regressor.

The previous two examples had relatively simple treatment and main effects. How well do these models perform in more complex situations? To see this I'm going to run them through some of the stress-tests given in section 6 of Nie and Wager (2020).

Case 3: no confounding, non-trivial main effects

In this case the treatment assignment is random and we're essentially running a randomized trial, but with complex main effects:

```
X <- matrix(
  rnorm(N * P),
  nrow = N,
  ncol = P,
  dimnames = list(NULL, paste0('X', 1:P))
)
W <- rbernoulli(N, p = 0.5)
T <- X[, 1] + log(1 + exp(X[, 2]))
E <- pmax(X[, 1] + X[, 2], X[, 3], 0) + pmax(X[, 4] + X[, 5], 0)
Y <- T * W + E + rnorm(N)
```

The empirical CATE and CATT are very close, since there's no confounding:

```
mean(T)
```

```
## [1] 0.804748
```

```
mean(T[W])
```

```
## [1] 0.8047635
```

Here are the estimates using a causal forest:

```
c.forest <- causal_forest(X, Y, W)
average_treatment_effect(c.forest, target.sample = 'all')
```

```
##   estimate    std.err
## 0.80974803 0.01518978
```

```
average_treatment_effect(c.forest, target.sample = 'treated')
```

```
##   estimate    std.err 
## 0.80902777 0.01520083
```

Next the `dmlmt`:

```
invisible(dmlmt(X, W, Y))
```

```
## 
##  Binary treatment
## 
## 
## 
##  Potential outcomes:
##                 PO     SE
## Treatment 0 1.3958 0.0139
## Treatment 1 2.2089 0.0179
## 
## Average effects
##              TE      SE     t         p    
## T1 - T0 0.81307 0.01876 43.34 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## # of obs on / off support: 19998  /  2
```

Next the one-model regressor:

```
ranger.model <- ranger(Y ~ ., data = data.frame(X, W, Y))
ranger_cate(ranger.model, X)
```

```
## [1] 0.7359536
```

And finally the two-model regressors:

```
model_treated <- ranger(Y ~ ., data = data.frame(X, W, Y)[W, ])
model_untreated <- ranger(Y ~ ., data = data.frame(X, W, Y)[!W, ])
ranger_cate_two_models(model_treated, model_untreated, X)
```

```
## [1] 0.8168369
```

All models perform rather well on this dataset with no confounders.

Case 4: confounding, non-trivial main effects

```
X <- matrix(
  runif(N * P),  # note the uniform distribution
  nrow = N,
  ncol = P,
  dimnames = list(NULL, paste0('X', 1:P))
)
trim <- function(x, eta) pmax(eta, pmin(x, 1 - eta))
W <- rbernoulli(N, trim(sinpi(X[, 1] * X[, 2]), 0.1))
T <- (X[, 1] + X[, 2]) / 2
E <- sinpi(X[, 1] * X[, 2]) + 2 * (X[, 3] - 0.5)^2 + X[, 4] + 0.5 * X[,
5]
Y <- T * W + E + rnorm(N)
```

Here are the empirical CATE and CATT:

```
mean(T)
```

```
## [1] 0.5006082
```

```
mean(T[W])
```

```
## [1] 0.5969957
```

Here are the estimates using a causal forest:

```
c.forest <- causal_forest(X, Y, W)
average_treatment_effect(c.forest, target.sample = 'all')
```

```
##   estimate    std.err
## 0.53728132 0.01858506
```

```
average_treatment_effect(c.forest, target.sample = 'treated')
```

```
##   estimate    std.err
## 0.63199708 0.02226546
```

Not too bad; perhaps a bit biased on the CATT estimate. Next the `dmlmt`:

```
invisible(dmlmt(X, W, Y))
```

```
##
##  Binary treatment
##
##
##
##  Potential outcomes:
##                  PO     SE
## Treatment 0 1.2626 0.0182
## Treatment 1 1.9945 0.0130
##
## Average effects
##              TE       SE       t         p
## T1 - T0 0.731896 0.022071 33.161 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## # of obs on / off support: 19958  /  42
```

Here the estimate is way too high. Let’s see now the one-model regressor:

```
ranger.model <- ranger(Y ~ ., data = data.frame(X, W, Y))
ranger_cate(ranger.model, X)
```

```
## [1] 0.5871552
```

And finally the two-model regressors:

```
model_treated <- ranger(Y ~ ., data = data.frame(X, W, Y)[W, ])
model_untreated <- ranger(Y ~ ., data = data.frame(X, W, Y)[!W, ])
ranger_cate_two_models(model_treated, model_untreated, X)
```

```
## [1] 0.6317095
```

Except for the causal forest, all models tend to overestimate the treatment effects.

In the final example, we have a complex confounding and non-trivial main effects, but a trivial treatment effect.

Case 5: confounding, trivial treatment effect, non-trivial main effects

```
X <- matrix(
  rnorm(N * P),
  nrow = N,
  ncol = P,
  dimnames = list(NULL, paste0('X', 1:P))
)
W <- rbernoulli(N, 1 / (1 + exp(X[, 2] + X[, 3])))
T <- 1
E <- 2 * log(1 + exp(X[, 1] + X[, 2] + X[, 3]))
Y <- T * W + E + rnorm(N)
```

In this case we have CATE = CATT = 1. Let’s see how the causal forest performs:

```
c.forest <- causal_forest(X, Y, W)
average_treatment_effect(c.forest, target.sample = 'all')
```

```
##   estimate    std.err
## 0.94210993 0.01783883
```

```
average_treatment_effect(c.forest, target.sample = 'treated')
```

```
##   estimate    std.err
## 0.94966382 0.02062717
```

Pretty good. Next the `dmlmt`:

```
invisible(dmlmt(X, W, Y))
```

```
##
##  Binary treatment
##
##
##
##  Potential outcomes:
##                 PO     SE
## Treatment 0 1.9719 0.0254
## Treatment 1 2.9588 0.0273
##
## Average effects
##               TE       SE      t         p
## T1 - T0 0.986893 0.032839 30.053 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## # of obs on / off support: 19855  /  145
```

Very good estimates too. How about the one-model regressor?

```
ranger.model <- ranger(Y ~ ., data = data.frame(X, W, Y))
ranger_cate(ranger.model, X)
```

```
## [1] 0.613593
```

Oops, not that good. Perhaps the two-model regressor does better?

```
model_treated <- ranger(Y ~ ., data = data.frame(X, W, Y)[W, ])
model_untreated <- ranger(Y ~ ., data = data.frame(X, W, Y)[!W, ])
ranger_cate_two_models(model_treated, model_untreated, X)
```

```
## [1] 0.7121187
```

Slightly better, but still a bit off.

Conclusion

The following table summarizes the empirical CATE in each case, and the CATE estimated by each algorithm:

Case	CATE	GRF	DMLMT	RF1	RF2
1	0.40	0.39	0.39	0.35	0.39
2	0.39	0.38	0.39	0.34	0.39
3	0.80	0.81	0.81	0.74	0.82
4	0.50	0.54	0.73	0.59	0.63
5	1.00	0.94	0.99	0.61	0.71

The causal forest from the `grf` package *always* dominates the other methods; like I said at the beginning, I'm not entirely sure why, but this quick study should alert you to the fact that **causal studies are hard**, because unlike traditional regression problems here **you have no ground truth** against which to cross-validate your model.

I would love to hear from anyone who could explain in simple terms why we see such a variety of estimation accuracies.