

Using Causal Forests to Estimate Heterogeneous Treatment Effects

Marcel Harald Wachter

August 23, 2022

Abstract

Randomized controlled trials gained increasing significance for economics in the past years and decades. With it the need of accurate estimations for heterogeneous treatment effects also gained importance. Oftentimes policy makers make decisions based on the estimated treatment effects. If the treatment effect differs based on certain observable characteristics it is important to quantify these effects. Traditionally heterogeneous treatment effects were estimated with linear interaction models. But in recent years machine learning techniques, e.g. causal forests, were developed to also estimate heterogeneous treatment effects and improve accuracy. This paper compares the results of estimating heterogeneous treatment effects with linear interaction models and causal forests. For the simulation study three different data generating processes that simulate observational studies and randomized controlled trials are considered. The results suggest that, compared to a linear interaction model, a causal forest provides the best performance in all considered setups. Subsequently a causal forest is used in an empirical application to estimate heterogeneous treatment effects and compare them to a linear interaction model.

Contents

1	Introduction	1
2	Causal Forests	1
3	Simulation Study	2
3.1	Data Generating Process	2
3.1.1	Setup 1	2
3.1.2	Setup 2	7
3.1.3	Setup 3	10
3.2	Simulation	21
3.2.1	Results	23
4	Empirical Application	25
5	Conclusion	31

1 Introduction

From the 1980s onwards, randomized controlled trials (RCTs), e.g. in development economics, have become increasingly popular as a way to identify causal effects (Cameron et al., 2016). Currently, the AEA RCT Registry (2022) lists 1886 RCTs as ongoing. From that it becomes clear what significance RCTs take in the current economic research.

The main objective of RCTs is to answer causal questions and to estimate causal effects of treatments on outcomes by overcoming the selection bias (Duflo et al., 2007, pp. 3899–3903). Following the potential outcome framework (Rubin, 1974; Rubin, 1977; Holland, 1986), the causal effect for unit i can be denoted by $\tau_i = Y_{1i} - Y_{0i}$ where Y_{1i} denotes the potential outcome if i receives the treatment and Y_{0i} if it does not. An additional parameter of interest is the average treatment effect (ATE) defined by $\tau_{ATE} = E[Y_{1i} - Y_{0i}]$ which describes the mean outcome difference between the treatment and control group, i.e. the average treatment effect across everyone in the population. One of the objectives of RCTs is to estimate the ATE. In addition to the ATE our interest often also lies on the size of the treatment effect for various subgroups. In the potential outcome framework this effect is called conditional average treatment effect (CATE) and is denoted by $\tau(x) = E[Y_{1i} - Y_{0i} | X_i = x]$ where X_i is a vector of covariates. But RCTs are not the only source of data for conducting estimations of heterogeneous treatment effects (HTEs), observational studies with quasi-experiments play an important role in causal inference too.

The effects of interventions are often heterogeneous depending on the population they affect (Duflo et al., 2007, p. 3948). The conclusions drawn from experiments may vary significantly depending on the heterogeneity of the effect. Traditionally, CATE are obtained by means of subgroup analyses or by a linear model where interactions between the treatment variable and explanatory variables of interest are specified. However both of these method have their drawbacks. Subgroup analyses suffer can suffer from low power. Linear models with interactions on the other hand can become unfeasible when the number of covariates is high. This stems from the fact that not only do we have to include interactions between treatment and covariates but also for all possible combinations of covariates with and without treatment. Machine learning techniques such as causal forests could improve the estimation of HTEs.

The objective of this term paper is to compare the estimation accuracy of causal forests and linear interaction model for HTEs. For this purpose three setups with different data generating processes, that approximate observational studies as well as RCTs, are considered. After depicting the accuracy and performance of each setup, a simulation study is conducted to obtain an empirical distribution of the estimates.

The remainder of the paper is organised as follows. Section 2 briefly describes causal forests. In Section 3 a simulation study is conducted. In Section 4 an empirical application is considered. Section 5 concludes.

2 Causal Forests

A causal forest, first proposed by Wager and Athey (2018), is essentially a random forest build from causal trees who averages the predictions of many of these causal trees. Causal trees are decision trees for treatment effects and can be viewed as a variation of regression trees. The functionality of regression trees is pretty straightforward. The idea is to segment the predictor space into a number of simple regions and fit a simple model in each one. Use p to denote the number of inputs and N for the number of observations. Our data can be described by (x_i, y_i) for $i = 1, 2, \dots, N$ with $x_i = (x_{i1}, x_{i2}, \dots, x_{ip})$. The algorithm for segmentation and fitting can be described as following. Starting from the entire predictor space a splitting variable j and split point s is considered and a pair of half-planes

$$R_1(j, s) = \{X \mid X_j \leq s\} \quad \text{and} \quad R_2(j, s) = \{X \mid X_j > s\} \quad (1)$$

defined. The splitting variable j and split point s are chosen, such that they solve

$$\min_{j,s} \left[\sum_{x_i \in R_1(j,s)} (y_i - \bar{y}_1(j,s))^2 + \sum_{x_i \in R_2(j,s)} (y_i - \bar{y}_2(j,s))^2 \right] \quad (2)$$

where $\bar{y}_1(j, s)$ and $\bar{y}_2(j, s)$ are the mean outcomes in $R_1(j, s)$ and $R_2(j, s)$ respectively. The same process is repeated for one of the regions created until a stopping criterion is reached. The response for every region R_1, R_2, \dots, R_M is modelled as a constant c_m :

$$f(x) = \sum_{m=1}^M c_m I(x \in R_m) \quad (3)$$

The best c_m is then simply the average y_i in region R_m . Causal trees are similar to this process, they differ however in the splitting criterion used which is now trying to find splits optimized for treatment heterogeneity. A causal forest then simply averages the treatment effects of causal trees and aggregates the prediction for the HTE.

3 Simulation Study

All analyses were conducted in R Version 4.2.1 (R Core Team, 2022). For estimating causal forests the package “grf” was used (Tibshirani et al., 2021). The data generating process is based on Wager and Athey (2018) and implemented with the package “htesim” from Tibshirani et al. (2021).

3.1 Data Generating Process

Following Wager and Athey (2018, p. 1238), three different data generating processes are considered. Each setup differs in their treatment propensity function which specifies the probability of treatment assignment conditional on observed baseline characteristics, the main effect function which specifies the main effect, and the treatment effect function which specifies the treatment effect. In total four covariates are considered. The models are trained on a training sample with 500 observations and tested on a test sample with 10000 observations. The data generating process is specified through following three functions:

$$m(x) = 2^{-1} \mathbb{E}[Y^{(0)} + Y^{(1)} | X = x] \quad (4)$$

for the main effect function.

$$\tau(x) = \mathbb{E}[Y^{(1)} - Y^{(0)} | X = x] \quad (5)$$

for the treatment effect function. And

$$e(x) = \mathbb{P}[W = 1 | X = x] \quad (6)$$

for the treatment propensity function. The covariates X are drawn from a uniform distribution

$$X \sim U([0, 1]^d) \quad (7)$$

and the outcome variable Y from

$$Y^{0/1} \sim \mathcal{N}(\mathbb{E}[Y^{0/1} | X], 1). \quad (8)$$

See also Wager and Athey (2018, p. 1238) for more details on the data generating process.

3.1.1 Setup 1

This setup resembles an observational study. The treatment propensity function is given by

$$e(X) = \frac{1}{4} (1 + \beta_{2,4}(X_1)),$$

the main effect function by

$$m(X) = 2X_1 - 1,$$

and the treatment effect function by

$$\tau(x) = 0.$$

These functions simulate an often found problem in observational studies that treatment assignment is correlated with potential outcomes (Wager & Athey, 2018, p. 1238).

Clear Workspace

```
rm(list = ls())
```

Loading packages

```
library(grf)
library(htesim)
library(lattice)
library(colorspace)
library(ggplot2)
library(gridExtra)
library(modelsummary)
library(DiagrammeRsvg)
library(rsvg)
library(readstata13)
```

Setting seed

```
options(scipen=999)
set.seed(123)
```

Specifying number of predictors and observations:

```
n <- 500
p <- 4
```

Data generating process

```
dgp1 <- dgp(p = pF_x1, m = mF_x1, t = 0, model = "normal", xmodel = "unif")
sim1 <- simulate.dgp(object = dgp1, nsim = n, dim = p, nsimtest = 10000)
x_sim1 <- data.matrix(frame = sim1[, c(1:p)])
Y_sim1 <- sim1$y
W_sim1 <- as.numeric(levels(sim1$trt))[sim1$trt]
```

Test Data

```
test1 <- attr(sim1, "testxdf")
test1$trt <- NULL
```

Train Forest and Linear Model

```
cf1 <- causal_forest(X = x_sim1, Y = Y_sim1, W = W_sim1)
tree1 <- get_tree(forest = cf1, index = 1)
lin1 <- lm(formula = y ~ (trt + X1 + X2 + X3 + X4)^2, data = sim1)
```

Predict CATE for test data

```
tau1_hat_cf <- predict(object = cf1, newdata = test1)
tau1_hat_cf <- as.numeric(tau1_hat_cf$predictions)

test_untr <- test_tr <- test1
test_untr$trt <- "0"
test_tr$trt <- "1"
tau1_hat_lin <- predict(lin1, test_tr) - predict(lin1, test_untr)
```

We receive the true effects of the test data by using

Table 1: Summary Statistics of the Simulated Data of Setup 1 (Numerical Variables)

	Unique (#)	Missing (%)	Mean	SD	Min	Median	Max	
X1	500	0	0.5	0.3	0.0	0.5	1.0	
X2	500	0	0.5	0.3	0.0	0.5	1.0	
X3	500	0	0.5	0.3	0.0	0.5	1.0	
X4	500	0	0.5	0.3	0.0	0.5	1.0	
y	500	0	0.0	1.2	-3.2	0.1	4.9	

```
tau1 <- predict(object = dgp1, newdata = test1[, "tfct"])
```

The performance is evaluated by the mean squared error (MSE)

```
mse1_cf <- mean((tau1 - tau1_hat_cf)^2)
mse1_lin <- mean((tau1 - tau1_hat_lin)^2)
```

Plots

```
res1 <- data.frame(test1, tau1, tau1_hat_cf, tau1_hat_lin, mse1_cf, mse1_lin)
plot1 <- ggplot(data = res1, aes(X1,X2)) + geom_point(aes(color = tau1)) +
  theme_bw() + scale_color_continuous_sequential(palette = "BurgYl") +
  theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank()) +
  theme(legend.position = "none") + coord_fixed() + ggtitle("True Effect")
plot2 <- ggplot(data = res1, aes(X1,X2)) + geom_point(aes(color = tau1_hat_cf)) +
  theme_bw() + scale_color_continuous_sequential(palette = "BurgYl") +
  theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank()) +
  theme(legend.position = "none") + coord_fixed() + ggtitle("Causal Forest")
plot3 <- ggplot(data = res1, aes(X1,X2)) + geom_point(aes(color = tau1_hat_lin)) +
  theme_bw() + scale_color_continuous_sequential(palette = "BurgYl") +
  theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank()) +
  theme(legend.position = "none") + coord_fixed() + ggtitle("Linear Int. Mod.")
bwtheme <- standard.theme("pdf", color=FALSE)
plot13 <- histogram(tau1, par.settings=bwtheme, xlab = "HTE")
plot14 <- histogram(tau1_hat_cf, par.settings=bwtheme, xlab = "HTE")
plot15 <- histogram(tau1_hat_lin, par.settings=bwtheme, xlab = "HTE")
tree1_plot <- plot(tree1)
cat(DiagrammeRsvg:::export_svg(tree1_plot), file = "./Figures/tree1.svg")
rsvg_pdf(svg = "./Figures/tree1.svg", file = "./Figures/tree1.pdf")

datasummary_skim(data = sim1, output = "kableExtra", type = "numeric",
                  title = "Summary Statistics of the Simulated Data of
Setup 1 (Numerical Variables)")

datasummary_skim(data = sim1, output = "kableExtra", type = "categorical",
                  title = "Summary Statistics of the Simulated Data of
Setup 1 (Categorical Variables)")

modelsummary(lin1, coef.omit = "^(?!trt)",
             gof.omit = "BIC|AIC|RMSE|Log.Lik.", output = "kableExtra",
             stars = TRUE,
             title = "Regression Table for the Linear Interaction Model of Setup 1")
```

Table 2: Summary Statistics of the Simulated Data of Setup 1 (Categorical Variables)

trt	N	%
0	240	48.0
1	260	52.0

Table 3: Regression Table for the Linear Interaction Model of Setup 1

Model 1	
trt1	-0.049 (0.340)
trt1 × X1	0.610+ (0.337)
trt1 × X2	-0.416 (0.323)
trt1 × X3	0.039 (0.336)
trt1 × X4	0.132 (0.326)
Num.Obs.	500
R2	0.309
R2 Adj.	0.287
F	14.411

+ p < 0.1, * p < 0.05, ** p < 0.01, *** p < 0.001

```
grid.arrange(plot1, plot2, plot3, plot13, plot14, plot15, ncol = 3)
```

Figure 1 depicts the true treatment effect $\tau(X_i)$ of 10000 random test examples X_i along with the treatment effects estimated by a causal forest and the linear interaction model for setup 1. The coordinates of the points correspond to the values of the first two covariates X_1 and X_2 . The size of the treatment effect is denoted by the colour, where dark means a high and bright a low treatment effect. Both methods struggle to estimate accurate results if one compares them to the true treatment effect. The ATE for both methods is fairly similar to the true ATE.

```
mean(tau1)

## [1] 0
mean(tau1_hat_cf)

## [1] 0.1288135
mean(tau1_hat_lin)

## [1] 0.1339017
```

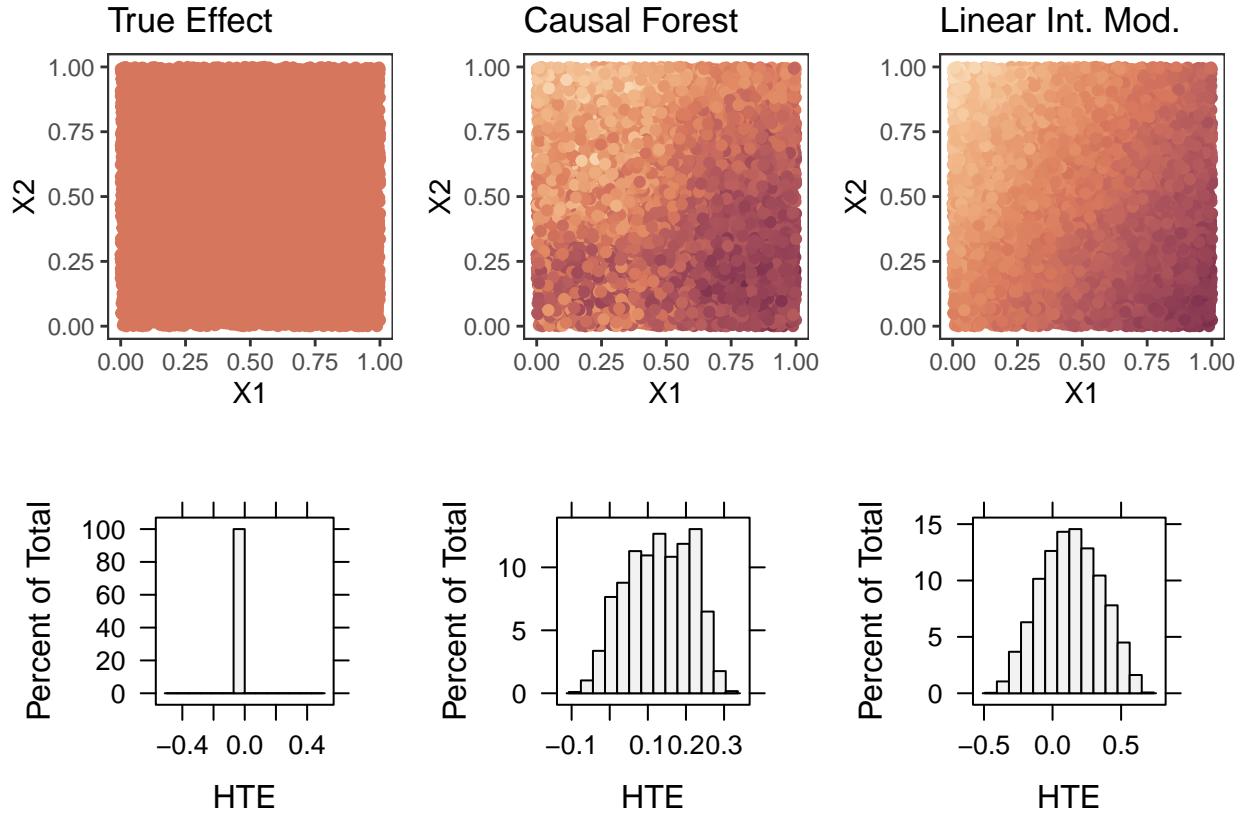
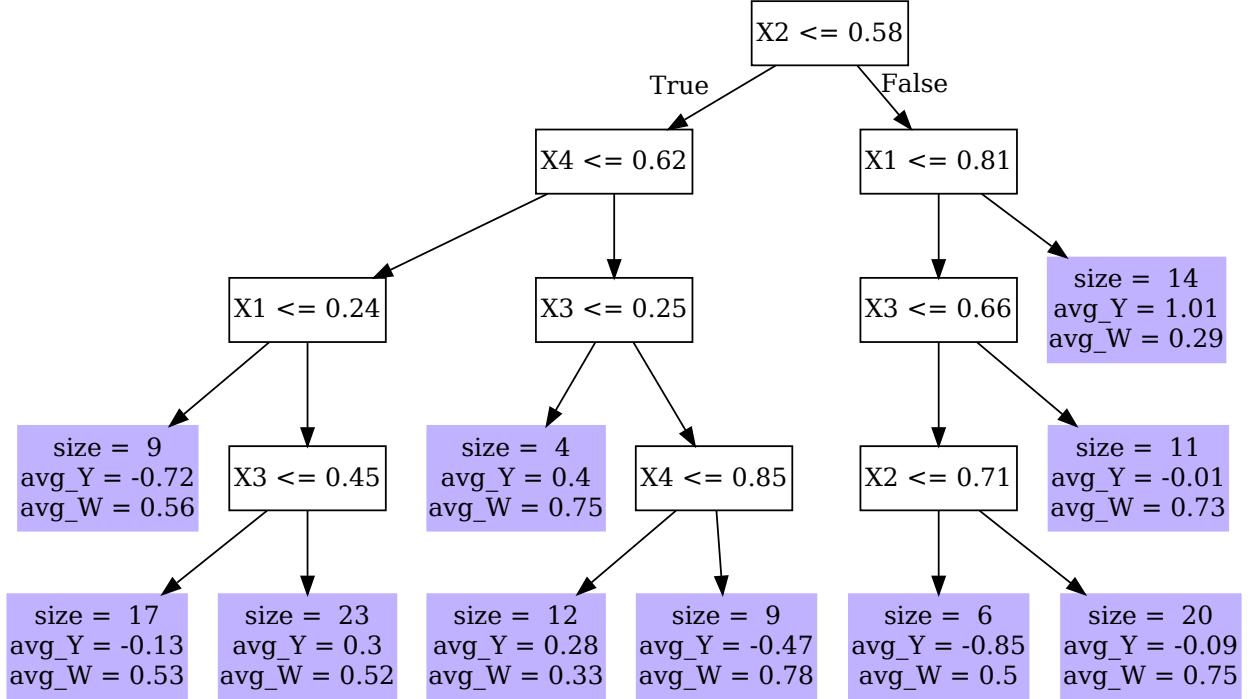


Figure 1: Treatment Effects Generated by Setup 1

Figure 2: This figure shows one tree from the causal forest of setup 1. In total, 2000 such trees are grown.



3.1.2 Setup 2

The following two setups and their results are similar to setup 1. The main differences are the changes in the data generating process to generate data similar to one a RCT produces.

This setup resembles a randomized experiment. The treatment propensity function is given by

$$e(x) = 0.5,$$

the main effect function by

$$m(x) = 0,$$

and the treatment effect function by

$$\tau(x) = \zeta(X1)\zeta(X2) \quad \text{where} \quad \zeta(x) = \frac{1}{1 + e^{-20(x-1/3)}}.$$

This setup tests how well the methods adapt to heterogeneity in the treatment effect.

```
dgp2 <- dgp(p = 0.5, m = 0, t = tF_exp_x1_x2, model = "normal", xmodel = "unif")
sim2 <- simulate.dgp(object = dgp2, nsim = n, dim = p, nsimtest = 10000)
x_sim2 <- data.matrix(frame = sim2[, c(1:p)])
Y_sim2 <- sim2$y
W_sim2 <- as.numeric(levels(sim2$trt))[sim2$trt]

test2 <- attr(sim2, "testxdf")
test2$trt <- NULL

cf2 <- causal_forest(X = x_sim2, Y = Y_sim2, W = W_sim2)
tree2 <- get_tree(forest = cf2, index = 1)
lin2 <- lm(formula = y ~ (trt + X1 + X2 + X3 + X4)^2, data = sim2)

tau2_hat_cf <- predict(object = cf2, newdata = test2)
tau2_hat_cf <- as.numeric(tau2_hat_cf$predictions)

test_untr<- test_tr <- test2
test_untr$trt <- "0"
test_tr$trt <- "1"
tau2_hat_lin <- predict(lin2, test_tr) - predict(lin2, test_untr)

tau2 <- predict(object = dgp2, newdata = test2)[, "tfct"]

mse2_cf <- mean((tau2 - tau2_hat_cf)^2)
mse2_lin <- mean((tau2 - tau2_hat_lin)^2)

res2 <- data.frame(test2, tau2, tau2_hat_cf, tau2_hat_lin)
plot4 <- ggplot(data = res2, aes(X1,X2)) + geom_point(aes(color = tau2)) +
  theme_bw() + scale_color_continuous_sequential(palette = "BurgYl") +
  theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank()) +
  theme(legend.position = "none") + coord_fixed() + ggtitle("True Effect")
plot5 <- ggplot(data = res2, aes(X1,X2)) + geom_point(aes(color = tau2_hat_cf)) +
  theme_bw() + scale_color_continuous_sequential(palette = "BurgYl") +
  theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank()) +
  theme(legend.position = "none") + coord_fixed() + ggtitle("Causal Forest")
plot6 <- ggplot(data = res2, aes(X1,X2)) + geom_point(aes(color = tau2_hat_lin)) +
  theme_bw() + scale_color_continuous_sequential(palette = "BurgYl") +
  theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank()) +
```

Table 4: Summary Statistics of the Simulated Data of Setup 2 (Numerical Variables)

	Unique (#)	Missing (%)	Mean	SD	Min	Median	Max	
X1	500	0	0.5	0.3	0.0	0.5	1.0	[REDACTED]
X2	500	0	0.5	0.3	0.0	0.5	1.0	[REDACTED]
X3	500	0	0.5	0.3	0.0	0.5	1.0	[REDACTED]
X4	500	0	0.5	0.3	0.0	0.5	1.0	[REDACTED]
y	500	0	1.3	1.8	-3.2	1.0	6.6	[REDACTED]

Table 5: Summary Statistics of the Simulated Data of Setup 2 (Categorical Variables)

	trt	N	%
0	273	54.6	
1	227	45.4	

```

theme(legend.position = "none") + coord_fixed() + ggtitle("Linear Int. Mod.")
plot16 <- histogram(tau2, par.settings=bwtheme, xlab = "HTE")
plot17 <- histogram(tau2_hat_cf, par.settings=bwtheme, xlab = "HTE")
plot18 <- histogram(tau2_hat_lin, par.settings=bwtheme, xlab = "HTE")
tree2_plot <- plot(tree2)
cat(DiagrammeRsvg::export_svg(tree2_plot), file = "./Figures/tree2.svg")
rsvg_pdf(svg = "./Figures/tree2.svg", file = "./Figures/tree2.pdf")

datasummary_skim(data = sim2, output = "kableExtra", type = "numeric",
                  title = "Summary Statistics of the Simulated Data of
Setup 2 (Numerical Variables)")

datasummary_skim(data = sim2, output = "kableExtra", type = "categorical",
                  title = "Summary Statistics of the Simulated Data of
Setup 2 (Categorical Variables)")

modelsummary(lin2, coef.omit = "^(?!\$trt)\$",
             gof.omit = "BIC|AIC|RMSE|Log.Lik.", output = "kableExtra",
             stars = TRUE,
             title = "Regression Table for the Linear Interaction Model of Setup 2")

grid.arrange(plot4, plot5, plot6, plot16, plot17, plot18, ncol = 3)

```

Figure 2 depicts the true treatment effect $\tau(X_i)$ of 10000 random test examples X_i along with the treatment effects estimated by a causal forest and the linear interaction model for setup 2. The coordinates of the points correspond to the values of the first two covariates X_1 and X_2 . The size of the treatment effect is denoted by the colour, where dark means a high and bright a low treatment effect. The predictions of the causal forest are highly accurate in this setting, while the linear interaction model lacks accuracy. This can also be seen in the histograms below. The histogram of the causal forest is far more similar to the histogram of the true effect than the one from the linear interaction model. In terms of the ATE both methods are fairly similar as can bee seen when the ATE is calculated for all three:

```
mean(tau2)
```

```
## [1] 2.776182
```

Table 6: Regression Table for the Linear Interaction Model of Setup 2

Model 1	
trt1	0.378 (0.330)
trt1 × X1	2.105*** (0.330)
trt1 × X2	2.272*** (0.324)
trt1 × X3	-0.237 (0.322)
trt1 × X4	0.656* (0.303)
Num.Obs.	500
R2	0.692
R2 Adj.	0.682
F	72.355

+ p < 0.1, * p < 0.05, ** p < 0.01, *** p < 0.001

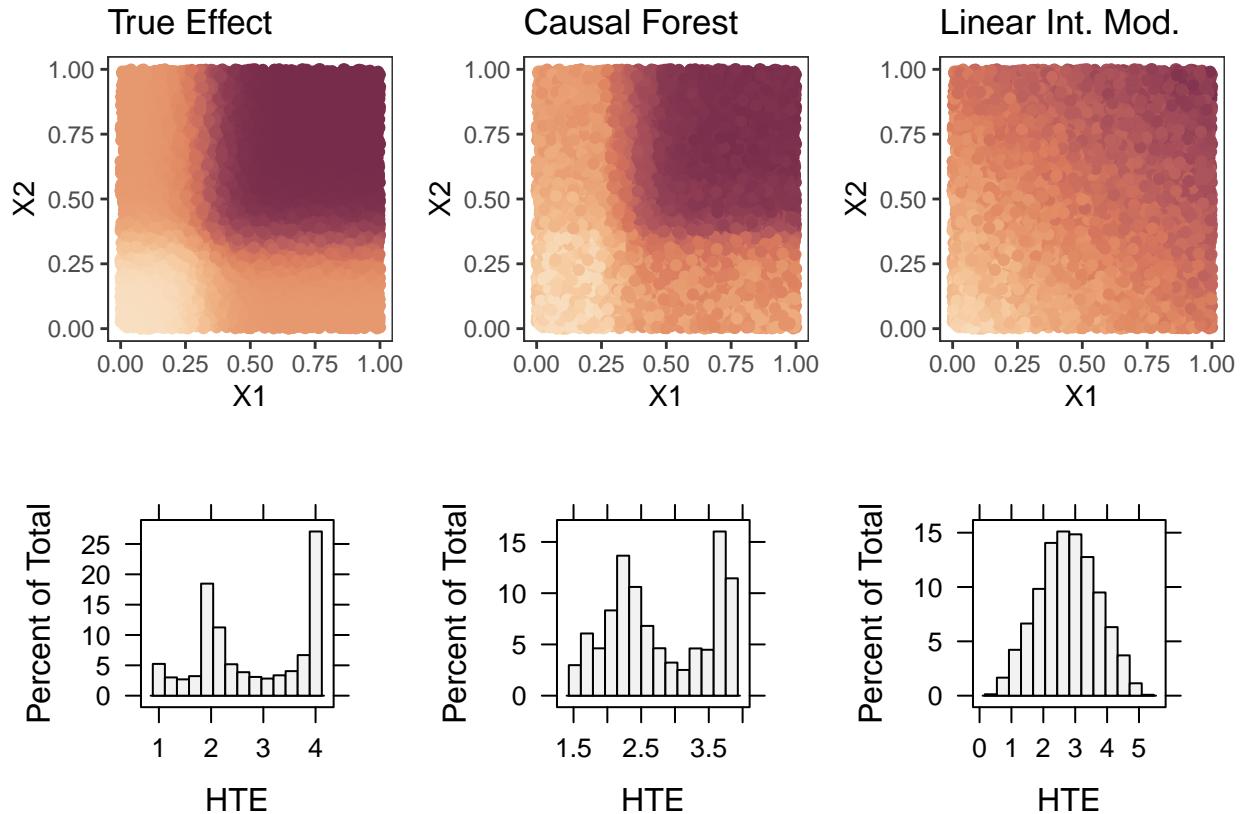


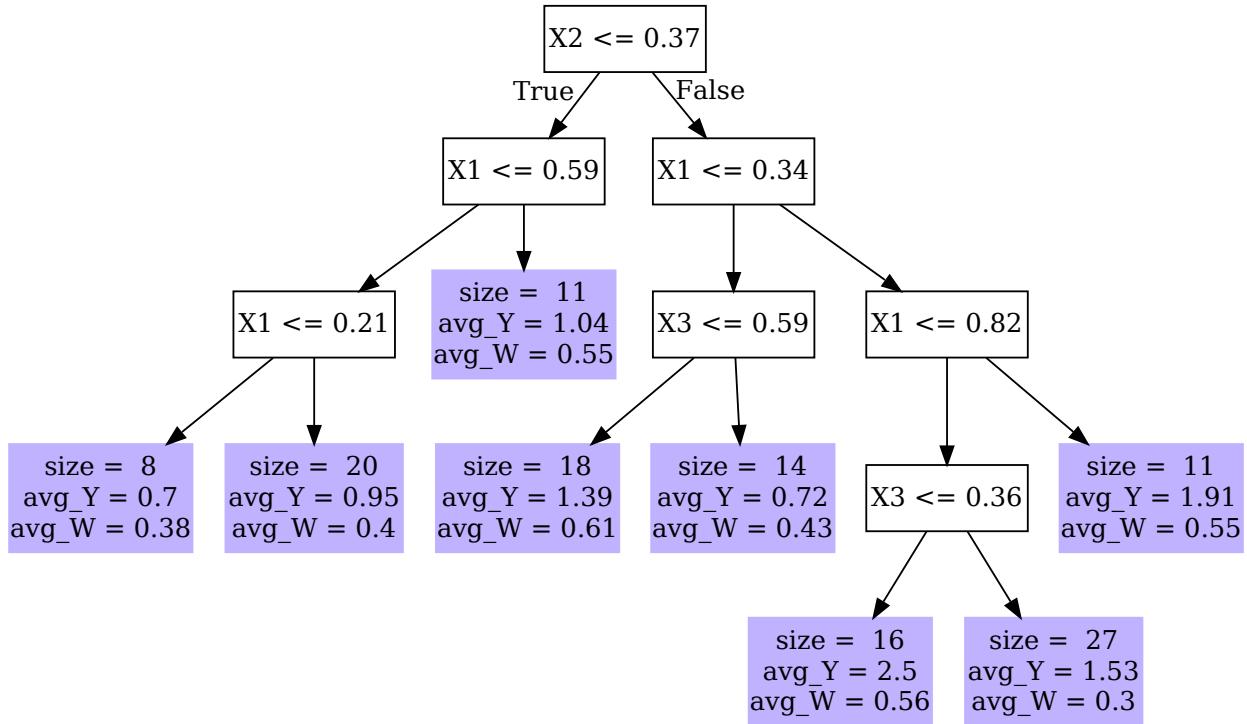
Figure 3: Treatment Effects Generated by Setup 2

```

mean(tau2_hat_cf)
## [1] 2.779024
mean(tau2_hat_lin)
## [1] 2.780543

```

Figure 4: This figure shows one tree from the causal forest of setup 2. In total, 2000 such trees are grown.



Setup 3 resembles a randomized experiment and is similar to Setup 2. The treatment propensity function is given by

$$e(x) = 0.5,$$

the main effect function by

$$m(x) = 0,$$

and the treatment effect function by

$$\tau(x) = \zeta(X1)\zeta(X2) \quad \text{where} \quad \zeta(x) = \frac{2}{1 + e^{-12(x-1/2)}}$$

which is different compared to setup 2 to account for the fact that random forests can fill the valleys and flatten the peaks of the true $\tau(x)$ function (Wager & Athey, 2018, p. 1238).

```

dgp3 <- dgp(p = 0.5, m = 0, t = tF_exp2_x1_x2, model = "normal", xmodel = "unif")
sim3 <- simulate.dgp(object = dgp3, nsim = n, dim = p, nsimtest = 10000)
x_sim3 <- data.matrix(frame = sim3[, c(1:4)])
Y_sim3 <- sim3$trt
W_sim3 <- as.numeric(levels(sim3$trt))[sim3$trt]

```

```

test3 <- attr(sim3, "testxdf")
test3$trt <- NULL

cf3 <- causal_forest(X = x_sim3, Y = Y_sim3, W = W_sim3)
tree3 <- get_tree(forest = cf3, index = 1)
lin3 <- lm(formula = y ~ (trt + X1 + X2 + X3 + X4)^2, data = sim3)

tau3_hat_cf <- predict(object = cf3, newdata = test3)
tau3_hat_cf <- as.numeric(tau3_hat_cf$predictions)

test_untr<- test_tr <- test3
test_untr$trt <- "0"
test_tr$trt <- "1"
tau3_hat_lin <- predict(lin3, test_tr) - predict(lin3, test_untr)

tau3 <- predict(object = dgp3, newdata = test3)[, "tfct"]

mse3_cf <- mean((tau3 - tau3_hat_cf)^2)
mse3_lin <- mean((tau3 - tau3_hat_lin)^2)

res3 <- data.frame(test3, tau3, tau3_hat_cf, tau3_hat_lin)
plot7 <- ggplot(data = res3, aes(X1,X2)) + geom_point(aes(color = tau3)) +
  theme_bw() + scale_color_continuous_sequential(palette = "BurgYl") +
  theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank()) +
  theme(legend.position = "none") + coord_fixed() + ggtitle("True Effect")
plot8 <- ggplot(data = res3, aes(X1,X2)) + geom_point(aes(color = tau3_hat_cf)) +
  theme_bw() + scale_color_continuous_sequential(palette = "BurgYl") +
  theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank()) +
  theme(legend.position = "none") + coord_fixed() + ggtitle("Causal Forest")
plot9 <- ggplot(data = res3, aes(X1,X2)) + geom_point(aes(color = tau3_hat_lin)) +
  theme_bw() + scale_color_continuous_sequential(palette = "BurgYl") +
  theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank()) +
  theme(legend.position = "none") + coord_fixed() + ggtitle("Linear Int. Mod.")
plot19 <- histogram(tau3, par.settings=bwtheme, xlab = "HTE")
plot20 <- histogram(tau3_hat_cf, par.settings=bwtheme, xlab = "HTE")
plot21 <- histogram(tau3_hat_lin, par.settings=bwtheme, xlab = "HTE")
tree3_plot <- plot(tree3)
cat(DiagrammeRsvg:::export_svg(tree3_plot), file = "./Figures/tree3.svg")
rsvg_pdf(svg = "./Figures/tree3.svg", file = "./Figures/tree3.pdf")

datasummary_skim(data = sim3, output = "kableExtra", type = "numeric",
                  title = "Summary Statistics of the Simulated Data of
Setup 3 (Numerical Variables)")

datasummary_skim(data = sim3, output = "kableExtra", type = "categorical",
                  title = "Summary Statistics of the Simulated Data of
Setup 3 (Categorical Variables)")

modelsummary(lin3, coef.omit = "^(?!trt)",
             gof.omit = "BIC|AIC|RMSE|Log.Lik.", output = "kableExtra",
             stars = TRUE,
             title = "Regression Table for the Linear Interaction Model of Setup 3")

```

Table 7: Summary Statistics of the Simulated Data of Setup 3 (Numerical Variables)

	Unique (#)	Missing (%)	Mean	SD	Min	Median	Max	
X1	500	0	0.5	0.3	0.0	0.5	1.0	
X2	500	0	0.5	0.3	0.0	0.5	1.0	
X3	500	0	0.5	0.3	0.0	0.5	1.0	
X4	500	0	0.5	0.3	0.0	0.5	1.0	
y	500	0	0.5	1.5	-3.7	0.3	5.3	

Table 8: Summary Statistics of the Simulated Data of Setup 3 (Categorical Variables)

trt	N	%
0	254	50.8
1	246	49.2

Table 9: Regression Table for the Linear Interaction Model of Setup 3

Model 1	
trt1	-2.199*** (0.386)
trt1 × X1	2.768*** (0.364)
trt1 × X2	2.653*** (0.371)
trt1 × X3	0.332 (0.355)
trt1 × X4	0.399 (0.375)
Num.Obs.	500
R2	0.434
R2 Adj.	0.417
F	24.789

+ p < 0.1, * p < 0.05, ** p < 0.01, *** p < 0.001

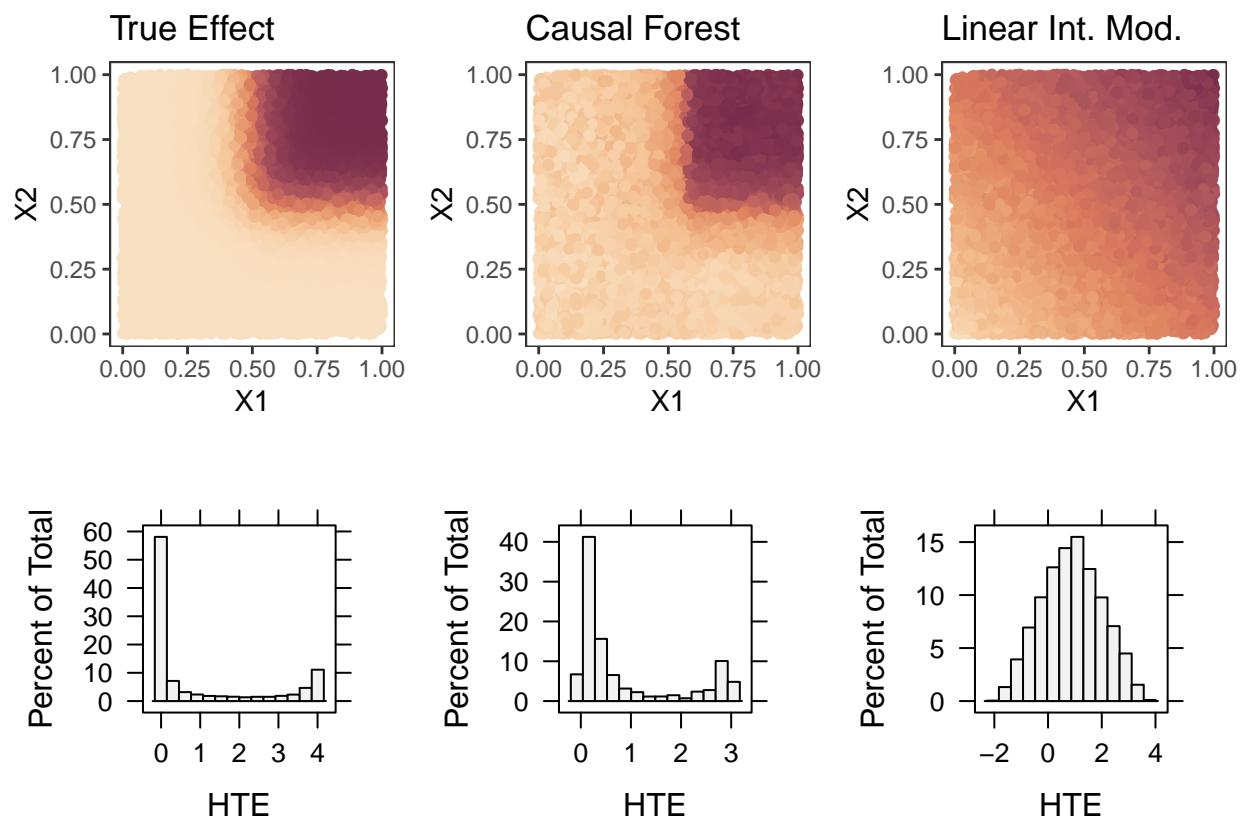


Figure 5: Treatment Effects Generated by Setup 3

```
grid.arrange(plot7, plot8, plot9, plot19, plot20, plot21, ncol = 3)
```

Figure 3 depicts the true treatment effect $\tau(X_i)$ of 10000 random test examples X_i along with the treatment effects estimated by a causal forest and the linear interaction model for setup 3. The coordinates of the points correspond to the values of the first two covariates X_1 and X_2 . The size of the treatment effect is denoted by the colour, where dark means a high and bright a low treatment effect. Again, the causal forest has an excellent performance while the linear interaction model lacks accuracy. You can also clearly see how the valleys of the true treatment effect are filled and the hills flattened as mentioned above. Again the ATE of both methods is quite near the true ATE but not as good as in the previous setup.

```
mean(tau3)
```

```
## [1] 1.001467
```

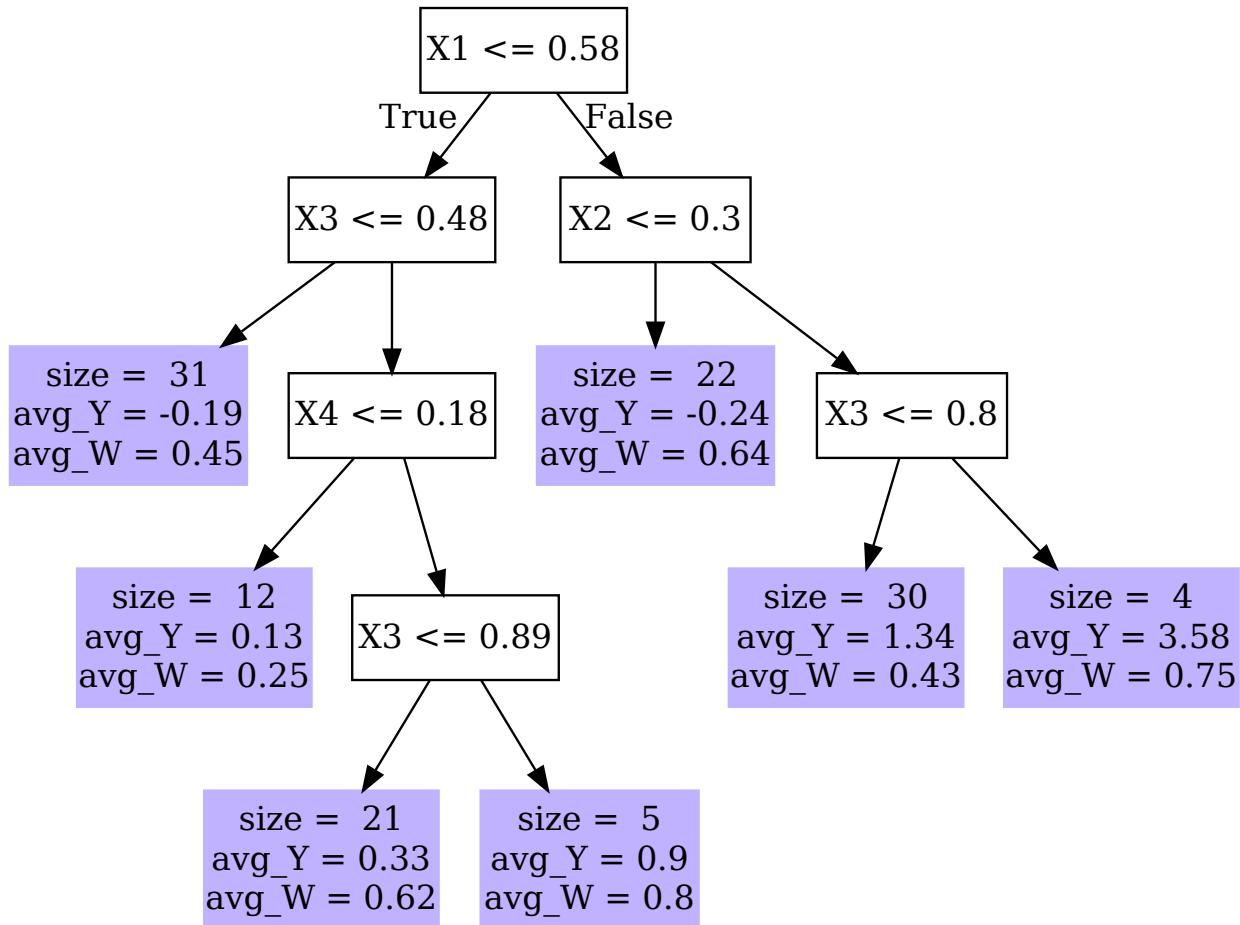
```
mean(tau3_hat_cf)
```

```
## [1] 0.8490406
```

```
mean(tau3_hat_lin)
```

```
## [1] 0.8872231
```

Figure 6: This figure shows one tree from the causal forest of setup 3. In total, 2000 such trees are grown.



These results in this section make it clear that the causal forest is superior to the linear interaction model in all considered setups. This can also be seen if we compare the MSE of both methods in all three setups.

```

mse1_cf
## [1] 0.02363871
mse1_lin

## [1] 0.06505173
mse2_cf

## [1] 0.09320533
mse2_lin

## [1] 0.2678735
mse3_cf

## [1] 0.2910437
mse3_lin

## [1] 0.8890762

```

How do the methods perform when the number of observations they are trained on varies? For this purpose I calculate the MSE of both methods for a range of training observations from 10 to 500. Results are shown in Figure 7, 8, and 9. I find that when the number of observations exceeds 200 the MSE of the causal forest is always lower compared to the linear interaction model. Depending on available computing power this simulation can take quite some time. Therefore the code is commented out and a file of the produced dataset is available in the data folder of the project.

```

# mse_cf_obs1 <- rep(NA, times = 500)
# mse_lin_obs1 <- rep(NA, times = 500)
# p <- 4
# for (i in 10:500) {
#   dgp1 <- dgp(p = pF_x1, m = mF_x1, t = 0, model = "normal", xmodel = "unif")
#   sim1 <- simulate.dgp(object = dgp1, nsim = i, dim = p, nsimtest = 1000)
#   x_sim1 <- data.matrix(frame = sim1[, c(1:p)])
#   Y_sim1 <- sim1$y
#   W_sim1 <- as.numeric(levels(sim1$trt))[sim1$trt]
#
#   test1 <- attr(sim1, "testxdf")
#   test1$trt <- NULL
#
#   cf1 <- causal_forest(X = x_sim1, Y = Y_sim1, W = W_sim1)
#   tree1 <- get_tree(forest = cf1, index = 1)
#   lin1 <- lm(formula = y ~ (trt + X1 + X2 + X3 + X4)^2, data = sim1)
#
#   tau1_hat_cf <- predict(object = cf1, newdata = test1)
#   tau1_hat_cf <- as.numeric(tau1_hat_cf$predictions)
#
#   test_untr <- test_tr <- test1
#   test_untr$trt <- "0"
#   test_tr$trt <- "1"
#   tau1_hat_lin <- predict(lin1, test_tr) - predict(lin1, test_untr)
#
#   tau1 <- predict(object = dgp1, newdata = test1)[, "tfct"]
#
#   mse_cf_obs1[i] <- mean((tau1 - tau1_hat_cf)^2)

```

```

#   mse_lin_obs1[i] <- mean((tau1 - tau1_hat_lin)^2)
# }
#
# #####
#
# mse_cf_obs2 <- rep(NA, times = 500)
# mse_lin_obs2 <- rep(NA, times = 500)
# mse_lin_obs4 <- matrix(data = NA, nrow = 500, ncol = 10)
# for (i in 10:500) {
#   dgp2 <- dgp(p = 0.5, m = 0, t = tF_exp_x1_x2, model = "normal", xmodel = "unif")
#   sim2 <- simulate.dgp(object = dgp2, nsim = i, dim = p, nsimtest = 1000)
#   x_sim2 <- data.matrix(frame = sim2[, c(1:p)])
#   Y_sim2 <- sim2$y
#   W_sim2 <- as.numeric(levels(sim2$trt))[sim2$trt]
#
#   test2 <- attr(sim2, "testxdf")
#   test2$trt <- NULL
#
#   cf2 <- causal_forest(X = x_sim2, Y = Y_sim2, W = W_sim2)
#   tree2 <- get_tree(forest = cf2, index = 1)
#   lin2 <- lm(formula = y ~ (trt + X1 + X2 + X3 + X4)^2, data = sim2)
#
#   tau2_hat_cf <- predict(object = cf2, newdata = test2)
#   tau2_hat_cf <- as.numeric(tau2_hat_cf$predictions)
#
#   test_untr<- test_tr <- test2
#   test_untr$trt <- "0"
#   test_tr$trt <- "1"
#   tau2_hat_lin <- predict(lin2, test_tr) - predict(lin2, test_untr)
#
#   tau2 <- predict(object = dgp2, newdata = test2)[, "tfct"]
#
#   mse_cf_obs2[i] <- mean((tau2 - tau2_hat_cf)^2)
#   mse_lin_obs2[i] <- mean((tau2 - tau2_hat_lin)^2)
# }

# #####
#
# mse_cf_obs3 <- rep(NA, times = 500)
# mse_lin_obs3 <- rep(NA, times = 500)
# for (i in 10:500) {
#   dgp3 <- dgp(p = 0.5, m = 0, t = tF_exp2_x1_x2, model = "normal", xmodel = "unif")
#   sim3 <- simulate.dgp(object = dgp3, nsim = i, dim = p, nsimtest = 1000)
#   x_sim3 <- data.matrix(frame = sim3[, c(1:4)])
#   Y_sim3 <- sim3$y
#   W_sim3 <- as.numeric(levels(sim3$trt))[sim3$trt]
#
#   test3 <- attr(sim3, "testxdf")
#   test3$trt <- NULL
#
#   cf3 <- causal_forest(X = x_sim3, Y = Y_sim3, W = W_sim3)
#   tree3 <- get_tree(forest = cf3, index = 1)
#   lin3 <- lm(formula = y ~ (trt + X1 + X2 + X3 + X4)^2, data = sim3)

```

```

#
# tau3_hat_cf <- predict(object = cf3, newdata = test3)
# tau3_hat_cf <- as.numeric(tau3_hat_cf$predictions)
#
# test_untr<- test_tr <- test3
# test_untr$trt <- "0"
# test_tr$trt <- "1"
# tau3_hat_lin <- predict(lin3, test_tr) - predict(lin3, test_untr)
#
# tau3 <- predict(object = dgp3, newdata = test3)[, "tfct"]
#
# mse_cf_obs3[i] <- mean((tau3 - tau3_hat_cf)^2)
# mse_lin_obs3[i] <- mean((tau3 - tau3_hat_lin)^2)
# }
# mse_cf_obs1 <- mse_cf_obs1[-(1:9)]
# mse_cf_obs2 <- mse_cf_obs2[-(1:9)]
# mse_cf_obs3 <- mse_cf_obs3[-(1:9)]
# mse_lin_obs1 <- mse_lin_obs1[-(1:9)]
# mse_lin_obs2 <- mse_lin_obs2[-(1:9)]
# mse_lin_obs3 <- mse_lin_obs3[-(1:9)]
# ntrain <- 10:500
# mse_obs <- data.frame(mse_cf_obs1, mse_cf_obs2, mse_cf_obs3, mse_lin_obs1,
#                         mse_lin_obs2, mse_lin_obs3, ntrain)
# saveRDS(mse_obs, file = "./Data/mse_obs.rds")
mse_obs <- readRDS("./Data/mse_obs.rds")

plot(mse_obs$ntrain, mse_obs$mse_cf_obs1, type = "l", xlab = "Number of Observations",
      ylab = "MSE")
lines(mse_obs$ntrain, mse_obs$mse_lin_obs1, col = 2)
legend("topright", legend = c("CF", "Lin"), lty = c(1,1), col = c(1,2))

plot(mse_obs$ntrain, mse_obs$mse_cf_obs2, type = "l", xlab = "Number of Observations",
      ylab = "MSE")
lines(mse_obs$ntrain, mse_obs$mse_lin_obs2, col = 2)
legend("topright", legend = c("CF", "Lin"), lty = c(1,1), col = c(1,2))

plot(mse_obs$ntrain, mse_obs$mse_cf_obs3, type = "l", xlab = "Number of Observations",
      ylab = "MSE")
lines(mse_obs$ntrain, mse_obs$mse_lin_obs3, col = 2)
legend("topright", legend = c("CF", "Lin"), lty = c(1,1), col = c(1,2))

```

Next we want see how the methods perform in a large number of simulations.

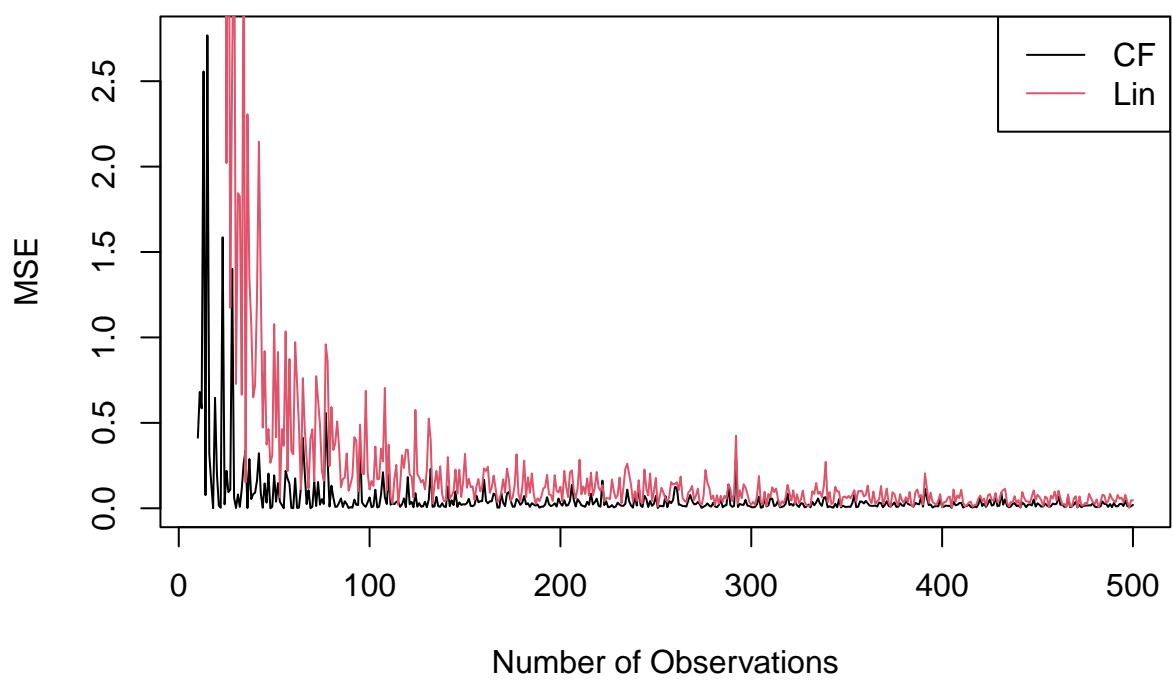


Figure 7: Computed MSE for Different Number of Training Observations (Setup 1)

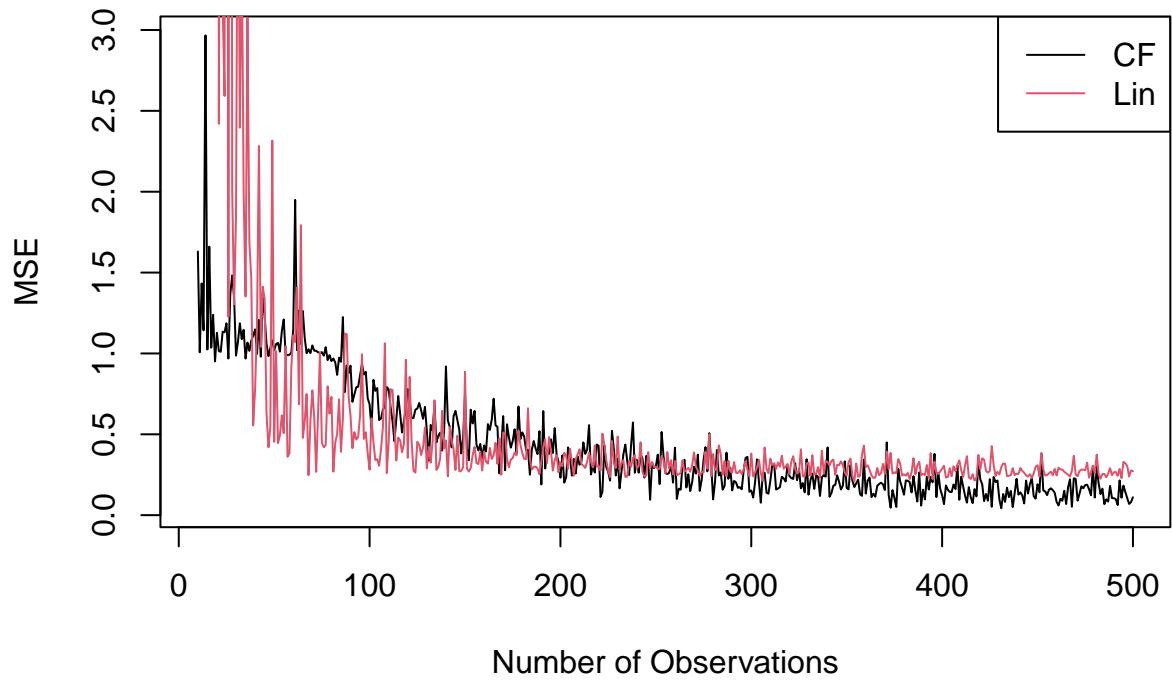


Figure 8: Computed MSE for Different Number of Training Observations (Setup 2)

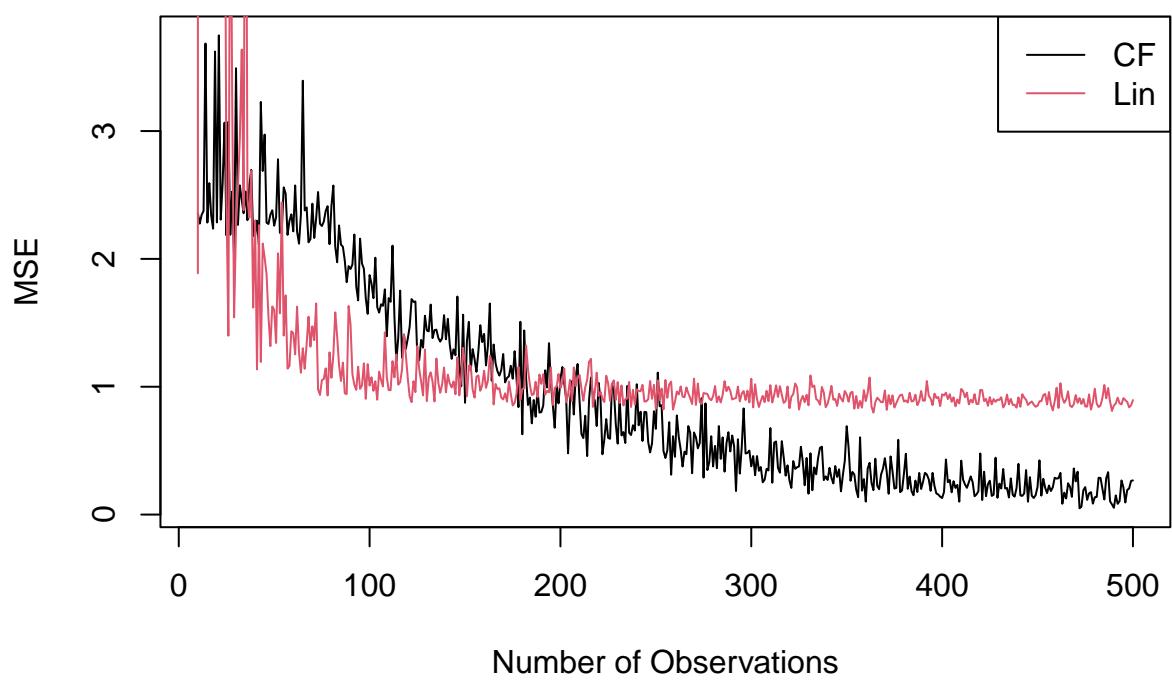


Figure 9: Computed MSE for Different Number of Training Observations (Setup 3)

3.2 Simulation

The structure of the simulation study follows Wager and Athey (2018) and Dandl et al. (2022). For the simulation study we compute 1000 simulations of each of the setups described above, thus we generate 3000 simulations in total. The number of observations in the test and training sample is 500 and 1000 respectively. The estimation is conducted twice (by the two functions below) once with the causal forest and once with the linear interaction model. Because we set the same seed for both data generating processes, both methods are computed and evaluated with the same data.

To evaluate the two models the MSE is computed for every simulation. Causal forests are grown with 2000 trees. In total, we obtain a dataset with 6000 observation that contains the parameters of the models as well as the MSE for each simulation. Again depending on the available computing power this simulation can take some time. Therefore the code is commented out and the produced dataset is available as a file in the data folder.

```
# n_small <- 1000
# n_large <- as.integer(3 * n_small)
# n_p <- 4L #Also change formula for lm at line 347
# obs_test <- 1000L
# Setup <- c(rep(x = "1", times = n_small), rep(x = "2", times = n_small),
#           rep(x = "3", times = n_small))
# setup <- factor(x = setup, levels = c(1, 2, 3),
#                   labels = c("Setup 1", "Setup 2", "Setup 3"))
# p <- c(rep(x = "pF_x1", times = n_small), rep(x = 0.5, times = n_small),
#        rep(x = 0.5, times = n_small))
# m <- c(rep(x = "mF_x1", times = n_small), rep(x = 0, times = n_small),
#        rep(x = 0, times = n_small))
# t <- c(rep(x = 0, times = n_small), rep(x = "tF_exp_x1_x2", times = n_small),
#        rep(x = "tF_exp2_x1_x2", times = n_small))
# sd <- rep(x = 1, times = n_large)
# ol <- rep(x = 0, times = n_large)
# model <- rep(x = "normal", times = n_large)
# xmodel <- rep(x = "unif", times = n_large)
# nsim <- rep(x = 500, times = n_large)
# dim <- rep(x = n_p, times = n_large)
# nsimtest <- rep(x = obs_test, times = n_large)
# method <- rep(x = 1, times = n_large)
# method <- factor(x = method, levels = 1, labels = "CF")
# id <- as.factor(c(rep(x = 1, times = n_small), rep(x = 2, times = n_small),
#                    rep(x = 3, times = n_small)))
# id <- factor(id, levels = c(1, 2, 3), labels = c("CF/1", "CF/2", "CF/3"))
# #sim_cf <- data.frame(setup, p, m, t, sd, ol, model, xmodel, nsim, dim, nsimtest,
# #                         method, id)
#
# set.seed(123)
# run_cf <- function(i) {
#   row <- sim_cf[1, ]
#   dgpA <- do.call(what = dgp,
#                   args = as.list(row)[c("p", "m", "t", "sd", "ol", "model",
#                                         "xmodel")])
#   simA <- do.call(what = simulate,
#                   args = c(list(object = dgpA,
#                                 as.list(row)[c("nsim", "dim", "seed", "nsimtest")])))
#   x_simA <- data.matrix(frame = simA[, c(1:n_p)])
#   Y_simA <- simA$y
}
```

```

# W_simA <- as.numeric(levels(simA$trt))[simA$trt]
#
# testA <- attr(simA, "testxdf")
# testA$trt <- NULL
#
# cfa <- causal_forest(X = x_simA, Y = Y_simA, W = W_simA)
#
# tauA_hat_cf <- predict(object = cfa, newdata = testA)
# tauA_hat_cf <- as.numeric(tauA_hat_cf$predictions)
#
# tauA <- predict(object = dgpA, newdata = testA)[, "tfct"]
#
# mse_cf <- mean((tauA - tauA_hat_cf)^2)
# return(mse_cf)
# }
# mse_cf <- sapply(X = 1:n_large, FUN = run_cf)
# sim_cf$mse <- mse_cf
#
# method <- rep(x = 2, times = n_large)
# method <- factor(x = method, levels = 2, labels = "Lin")
# id <- c(rep(x = 4, times = n_small), rep(x = 5, times = n_small),
#         rep(x = 6, times = n_small))
# id <- factor(id, levels = c(4, 5, 6), labels = c("Lin/1", "Lin/2", "Lin/3"))
# sim_lin <- data.frame(setup, p, m, t, sd, ol, model, xmodel, nsim, dim,
#                         nsimtest, method, id)
#
# set.seed(123)
# run_lin <- function(i) {
#   row <- sim_lin[1, ]
#   dgpA <- do.call(what = dgp,
#                   args = as.list(row)[c("p", "m", "t", "sd", "ol", "model",
#                                         "xmodel")])
#   simA <- do.call(what = simulate,
#                   args = c(list(object = dgpA),
#                           as.list(row)[c("nsim", "dim", "nsimtest")]))
#
#   testA <- attr(simA, "testxdf")
#   testA$trt <- NULL
#
#   linA <- lm(formula = y ~ (trt + X1 + X2 + X3 + X4)^2, data = simA)
#
#   test_untr<- test_tr <- testA
#   test_untr$trt <- "0"
#   test_tr$trt <- "1"
#   tauA_hat_lin <- predict(linA, test_tr) - predict(linA, test_untr)
#
#   tauA <- predict(object = dgpA, newdata = testA)[, "tfct"]
#
#   mse_lin <- mean((tauA - tauA_hat_lin)^2)
#   return(mse_lin)
# }
# mse_lin <- sapply(X = 1:n_large, FUN = run_lin)
# sim_lin$mse <- mse_lin

```

```

# 
# sim <- rbind(sim_cf, sim_lin)
# saveRDS(sim, file = "./Data/sim.rds")
sim <- readRDS(file = "./Data/sim.rds")

```

3.2.1 Results

```

plot10 <- bwplot(x = mse ~ method | setup, data = sim, par.settings=bwtheme)
plot10

```

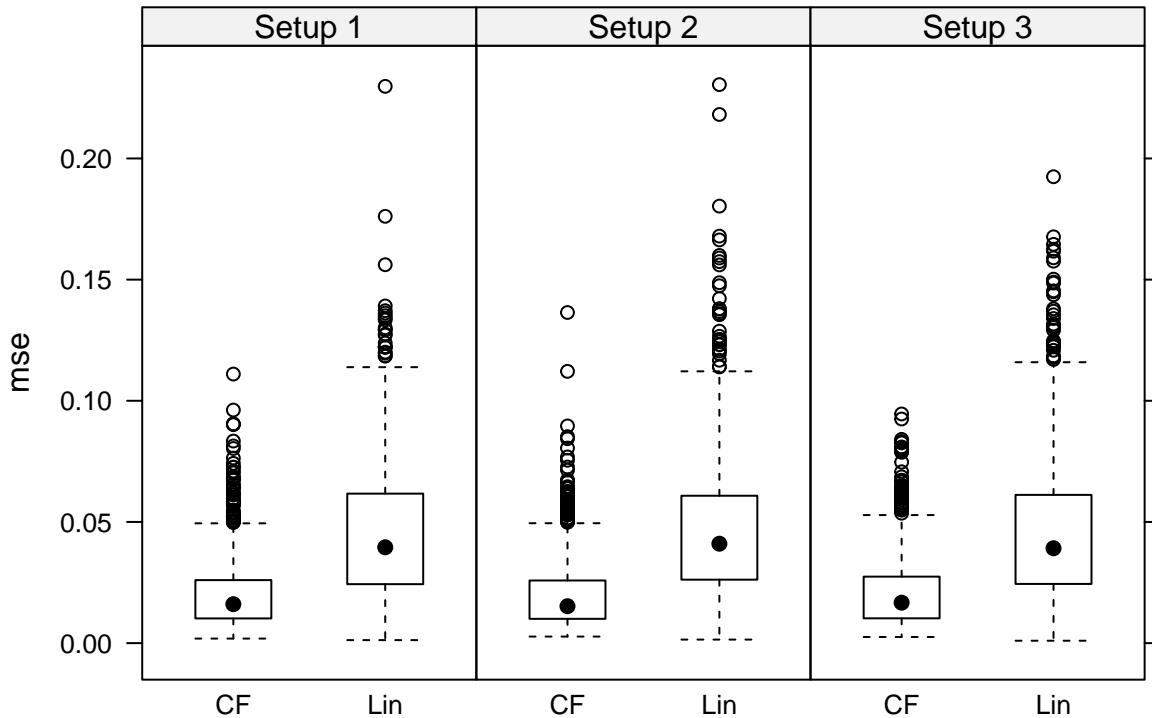


Figure 10: MSE of the Two Methods for All Setups

```

msetab <- aggregate(x = mse ~ id, data = sim, FUN = mean)
knitr::kable(msetab, caption = "Average MSE of the Two Methods for All Setups",
             col.names = c("Method/Setup", "MSE"))

```

Figure 10 and Table 10 describe the average MSE for both methods and all setups. The histogram and the calculated MSE confirms what we already saw in the section before: The estimations of the causal forest are closer to the true HTE than the estimations by the linear interaction model. This result holds for all three setups.

Table 10: Average MSE of the Two Methods for All Setups

Method/Setup	MSE
CF/1	0.0206419
CF/2	0.0203078
CF/3	0.0212736
Lin/1	0.0455275
Lin/2	0.0465423
Lin/3	0.0460807

4 Empirical Application

To apply and compare the two methods in an empirical setting we consider the study from Gazeaud et al. (2021b) where they investigate the link between cash transfers and international migration. The authors conducted an RCT in Comoros where randomly selected households received up to US\$320 as part of a cash-for-work program. The data is from Gazeaud et al. (2021c). The most important variables are described in Table 11. Summary statistics are reported in Table 12 and 13.

Table 11: Description of Variables

Variable Name	Description
migmhr_dum	Migration to Mayotte
treated	Treatment indicator
bl_network_mayotte_hh	Migration network
bl_mig_willingness	Willing to migrate
bl_working_age_adult_n	Working age adults (N)
bl_ihsconsopea	Consumption
bl_schooling	Schooling
bl_treated_tot_cont	CFW rounds (N)

```

migration_prepared <- read.dta13(file = "./Data/migration_prepared.dta",
                                    generate.factors = TRUE, convert.factors = TRUE)
migration_prepared <- migration_prepared[migration_prepared$attrition != 1, ]

migration_prepared <- migration_prepared[, c("bl_network_mayotte_hh",
                                              "bl_mig_willingness",
                                              "bl_working_age_adult_n",
                                              "bl_ihsconsopea", "bl_schooling",
                                              "bl_treated_tot_cont",
                                              "treated", "migmhr_dum")]

X_mig <- data.matrix(migration_prepared[, -c(7, 8)])
Y_mig <- migration_prepared$migmhr_dum
W_mig <- migration_prepared$treated

lm1_mig <- lm(formula = migmhr_dum ~ (treated + bl_network_mayotte_hh +
                                         bl_mig_willingness +
                                         bl_working_age_adult_n +
                                         bl_ihsconsopea + bl_schooling +
                                         bl_treated_tot_cont)^2, data = migration_prepared)

test_tr_mig <- test_untr_mig <- migration_prepared[, -8]
test_tr_mig$treated <- 0
test_untr_mig$treated <- 1

tau_hat_lin_mig <- predict(lm1_mig,
                            newdata = test_tr_mig) - predict(lm1_mig,
                                                               newdata = test_untr_mig)

cf_mig <- causal_forest(X = X_mig, Y = Y_mig, W = W_mig)
tree_mig <- get_tree(forest = cf_mig, index = 1)

tau_hat_cf_mig <- predict(object = cf_mig, newdata = X_mig)
tau_hat_cf_mig <- as.numeric(tau_hat_cf_mig$predictions)

```

Table 12: Summary Statistics for Numerical Variables

	Unique (#)	Missing (%)	Mean	SD	Min	Median	Max
bl_network_mayotte_hh	2	0	0.15	0.35	0.00	0.00	1.00
bl_working_age_adult_n	15	0	3.29	1.89	0.00	3.00	14.00
bl_ihsconopea	2177	0	7.15	0.99	0.00	7.19	10.77
bl_schooling	4	0	0.69	0.92	0.00	0.00	3.00
bl_treated_tot_cont	5	0	4.73	1.35	3.00	5.00	7.00
treated	2	0	0.61	0.49	0.00	1.00	1.00
migmhr_dum	2	0	0.10	0.29	0.00	0.00	1.00

Table 13: Summary Statistics for Categorical Variables

bl_mig_willingness	N	%
Ne sait pas	0	0.00
Ne souhaite pas repondre	0	0.00
Non	1703	78.08
Oui	478	21.92

```

res_mig <- migration_prepared
res_mig$tau_hat_cf_mig <- tau_hat_cf_mig
res_mig$tau_hat_lin_mig <- tau_hat_lin_mig

datasummary_skim(data = migration_prepared, output = "kableExtra",
                  histogram = FALSE, fmt = 2,
                  title = "Summary Statistics for Numerical Variables",
                  type = "numeric")

datasummary_skim(data = migration_prepared, output = "kableExtra",
                  histogram = FALSE, fmt = 2,
                  title = "Summary Statistics for Categorical Variables",
                  type = "categorical")

```

In Table A14 of Gazeaud et al. (2021a) the author estimate the heterogeneous treatment effects. No coefficient is significant, but as the authors notes it seems that the effect is stronger for people who are willing to migrate and receive more cash-for-work funds.

In addition to this table a linear model with interactions of all explanatory variables in Table 11 is estimated. Results are reported in Table 14 and support the prediction of the author for the most important variables.

```

modelsummary(lm1_mig, vcov = "HC3", coef omit = "^(?!treated)",
             gof omit = "BIC|AIC|RMSE|Log.Lik.", output = "kableExtra",
             stars = TRUE,
             title = "Regression Table With Interactions of the Variables in Table 11")

```

The plotted histograms and boxplots of the estimated HTE for both methods, seen in Figure 11 and 12, show that the causal forest predicts overall positive HTE but they are close to zero. The linear interaction model predicts overall negative HTE ranging from -0.08 to 0.

```

plot11 <- histogram(tau_hat_cf_mig, type = "count", xlab = "Estimated HTE",
                     main = "Causal Forest", par.settings=bwtheme)
plot12 <- histogram(tau_hat_lin_mig, type = "count", xlab = "Estimated HTE",

```

Table 14: Regression Table With Interactions of the Variables in Table 11

	Model 1
treated	-0.056 (0.112)
treated × bl_network_mayotte_hh	-0.004 (0.047)
treated × bl_mig_willingnessOui	0.034 (0.034)
treated × bl_working_age_adult_n	0.000 (0.008)
treated × bl_ihsconsopea	0.002 (0.012)
treated × bl_schooling	-0.002 (0.013)
treated × bl_treated_tot_cont	0.013 (0.009)
Num.Obs.	2181
R2	0.119
R2 Adj.	0.107
F	5.487
Std.Errors	HC3

+ p < 0.1, * p < 0.05, ** p < 0.01, *** p < 0.001

```
main = "Linear Interaction Model", par.settings=bwtheme)
grid.arrange(plot11, plot12, ncol = 2)

par(mfrow = c(1, 2))
boxplot(tau_hat_cf_mig, ylab = "Estimated HTE", main = "Causal Forest")
boxplot(tau_hat_lin_mig, ylab = "Estimated HTE", main = "Linear Interaction Model")
```

Plotting One Causal Tree

```
tree_mig_plot <- plot(tree_mig)
cat(DiagrammeRsvg::export_svg(tree_mig_plot), file = './Figures/tree_mig.svg')
rsvg_pdf(svg = './Figures/tree_mig.svg', file = './Figures/tree_mig.pdf')
```

Looking at the variable importance estimated by the causal forest we can see that contrary to the linear model in the paper, the most important variables are consumption, the number of working age adults, and additional cash-for-work funds.

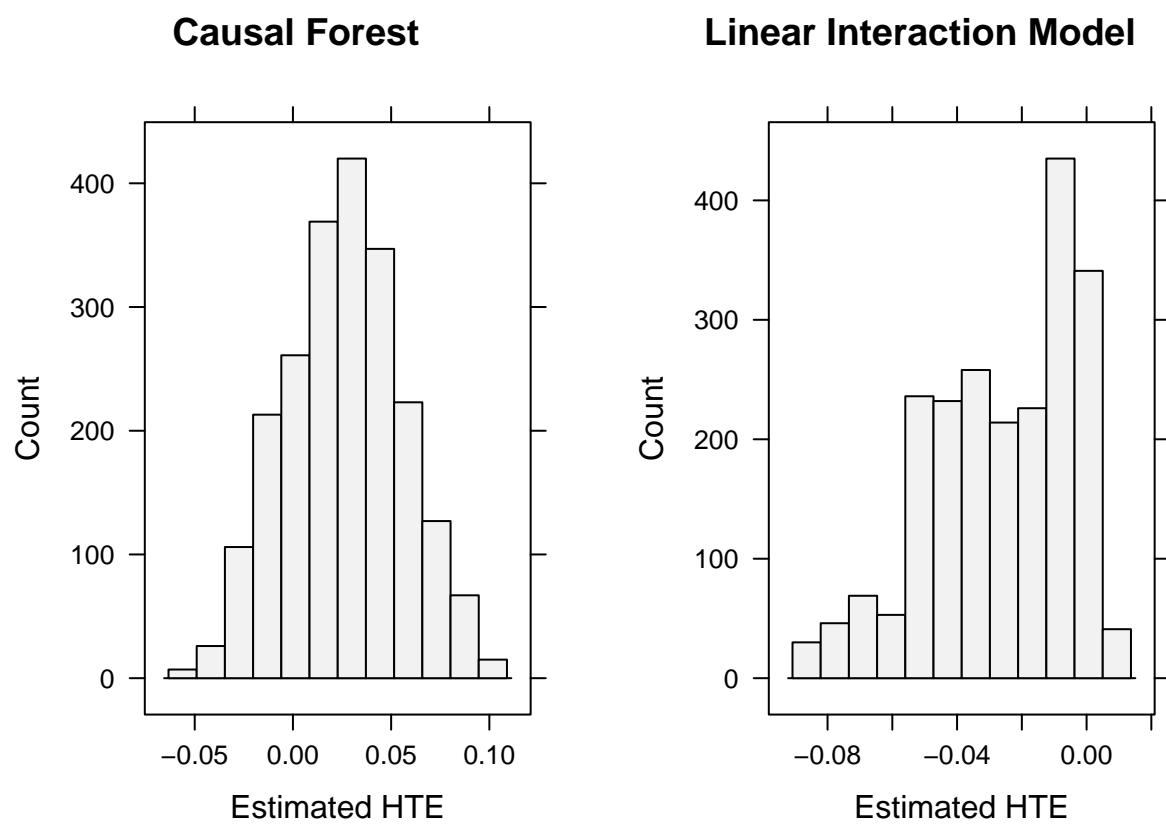


Figure 11: Histogram of the Estimated HTEs for Both Methods

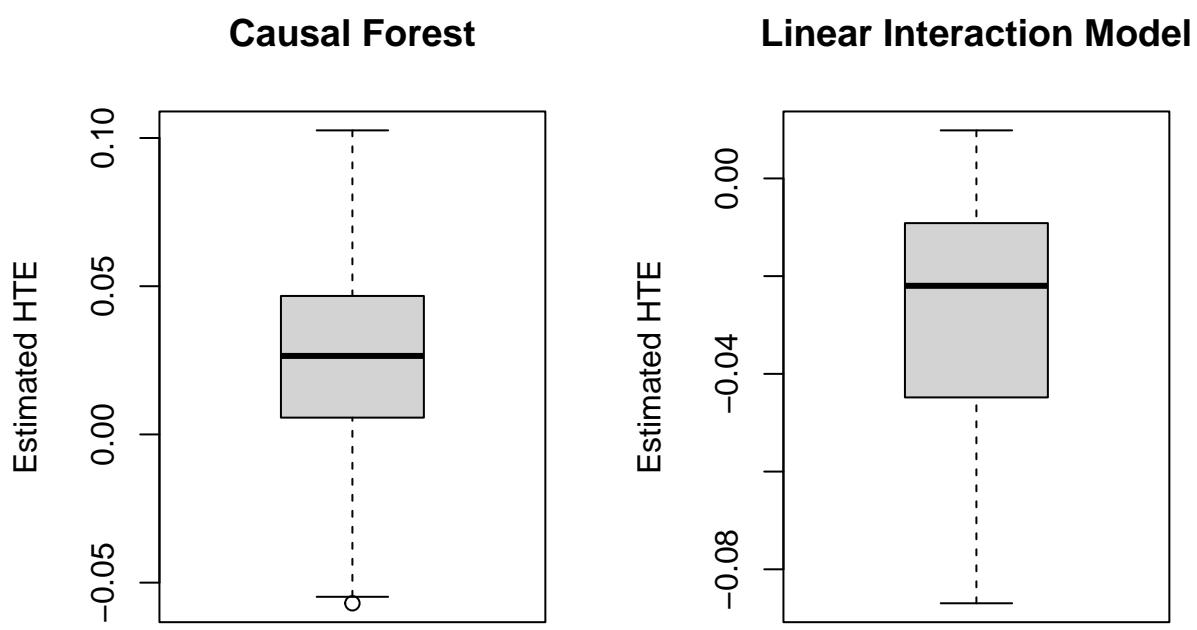


Figure 12: Boxplot of the Estimated HTE for both methods

```
variable_importance(forest = cf_mig)
```

```
## [1] 0.05959137
## [2,] 0.04772656
## [3,] 0.21908530
## [4,] 0.44134217
## [5,] 0.04846725
## [6,] 0.18378735
```

Calculating the estimated ATE of the causal forest confirms what we saw in the histogram above that the ATE of the causal forest is positive but close to zero and the ATE of the linear interaction model is negative and close to zero.

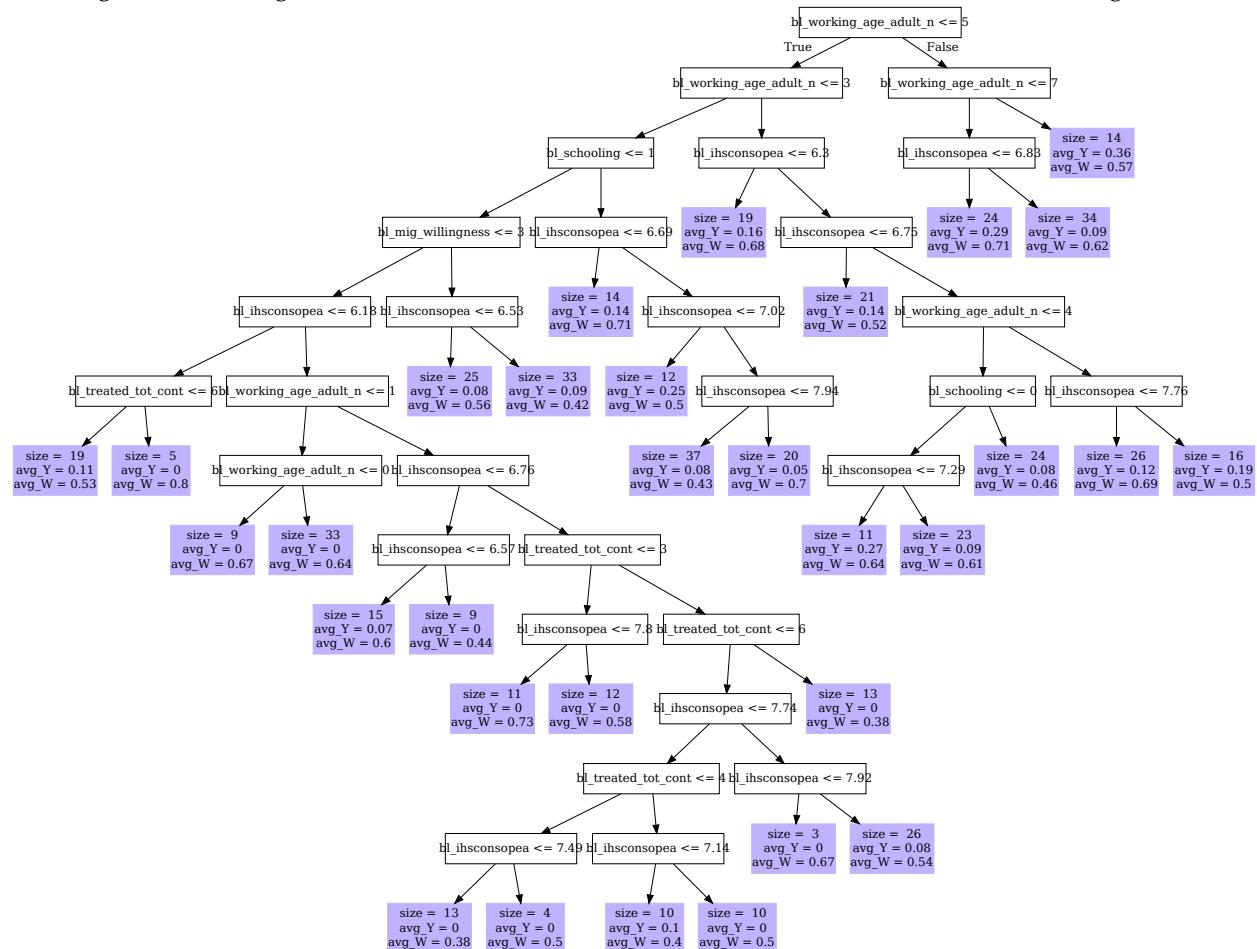
```
mean(tau_hat_cf_mig)
```

```
## [1] 0.02590356
```

```
mean(tau_hat_lin_mig)
```

```
## [1] -0.02625057
```

Figure 13: This figure shows one tree from the causal forest. In total, 2000 such trees are grown.



5 Conclusion

Throughout this analysis it became clear that a causal forest outperforms the linear interaction model when the interest lies in HTE. Three data generating processes were considered that resemble observational studies and RCTs in the real world. For every data generating process the causal forest delivered more accurate predictions of the true treatment effect. For the empirical application we found conflicting results. Overall the HTEs estimated by the causal forest are positive, while the HTEs estimated by the linear interaction model are negative. Contrary to the author the most important variables are not the willingness to migrate and more cash-for-work funds, but rather consumption and the number of working age adults with additional cash-for-work funds only at the third place.

References

- AEA RCT Registry. (2022). *Registrations in the AEA RCT Registry* (Version V1). Harvard Dataverse. <https://doi.org/10.7910/DVN/IHSCI6>
- Cameron, D. B., Mishra, A., & Brown, A. N. (2016). The Growth Of Impact Evaluation For International Development: How Much Have We Learned? *Journal of Development Effectiveness*, 8(1), 1–21. <https://doi.org/10.1080/19439342.2015.1034156>
- Dandl, S., Hothorn, T., Seibold, H., Sverdrup, E., Wager, S., & Zeileis, A. (2022). What Makes Forest-Based Heterogeneous Treatment Effect Estimators Work? *Working Paper*. <https://doi.org/10.48550/arXiv.2206.10323>
- Duflo, E., Glennerster, R., & Kremer, M. (2007). Using Randomization in Development Economics Research: A Toolkit. In T. P. Schultz & J. A. Strauss (Eds.), *Handbook of Development Economics* (pp. 3895–3962). Elsevier. [https://doi.org/10.1016/S1573-4471\(07\)04061-2](https://doi.org/10.1016/S1573-4471(07)04061-2)
- Gazeaud, J., Mvukiyehe, E., & Sterck, O. (2021a). Cash Transfers and Migration: Theory and Evidence from a Randomized Controlled Trial. *Working Paper*. <https://hal.archives-ouvertes.fr/hal-03512282>
- Gazeaud, J., Mvukiyehe, E., & Sterck, O. (2021b). Cash Transfers and Migration: Theory and Evidence from a Randomized Controlled Trial. *The Review of Economics and Statistics*, Forthcoming, 1–45. https://doi.org/10.1162/rest_a_01041
- Gazeaud, J., Mvukiyehe, E., & Sterck, O. (2021c). *Replication Data for: Cash Transfers and Migration: Theory and Evidence from a Randomized Controlled Trial* (Version V1). Harvard Dataverse. <https://doi.org/10.7910/DVN/DDVBFE>
- Holland, P. W. (1986). Statistics and Causal Inference. *Journal of the American Statistical Association*, 81(396), 945–960. <https://doi.org/10.2307/2289064>
- R Core Team. (2022). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing. Vienna, Austria. <https://www.R-project.org/>
- Rubin, D. B. (1974). Estimating Causal Effects Of Treatments In Randomized And Nonrandomized Studies. *Journal of Educational Psychology*, 66(5), 688–701. <https://doi.org/10.1037/h0037350>
- Rubin, D. B. (1977). Assignment To Treatment Group On The Basis Of A Covariate. *Journal of Educational Statistics*, 2(1), 1–26. <https://doi.org/10.3102/10769986002001001>
- Tibshirani, J., Athey, S., Sverdrup, E., & Wager, S. (2021). *htesim: Conducting Extensive Simulation Studies of Heterogeneous Treatment Effect Estimation* [R package version 0.0.9000]. <https://github.com/dandls/htesim>
- Wager, S., & Athey, S. (2018). Estimation and Inference of Heterogeneous Treatment Effects using Random Forests. *Journal of the American Statistical Association*, 113(523), 1228–1242. <https://doi.org/10.1080/01621459.2017.1319839>