Simulation Study in Thesis

Juho Lahteenmaa

27 April 2020

Contents

1 Abstract **2**

2 Functions **2**

2.1 Randomized Controlled Trial with Constant Treatment Effect 2

2.2 Constant Treatment Effect with Unconfounded Assignment, Including Covariates Affecting to Output 5

2.3 Constant Treatment Effect with Confounded Assignment, Including Covariates Affecting to Output 5

2.4 Constant Treatment Effect with Confounded Assignment, Including Covariates Affecting to Output and a Pure Local M-Structure 7

2.5 Constant Treatment Effect with Confounded Assignment, Including Covariates Affecting to Output and a Impure Local M-Structure 9

2.6 Randomized Controlled Trial with Heterogeneous Treatment Effect, Including Covariates Affecting to Output 10

2.7 Heterogeneous Treatment Effect with the Confounded Assignment, Including Covariates Affecting to Output 11

2.8 Heterogeneous Treatment Effect with Confounded Assignment, Including Covariates Affecting to Output and a Pure Local M-Structure 13

2.9 Heterogeneous Treatment Effect with Confounded Assignment, Including Covariates Affecting to Output and a Impure Local M-Structure 13

3 Causal Forest **19**

3.1 Randomized Controlled Trial with Constant Treatment Effect 20

3.2 Constant Treatment Effect with Unconfounded Assignment, Including Covariates Affecting to Output 29

3.3 Constant Treatment Effect with Confounded Assignment, Including Covariates Affecting to Output 38

3.4 Constant Treatment Effect with Confounded Assignment, Including Covariates Affecting to Output and a Pure Local M-Structure 53

3.5 Constant Treatment Effect with Confounded Assignment, Including Covariates Affecting to Output and a Impure Local M-Structure 63

3.6 Randomized Controlled Trial with Heterogeneous Treatment Effect, Including Covariates Affecting to Output 72

3.7 Heterogeneous Treatment Effect with the Confounded Assignment, Including Covariates Affecting to Output 81

3.8 Heterogeneous Treatment Effect with Confounded Assignment, Including Covariates Affecting to Output and a Pure Local M-Structure 90

3.9 Heterogeneous Treatment Effect with Confounded Assignment, Including Covariates Affecting to Output and a Impure Local M-Structure 99

```
require(dplyr)
require(Rlab)
require(ggplot2)
require(plotly)
require(grf)
require(forcats)
require(tidyr)
require(gridExtra)

set.seed(101)

setwd("C:/Users/juhol/OneDrive/Documents/Tyokansio/Gradu/Koodit")
```

1 Abstract

Simulation study consists nine different SCMs (and the responding DAGs). Let's define the sample size to equal N = 10 000.

```
n <- 10000
```

In the first part of this report, the simulation functions will be defined.In the second part the *causal forest* algorithm will be ran.

2 Functions

2.1 Randomized Controlled Trial with Constant Treatment Effect

```
#New exogeneous parameters and variables

w_1 <- rbern(n, 1/2)

u_y <- rnorm(n, mean = 100, sd = 10)

tau <- 10

# Variables X_1, ..., X_10

X <- matrix(0, nrow = n, ncol = 10)

for (k in 1:10) {


  if(k%%2 != 0){    #if k is odd
```

4.1.2 Randomized Controlled Trial with Constant Treatment Effect, Including Covariates Affecting to Output

$$
\begin{array}{ccccc}
 & & U_Y & & \\
 & & \downarrow & & \\
U_W & \longrightarrow W \xrightarrow[\tau]{} & Y & & \\
 & & \uparrow & & \\
 & & X_1, \ldots, X_{10} & &
\end{array}
\tag{4.4}
$$

$$
f_W(U_W) = \text{Bern}\left(\frac{1}{2}\right) \tag{4.5}
$$

$$
f_Y(W, X_1, \ldots, X_{10}, U_Y) = \tau \cdot W + \sum_{k=1}^{10} \beta_{X_k} X_k + U_Y \tag{4.6}
$$

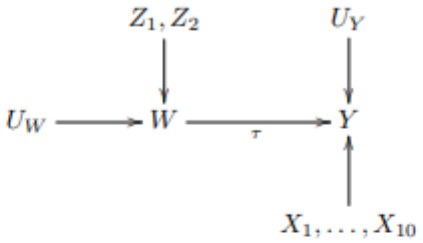
, where

$$
\begin{cases}
U_Y & \sim \text{Norm}(\mu_Y(0), \sigma_Y^2) \\
X_K & \sim \text{Norm}(\mu_K, \sigma_K^2) \text{ when } k \text{ is odd and} \\
X_K & \sim \text{Bern}\left(\frac{1}{2}\right) \text{ when } k \text{ is even}
\end{cases}
$$

Figure 1: Simulation 1

```
    for (i in 1:n) {

      X[i, k] <- rnorm(1, mean = 0, sd = 1)
    }

  } else {         #if k is even

      for (i in 1:n) {

        X[i, k] <- rbern(1, 1/2)

    }

  }
}

beta_X_k <- runif(10, min = -3, max = 3)

#SCM

y_1 <- tau*w_1 + X%*%beta_X_k + u_y  #Output
```

4.1.3 Constant Treatment Effect with Unconfounded Assignment, Including Covariates Affecting to Output



(4.7)

$$f_W(Z_1, Z_2, U_W) = \frac{1}{1 + \exp\left\{-\left(\sum_{j=1}^{2} \gamma_{Z_j} Z_j + U_W\right)\right\}} \tag{4.8}$$

$$f_Y(W, X_1, \ldots, X_{10}, U_Y) = \tau \cdot W + \sum_{k=1}^{10} \beta_{X_k} X_k + U_Y \tag{4.9}$$

, where

$$\begin{cases} U_W & \sim \text{Norm}(0, \sigma_W^2) \\ U_Y & \sim \text{Norm}(\mu_Y(0), \sigma_Y^2) \\ Z_1, Z_2 & \sim \text{Bern}\left(\frac{1}{2}\right) \\ X_K & \sim \text{Norm}(\mu_K, \sigma_K^2) \text{ when } k \text{ is odd and} \\ X_K & \sim \text{Bern}\left(\frac{1}{2}\right) \text{ when } k \text{ is even} \end{cases}$$

Figure 2: Simulation 2

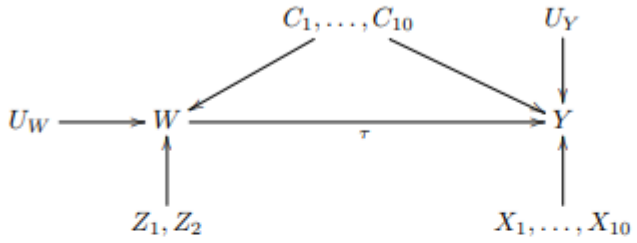
2.2 Constant Treatment Effect with Unconfounded Assignment, Including Covariates Affecting to Output

```
#New exogeneous parameters and variables

# Variables Z_1, Z_2

z_1 <- rbern(n, 1/2)

z_2 <- rbern(n, 1/2)

Z <- cbind(z_1, z_2)

u_w <- rnorm(n = n, mean = 0, sd = 1)

gamma_z <- runif(2, min = -1, max =  1)


#SCMs

w_2_propensity <- 1/(1 + exp(-(Z%*%gamma_z + u_w))) #Underlying propensity scores

w_2 <- rep.int(0, n)

for (i in 1:n) {                      # Treatment assignment
  w_2[i] <- rbern(1, w_2_propensity[i])
}

y_2 <-  tau*w_2 + X%*%beta_X_k + u_y  #Output
```

2.3 Constant Treatment Effect with Confounded Assignment, Including Covariates Affecting to Output

```
#New exogeneous parameters and variables

#Confounders C_1, ..., C_10

C <- matrix(0, nrow = n, ncol = 10)

for (l in 1:10) {


  if(l%%2 != 0){    #if k is odd

    for (i in 1:n) {

      C[i,l] <- rnorm(1, mean = 0, sd = 1)
    }

  } else {        #if k is even
```

4.1.4 Constant Treatment Effect with Confounded Assignment, Including Covariates Affecting to Output



(4.10)

$$f_W(Z_1, Z_2, C_1, \ldots, C_{10}, U_W) = \frac{1}{1 + \exp\left\{-\left(\sum_{j=1}^{2} \gamma_{Z_j} Z_j + \sum_{l=1}^{10} \gamma_{C_l} C_l + U_Y\right)\right\}} \tag{4.11}$$

$$f_Y(W, X_1, \ldots, X_{10}, C_1, \ldots, C_{10}, U_Y) = \tau \cdot W + \sum_{k=1}^{10} \beta_{X_k} X_k + \sum_{l=1}^{10} \beta_{C_l} C_l + U_Y \tag{4.12}$$

Figure 3: Simulation 3-1

, where

$$\begin{cases} U_Y & \sim \text{Norm}(\mu_Y(0), \sigma_Y^2) \\ U_W & \sim \text{Norm}(0, \sigma_W^2) \\ Z_1, Z_2 & \sim \text{Bern}\left(\frac{1}{2}\right) \\ X_K & \sim \text{Norm}(\mu_K, \sigma_K^2) \text{ when } k \text{ is odd and} \\ X_K & \sim \text{Bern}\left(\frac{1}{2}\right) \text{ when } k \text{ is even} \\ C_L & \sim \text{Norm}(\mu_L, \sigma_L^2) \text{ when } l \text{ is odd and} \\ C_L & \sim \text{Bern}\left(\frac{1}{2}\right) \text{ when } l \text{ is even} \end{cases}$$

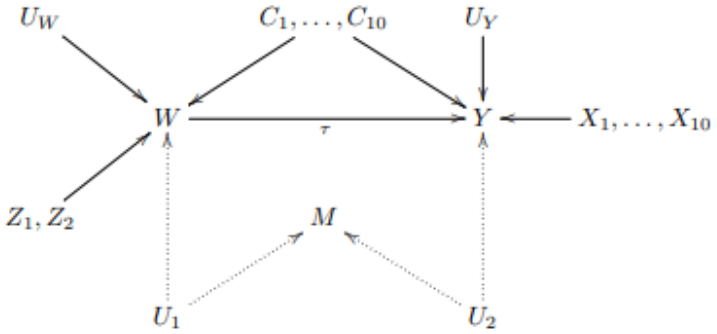
Figure 4: Simulation 3-2

```
      for (i in 1:n) {

        C[i, l] <- rbern(1, 1/2)

    }

  }
}


gamma_C_l <- runif(10, min = -0.5, max = 0.5)    #Coefficients for confounders for W

beta_C_l <- runif(10, min = -3, max = 3)    #Coefficients for confounders for Y


#SCMs

w_3_propensity <- 1/(1 + exp(-(Z%*%gamma_z + C%*%gamma_C_l + u_w))) #Underlying propensity scores

w_3 <- rep.int(0, n)

for (i in 1:n) {                        # Treatment assignment
  w_3[i] <- rbern(1, w_2_propensity[i])
}

y_3 <-  tau*w_3 + X%*%beta_X_k + C%*%beta_C_l + u_y  #Output
```

2.4 Constant Treatment Effect with Confounded Assignment, Including Covariates Affecting to Output and a Pure Local M-Structure

```
#New exogeneous parameters and variables

# Unobserved covariates U_1 and U_2
u_1 <- rbern(n, 1/2)
u_2 <- rbern(n, 1/2)

#Coefficients
gamma_U_1 <- 0.5
beta_U_2 <- 5

delta_1 <- 5
delta_2 <- 5

#SCMs

w_4_propensity <- 1/(1 + exp(-(Z%*%gamma_z + C%*%gamma_C_l + gamma_U_1*u_1 + u_w))) #Underlying propens

w_4 <- rep.int(0, n)

for (i in 1:n) {                        # Treatment assignment
  w_4[i] <- rbern(1, w_2_propensity[i])
```

4.1.6 Constant Treatment Effect with Confounded Assignment, Including Covariates Affecting to Output and a Pure Local M-Structure



(4.17)

$$f_W(Z_1, Z_2, C_1, \ldots, C_{10}, U_1, U_W) = \frac{1}{1 + \exp\left\{-\left(\sum_{j=1}^{2} \gamma_{Z_j} Z_j + \sum_{l=1}^{10} \gamma_{C_l} C_l + \gamma_{U_1} U_1 + U_Y\right)\right\}} \tag{4.18}$$

$$f_M = \delta_1 U_1 + \delta_2 U_2 \tag{4.19}$$

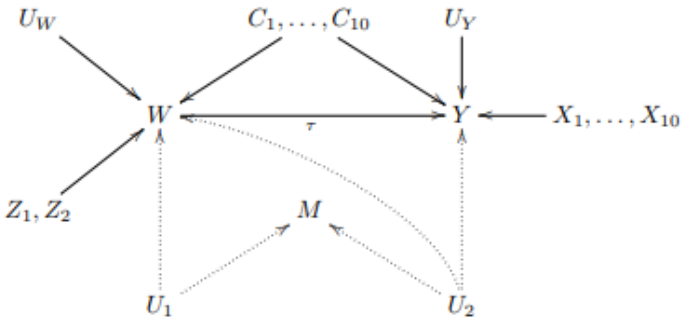
$$f_Y(W, X_1, \ldots, X_{10}, C_1, \ldots, C_{10}, I, U_2, U_Y) = \tau \cdot W + \sum_{k=1}^{10} \beta_{X_k} X_k + \sum_{l=1}^{10} \beta_{C_l} C_l + \beta_{U_2} U_2 + U_Y \tag{4.20}$$

, where

$$\begin{cases} U_Y & \sim \text{Norm}(\mu_Y(0), \sigma_Y^2) \\ U_W & \sim \text{Norm}(0, \sigma_W^2) \\ Z_1, Z_2 & \sim \text{Bern}\left(\frac{1}{2}\right) \\ X_K & \sim \text{Norm}(\mu_K, \sigma_K^2) \text{ when } k \text{ is odd and} \\ X_K & \sim \text{Bern}\left(\frac{1}{2}\right) \text{ when } k \text{ is even} \\ C_L & \sim \text{Norm}(\mu_L, \sigma_L^2) \text{ when } l \text{ is odd and} \\ C_L & \sim \text{Bern}\left(\frac{1}{2}\right) \text{ when } l \text{ is even} \\ U_1, U_2 & \sim \text{Bern}\left(\frac{1}{2}\right) \end{cases}$$

Figure 5: Simulation 4

4.1.7 Constant Treatment Effect with Confounded Assignment, Including Covariates Affecting to Output and a Impure Local M-Structure



(4.21)

$$f_W(Z_1, Z_2, C_1, \ldots, C_{10}, U_1, U_2, U_W) = \frac{1}{1 + \exp\left\{-\left(\sum_{j=1}^{2} \gamma_{Z_j} Z_j + \sum_{l=1}^{10} \gamma_{C_l} C_l + \sum_{m=1}^{2} \gamma_{U_m} U_m + U_Y\right)\right\}} \tag{4.22}$$

$$f_M = \delta_1 U_1 + \delta_2 U_2 \tag{4.23}$$

$$f_Y(W, X_1, \ldots, X_{10}, C_1, \ldots, C_{10}, U_2, U_Y) = \tau \cdot W + \sum_{k=1}^{10} \beta_{X_k} X_k + \sum_{l=1}^{10} \beta_{C_l} C_l + \beta_{U_2} U_2 + U_Y \tag{4.24}$$

, where

$$\begin{cases} U_Y & \sim \text{Norm}(\mu_Y(0), \sigma_Y^2) \\ U_W & \sim \text{Norm}(0, \sigma_W^2) \\ Z_1, Z_2 & \sim \text{Bern}\left(\frac{1}{2}\right) \\ X_K & \sim \text{Norm}(\mu_K, \sigma_K^2) \text{ when } k \text{ is odd and} \\ X_K & \sim \text{Bern}\left(\frac{1}{2}\right) \text{ when } k \text{ is even} \\ C_L & \sim \text{Norm}(\mu_L, \sigma_L^2) \text{ when } l \text{ is odd and} \\ C_L & \sim \text{Bern}\left(\frac{1}{2}\right) \text{ when } l \text{ is even} \\ U_1, U_2 & \sim \text{Bern}\left(\frac{1}{2}\right) \end{cases}$$

Figure 6: Simulation 5

```
}

m <- delta_1*u_1 + delta_2*u_2          # Observed collider M


y_4 <-  tau*w_4 + X%*%beta_X_k + C%*%beta_C_l + beta_U_2*u_2 + u_y  #Output
```

2.5 Constant Treatment Effect with Confounded Assignment, Including Covariates Affecting to Output and a Impure Local M-Structure

```
#New coefficient

gamma_U_2 <- 0.2
```

4.1.8 Randomized Controlled Trial with Heterogeneous Treatment Effect, Including Covariates Affecting to Output

$$U_Y \tag{4.25}$$

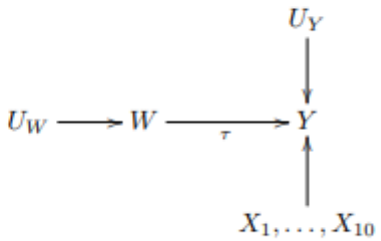


Figure 7: Simulation 6-1

$$f_W(U_W) = \text{Bern}\left(\frac{1}{2}\right) \tag{4.26}$$

$$f_Y(W, X_1, \ldots, X_{10}, U_Y) = \tau(x_1, x_2, x_3) \cdot W + \sum_{k=1}^{10} \beta_{X_k} X_k + U_Y \tag{4.27}$$

, where

$$\begin{cases} U_Y & \sim \text{Norm}(\mu_Y(0), \sigma_Y^2) \\ X_K & \sim \text{Norm}(\mu_K, \sigma_K^2) \text{ when } k \text{ is odd and} \\ X_K & \sim \text{Bern}\left(\frac{1}{2}\right) \text{ when } k \text{ is even} \end{cases}$$

and $\tau(x_1, x_2, x_3) = \alpha_0 + \alpha_1 x_1 + \alpha_2 x_2 x_3$.

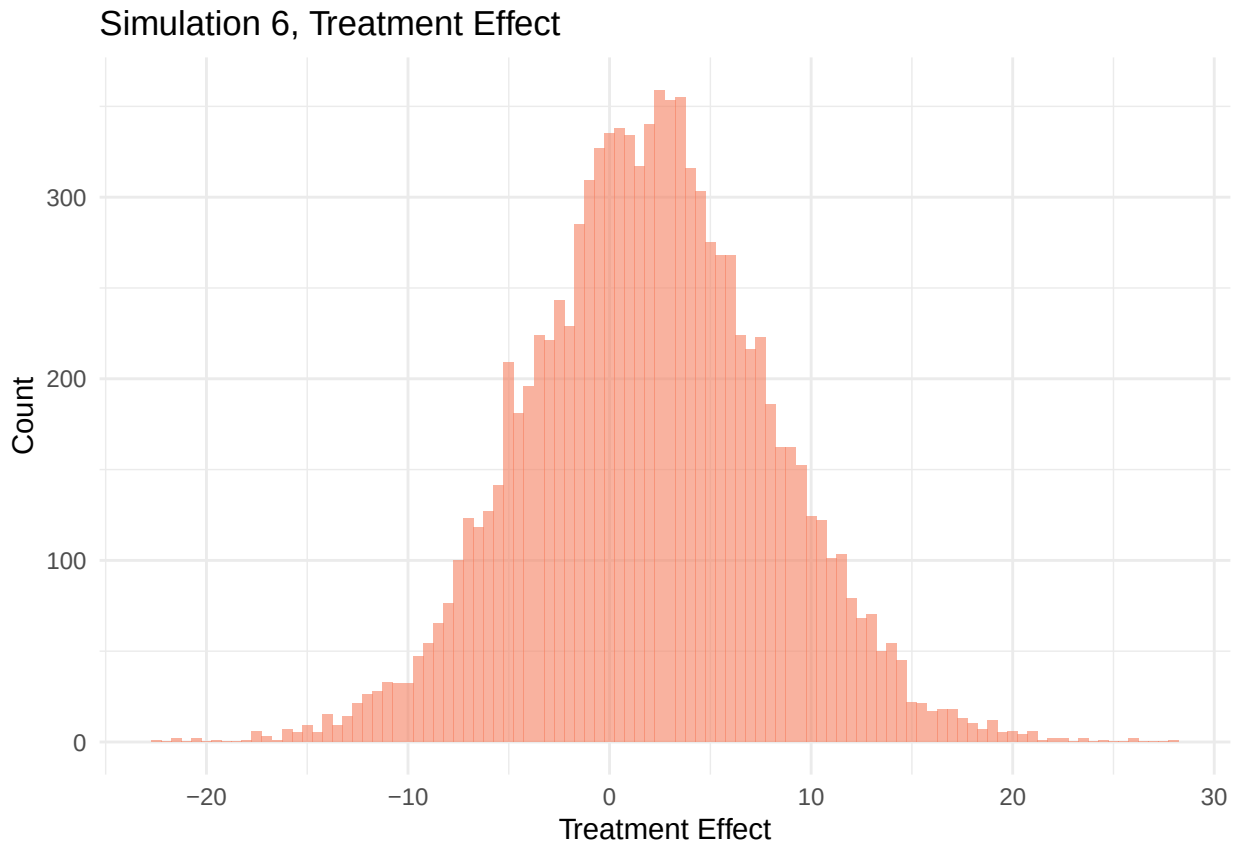
Figure 8: Simulation 6-2

```
#SCMs

w_5_propensity <- 1/(1 + exp(-(Z%*%gamma_z + C%*%gamma_C_l + gamma_U_1*u_1 + gamma_U_2*u_2 + u_w))) #Un

w_5 <- rep.int(0, n)

for (i in 1:n) {                        # Treatment assignment
  w_5[i] <- rbern(1, w_2_propensity[i])
}

y_5 <-  tau*w_5 + X%*%beta_X_k + C%*%beta_C_l + beta_U_2*u_2 + u_y      #Output
```

2.6 Randomized Controlled Trial with Heterogeneous Treatment Effect, Including Covariates Affecting to Output

```
#Heterogeneous treatment effect

#Alphas
```

```r
alp_0 <- 2
alp_1 <- 5
alp_2 <- -5

tau_1 <- rep.int(alp_0,n) + alp_1*X[,1] + alp_2*X[,2]*X[,3]  #Treatment effect

ggplot(data = as.data.frame(tau_1), aes(x = tau_1)) +
  geom_histogram(fill="#f68060", alpha=.6, binwidth = 0.5) +
  labs(title ="Simulation 6, Treatment Effect", x = "Treatment Effect", y = "Count") +
  theme_minimal()
```

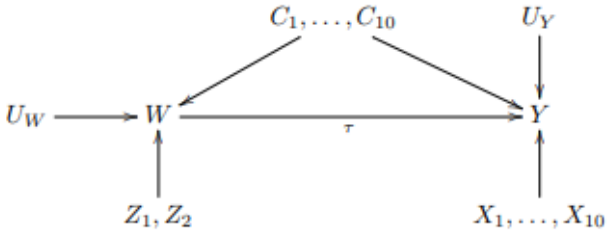



```r
#SCM

y_6 <- tau_1*w_1 + X%*%beta_X_k + u_y  #Output
```

2.7 Heterogeneous Treatment Effect with the Confounded Assignment, Including Covariates Affecting to Output

```r
#New alpha

alp_3 <- 3

tau_2 <- rep.int(alp_0,n) + alp_1*X[,1] + alp_2*X[,2]*X[,3] + alp_3*C[,1]   #Treatment effect
```

4.1.9 Heterogeneous Treatment Effect with Confounded Assignment, Including Covariates Affecting to Output



(4.28)

$$f_W(Z_1, Z_2, C_1, \ldots, C_{10}, U_W) = \frac{1}{1 + \exp\left\{-\left(\sum_{j=1}^{2} \gamma_{Z_j} Z_j + \sum_{l=1}^{10} \gamma_{C_l} C_l + U_Y\right)\right\}} \tag{4.29}$$

$$f_Y(W, X_1, \ldots, X_{10}, C_1, \ldots, C_{10}, U_Y) = \tau(x_1, x_2, x_3, c_1) \cdot W + \sum_{k=1}^{10} \beta_{X_k} X_k + \sum_{l=1}^{10} \beta_{C_l} C_l + U_Y \tag{4.30}$$

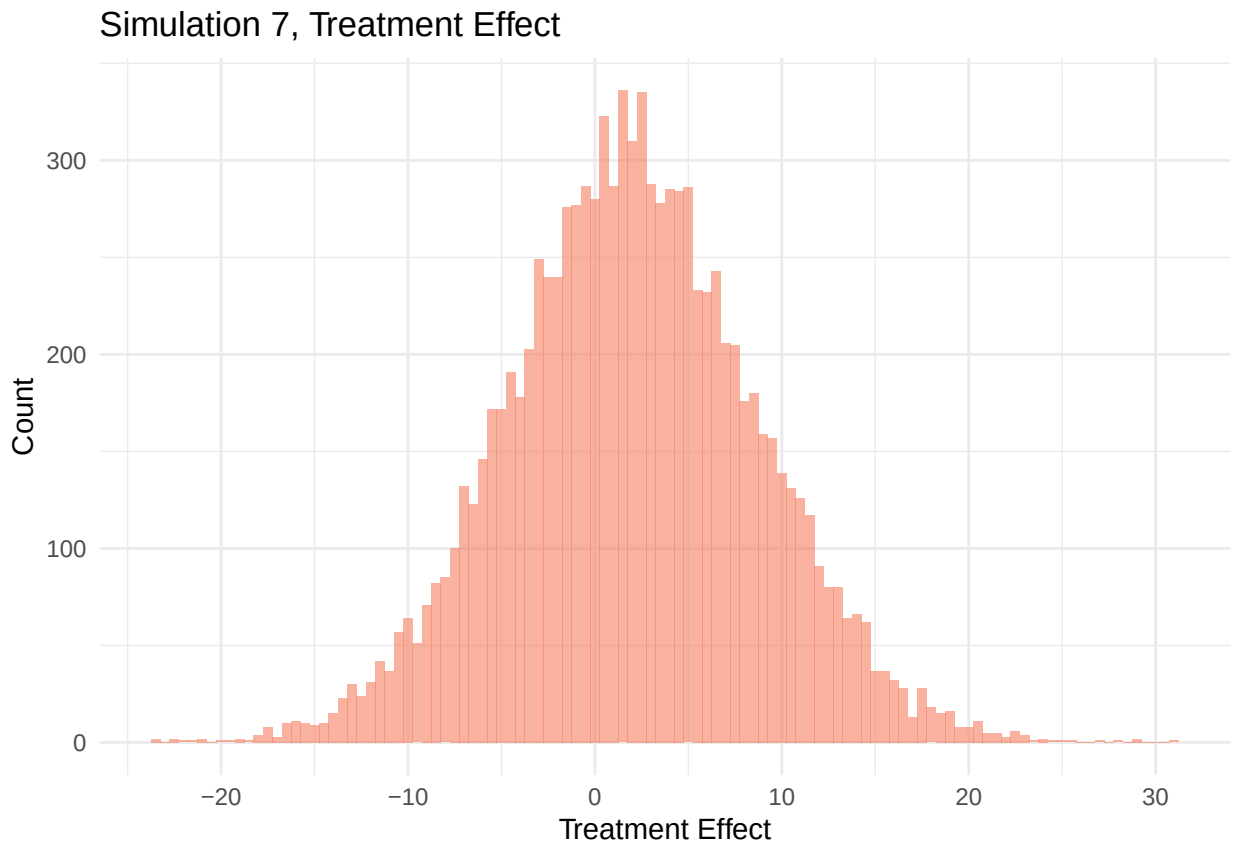
, where

$$\begin{cases} U_Y & \sim \text{Norm}(\mu_Y(0), \sigma_Y^2) \\ U_W & \sim \text{Norm}(0, \sigma_W^2) \\ Z_1, Z_2 & \sim \text{Bern}\left(\frac{1}{2}\right) \\ X_K & \sim \text{Norm}(\mu_K, \sigma_K^2) \text{ when } k \text{ is odd and} \\ X_K & \sim \text{Bern}\left(\frac{1}{2}\right) \text{ when } k \text{ is even} \\ C_L & \sim \text{Norm}(\mu_L, \sigma_L^2) \text{ when } l \text{ is odd and} \\ C_L & \sim \text{Bern}\left(\frac{1}{2}\right) \text{ when } l \text{ is even} \end{cases}$$

and $\tau(x_1, x_2, x_3, c_1) = \alpha_0 + \alpha_1 x_1 + \alpha_2 x_2 x_3 + \alpha_3 c_1$.

Figure 9: Simulation 7

```r
ggplot(data = as.data.frame(tau_2), aes(x = tau_2)) +
  geom_histogram(fill="#f68060", alpha=.6,binwidth = 0.5) +
  labs(title ="Simulation 7, Treatment Effect", x = "Treatment Effect", y = "Count") +
  theme_minimal()
```



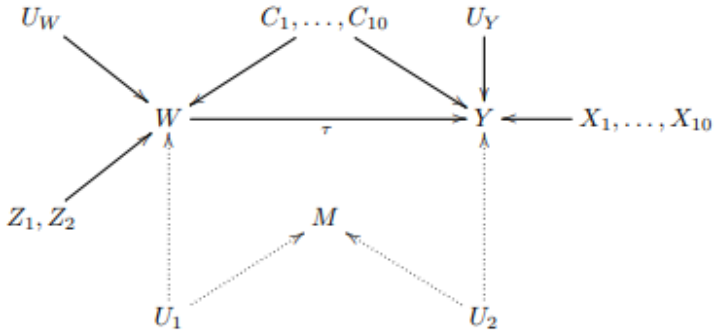

```r
#SCM

y_7 <-  tau_2*w_3 + X%*%beta_X_k + C%*%beta_C_l + u_y  #Output
```

2.8 Heterogeneous Treatment Effect with Confounded Assignment, Including Covariates Affecting to Output and a Pure Local M-Structure

```r
#SCM

y_8 <-  tau_2*w_4 + X%*%beta_X_k + C%*%beta_C_l + beta_U_2*u_2 + u_y  #Output
```

2.9 Heterogeneous Treatment Effect with Confounded Assignment, Including Covariates Affecting to Output and a Impure Local M-Structure

4.1.11 Heterogeneous Treatment Effect with Confounded Assignment, Including Covariates Affecting to Output and a Pure Local M-Structure



(4.35)

$$f_W(Z_1, Z_2, C_1, \ldots, C_{10}, U_1, U_W) = \frac{1}{1 + \exp\left\{-\left(\sum_{j=1}^{2} \gamma_{Z_j} Z_j + \sum_{l=1}^{10} \gamma_{C_l} C_l + \gamma_{U_1} U_1 + U_Y\right)\right\}} \tag{4.36}$$

$$f_M = \delta_1 U_1 + \delta_2 U_2 \tag{4.37}$$

$$f_Y(W, X_1, \ldots, X_{10}, C_1, \ldots, C_{10}, U_2, U_Y) = \tau(x_1, x_2, x_3, c_1) \cdot W + \sum_{k=1}^{10} \beta_{X_k} X_k + \sum_{l=1}^{10} \beta_{C_l} C_l + \beta_{U_2} U_2 + U_Y \tag{4.38}$$

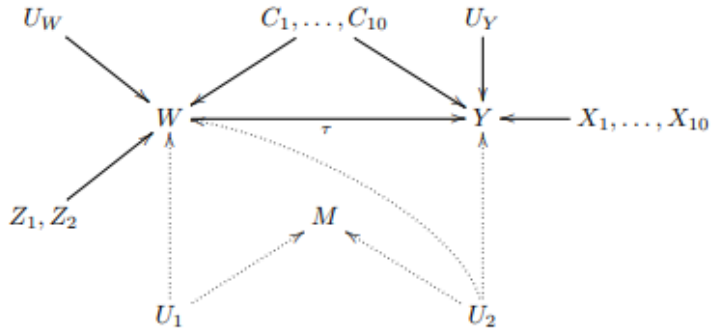
, where

$$\begin{cases} U_Y & \sim \text{Norm}(\mu_Y(0), \sigma_Y^2) \\ U_W & \sim \text{Norm}(0, \sigma_W^2) \\ Z_1, Z_2 & \sim \text{Bern}\left(\frac{1}{2}\right) \\ X_K & \sim \text{Norm}(\mu_K, \sigma_K^2) \text{ when } k \text{ is odd and} \\ X_K & \sim \text{Bern}\left(\frac{1}{2}\right) \text{ when } k \text{ is even} \\ C_L & \sim \text{Norm}(\mu_L, \sigma_L^2) \text{ when } l \text{ is odd and} \\ C_L & \sim \text{Bern}\left(\frac{1}{2}\right) \text{ when } l \text{ is even} \\ U_1, U_2 & \sim \text{Bern}\left(\frac{1}{2}\right) \end{cases}$$

and $\tau(x_1, x_2, x_3, c_1) = \alpha_0 + \alpha_1 x_1 + \alpha_2 x_2 x_3 + \alpha_3 c_1$.

Figure 10: Simulation 8

4.1.12 Heterogeneous Treatment Effect with Confounded Assignment, Including Covariates Affecting to Output and a Impure Local M-Structure



(4.39)

$$f_W(Z_1, Z_2, C_1, \ldots, C_{10}, U_1, U_2, U_W) = \frac{1}{1 + \exp\left\{-\left(\sum_{j=1}^{2} \gamma_{Z_j} Z_j + \sum_{l=1}^{10} \gamma_{C_l} C_l + \sum_{m=1}^{2} \gamma_{U_m} U_m + U_Y\right)\right\}} \tag{4.40}$$

$$f_M = \delta_1 U_1 + \delta_2 U_2 \tag{4.41}$$

$$f_Y(W, X_1, \ldots, X_{10}, C_1, \ldots, C_{10}, U_2, U_Y) = \tau(x_1, x_2, x_3, c_1) \cdot W + \sum_{k=1}^{10} \beta_{X_k} X_k + \sum_{l=1}^{10} \beta_{C_l} C_l + \beta_{U_2} U_2 + U_Y \tag{4.42}$$

, where

$$\begin{cases} U_Y & \sim \text{Norm}(\mu_Y(0), \sigma_Y^2) \\ U_W & \sim \text{Norm}(0, \sigma_W^2) \\ Z_1, Z_2 & \sim \text{Bern}\left(\frac{1}{2}\right) \\ X_K & \sim \text{Norm}(\mu_K, \sigma_K^2) \text{ when } k \text{ is odd and} \\ X_K & \sim \text{Bern}\left(\frac{1}{2}\right) \text{ when } k \text{ is even} \\ C_L & \sim \text{Norm}(\mu_L, \sigma_L^2) \text{ when } l \text{ is odd and} \\ C_L & \sim \text{Bern}\left(\frac{1}{2}\right) \text{ when } l \text{ is even} \\ U_1, U_2 & \sim \text{Bern}\left(\frac{1}{2}\right) \end{cases}$$

and $\tau(x_1, x_2, x_3, c_1) = \alpha_0 + \alpha_1 x_1 + \alpha_2 x_2 x_3 + \alpha_3 c_1$.

Figure 11: Simulation 9

```
#SCM

y_9 <- tau_2*w_5 + X%*%beta_X_k + C%*%beta_C_l + beta_U_2*u_2 + u_y      #Output
```

All the parameters, table:

```
#Betas

betas <- as.data.frame(c(beta_X_k, beta_C_l, beta_U_2))

rownames(betas) <- c("$\\beta_{X_1}$",
                     "$\\beta_{X_2}$",
                     "$\\beta_{X_3}$",
                     "$\\beta_{X_4}$",
                     "$\\beta_{X_5}$",
                     "$\\beta_{X_6}$",
                     "$\\beta_{X_7}$",
                     "$\\beta_{X_8}$",
                     "$\\beta_{X_9}$",
                     "$\\beta_{X_10}$",
                     "$\\beta_{C_1}$",
                     "$\\beta_{C_2}$",
                     "$\\beta_{C_3}$",
                     "$\\beta_{C_4}$",
                     "$\\beta_{C_5}$",
                     "$\\beta_{C_6}$",
                     "$\\beta_{C_7}$",
                     "$\\beta_{C_8}$",
                     "$\\beta_{C_9}$",
                     "$\\beta_{C_10}$",
                     "$\\beta_{U_2}$")

colnames(betas) <- c("Value")

knitr::kable(betas, escape = FALSE)
```

	Value
β_{X_1}	1.1846881
β_{X_2}	1.6731552
β_{X_3}	2.9565855
β_{X_4}	-1.8355449
β_{X_5}	2.8452661
β_{X_6}	-1.1567457
β_{X_7}	-1.5805033
β_{X_8}	1.4988092
β_{X_9}	1.3516565
β_{X_10}	2.8661819
β_{C_1}	2.9628036
β_{C_2}	0.8623509
β_{C_3}	-0.0993400
β_{C_4}	-1.1987373
β_{C_5}	-2.5123647

	Value
β_{C_6}	-1.7654443
β_{C_7}	0.2969675
β_{C_8}	0.9698760
β_{C_9}	0.9790413
β_{C_10}	0.3159898
β_{U_2}	5.0000000

```
gammas <- as.data.frame(c(gamma_z, gamma_C_l, gamma_U_1))

rownames(gammas) <- c("$\\gamma_{Z_1}$",
                      "$\\gamma_{Z_2}$",
                      "$\\gamma_{C_1}$",
                      "$\\gamma_{C_2}$",
                      "$\\gamma_{C_3}$",
                      "$\\gamma_{C_4}$",
                      "$\\gamma_{C_5}$",
                      "$\\gamma_{C_6}$",
                      "$\\gamma_{C_7}$",
                      "$\\gamma_{C_8}$",
                      "$\\gamma_{C_9}$",
                      "$\\gamma_{C_{10}}$",
                      "$\\gamma_{U_1}$")

colnames(gammas) <- c("Value")

knitr::kable(gammas, escape = FALSE)
```

	Value
γ_{Z_1}	0.3103331
γ_{Z_2}	-0.8675404
γ_{C_1}	0.2809952
γ_{C_2}	0.0842018
γ_{C_3}	-0.2081202
γ_{C_4}	-0.0699206
γ_{C_5}	-0.1153378
γ_{C_6}	-0.3488950
γ_{C_7}	-0.2784899
γ_{C_8}	0.3364730
γ_{C_9}	-0.3641025
$\gamma_{C_{10}}$	-0.1293683
γ_{U_1}	0.5000000

```
deltas <- as.data.frame(c(delta_1, delta_2))

rownames(deltas) <- c("$\\delta_1$", "$\\delta_2$")

colnames(deltas) <- c("Value")

knitr::kable(deltas, escape = FALSE)
```

	Value
δ_1	5
δ_2	5

As LaTeX:

```
knitr::kable(betas, format = "latex", escape = FALSE)
```

	Value
β_{X_1}	1.1846881
β_{X_2}	1.6731552
β_{X_3}	2.9565855
β_{X_4}	-1.8355449
β_{X_5}	2.8452661
β_{X_6}	-1.1567457
β_{X_7}	-1.5805033
β_{X_8}	1.4988092
β_{X_9}	1.3516565
β_{X_10}	2.8661819
β_{C_1}	2.9628036
β_{C_2}	0.8623509
β_{C_3}	-0.0993400
β_{C_4}	-1.1987373
β_{C_5}	-2.5123647
β_{C_6}	-1.7654443
β_{C_7}	0.2969675
β_{C_8}	0.9698760
β_{C_9}	0.9790413
β_{C_10}	0.3159898
β_{U_2}	5.0000000

```
knitr::kable(gammas, format = "latex", escape = FALSE)
```

	Value
γ_{Z_1}	0.3103331
γ_{Z_2}	-0.8675404
γ_{C_1}	0.2809952
γ_{C_2}	0.0842018
γ_{C_3}	-0.2081202
γ_{C_4}	-0.0699206
γ_{C_5}	-0.1153378
γ_{C_6}	-0.3488950
γ_{C_7}	-0.2784899
γ_{C_8}	0.3364730
γ_{C_9}	-0.3641025
$\gamma_{C_{10}}$	-0.1293683
γ_{U_1}	0.5000000

```
knitr::kable(deltas, format = "latex" , escape = FALSE)
```

	Value
δ_1	5
δ_2	5

3 Causal Forest

In the simulation study, we are using the *causal forest* . The function is in *grf* package which can be found from CRAN. See also this.

In each subsection, the causal boosting algorithm will be ran. The performance in each simulations are tested with *root mean squared error*:

$$\widehat{\mathrm{RMSE}} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left[(\tau(x) - \hat{\tau}(x)\right]^2}$$

```
fun.rmse <- function(predicted, true){

  rmse <- sqrt(mean((true - predicted)^2))

  rmse
}
```

Also the *proportional difference in the ATE estimates* for the whole population is calcucated (with 95 % confidence interval):

$$\frac{\tau_{\mathrm{ATE}} - \hat{\tau}_{ATE}}{\tau_{ATE}}$$

```
fun.diff_ATE <- function(ATE_est, ATE_true){

  print("Proportional mean of the differences:")

  diff_mean <- (ATE_true - ATE_est[1])/ATE_true

  print(diff_mean)

  print("Proportional mean of the differences, 95 % confidence intervals:")

  print(c((diff_mean - ATE_est[2]/ATE_true), (diff_mean + ATE_est[2]/ATE_true)))

}
```

The 95 % coverage of the CATEs:

```
# Is the real value in the 95% confidence interval

fun.coverage <- function(predicted, true){

  cate <- predicted$predictions

  std_error <- sqrt(predicted$variance.estimates)

  confint <- matrix(c((cate - qnorm(1 - 0.025)*std_error),
                      (cate + qnorm(1 - 0.025)*std_error)), ncol = 2)

  fun.is_in_confint <- function(value, min, max){

    (value >= min)&(value <= max)
```

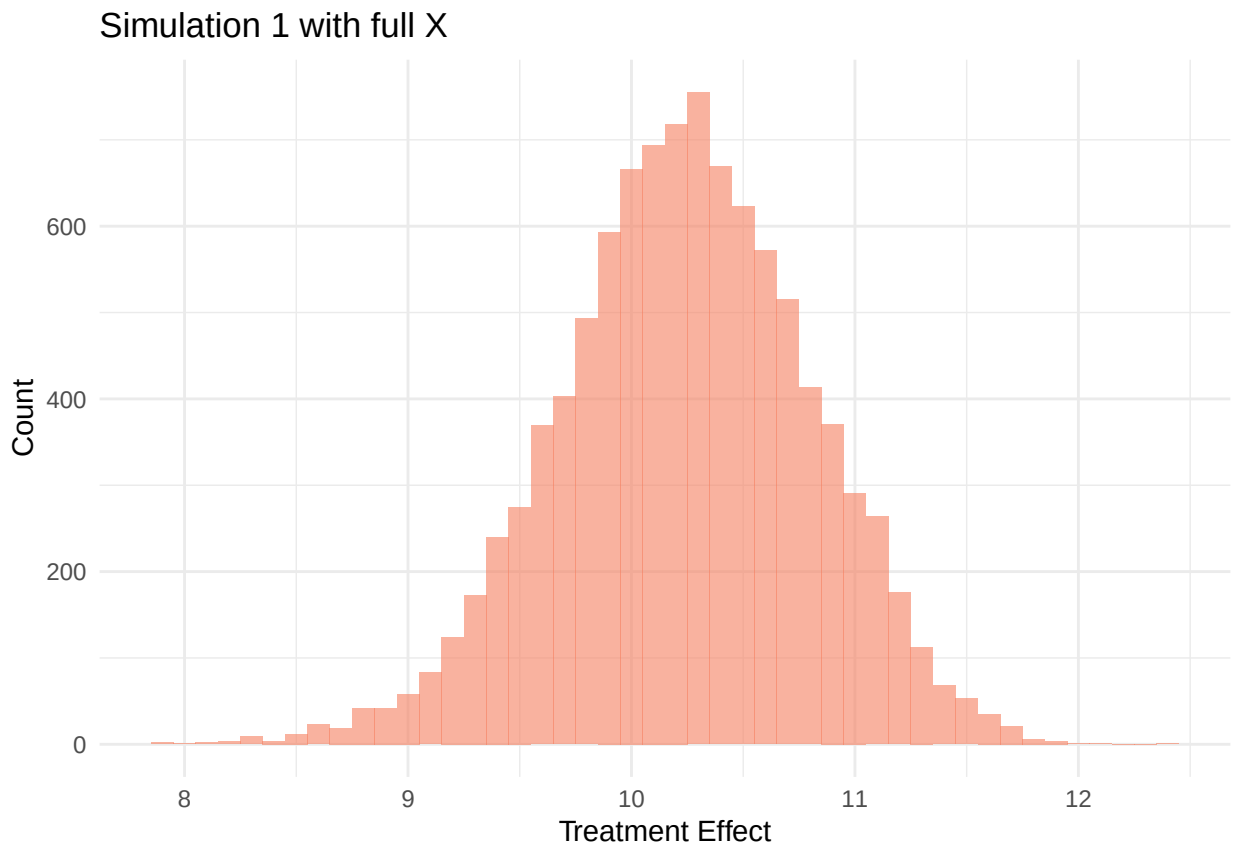
```
  }

  in_confint <- cbind(true, confint) %>%
    as.data.frame()  %>%
    mutate(in_confint = fun.is_in_confint(true, V2, V3)) %>%
    select(in_confint)

  length(which(in_confint$in_confint == TRUE))/length(in_confint$in_confint)

}
```

Also the running times for the algorithm are given for each simulation. Let's remark the adjusted set with $\mathbb{S}$.

3.1 Randomized Controlled Trial with Constant Treatment Effect

$X \in \mathbb{S}$

```
# Estimate causal forest
start_time_1 <- Sys.time() #Recording the running time

#Fitting the model
cf1.full <- grf::causal_forest(X, y_1, w_1)


end_time_1 <- Sys.time()

#Predicted values

pred_tau_1.full <- predict(cf1.full, estimate.variance = TRUE)

plot_pred_tau_1 <- ggplot(data = as.data.frame(pred_tau_1.full$predictions),
                          aes(x = (pred_tau_1.full$predictions))) +
  geom_histogram(fill="#f68060", alpha=.6,binwidth = 0.1) +
  labs(title ="Simulation 1 with full X",
       x = "Treatment Effect", y = "Count") +
  theme_minimal()

plot_pred_tau_1
```



Runing time:

```
start_time_1 - end_time_1
```

```
## Time difference of -35.44318 secs
```

RMSE:

```
rmse_1.full <- fun.rmse(predicted = pred_tau_1.full$predictions, true = tau)

rmse_1.full
```

```
## [1] 0.610858
```

Coverage:

```
coverage_1.full <- fun.coverage(pred_tau_1.full, tau)

coverage_1.full
```

```
## [1] 0.9917
```

Estimated ATE:

```r
ATE_est_1.full <- average_treatment_effect(cf1.full)


ATE_est_1.full
```

```
##  estimate   std.err
## 10.242091  0.203769
```

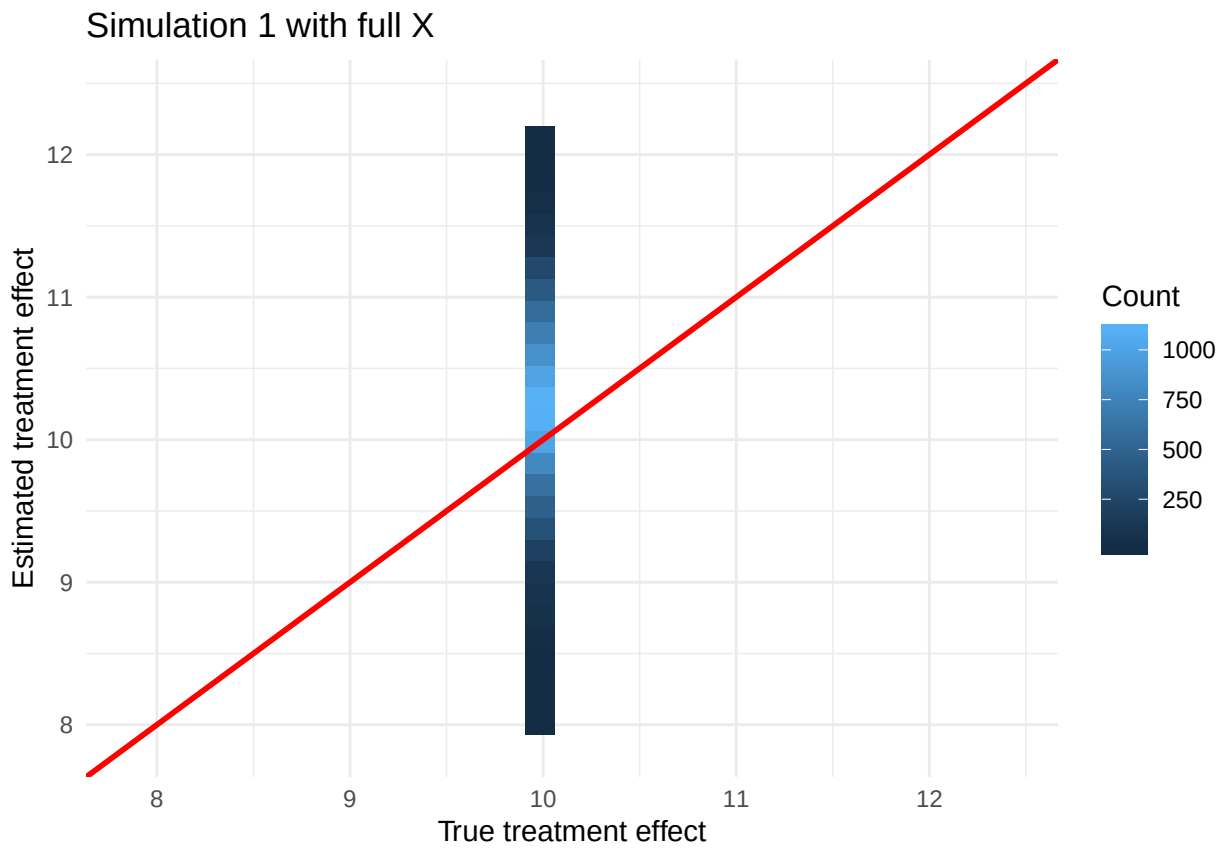
The *proportional difference in the ATE estimates* for the whole population:

```r
fun.diff_ATE(ATE_est = ATE_est_1.full, ATE_true = tau)
```

```
## [1] "Proportional mean of the differences:"
##    estimate
## -0.02420905
## [1] "Proportional mean of the differences, 95 % confidence intervals:"
##     estimate     estimate
## -0.044585947 -0.003832155
```

```r
true_vs_pred_1.full <- as.data.frame(cbind(tau, pred_tau_1.full$predictions))

colnames(true_vs_pred_1.full) <- c("tau", "pred_tau")

true_vs_pred_1.full$tau <- true_vs_pred_1.full$tau + rnorm(n, 0 , sd = 0.0001)

plot1 <- ggplot(data = true_vs_pred_1.full, aes(x = tau, y = pred_tau)) +
  geom_bin2d() +
  geom_abline(slope = 1, intercept = 0, colour = "red", size = 1) +
  theme_minimal() +
  labs(title = "Simulation 1 with full X",
       x = "True treatment effect", y = "Estimated treatment effect",
       fill = "Count") +
  lims(x = c(min(true_vs_pred_1.full$pred_tau),
             max(true_vs_pred_1.full$pred_tau)),
       y = c(min(true_vs_pred_1.full$pred_tau),
             max(true_vs_pred_1.full$pred_tau)))

plot1
```

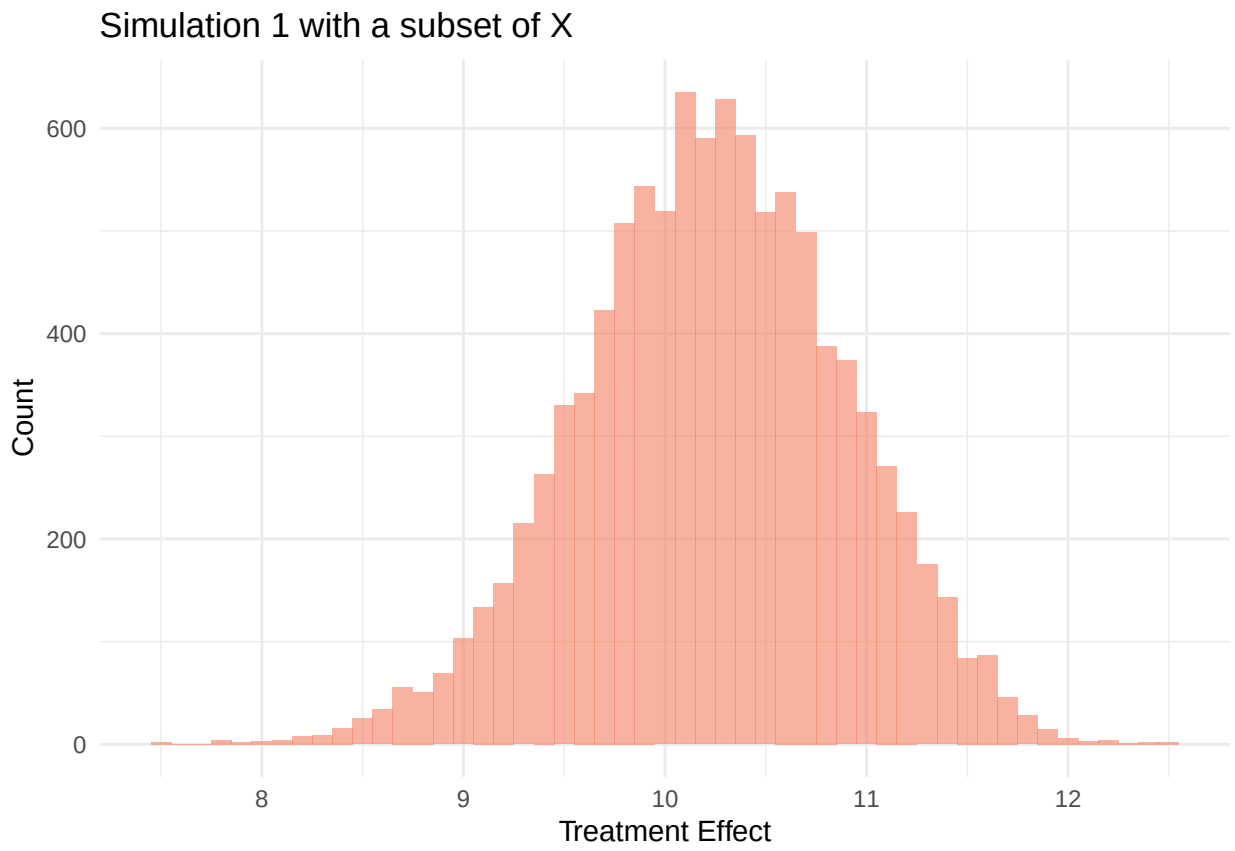
```
## Warning: Removed 1 rows containing missing values (geom_tile).
```



Let's try to use only the most important (*"A simple weighted sum of how many times feature i was split on at each depth in the forest"*) variables (over the median):

```
important_var_1 <- which(variable_importance(cf1.full) >= median(variable_importance(cf1.full))) #Varia
```

```
# Estimate causal forest
start_time_1_2 <- Sys.time() #Recording the running time

#Fitting the model
cf1.important <- grf::causal_forest(X[, important_var_1], y_1, w_1)


end_time_1_2 <- Sys.time()

#Predicted values

pred_tau_1.important <- predict(cf1.important, estimate.variance = TRUE)

plot_pred_tau_1_2 <- ggplot(data = as.data.frame(pred_tau_1.important$predictions),
                      aes(x = (pred_tau_1.important$predictions))) +
  geom_histogram(fill="#f68060", alpha=.6, binwidth = 0.1) +
  labs(title ="Simulation 1 with a subset of X",
       x = "Treatment Effect", y = "Count") +
  theme_minimal()

plot_pred_tau_1_2
```



Runing time:

```
start_time_1_2 - end_time_1_2
```

```
## Time difference of -30.76669 secs
```

RMSE:

```
rmse_1.important <- fun.rmse(predicted = pred_tau_1.important$predictions, true = tau)

rmse_1.important
```

```
## [1] 0.6979229
```

Coverage:

```
coverage_1.important <- fun.coverage(pred_tau_1.important, tau)

coverage_1.important
```

```
## [1] 0.9875
```

Estimated ATE:

```r
ATE_est_1.important <- average_treatment_effect(cf1.important)


ATE_est_1.important
```

```
##   estimate    std.err
## 10.2259539  0.2062475
```

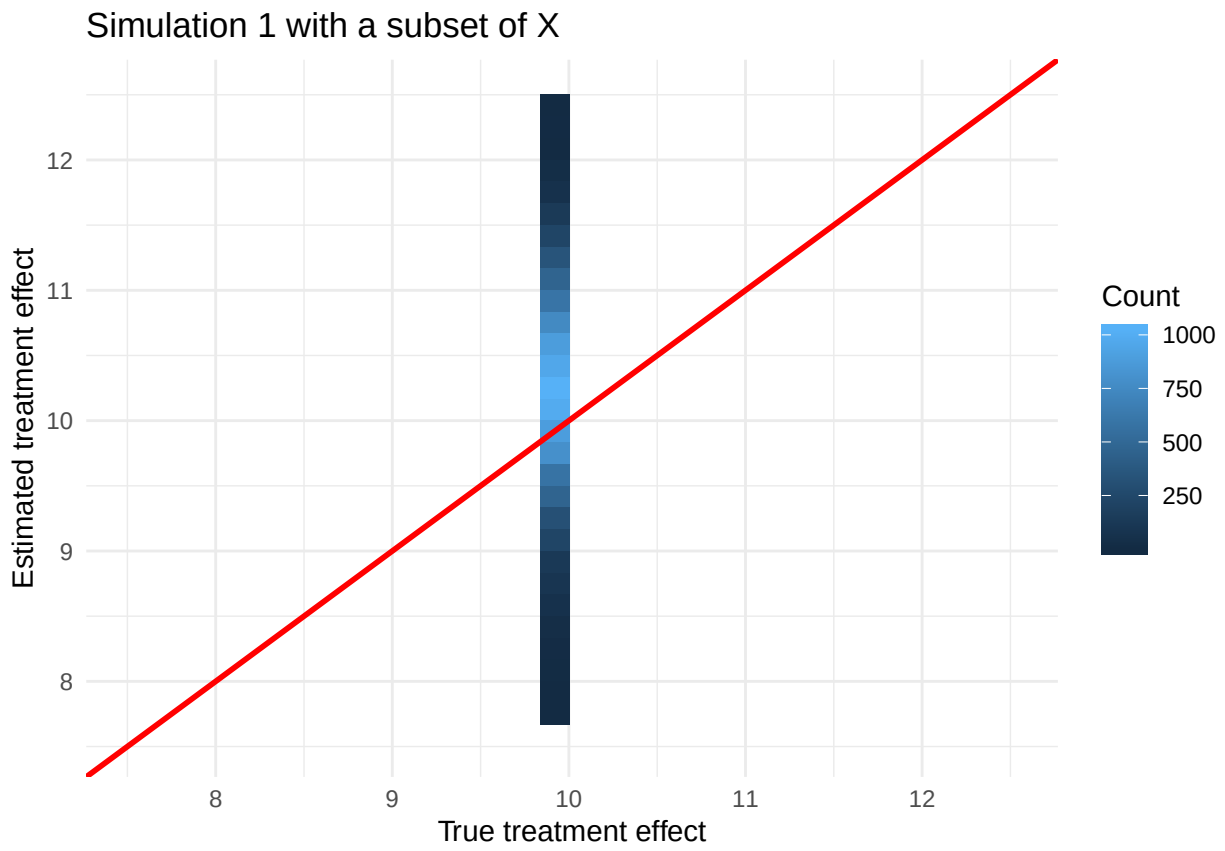
The *proportional difference in the ATE estimates* for the whole population:

```r
fun.diff_ATE(ATE_est = ATE_est_1.important, ATE_true = tau)
```

```
## [1] "Proportional mean of the differences:"
##    estimate
## -0.02259539
## [1] "Proportional mean of the differences, 95 % confidence intervals:"
##     estimate     estimate
## -0.043220144 -0.001970645
```

```r
true_vs_pred_1.important <- as.data.frame(cbind(tau, pred_tau_1.important$predictions))

colnames(true_vs_pred_1.important) <- c("tau", "pred_tau")

true_vs_pred_1.important$tau <- true_vs_pred_1.important$tau + rnorm(n, 0 , sd = 0.0001)

plot1_2 <- ggplot(data = true_vs_pred_1.important, aes(x = tau, y = pred_tau)) +
  geom_bin2d() +
  geom_abline(slope = 1, intercept = 0, colour = "red", size = 1) +
  theme_minimal() +
  labs(title = "Simulation 1 with a subset of X",
       x = "True treatment effect", y = "Estimated treatment effect",
       fill = "Count") +
  lims(x = c(min(true_vs_pred_1.important$pred_tau),
             max(true_vs_pred_1.important$pred_tau)),
       y = c(min(true_vs_pred_1.important$pred_tau),
             max(true_vs_pred_1.important$pred_tau)))

plot1_2
```

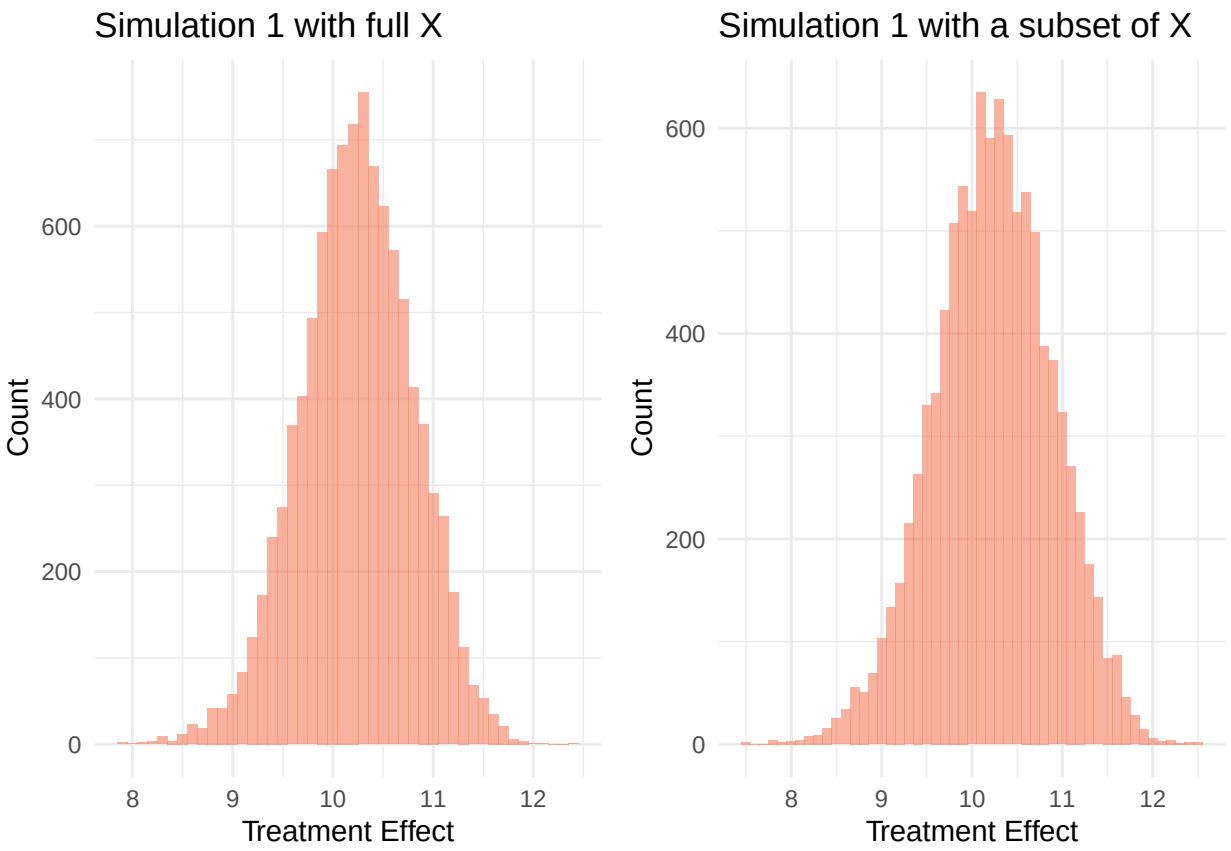
```
## Warning: Removed 1 rows containing missing values (geom_tile).
```



Summary

Predicted $\hat{\tau}s$:

```
grid.arrange(plot_pred_tau_1, plot_pred_tau_1_2, nrow = 1)
```

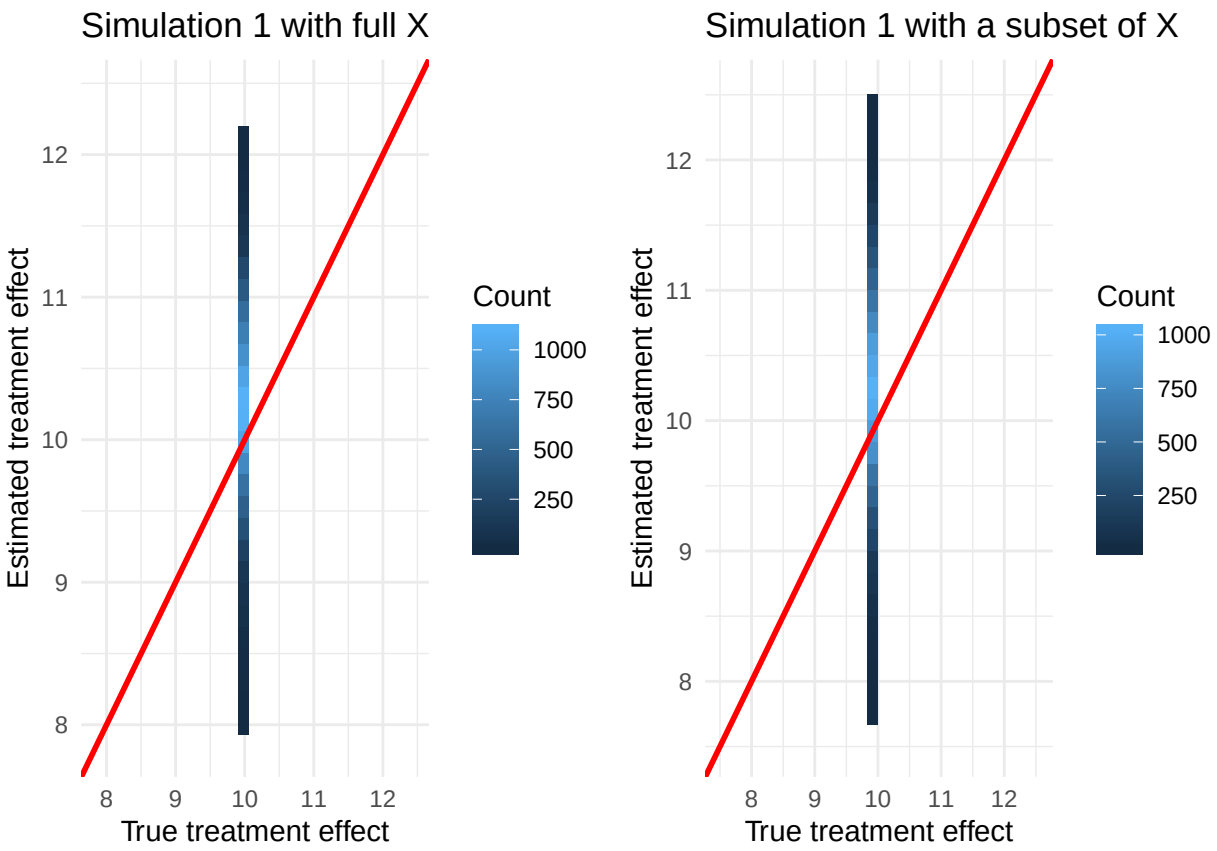


True τ vs. predicted $\hat{\tau}$:

```
grid.arrange(plot1, plot1_2, nrow = 1)
```

```
## Warning: Removed 1 rows containing missing values (geom_tile).
```

```
## Warning: Removed 1 rows containing missing values (geom_tile).
```



RMSEs:

```
rmse_1 <- as.data.frame(c(rmse_1.full, rmse_1.important))

rownames(rmse_1) <- c("Full $X$", "Subset of $X$")

colnames(rmse_1) <- c("RMSE")

knitr::kable(rmse_1, escape = FALSE)
```

	RMSE
Full X	0.6108580
Subset of X	0.6979229

Coverages:

```
coverage_1 <- as.data.frame(c(coverage_1.full, coverage_1.important))

rownames(coverage_1) <- c("Full $X$", "Subset of $X$")

colnames(coverage_1) <- c("Coverage")

knitr::kable(coverage_1, escape = FALSE)
```

	Coverage
Full X	0.9917
Subset of X	0.9875

LaTeX:

```
knitr::kable(rmse_1, format = "latex",escape = FALSE)
```

	RMSE
Full X	0.6108580
Subset of X	0.6979229

```
knitr::kable(coverage_1, format = "latex",escape = FALSE)
```

	Coverage
Full X	0.9917
Subset of X	0.9875

3.2 Constant Treatment Effect with Unconfounded Assignment, Including Covariates Affecting to Output

Let's do the first estimation without the Zs ($X \in \mathbb{S}$):

```
# Estimate causal forest
start_time_2 <- Sys.time() #Recording the running time

#Fitting the model
cf2.no_z <- grf::causal_forest(X, y_2, w_2)


end_time_2 <- Sys.time()

#Predicted values

pred_tau_2.no_z <-predict(cf2.no_z, estimate.variance = TRUE)

plot_pred_tau_2 <- ggplot(data = as.data.frame(pred_tau_2.no_z$predictions),
       aes(x = (pred_tau_2.no_z$predictions))) +
  geom_histogram(fill="#f68060", alpha=.6,binwidth = 0.1) +
  labs(title ="Simulation 2 without Z",
       x = "Treatment Effect", y = "Count") +
  theme_minimal()

plot_pred_tau_2
```

Runing time:

```
start_time_2 - end_time_2
```

```
## Time difference of -38.83714 secs
```

RMSE:

```
rmse_2.no_z <- fun.rmse(predicted = pred_tau_2.no_z$predictions, true = tau)

rmse_2.no_z
```

```
## [1] 0.6930577
```

Coverage:

```
coverage_2.no_z <- fun.coverage(pred_tau_2.no_z, tau)

coverage_2.no_z
```

```
## [1] 0.9842
```

Estimated ATE:

```r
ATE_est_2.no_z <- average_treatment_effect(cf2.no_z)

ATE_est_2.no_z
```

```
##  estimate   std.err
## 9.9265084 0.2053532
```

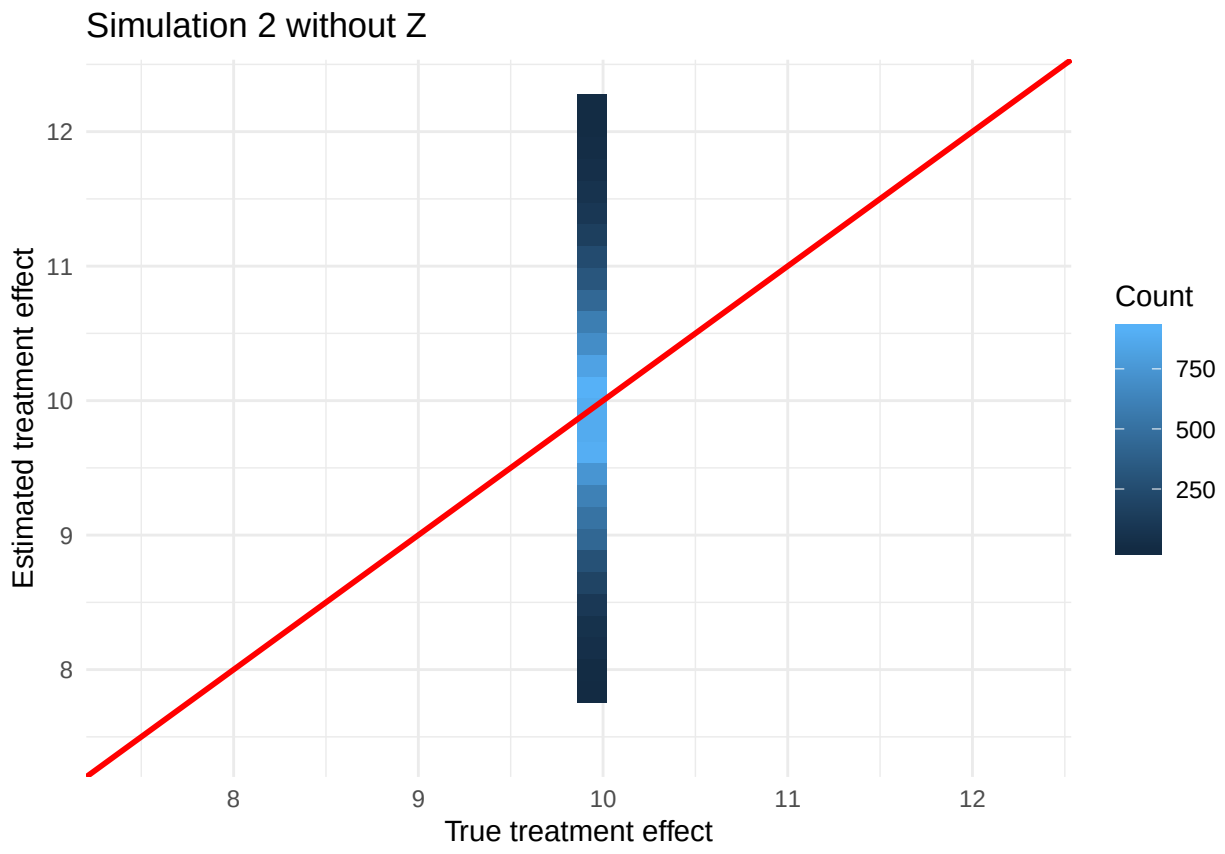
The *proportional difference in the ATE estimates* for the whole population:

```r
fun.diff_ATE(ATE_est = ATE_est_2.no_z, ATE_true = tau)
```

```
## [1] "Proportional mean of the differences:"
##    estimate
## 0.007349159
## [1] "Proportional mean of the differences, 95 % confidence intervals:"
##    estimate    estimate
## -0.01318616  0.02788447
```

```r
true_vs_pred_2.no_z <- as.data.frame(cbind(tau, pred_tau_2.no_z$predictions))

colnames(true_vs_pred_2.no_z) <- c("tau", "pred_tau")

true_vs_pred_2.no_z$tau <- true_vs_pred_2.no_z$tau + rnorm(n, 0 , sd = 0.0001)

plot2 <- ggplot(data = true_vs_pred_2.no_z, aes(x = tau, y = pred_tau)) +
  geom_bin2d() +
  geom_abline(slope = 1, intercept = 0, colour = "red", size = 1) +
  theme_minimal() +
  labs(title = "Simulation 2 without Z",
       x = "True treatment effect", y = "Estimated treatment effect", fill = "Count") +
  lims(x = c(min(true_vs_pred_2.no_z$pred_tau), max(true_vs_pred_2.no_z$pred_tau)),
       y = c(min(true_vs_pred_2.no_z$pred_tau), max(true_vs_pred_2.no_z$pred_tau)))

plot2
```

```
## Warning: Removed 1 rows containing missing values (geom_tile).
```



$X \cup Z \in \mathbb{S}$s.

```
# Estimate causal forest
start_time_3 <- Sys.time() #Recording the running time

#Fitting the model
cf2.with_z <- grf::causal_forest(cbind(X, Z), y_2, w_2, orthog.boosting = FALSE)


end_time_3 <- Sys.time()

#Predicted values

pred_tau_2.with_z <-predict(cf2.with_z, estimate.variance = TRUE)

plot_pred_tau_3 <- ggplot(data = as.data.frame(pred_tau_2.with_z$predictions),
       aes(x = (pred_tau_2.with_z$predictions))) +
  geom_histogram(fill="#f68060", alpha=.6, binwidth = 0.1) +
  labs(title ="Simulation 2 with Z",
       x = "Treatment Effect", y = "Count") +
  theme_minimal()

plot_pred_tau_3
```

Runing time:

```
start_time_3 - end_time_3
```

```
## Time difference of -36.1214 secs
```

RMSE:

```
rmse_2.with_z <- fun.rmse(predicted = pred_tau_2.with_z$predictions, true = tau)

rmse_2.with_z
```

```
## [1] 0.6742028
```

Coverage:

```
coverage_2.with_z <- fun.coverage(pred_tau_2.with_z, tau)

coverage_2.with_z
```

```
## [1] 0.9877
```

Estimated ATE:

```r
ATE_est_2.with_z <- average_treatment_effect(cf2.with_z)

ATE_est_2.with_z
```

```
##  estimate   std.err
## 9.9278134 0.2096408
```

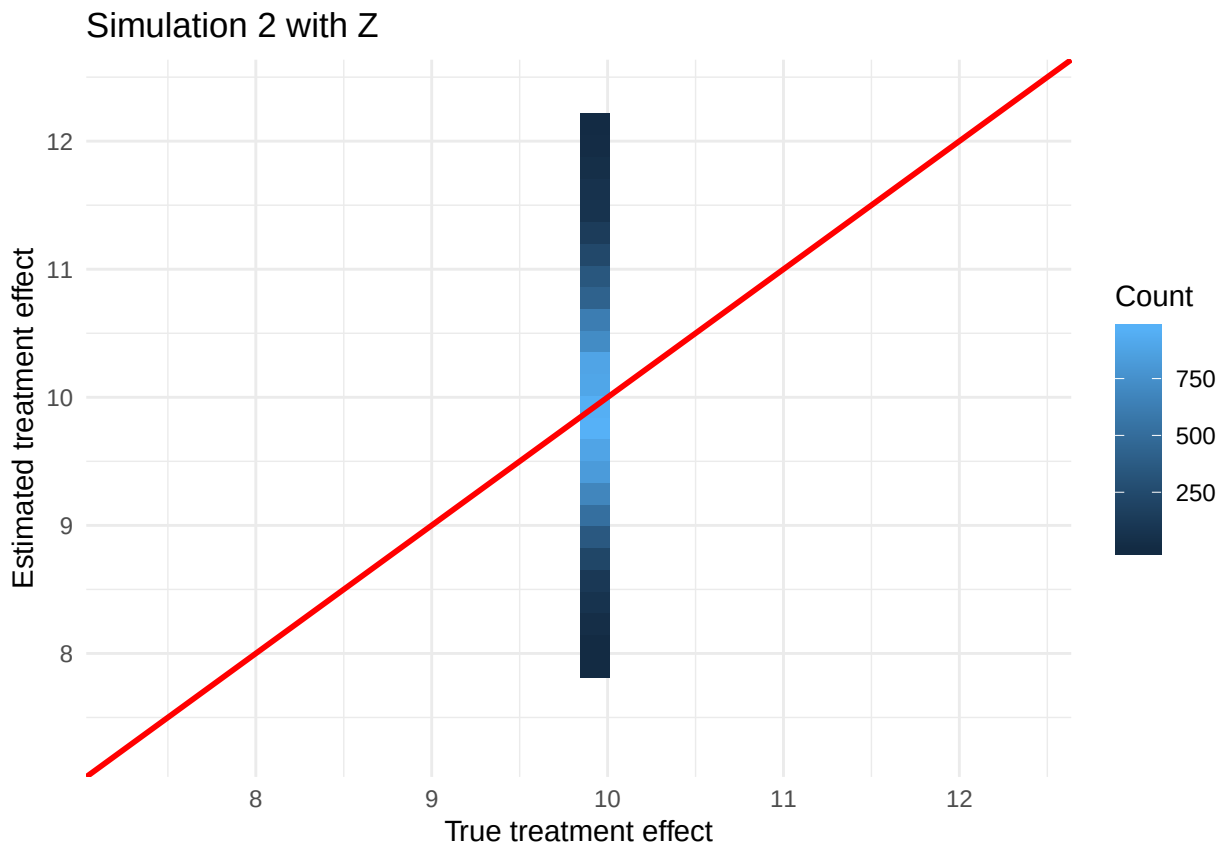
The *proportional difference in the ATE estimates* for the whole population:

```r
fun.diff_ATE(ATE_est = ATE_est_2.with_z, ATE_true = tau)
```

```
## [1] "Proportional mean of the differences:"
##    estimate
## 0.007218663
## [1] "Proportional mean of the differences, 95 % confidence intervals:"
##    estimate    estimate
## -0.01374542  0.02818274
```

```r
true_vs_pred_2.with_z <- as.data.frame(cbind(tau, pred_tau_2.with_z$predictions))

colnames(true_vs_pred_2.with_z) <- c("tau", "pred_tau")

true_vs_pred_2.with_z$tau <- true_vs_pred_2.with_z$tau + rnorm(n, 0 , sd = 0.0001)

plot3 <- ggplot(data = true_vs_pred_2.with_z, aes(x = tau, y = pred_tau)) +
  geom_bin2d() +
  geom_abline(slope = 1, intercept = 0, colour = "red", size = 1) +
  theme_minimal() +
  labs(title = "Simulation 2 with Z",
       x = "True treatment effect", y = "Estimated treatment effect", fill = "Count") +
  lims(x = c(min(true_vs_pred_2.with_z$pred_tau),
             max(true_vs_pred_2.with_z$pred_tau)),
       y = c(min(true_vs_pred_2.with_z$pred_tau),
             max(true_vs_pred_2.with_z$pred_tau)))

plot3
```

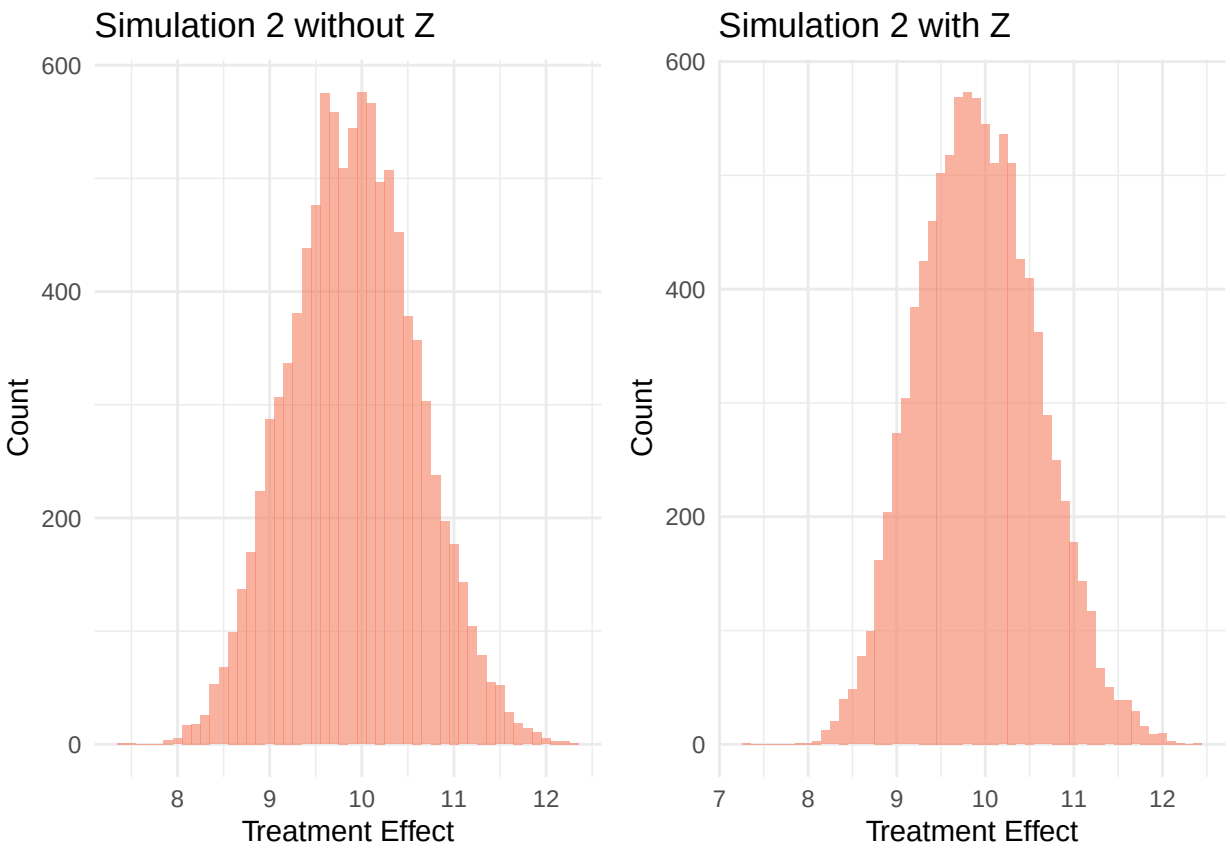
```
## Warning: Removed 1 rows containing missing values (geom_tile).
```



Summary

Predicted $\hat{\tau}s$:

```
grid.arrange(plot_pred_tau_2, plot_pred_tau_3, nrow = 1)
```

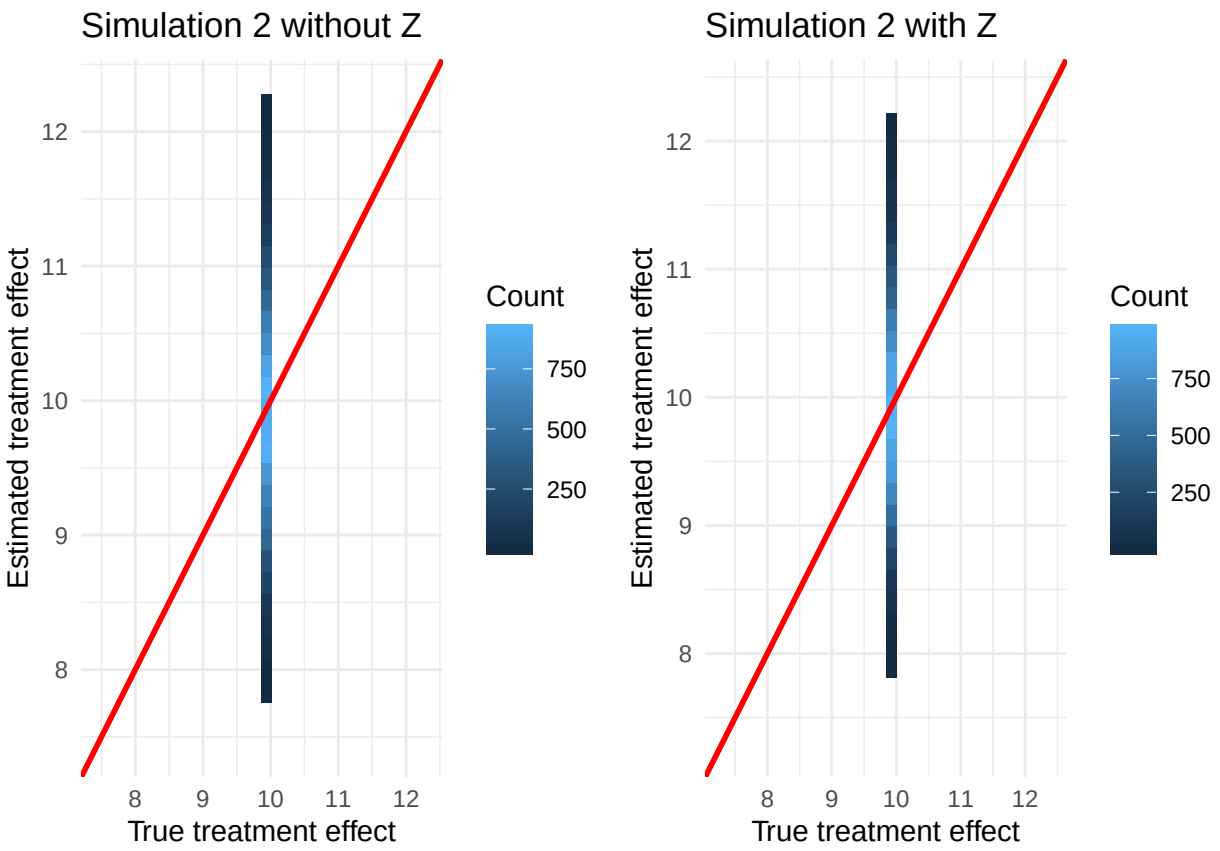


True τ vs. predicted $\hat{\tau}$:

```
grid.arrange(plot2, plot3, nrow = 1)
```

```
## Warning: Removed 1 rows containing missing values (geom_tile).
```

```
## Warning: Removed 1 rows containing missing values (geom_tile).
```



RMSEs:

```
rmse_2 <- as.data.frame(c(rmse_2.no_z, rmse_2.with_z))

rownames(rmse_2) <- c("Without $Z$", "With $Z$")

colnames(rmse_2) <- c("RMSE")

knitr::kable(rmse_2, escape = FALSE)
```

	RMSE
Without Z	0.6930577
With Z	0.6742028

Coverages:

```
coverage_2 <- as.data.frame(c(coverage_2.no_z, coverage_2.with_z))

rownames(coverage_2) <- c("Without $Z$", "With $Z$")

colnames(coverage_2) <- c("Coverage")

knitr::kable(coverage_2, escape = FALSE)
```

	Coverage
Without Z	0.9842
With Z	0.9877

LaTeX:

```
knitr::kable(rmse_2, format = "latex",escape = FALSE)
```

	RMSE
Without Z	0.6930577
With Z	0.6742028

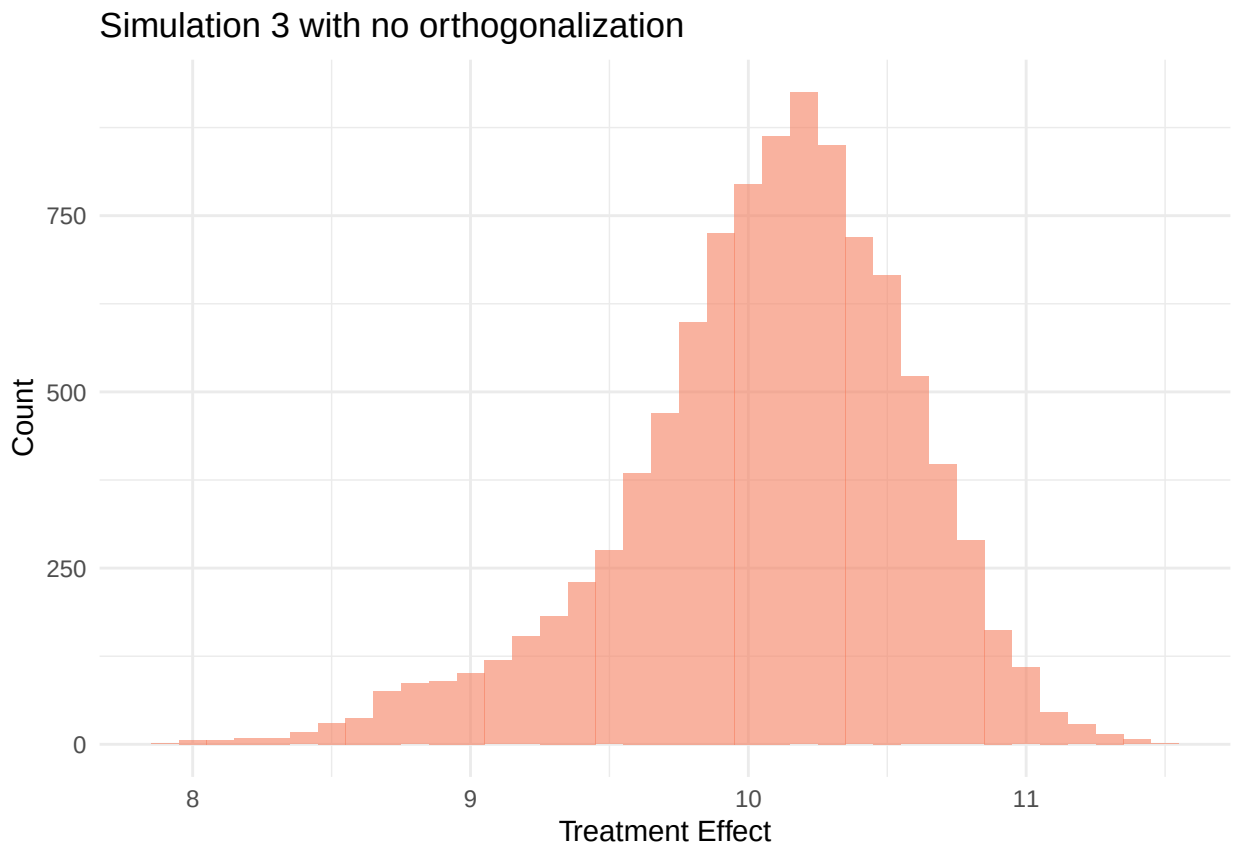
```
knitr::kable(coverage_2, format = "latex",escape = FALSE)
```

	Coverage
Without Z	0.9842
With Z	0.9877

3.3 Constant Treatment Effect with Confounded Assignment, Including Covariates Affecting to Output

Let's do the first estimation with all the observed variables $X \cup Z \cup C \in \mathbb{S}$. orthogonalization is not used in the first test:

```
# Estimate causal forest
start_time_4 <- Sys.time() #Recording the running time

#Fitting the model
cf3.no_ort <- grf::causal_forest(cbind(X, Z, C), y_3, w_3, orthog.boosting = FALSE)


end_time_4 <- Sys.time()

#Predicted values

pred_tau_3.no_ort <-predict(cf3.no_ort, estimate.variance = TRUE)

plot_pred_tau_4 <- ggplot(data = as.data.frame(pred_tau_3.no_ort$predictions),
       aes(x = (pred_tau_3.no_ort$predictions))) +
  geom_histogram(fill="#f68060", alpha=.6, binwidth = 0.1) +
  labs(title ="Simulation 3 with no orthogonalization",
       x = "Treatment Effect", y = "Count") +
  theme_minimal()

plot_pred_tau_4
```



Runing time:

```
start_time_4 - end_time_4
```

```
## Time difference of -1.07855 mins
```

RMSE:

```
rmse_3.no_ort <- fun.rmse(predicted = pred_tau_3.no_ort$predictions, true = tau)

rmse_3.no_ort
```

```
## [1] 0.5160037
```

Coverage:

```
coverage_3.no_ort <- fun.coverage(pred_tau_3.no_ort, tau)

coverage_3.no_ort
```

```
## [1] 0.9976
```

Estimated ATE:

```r
ATE_est_3.no_ort <- average_treatment_effect(cf3.no_ort)

ATE_est_3.no_ort
```

```
##   estimate    std.err
## 10.0475465  0.2168458
```

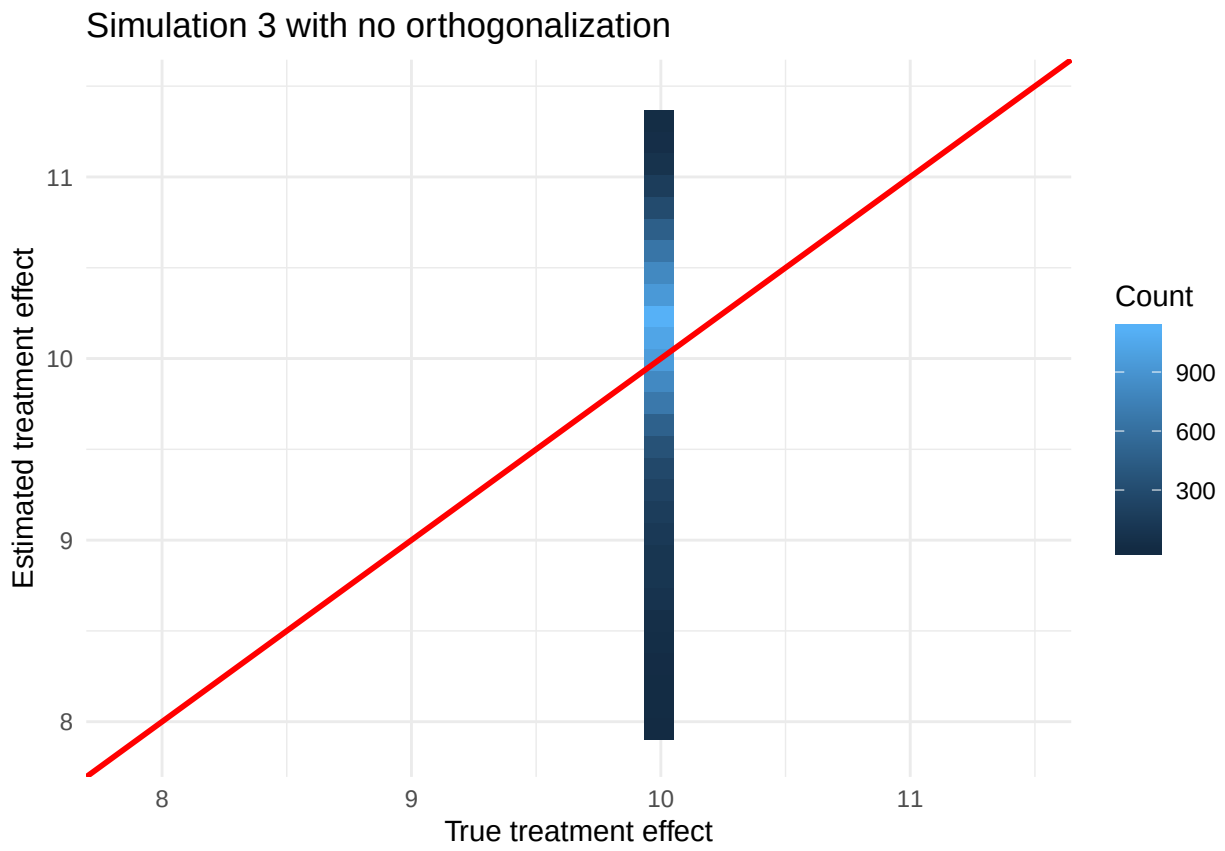
The *proportional difference in the ATE estimates* for the whole population:

```r
fun.diff_ATE(ATE_est = ATE_est_3.no_ort, ATE_true = tau)
```

```
## [1] "Proportional mean of the differences:"
##     estimate
## -0.004754651
## [1] "Proportional mean of the differences, 95 % confidence intervals:"
##    estimate    estimate
## -0.02643923  0.01692993
```

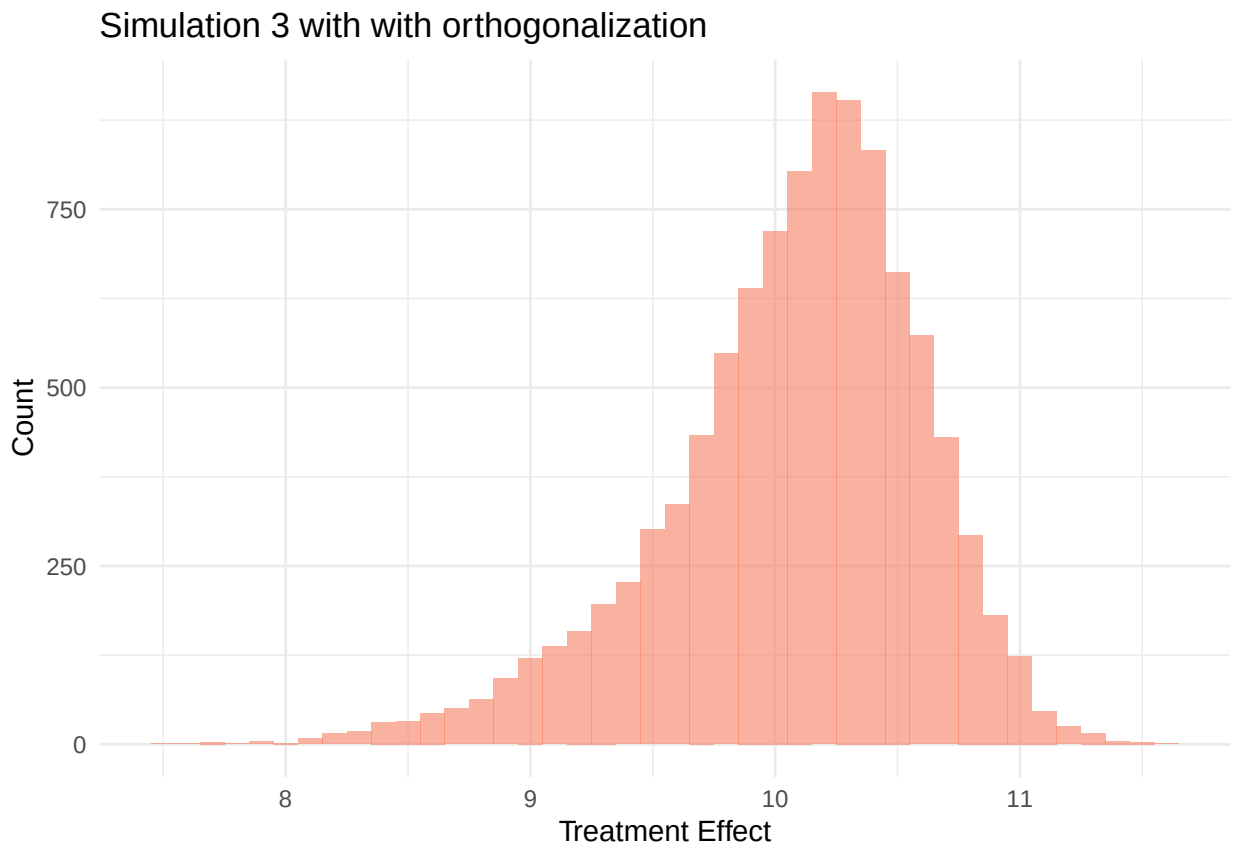
```r
true_vs_pred_3.no_ort <- as.data.frame(cbind(tau, pred_tau_3.no_ort$predictions))

colnames(true_vs_pred_3.no_ort) <- c("tau", "pred_tau")

true_vs_pred_3.no_ort$tau <- true_vs_pred_3.no_ort$tau + rnorm(n, 0 , sd = 0.0001)

plot4 <- ggplot(data = true_vs_pred_3.no_ort, aes(x = tau, y = pred_tau)) +
  geom_bin2d() +
  geom_abline(slope = 1, intercept = 0, colour = "red", size = 1) +
  theme_minimal() +
  labs(title = "Simulation 3 with no orthogonalization",
       x = "True treatment effect", y = "Estimated treatment effect", fill = "Count") +
  lims(x = c(min(true_vs_pred_3.no_ort$pred_tau), max(true_vs_pred_3.no_ort$pred_tau)),
       y = c(min(true_vs_pred_3.no_ort$pred_tau), max(true_vs_pred_3.no_ort$pred_tau)))

plot4
```

```
## Warning: Removed 1 rows containing missing values (geom_tile).
```



$X \cup Z \cup C \in \mathbb{S}$. Orthogonalizarion is used in the second test:

```
# Estimate causal forest
start_time_5 <- Sys.time() #Recording the running time

#Fitting the model
cf3.with_ort <- grf::causal_forest(cbind(X, Z, C), y_3, w_3, orthog.boosting = TRUE)


end_time_5 <- Sys.time()

#Predicted values

pred_tau_3.with_ort <-predict(cf3.with_ort, estimate.variance = TRUE)

plot_pred_tau_5 <- ggplot(data = as.data.frame(pred_tau_3.with_ort$predictions)
                          , aes(x = (pred_tau_3.with_ort$predictions))) +
  geom_histogram(fill="#f68060", alpha=.6, binwidth = 0.1) +
  labs(title ="Simulation 3 with with orthogonalization",
       x = "Treatment Effect", y = "Count") +
  theme_minimal()

plot_pred_tau_5
```



Runing time:

```
start_time_5 - end_time_5
```

```
## Time difference of -1.204727 mins
```

RMSE:

```
rmse_3.with_ort <- fun.rmse(predicted = pred_tau_3.with_ort$predictions, true = tau)

rmse_3.with_ort
```

```
## [1] 0.534276
```

Coverage:

```
coverage_3.with_ort <- fun.coverage(pred_tau_3.with_ort, tau)

coverage_3.with_ort
```

```
## [1] 0.9966
```

Estimated ATE:

```
ATE_est_3.with_ort <- average_treatment_effect(cf3.with_ort)

ATE_est_3.with_ort
```

```
##   estimate    std.err
## 10.0927628  0.2105682
```

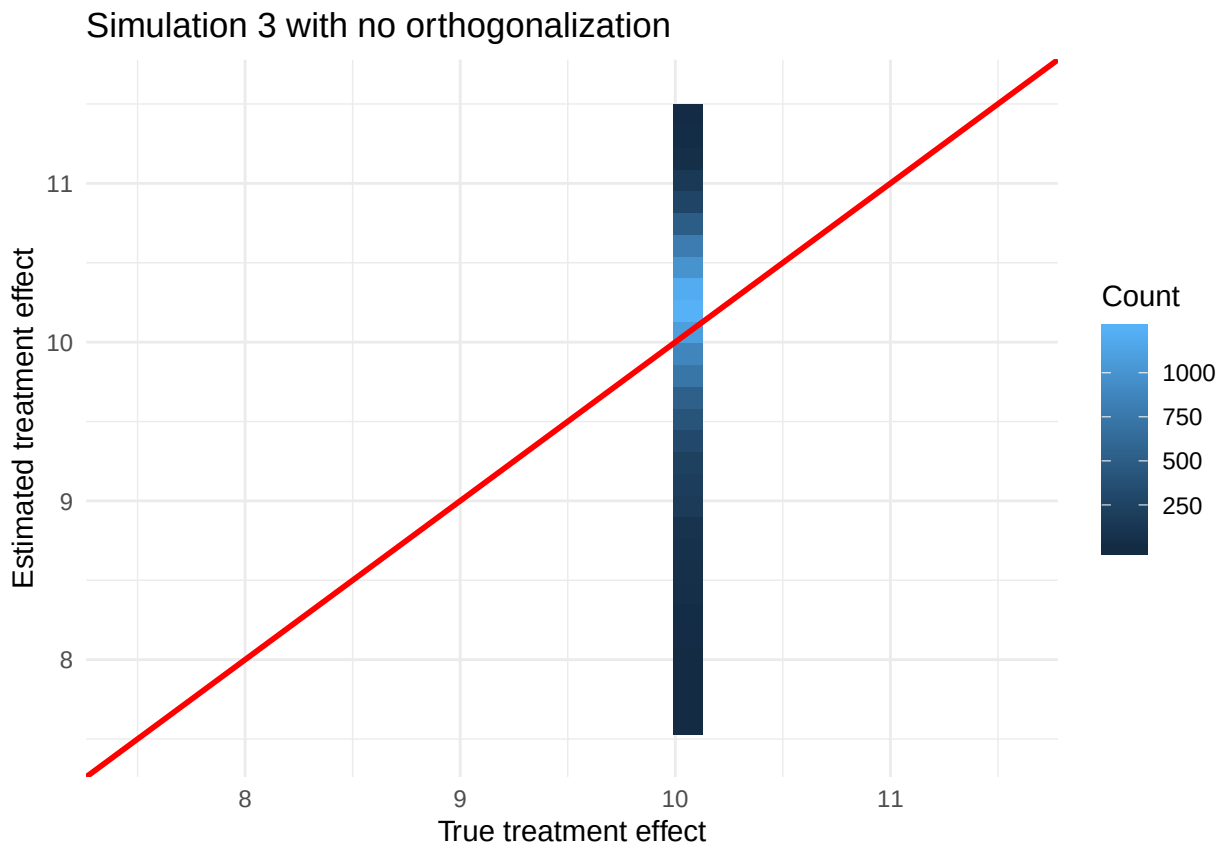
The *proportional difference in the ATE estimates* for the whole population:

```
fun.diff_ATE(ATE_est = ATE_est_3.with_ort, ATE_true = tau)
```

```
## [1] "Proportional mean of the differences:"
##     estimate
## -0.009276278
## [1] "Proportional mean of the differences, 95 % confidence intervals:"
##    estimate    estimate
## -0.03033310  0.01178054
```

```
true_vs_pred_3.with_ort <- as.data.frame(cbind(tau, pred_tau_3.with_ort$predictions))

colnames(true_vs_pred_3.with_ort) <- c("tau", "pred_tau")

true_vs_pred_3.with_ort$tau <- true_vs_pred_3.with_ort$tau + rnorm(n, 0 , sd = 0.0001)

plot5 <- ggplot(data = true_vs_pred_3.with_ort, aes(x = tau, y = pred_tau)) +
  geom_bin2d() +
  geom_abline(slope = 1, intercept = 0, colour = "red", size = 1) +
  theme_minimal() +
  labs(title = "Simulation 3 with no orthogonalization",
       x = "True treatment effect", y = "Estimated treatment effect", fill = "Count") +
  lims(x = c(min(true_vs_pred_3.with_ort$pred_tau), max(true_vs_pred_3.with_ort$pred_tau)),
       y = c(min(true_vs_pred_3.with_ort$pred_tau), max(true_vs_pred_3.with_ort$pred_tau)))

plot5
```

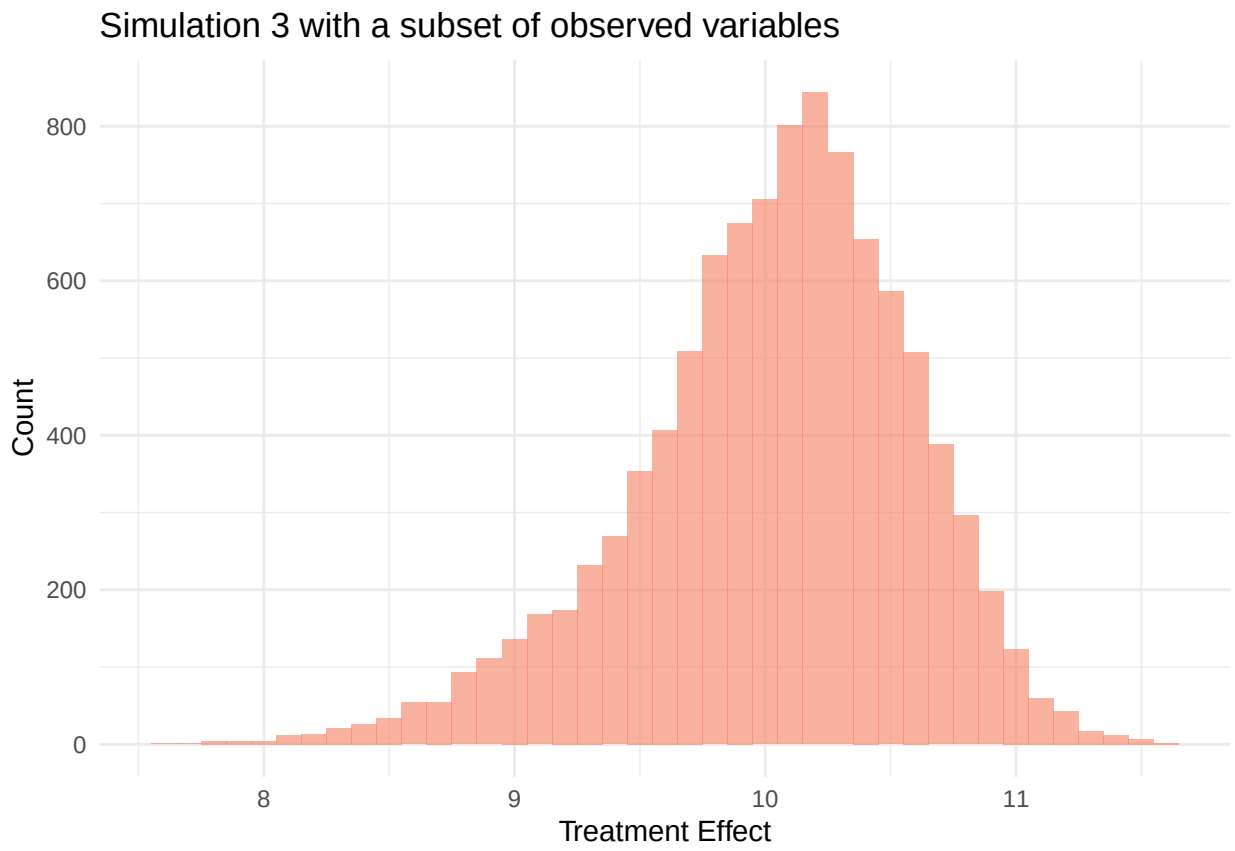
```
## Warning: Removed 1 rows containing missing values (geom_tile).
```



As the third test, let's try to use only the most important variables (over the median):

```
important_var_3 <- which(variable_importance(cf3.with_ort) >= median(variable_importance(cf3.with_ort)))
```

```
# Estimate causal forest
start_time_6 <- Sys.time() #Recording the running time

#Fitting the model
cf3.important <- grf::causal_forest(cbind(X, Z, C)[, important_var_3], y_3, w_3, orthog.boosting = TRUE)


end_time_6 <- Sys.time()

#Predicted values

pred_tau_3.important <- predict(cf3.important, estimate.variance = TRUE)

plot_pred_tau_6 <- ggplot(data = as.data.frame(pred_tau_3.important$predictions),
       aes(x = (pred_tau_3.important$predictions))) +
  geom_histogram(fill="#f68060", alpha=.6, binwidth = 0.1) +
  labs(title ="Simulation 3 with a subset of observed variables",
       x = "Treatment Effect", y = "Count") +
  theme_minimal()

plot_pred_tau_6
```



Runing time:

```
start_time_6 - end_time_6
```

```
## Time difference of -51.46338 secs
```

RMSE:

```
rmse_3.important <- fun.rmse(predicted = pred_tau_3.important$predictions, true = tau)

rmse_3.important
```

```
## [1] 0.5526428
```

Coverage:

```
coverage_3.important <- fun.coverage(pred_tau_3.important, tau)

coverage_3.important
```

```
## [1] 0.9972
```

Estimated ATE:

```r
ATE_est_3.important <- average_treatment_effect(cf3.important)

ATE_est_3.important
```

```
##   estimate    std.err
## 10.0280011  0.2138981
```

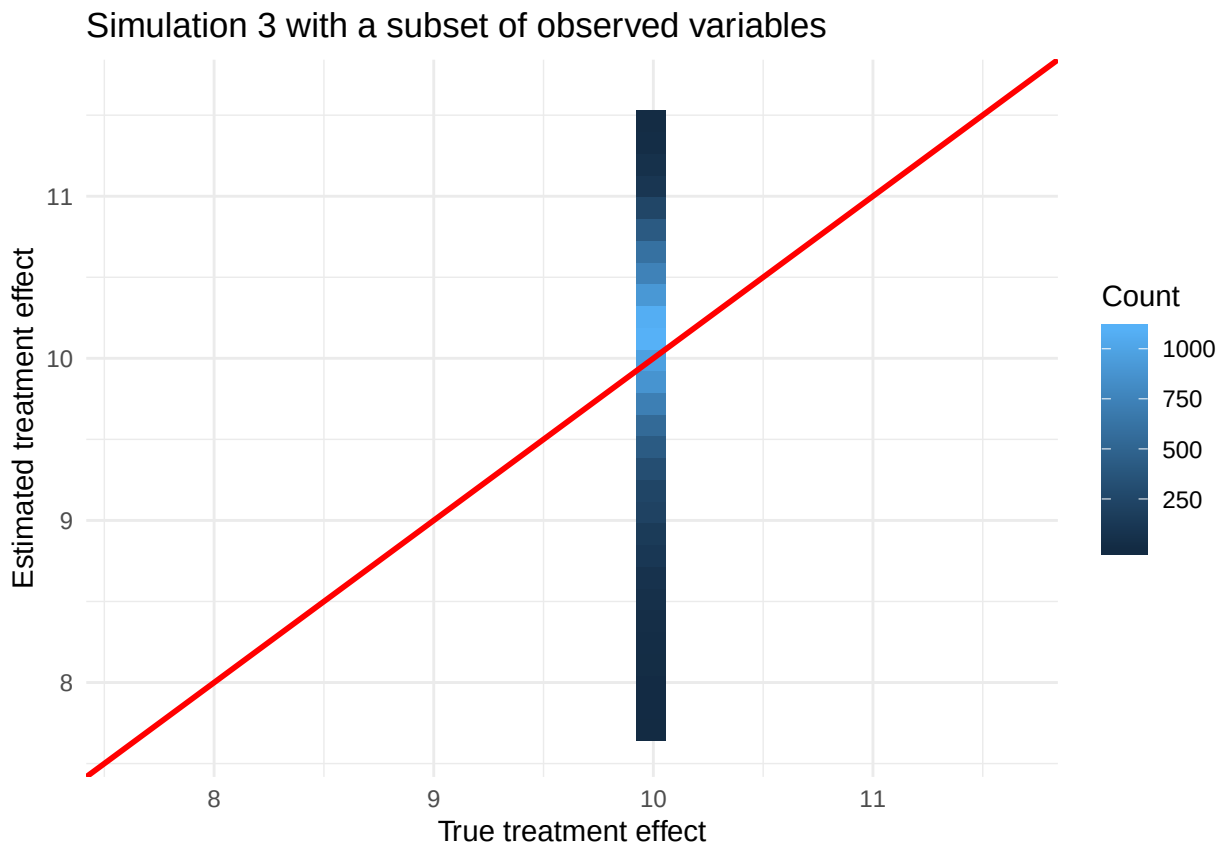
The *proportional difference in the ATE estimates* for the whole population:

```r
fun.diff_ATE(ATE_est = ATE_est_3.important, ATE_true = tau)
```

```
## [1] "Proportional mean of the differences:"
##     estimate
## -0.002800106
## [1] "Proportional mean of the differences, 95 % confidence intervals:"
##    estimate    estimate
## -0.02418992  0.01858971
```

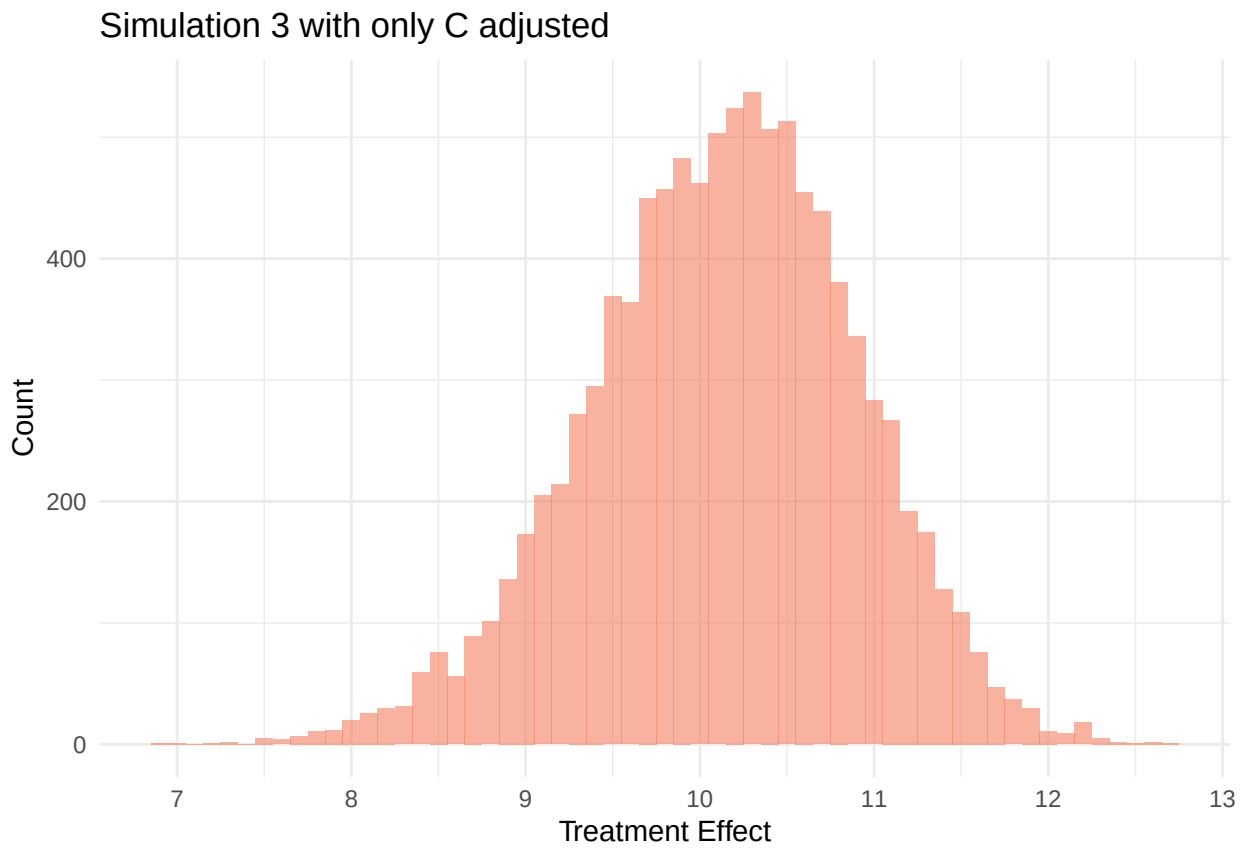
```r
true_vs_pred_3.important <- as.data.frame(cbind(tau, pred_tau_3.important$predictions))

colnames(true_vs_pred_3.important) <- c("tau", "pred_tau")

true_vs_pred_3.important$tau <- true_vs_pred_3.important$tau + rnorm(n, 0 , sd = 0.0001)

plot6 <- ggplot(data = true_vs_pred_3.important, aes(x = tau, y = pred_tau)) +
  geom_bin2d() +
  geom_abline(slope = 1, intercept = 0, colour = "red", size = 1) +
  theme_minimal() +
  labs(title = "Simulation 3 with a subset of observed variables",
       x = "True treatment effect", y = "Estimated treatment effect", fill = "Count") +
  lims(x = c(min(true_vs_pred_3.important$pred_tau), max(true_vs_pred_3.important$pred_tau)),
       y = c(min(true_vs_pred_3.important$pred_tau), max(true_vs_pred_3.important$pred_tau)))

plot6
```

```
## Warning: Removed 1 rows containing missing values (geom_tile).
```



As the last test, let's test the backdoor criterion (adjusting only for $C \in \mathbb{S}$):

```
# Estimate causal forest
start_time_7 <- Sys.time() #Recording the running time

#Fitting the model
cf3.bd <- grf::causal_forest(C, y_3, w_3, orthog.boosting = TRUE)


end_time_7 <- Sys.time()

#Predicted values

pred_tau_3.bd <- predict(cf3.bd, estimate.variance = TRUE)

plot_pred_tau_7 <- ggplot(data = as.data.frame(pred_tau_3.bd$predictions),
       aes(x = (pred_tau_3.bd$predictions))) +
  geom_histogram(fill="#f68060", alpha=.6, binwidth = 0.1) +
  labs(title ="Simulation 3 with only C adjusted",
       x = "Treatment Effect", y = "Count") +
  theme_minimal()

plot_pred_tau_7
```



Runing time:

```
start_time_7 - end_time_7
```

```
## Time difference of -39.08847 secs
```

RMSE:

```
rmse_3.bd <- fun.rmse(predicted = pred_tau_3.bd$predictions, true = tau)

rmse_3.bd
```

```
## [1] 0.7784988
```

Coverage:

```
coverage_3.bd <- fun.coverage(pred_tau_3.bd, tau)

coverage_3.bd
```

```
## [1] 0.9815
```

Estimated ATE:

```r
ATE_est_3.bd <- average_treatment_effect(cf3.bd)

ATE_est_3.bd
```

```
##   estimate    std.err
## 10.1332955  0.2297555
```

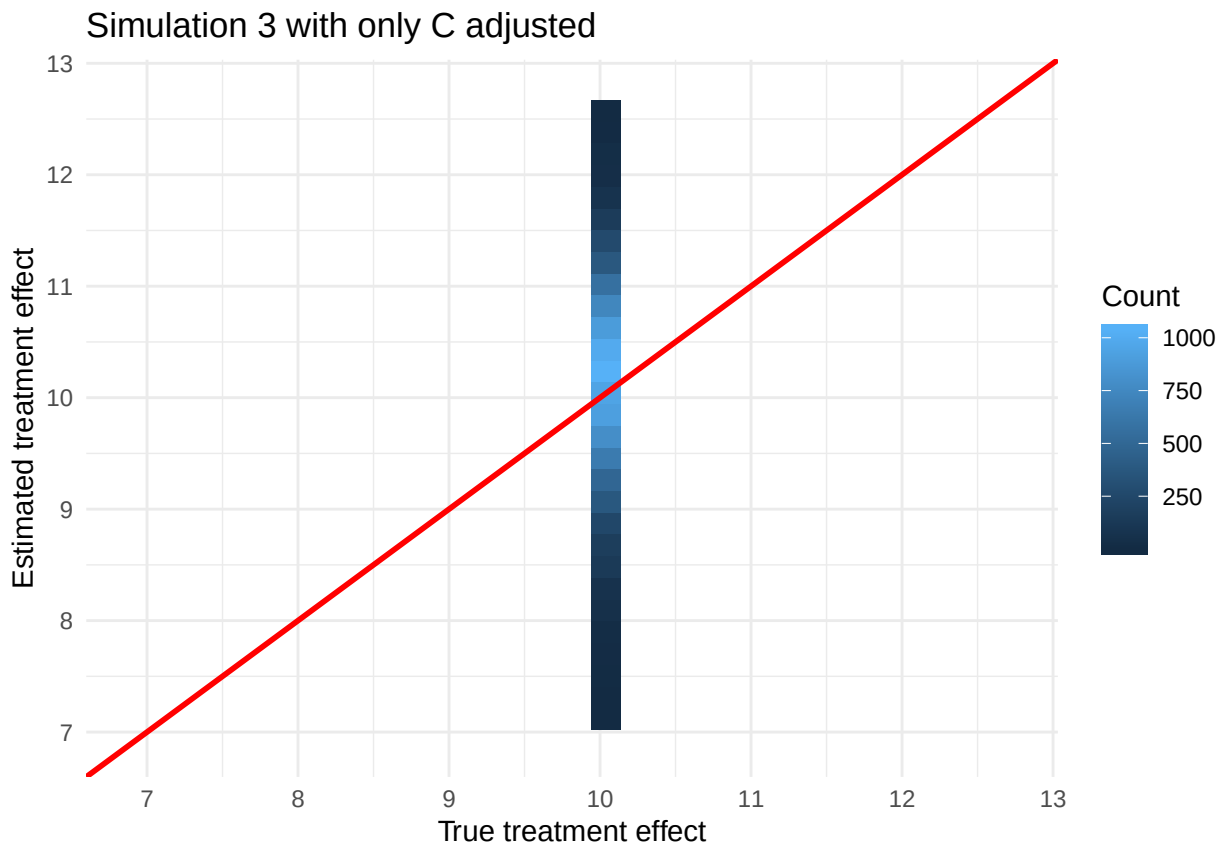
The *proportional difference in the ATE estimates* for the whole population:

```r
fun.diff_ATE(ATE_est = ATE_est_3.bd, ATE_true = tau)
```

```
## [1] "Proportional mean of the differences:"
##    estimate
## -0.01332955
## [1] "Proportional mean of the differences, 95 % confidence intervals:"
##     estimate     estimate
## -0.036305105  0.009646001
```

```r
true_vs_pred_3.bd <- as.data.frame(cbind(tau, pred_tau_3.bd$predictions))

colnames(true_vs_pred_3.bd) <- c("tau", "pred_tau")

true_vs_pred_3.bd$tau <- true_vs_pred_3.bd$tau + rnorm(n, 0 , sd = 0.0001)

plot7 <- ggplot(data = true_vs_pred_3.bd, aes(x = tau, y = pred_tau)) +
  geom_bin2d() +
  geom_abline(slope = 1, intercept = 0, colour = "red", size = 1) +
  theme_minimal() +
  labs(title = "Simulation 3 with only C adjusted",
       x = "True treatment effect", y = "Estimated treatment effect", fill = "Count") +
  lims(x = c(min(true_vs_pred_3.bd$pred_tau), max(true_vs_pred_3.bd$pred_tau)),
       y = c(min(true_vs_pred_3.bd$pred_tau), max(true_vs_pred_3.bd$pred_tau)))

plot7
```

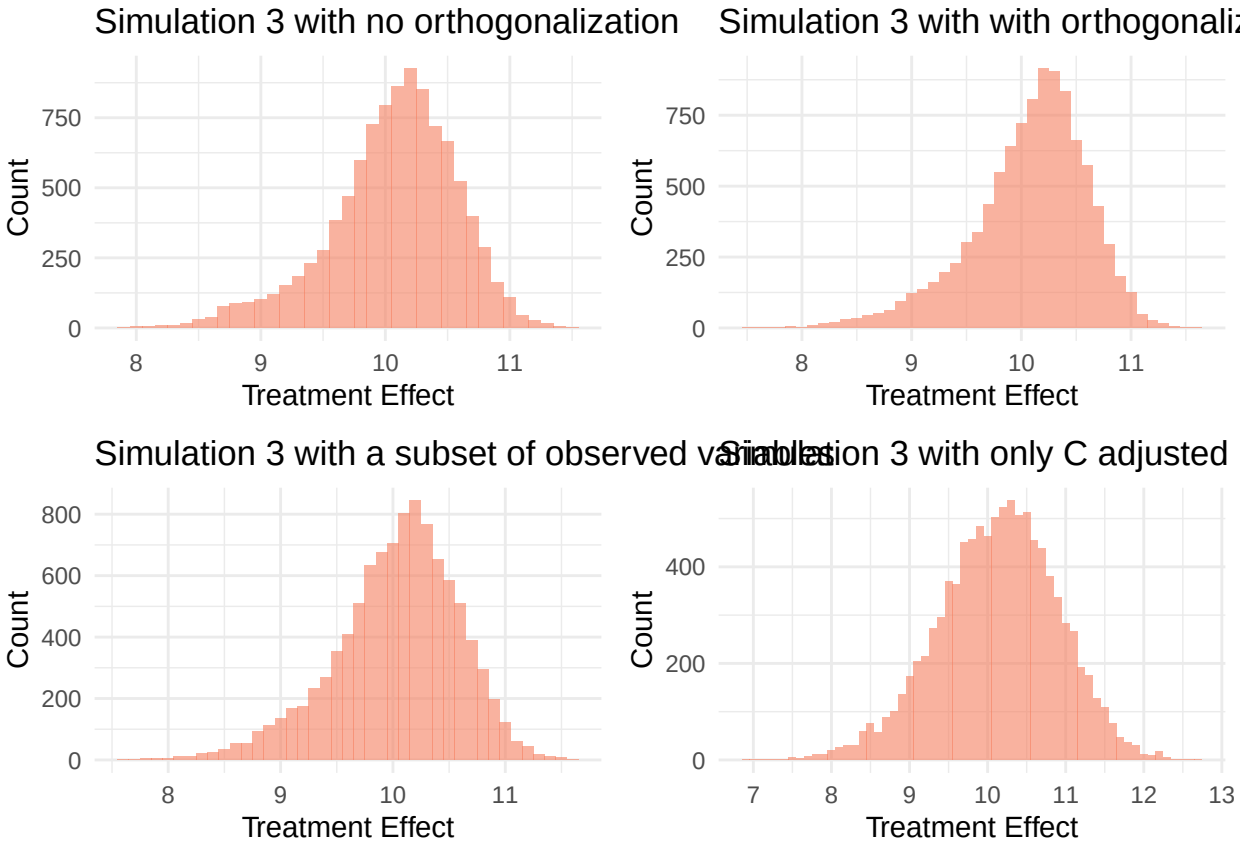
```
## Warning: Removed 1 rows containing missing values (geom_tile).
```



Summary

Predicted $\hat{\tau}s$:

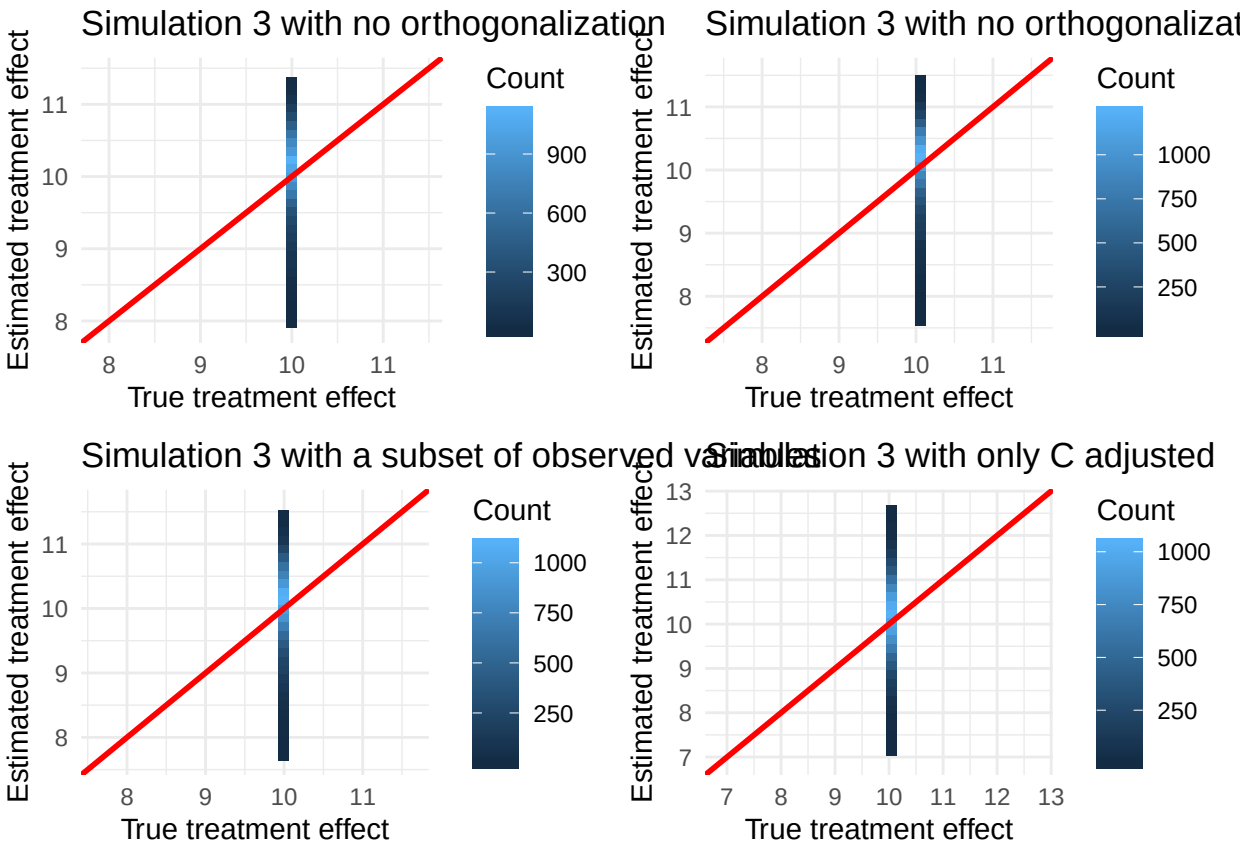
```
grid.arrange(plot_pred_tau_4, plot_pred_tau_5, plot_pred_tau_6, plot_pred_tau_7, nrow = 2)
```



True τ vs. predicted $\hat{\tau}$:

```
grid.arrange(plot4, plot5, plot6, plot7, nrow = 2)
```

```
## Warning: Removed 1 rows containing missing values (geom_tile).

## Warning: Removed 1 rows containing missing values (geom_tile).

## Warning: Removed 1 rows containing missing values (geom_tile).

## Warning: Removed 1 rows containing missing values (geom_tile).
```



RMSEs:

```
rmse_3 <- as.data.frame(c(rmse_3.no_ort, rmse_3.with_ort, rmse_3.important, rmse_3.bd))

rownames(rmse_3) <- c("No orthogonalization", "With orthogonalization",
                      "Subset of observed variables", "Only confounders adjusted")

colnames(rmse_3) <- c("RMSE")

knitr::kable(rmse_3, escape = FALSE)
```

	RMSE
No orthogonalization	0.5160037
With orthogonalization	0.5342760
Subset of observed variables	0.5526428
Only confounders adjusted	0.7784988

Coverages:

```
coverage_3 <- as.data.frame(c(coverage_3.no_ort, coverage_3.with_ort,
                              coverage_3.important, coverage_3.bd))

rownames(coverage_3) <- c("No orthogonalization", "With orthogonalization",
                      "Subset of observed variables", "Only confounders adjusted")
```

```r
colnames(coverage_3) <- c("Coverage")

knitr::kable(coverage_3, escape = FALSE)
```

	Coverage
No orthogonalization	0.9976
With orthogonalization	0.9966
Subset of observed variables	0.9972
Only confounders adjusted	0.9815

LaTeX:

```r
knitr::kable(rmse_3, format = "latex",escape = FALSE)
```

	RMSE
No orthogonalization	0.5160037
With orthogonalization	0.5342760
Subset of observed variables	0.5526428
Only confounders adjusted	0.7784988

```r
knitr::kable(coverage_3, format = "latex",escape = FALSE)
```

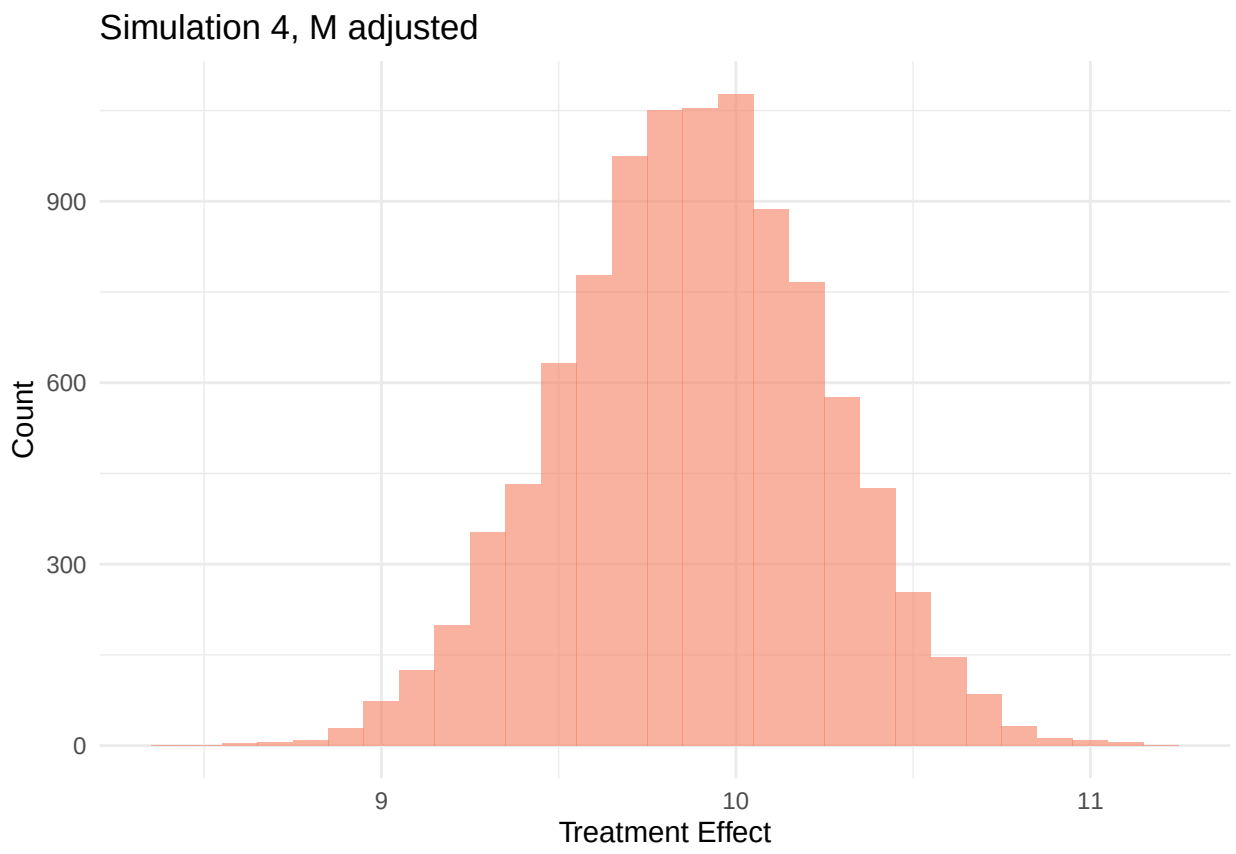
	Coverage
No orthogonalization	0.9976
With orthogonalization	0.9966
Subset of observed variables	0.9972
Only confounders adjusted	0.9815

3.4 Constant Treatment Effect with Confounded Assignment, Including Covariates Affecting to Output and a Pure Local M-Structure

Let's adjust the collider M (incorrectly) and then leave it out from the adjusted set $\mathbb{S}$ ($X \cup Z \cup C \cup M \in \mathbb{S}$).

```r
# Estimate causal forest
start_time_8 <- Sys.time() #Recording the running time

#Fitting the model
cf4.with_M <- grf::causal_forest(cbind(X, Z, C, m), y_4, w_4, orthog.boosting = TRUE)


end_time_8 <- Sys.time()

#Predicted values

pred_tau_4.with_M <- predict(cf4.with_M, estimate.variance = TRUE)

plot_pred_tau_8 <- ggplot(data = as.data.frame(pred_tau_4.with_M$predictions),
       aes(x = (pred_tau_4.with_M$predictions))) +
  geom_histogram(fill="#f68060", alpha=.6, binwidth = 0.1) +
```

```
  labs(title ="Simulation 4, M adjusted",
       x = "Treatment Effect", y = "Count") +
  theme_minimal()

plot_pred_tau_8
```



Runing time:

```
start_time_8 - end_time_8
```

```
## Time difference of -1.218691 mins
```

RMSE:

```
rmse_4.with_M <- fun.rmse(predicted = pred_tau_4.with_M$predictions, true = tau)

rmse_4.with_M
```

```
## [1] 0.3844006
```

Coverage:

```r
coverage_4.with_M <- fun.coverage(pred_tau_4.with_M, tau)

coverage_4.with_M
```

```
## [1] 0.9998
```

Estimated ATE:

```r
ATE_est_4.with_M <- average_treatment_effect(cf4.with_M)

ATE_est_4.with_M
```

```
##  estimate   std.err
## 9.8655052 0.2150415
```

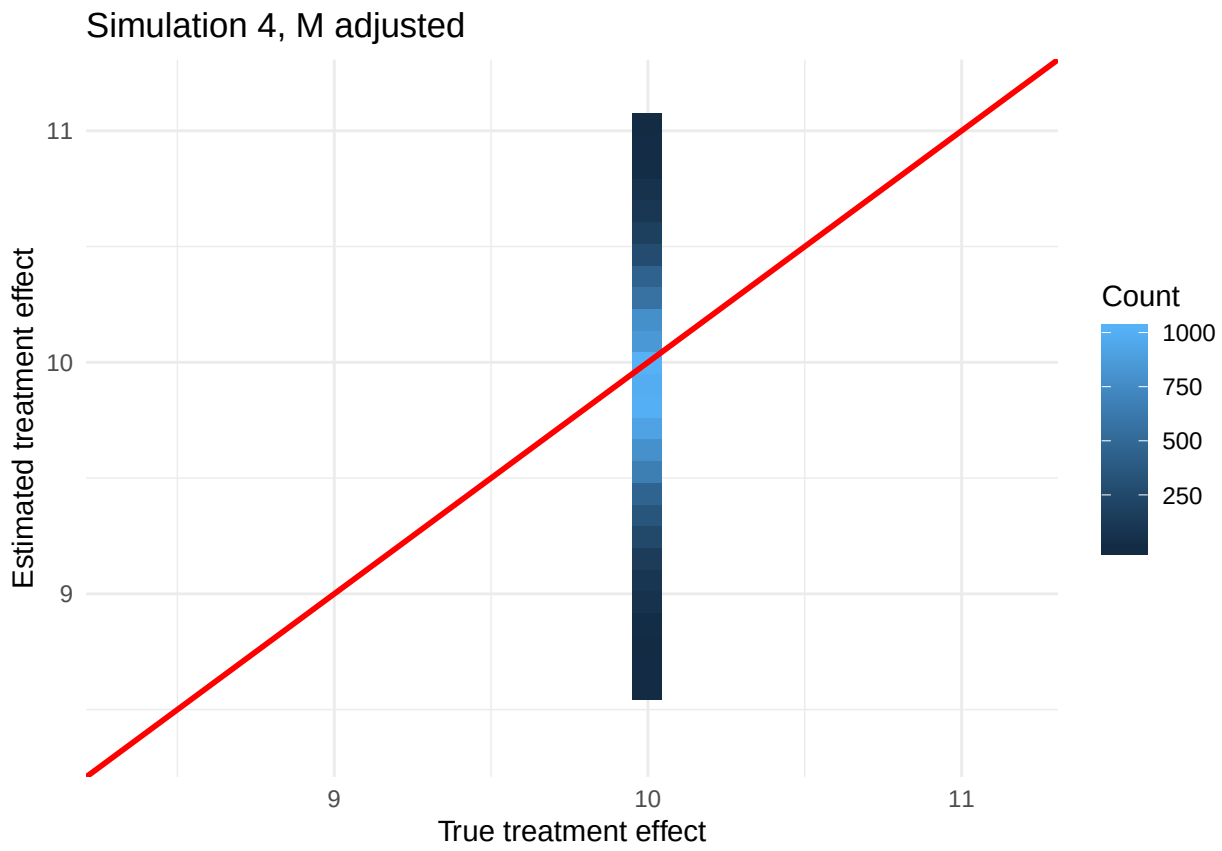
The *proportional difference in the ATE estimates* for the whole population:

```r
fun.diff_ATE(ATE_est = ATE_est_4.with_M, ATE_true = tau)
```

```
## [1] "Proportional mean of the differences:"
##   estimate
## 0.01344948
## [1] "Proportional mean of the differences, 95 % confidence intervals:"
##     estimate     estimate
## -0.008054669  0.034953622
```

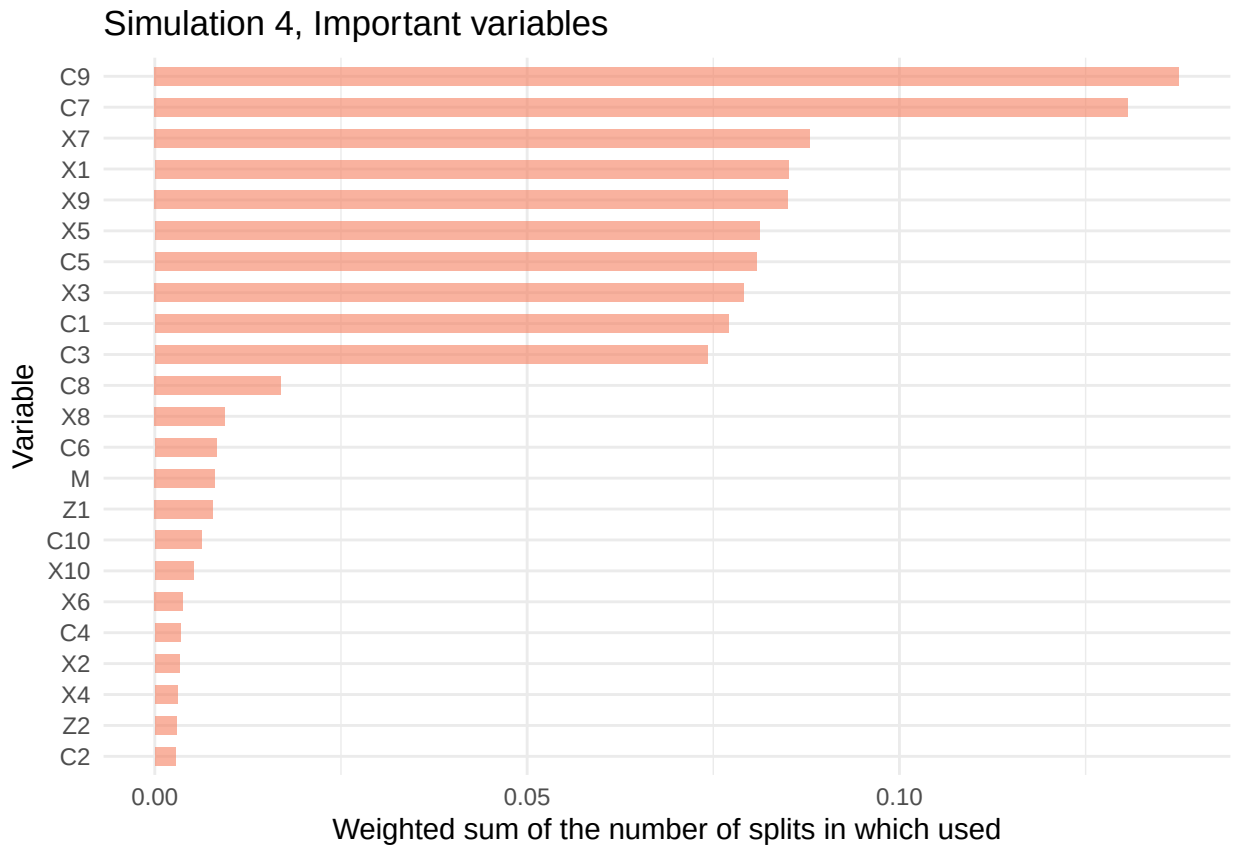
```r
true_vs_pred_4.with_M <- as.data.frame(cbind(tau, pred_tau_4.with_M$predictions))

colnames(true_vs_pred_4.with_M) <- c("tau", "pred_tau")

true_vs_pred_4.with_M$tau <- true_vs_pred_4.with_M$tau + rnorm(n, 0 , sd = 0.0001)

plot8 <- ggplot(data = true_vs_pred_4.with_M, aes(x = tau, y = pred_tau)) +
  geom_bin2d() +
  geom_abline(slope = 1, intercept = 0, colour = "red", size = 1) +
  theme_minimal() +
  labs(title = "Simulation 4, M adjusted",
       x = "True treatment effect", y = "Estimated treatment effect", fill = "Count") +
  lims(x = c(min(true_vs_pred_4.with_M$pred_tau), max(true_vs_pred_4.with_M$pred_tau)),
       y = c(min(true_vs_pred_4.with_M$pred_tau), max(true_vs_pred_4.with_M$pred_tau)))

plot8
```

```
## Warning: Removed 1 rows containing missing values (geom_tile).
```



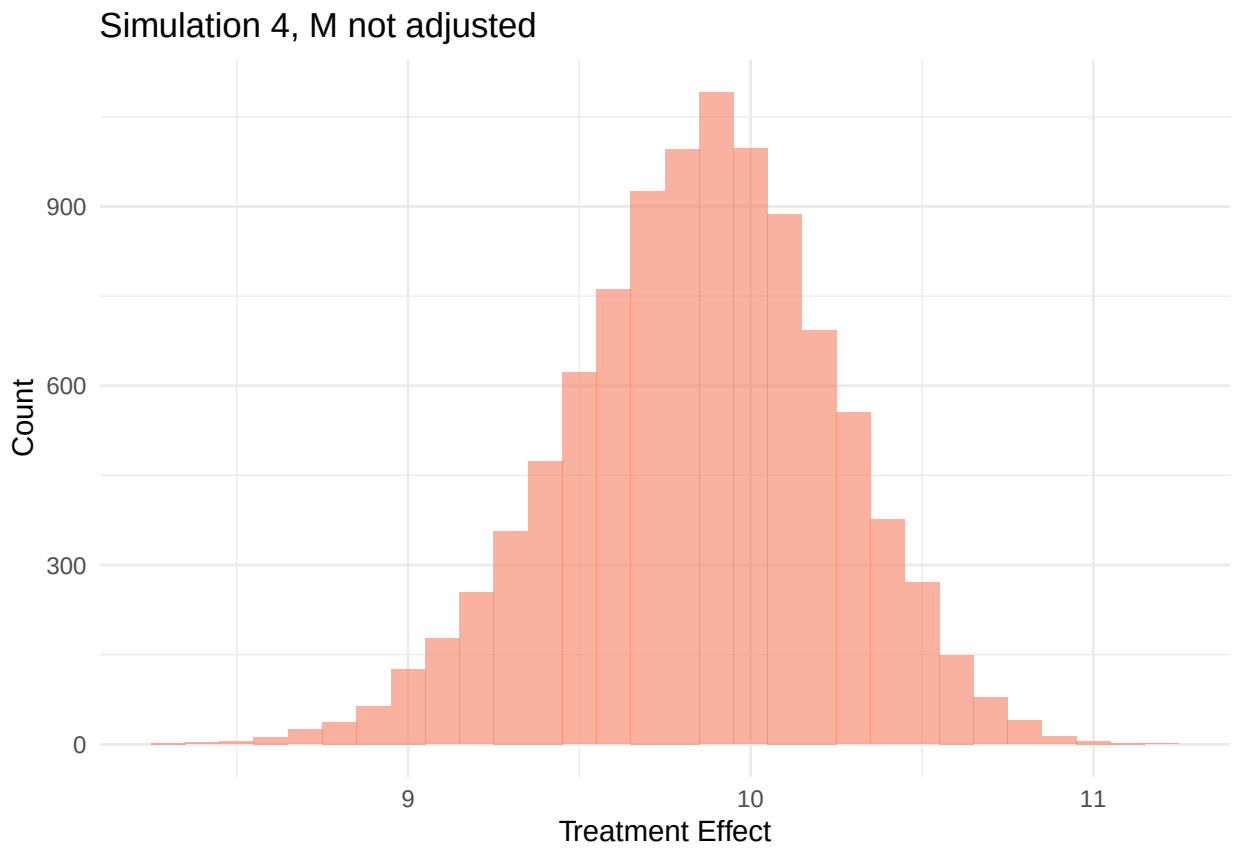
How "important" variable M was?

```
names4.with_M.X <- paste("X", 1:10, sep = "")

names4.with_M.Z <- paste("Z", 1:2, sep = "")

names4.with_M.C <- paste("C", 1:10, sep = "")

names4.with_M <- list(names4.with_M.X, names4.with_M.Z, names4.with_M.C, c("M"))

important_var_4.with_M <- as.data.frame(grf::variable_importance(cf4.with_M), row.names = Reduce(c, nam

important_var_4.with_M %>%
  mutate(name = fct_reorder(row.names(important_var_4.with_M), V1)) %>%
  ggplot(aes(x = name, y = V1)) +
  geom_bar(stat="identity", fill="#f68060", alpha=.6, width=.6) +
  coord_flip() +
  labs(title ="Simulation 4, Important variables", x = "Variable", y = "Weighted sum of the number of sp
  theme_minimal()
```



The following estimation is done by excluding the M from the ajusted set $\mathbb{S}$ ($X \cup Z \cup C \in \mathbb{S}$):

```
# Estimate causal forest
start_time_9 <- Sys.time() #Recording the running time

#Fitting the model
cf4.no_M <- grf::causal_forest(cbind(X, Z, C), y_4, w_4, orthog.boosting = TRUE)


end_time_9 <- Sys.time()

#Predicted values

pred_tau_4.no_M <- predict(cf4.no_M, estimate.variance = TRUE)

plot_pred_tau_9 <- ggplot(data = as.data.frame(pred_tau_4.no_M$predictions),
                          aes(x = (pred_tau_4.no_M$predictions))) +
  geom_histogram(binwidth = 0.1, fill="#f68060", alpha=.6) +
  labs(title ="Simulation 4, M not adjusted",
       x = "Treatment Effect", y = "Count") +
  theme_minimal()

plot_pred_tau_9
```



Runing time:

```
start_time_9 - end_time_9
```

```
## Time difference of -1.202667 mins
```

RMSE:

```
rmse_4.no_M <- fun.rmse(predicted = pred_tau_4.no_M$predictions, true = tau)

rmse_4.no_M
```

```
## [1] 0.4186501
```

Coverage:

```
coverage_4.no_M <- fun.coverage(pred_tau_4.no_M, tau)

coverage_4.no_M
```

```
## [1] 0.9994
```

Estimated ATE:

```r
ATE_est_4.no_M <- average_treatment_effect(cf4.no_M)

ATE_est_4.no_M
```

```
##  estimate   std.err
## 9.8462622 0.2171587
```

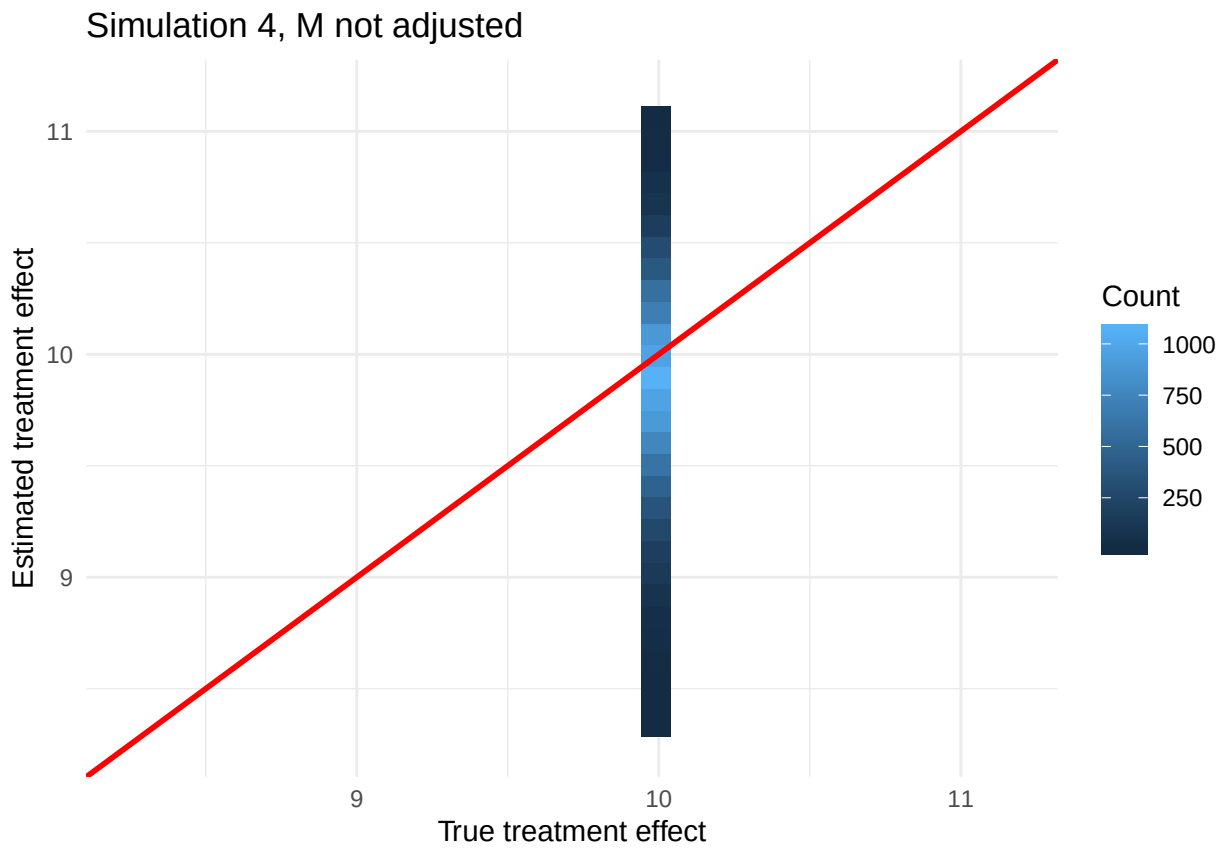
The *proportional difference in the ATE estimates* for the whole population:

```r
fun.diff_ATE(ATE_est = ATE_est_4.no_M, ATE_true = tau)
```

```
## [1] "Proportional mean of the differences:"
##   estimate
## 0.01537378
## [1] "Proportional mean of the differences, 95 % confidence intervals:"
##     estimate     estimate
## -0.006342088  0.037089648
```

```r
true_vs_pred_4.no_M <- as.data.frame(cbind(tau, pred_tau_4.no_M$predictions))

colnames(true_vs_pred_4.no_M) <- c("tau", "pred_tau")

true_vs_pred_4.no_M$tau <- true_vs_pred_4.no_M$tau + rnorm(n, 0 , sd = 0.0001)

plot9 <- ggplot(data = true_vs_pred_4.no_M, aes(x = tau, y = pred_tau)) +
  geom_bin2d() +
  geom_abline(slope = 1, intercept = 0, colour = "red", size = 1) +
  theme_minimal() +
  labs(title = "Simulation 4, M not adjusted",
       x = "True treatment effect", y = "Estimated treatment effect", fill = "Count") +
  lims(x = c(min(true_vs_pred_4.no_M$pred_tau), max(true_vs_pred_4.no_M$pred_tau)),
       y = c(min(true_vs_pred_4.no_M$pred_tau), max(true_vs_pred_4.no_M$pred_tau)))

plot9
```

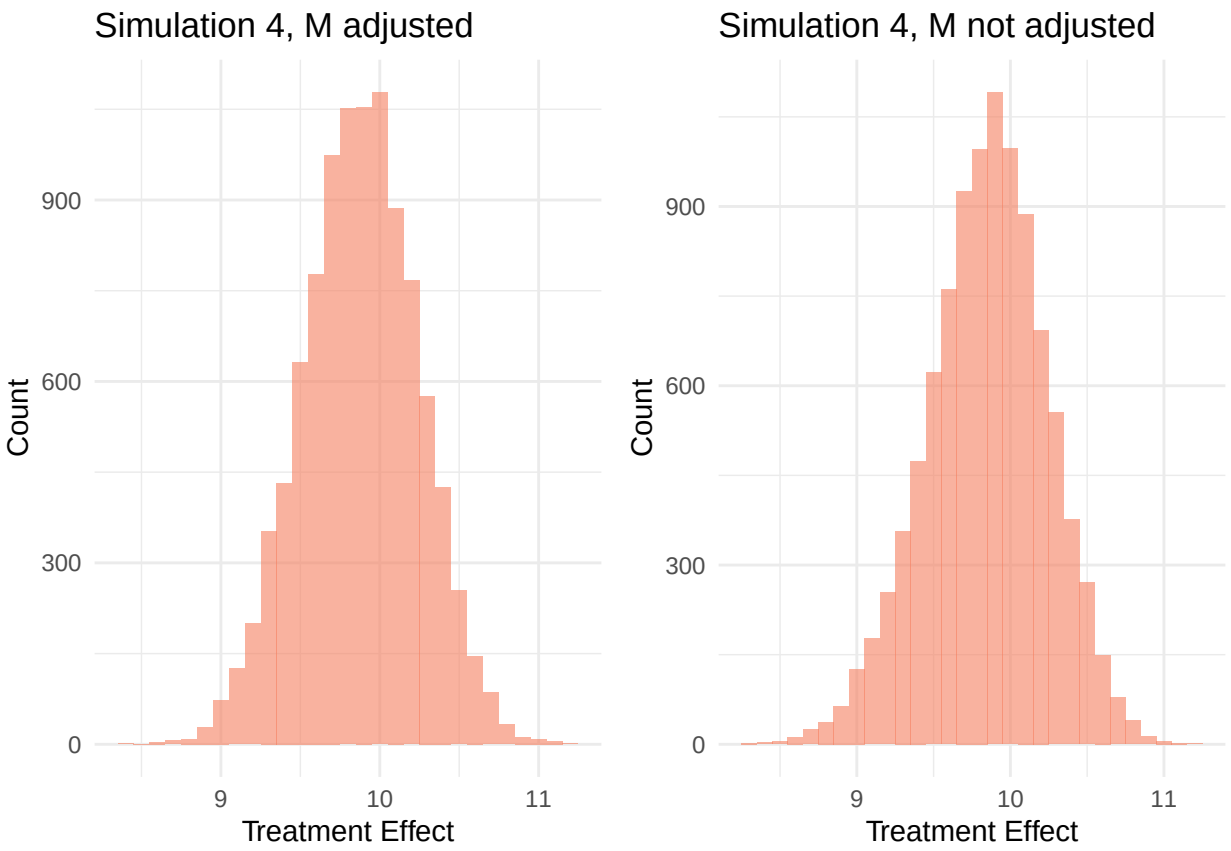
```
## Warning: Removed 1 rows containing missing values (geom_tile).
```



Summary

Predicted $\hat{\tau}s$:

```
grid.arrange(plot_pred_tau_8, plot_pred_tau_9, nrow = 1)
```

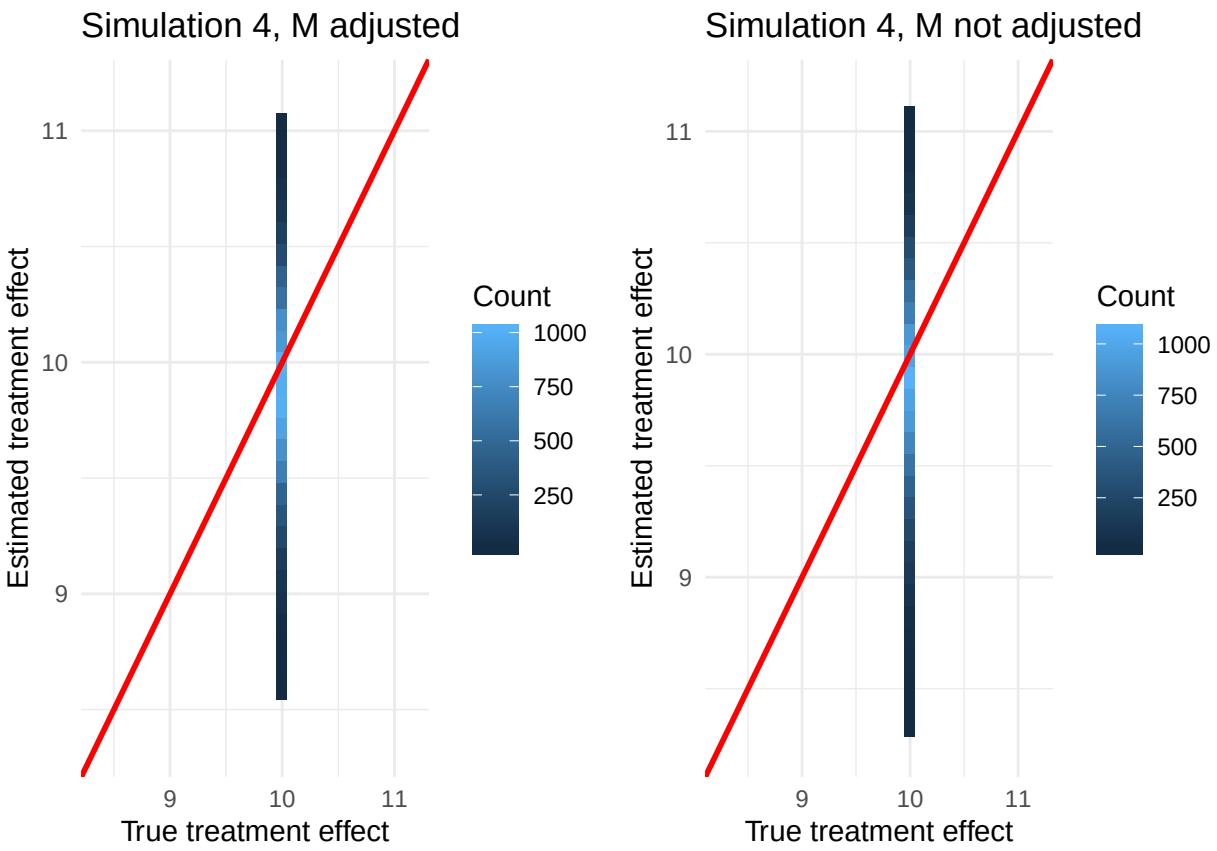


True τ vs. predicted $\hat{\tau}$:

```
grid.arrange(plot8, plot9, nrow = 1)
```

```
## Warning: Removed 1 rows containing missing values (geom_tile).
```

```
## Warning: Removed 1 rows containing missing values (geom_tile).
```



RMSEs:

```
rmse_4 <- as.data.frame(c(rmse_4.with_M, rmse_4.no_M))

rownames(rmse_4) <- c("$M$ adjusted", "$M$ not adjusted")

colnames(rmse_4) <- c("RMSE")

knitr::kable(rmse_4, escape = FALSE)
```

	RMSE
M adjusted	0.3844006
M not adjusted	0.4186501

Coverages:

```
coverage_4 <- as.data.frame(c(coverage_4.with_M, coverage_4.no_M))

rownames(coverage_4) <-  c("$M$ adjusted", "$M$ not adjusted")

colnames(coverage_4) <- c("Coverage")

knitr::kable(coverage_4, escape = FALSE)
```

	Coverage
M adjusted	0.9998
M not adjusted	0.9994

LaTeX:

```
knitr::kable(rmse_4, format = "latex",escape = FALSE)
```

	RMSE
M adjusted	0.3844006
M not adjusted	0.4186501

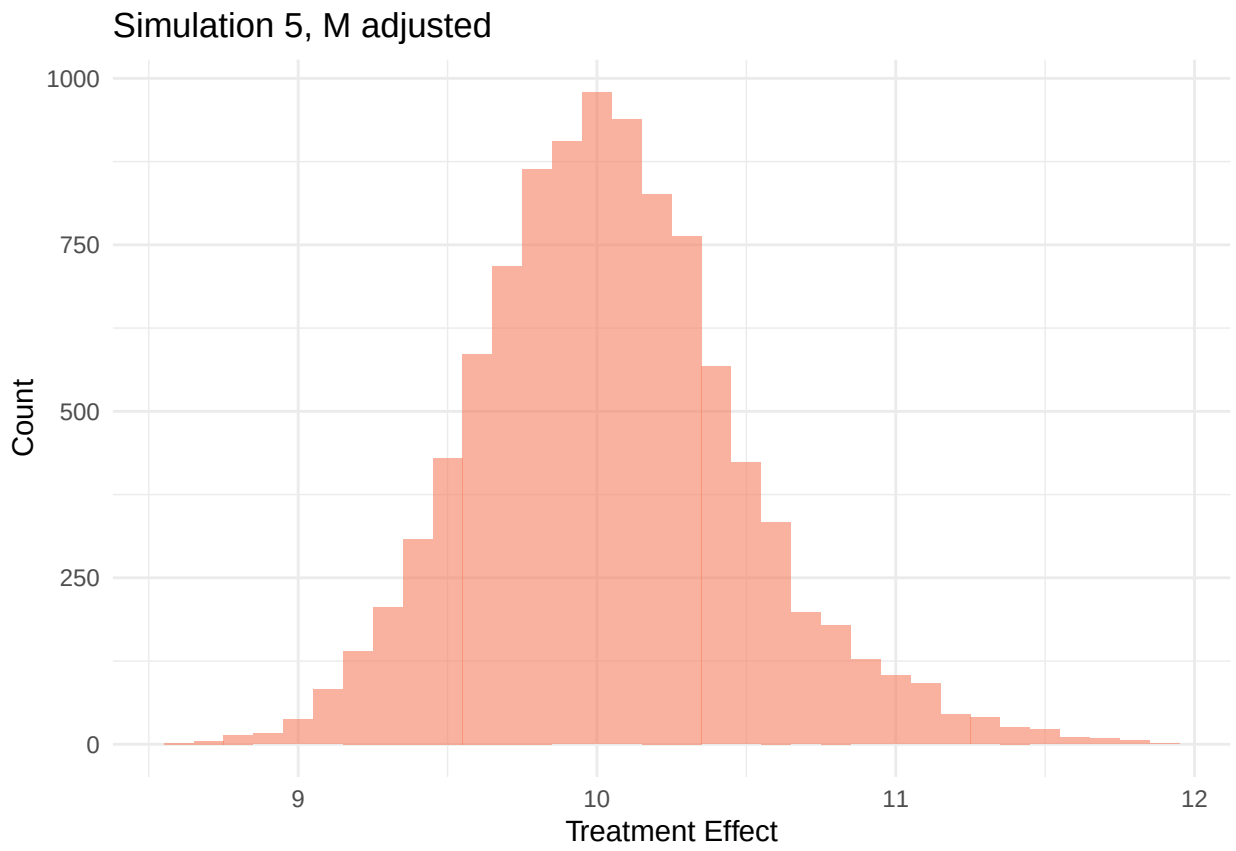
```
knitr::kable(coverage_4, format = "latex",escape = FALSE)
```

	Coverage
M adjusted	0.9998
M not adjusted	0.9994

3.5 Constant Treatment Effect with Confounded Assignment, Including Covariates Affecting to Output and a Impure Local M-Structure

The following estimation with $X \cup Z \cup C \cup M \in \mathbb{S}$:

```
# Estimate causal forest
start_time_10 <- Sys.time() #Recording the running time

#Fitting the model
cf5.with_M <- grf::causal_forest(cbind(X, Z, C, m), y_5, w_5, orthog.boosting = TRUE)


end_time_10 <- Sys.time()

#Predicted values

pred_tau_5.with_M <- predict(cf5.with_M, estimate.variance = TRUE)

plot_pred_tau_10 <- ggplot(data = as.data.frame(pred_tau_5.with_M$predictions),
       aes(x = (pred_tau_5.with_M$predictions))) +
  geom_histogram(binwidth = 0.1, fill="#f68060", alpha=.6) +
  labs(title ="Simulation 5, M adjusted",
       x = "Treatment Effect", y = "Count") +
  theme_minimal()

plot_pred_tau_10
```



Runing time:

```
start_time_10 - end_time_10
```

```
## Time difference of -1.234566 mins
```

RMSE:

```
rmse_5.with_M <- fun.rmse(predicted = pred_tau_5.with_M$predictions, true = tau)

rmse_5.with_M
```

```
## [1] 0.4490187
```

Coverage:

```
coverage_5.with_M <- fun.coverage(pred_tau_5.with_M, tau)

coverage_5.with_M
```

```
## [1] 0.9996
```

Estimated ATE:

```r
ATE_est_5.with_M <- average_treatment_effect(cf5.with_M)

ATE_est_5.with_M
```

```
##   estimate    std.err
## 10.0542646  0.2154077
```

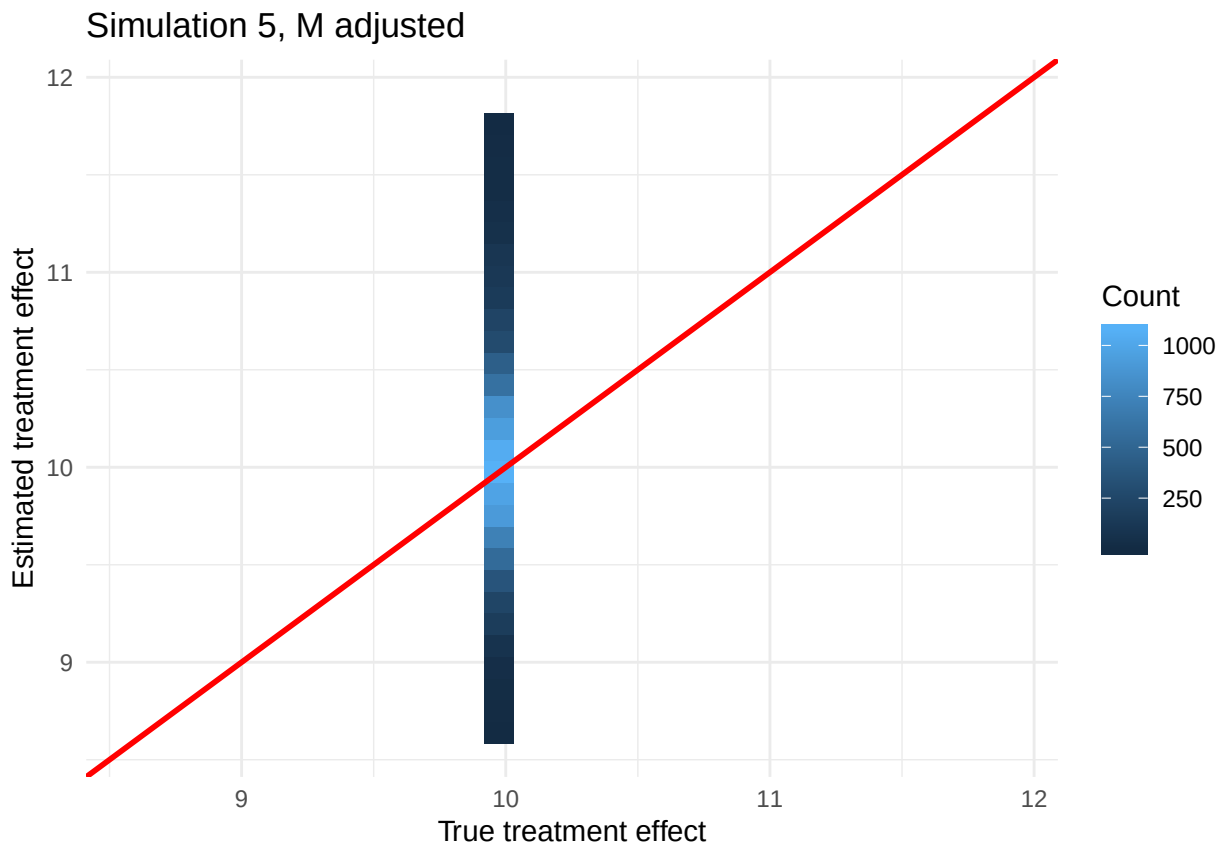
The *proportional difference in the ATE estimates* for the whole population:

```r
fun.diff_ATE(ATE_est = ATE_est_5.with_M, ATE_true = tau)
```

```
## [1] "Proportional mean of the differences:"
##     estimate
## -0.005426457
## [1] "Proportional mean of the differences, 95 % confidence intervals:"
##    estimate    estimate
## -0.02696723  0.01611432
```

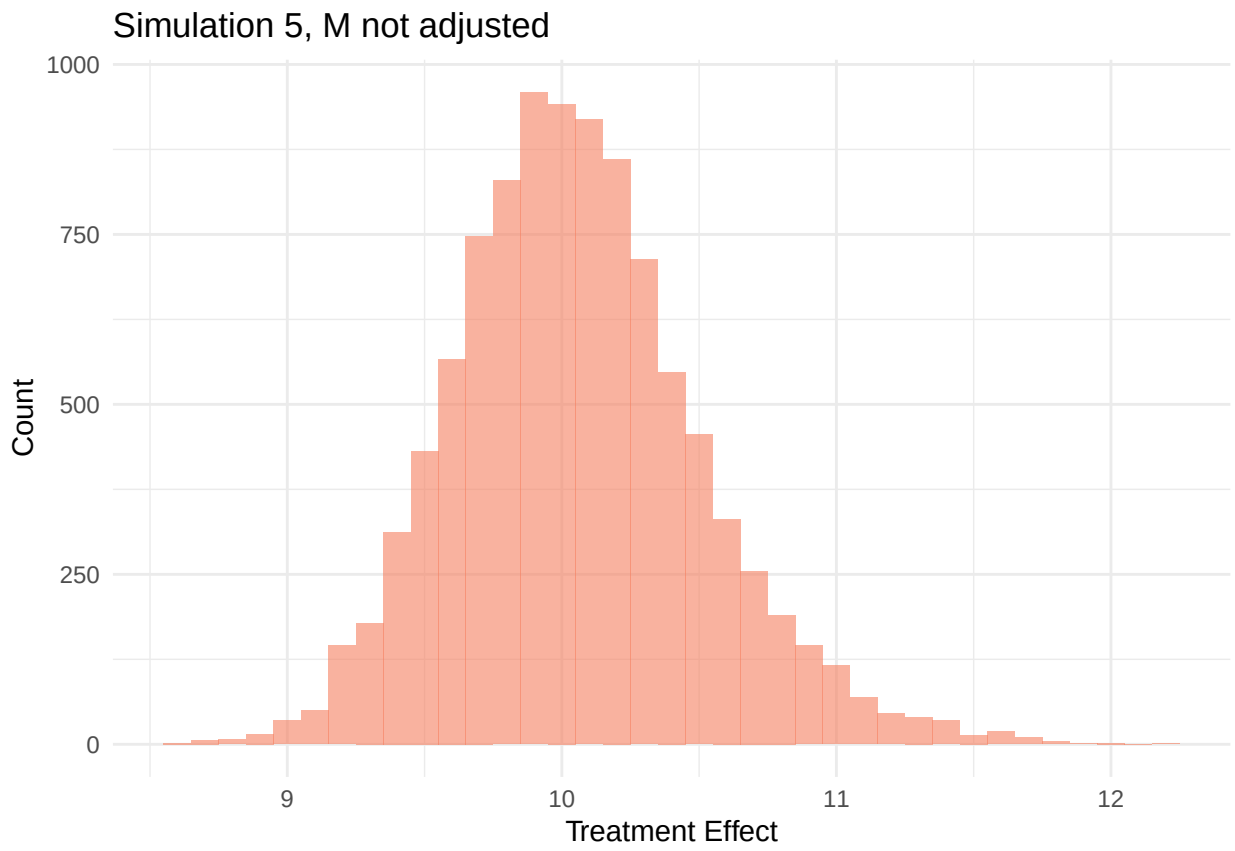
```r
true_vs_pred_5.with_M <- as.data.frame(cbind(tau, pred_tau_5.with_M$predictions))

colnames(true_vs_pred_5.with_M) <- c("tau", "pred_tau")

true_vs_pred_5.with_M$tau <- true_vs_pred_5.with_M$tau + rnorm(n, 0 , sd = 0.0001)

plot10 <- ggplot(data = true_vs_pred_5.with_M, aes(x = tau, y = pred_tau)) +
  geom_bin2d() +
  geom_abline(slope = 1, intercept = 0, colour = "red", size = 1) +
  theme_minimal() +
  labs(title = "Simulation 5, M adjusted",
       x = "True treatment effect", y = "Estimated treatment effect", fill = "Count") +
  lims(x = c(min(true_vs_pred_5.with_M$pred_tau), max(true_vs_pred_5.with_M$pred_tau)),
       y = c(min(true_vs_pred_5.with_M$pred_tau), max(true_vs_pred_5.with_M$pred_tau)))

plot10
```

```
## Warning: Removed 1 rows containing missing values (geom_tile).
```



The following estimation with $M \notin \mathbb{S}$ ($X \cup Z \cup C \in \mathbb{S}$):

```
# Estimate causal forest
start_time_11 <- Sys.time() #Recording the running time

#Fitting the model
cf5.no_M <- grf::causal_forest(cbind(X, Z, C), y_5, w_5, orthog.boosting = TRUE)


end_time_11 <- Sys.time()

#Predicted values

pred_tau_5.no_M <- predict(cf5.no_M, estimate.variance = TRUE)

plot_pred_tau_11 <- ggplot(data = as.data.frame(pred_tau_5.no_M$predictions),
       aes(x = (pred_tau_5.no_M$predictions))) +
  geom_histogram(binwidth = 0.1, fill="#f68060", alpha=.6) +
  labs(title ="Simulation 5, M not adjusted",
       x = "Treatment Effect", y = "Count") +
  theme_minimal()

plot_pred_tau_11
```



Runing time:

```
start_time_11 - end_time_11
```

```
## Time difference of -1.262756 mins
```

RMSE:

```
rmse_5.no_M <- fun.rmse(predicted = pred_tau_5.no_M$predictions, true = tau)

rmse_5.no_M
```

```
## [1] 0.4513016
```

Coverage:

```
coverage_5.no_M <- fun.coverage(pred_tau_5.no_M, tau)

coverage_5.no_M
```

```
## [1] 0.9986
```

Estimated ATE:

```r
ATE_est_5.no_M <- average_treatment_effect(cf5.no_M)

ATE_est_5.no_M
```

```
##   estimate    std.err
## 10.0524049  0.2171627
```

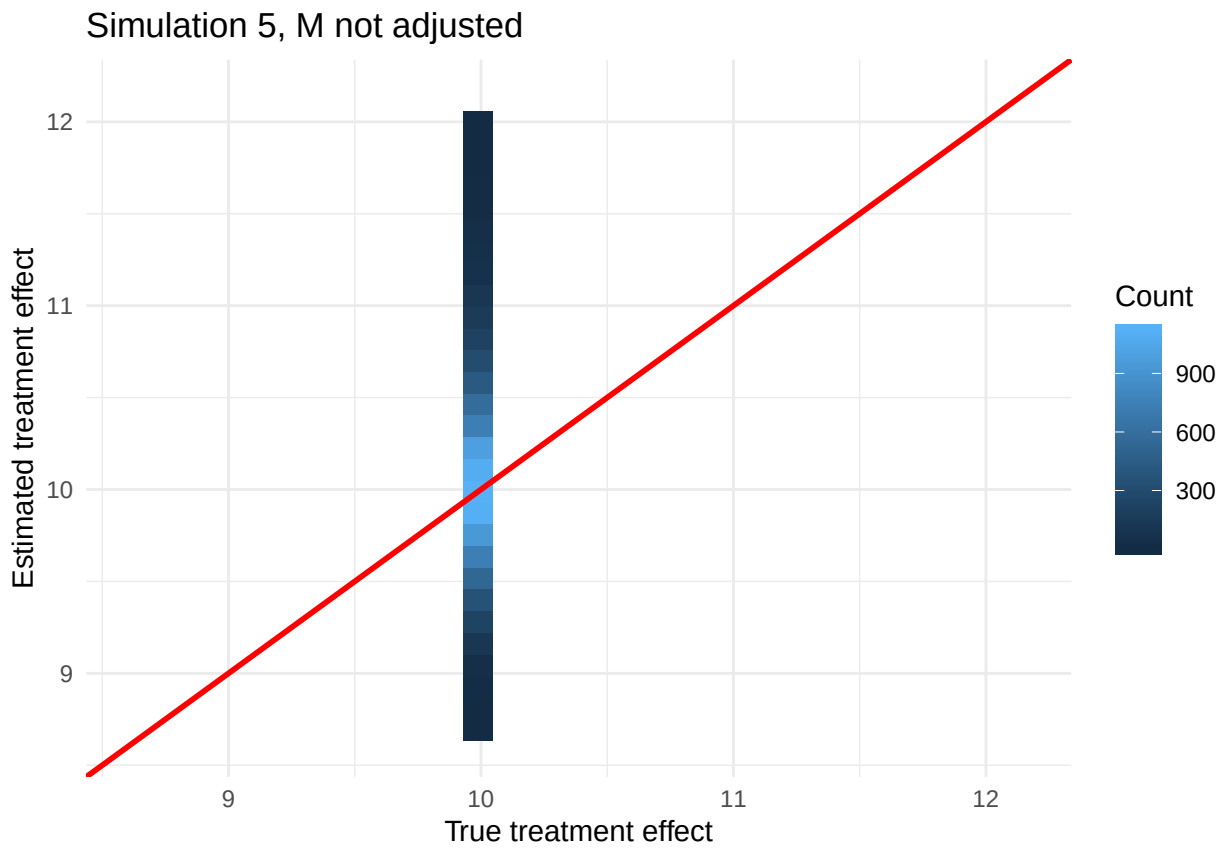
The *proportional difference in the ATE estimates* for the whole population:

```r
fun.diff_ATE(ATE_est = ATE_est_5.no_M, ATE_true = tau)
```

```
## [1] "Proportional mean of the differences:"
##     estimate
## -0.005240495
## [1] "Proportional mean of the differences, 95 % confidence intervals:"
##    estimate    estimate
## -0.02695676  0.01647577
```

```r
true_vs_pred_5.no_M <- as.data.frame(cbind(tau, pred_tau_5.no_M$predictions))

colnames(true_vs_pred_5.no_M) <- c("tau", "pred_tau")

true_vs_pred_5.no_M$tau <- true_vs_pred_5.no_M$tau + rnorm(n, 0 , sd = 0.0001)

plot11 <- ggplot(data = true_vs_pred_5.no_M, aes(x = tau, y = pred_tau)) +
  geom_bin2d() +
  geom_abline(slope = 1, intercept = 0, colour = "red", size = 1) +
  theme_minimal() +
  labs(title = "Simulation 5, M not adjusted",
       x = "True treatment effect", y = "Estimated treatment effect", fill = "Count") +
  lims(x = c(min(true_vs_pred_5.no_M$pred_tau), max(true_vs_pred_5.no_M$pred_tau)),
       y = c(min(true_vs_pred_5.no_M$pred_tau), max(true_vs_pred_5.no_M$pred_tau)))

plot11
```

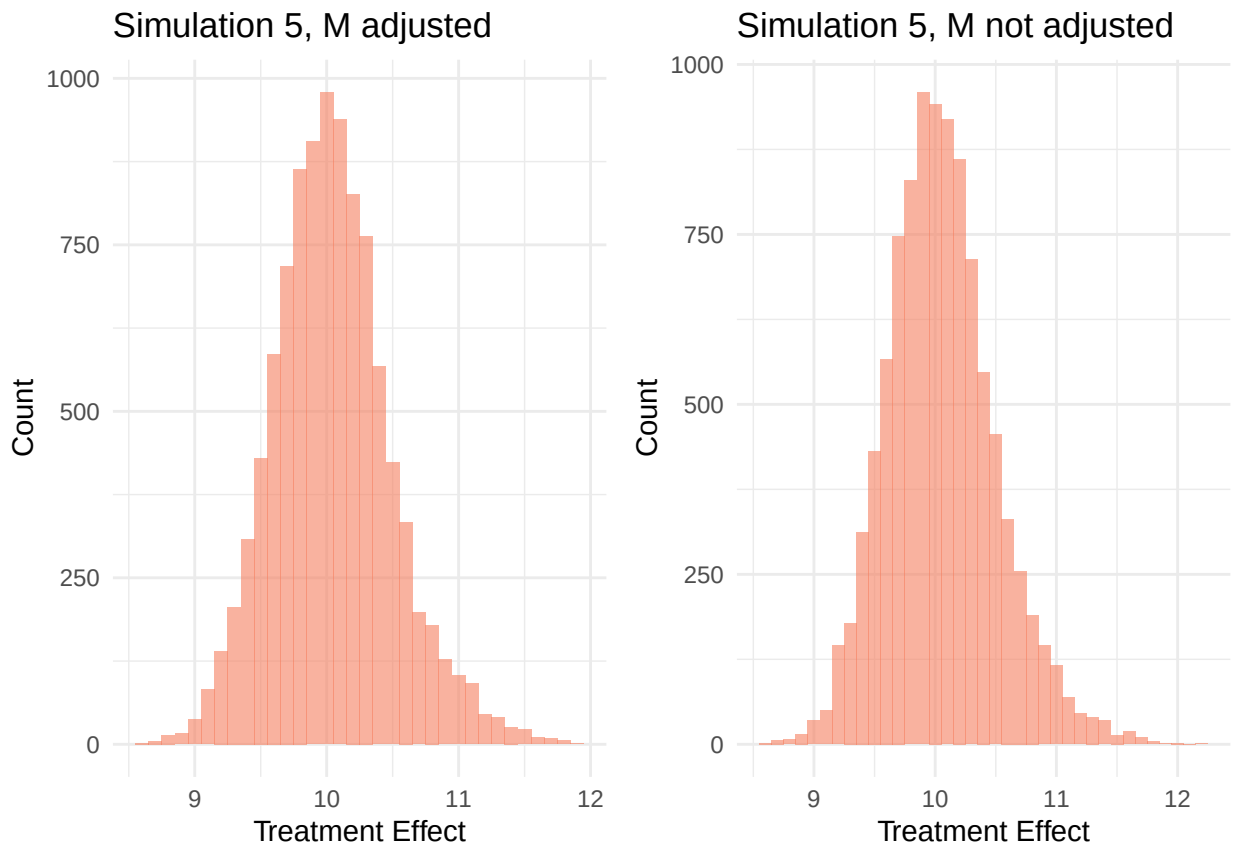
```
## Warning: Removed 1 rows containing missing values (geom_tile).
```



Summary

Predicted $\hat{\tau}s$:

```
grid.arrange(plot_pred_tau_10, plot_pred_tau_11, nrow = 1)
```

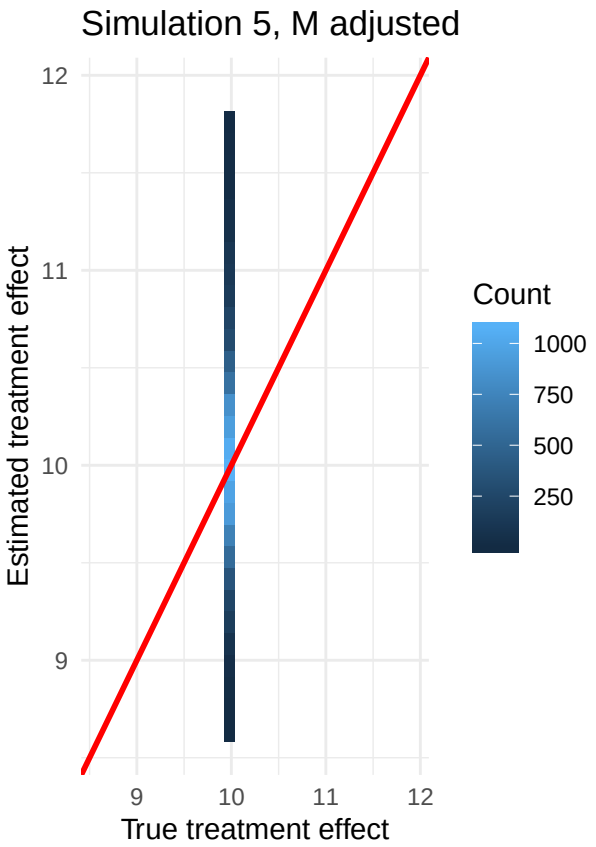


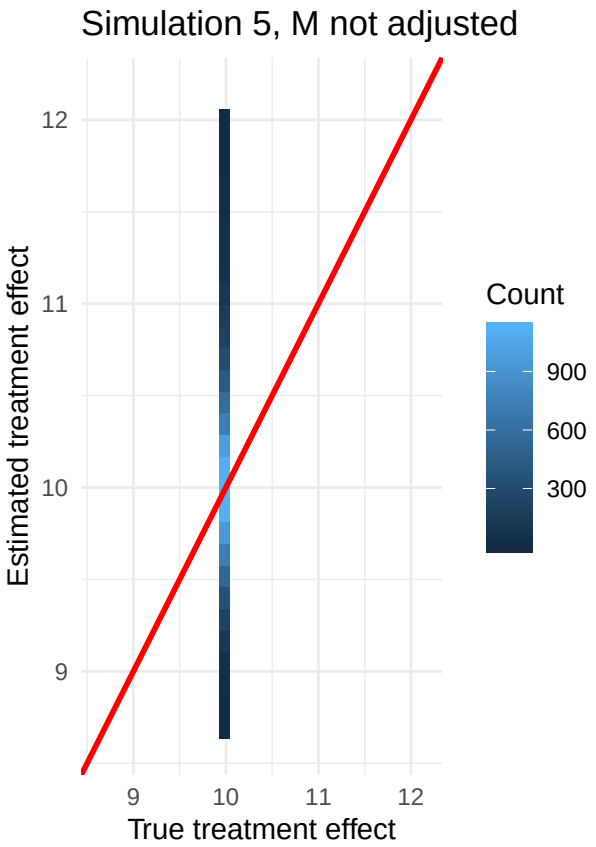
True τ vs. predicted $\hat{\tau}$:

```
grid.arrange(plot10, plot11, nrow = 1)
```

```
## Warning: Removed 1 rows containing missing values (geom_tile).
```

```
## Warning: Removed 1 rows containing missing values (geom_tile).
```





RMSEs:

```
rmse_5 <- as.data.frame(c(rmse_5.with_M, rmse_5.no_M))

rownames(rmse_5) <- c("$M$ adjusted", "$M$ not adjusted")

colnames(rmse_5) <- c("RMSE")

knitr::kable(rmse_5, escape = FALSE)
```

	RMSE
M adjusted	0.4490187
M not adjusted	0.4513016

Coverages:

```
coverage_5 <- as.data.frame(c(coverage_5.with_M, coverage_5.no_M))

rownames(coverage_5) <-  c("$M$ adjusted", "$M$ not adjusted")

colnames(coverage_5) <- c("Coverage")

knitr::kable(coverage_5, escape = FALSE)
```

	Coverage
M adjusted	0.9996
M not adjusted	0.9986

LaTeX:

```
knitr::kable(rmse_5, format = "latex",escape = FALSE)
```

	RMSE
M adjusted	0.4490187
M not adjusted	0.4513016

```
knitr::kable(coverage_5, format = "latex",escape = FALSE)
```

	Coverage
M adjusted	0.9996
M not adjusted	0.9986

3.6 Randomized Controlled Trial with Heterogeneous Treatment Effect, Including Covariates Affecting to Output

The first estimation for a heterogeneous treatment effect. At first, let's test with the full set of observed variables ($X \in \mathbb{S}$)::

```
# Estimate causal forest
start_time_12 <- Sys.time() #Recording the running time

#Fitting the model
cf6.full <- grf::causal_forest(X, y_6, w_1, orthog.boosting = FALSE)


end_time_12 <- Sys.time()

#Predicted values

pred_tau_6.full <- predict(cf6.full, estimate.variance = TRUE)

plot_pred_tau_12 <- cbind(pred_tau_6.full$predictions, tau_1) %>%
  as.data.frame() %>%
  dplyr::rename(Predicted = V1) %>%
  dplyr::rename(True = tau_1) %>%
  gather(key = "key", value = "value") %>%
  ggplot(aes(x = value, fill = key)) +
  geom_histogram(binwidth = 0.1, alpha=.6) +
  scale_fill_manual(name = "", values=c("#f68060", "#57b2eb")) +
  labs(title ="Simulation 6 with the whole set X",
       x = "Treatment Effect", y = "Count") +
  theme_minimal()

plot_pred_tau_12
```

Runing time:

```
start_time_12 - end_time_12
```

```
## Time difference of -35.61582 secs
```

RMSE:

```
rmse_6.full <- fun.rmse(predicted = pred_tau_6.full$predictions, true = tau_1)

rmse_6.full
```

```
## [1] 2.274664
```

Coverage:

```
coverage_6.full <- fun.coverage(pred_tau_6.full, tau_1)

coverage_6.full
```

```
## [1] 0.7734
```

Estimated ATE:

```r
ATE_est_6.full <- average_treatment_effect(cf6.full)

ATE_est_6.full
```

```
##  estimate   std.err
## 2.2208766 0.2069256
```

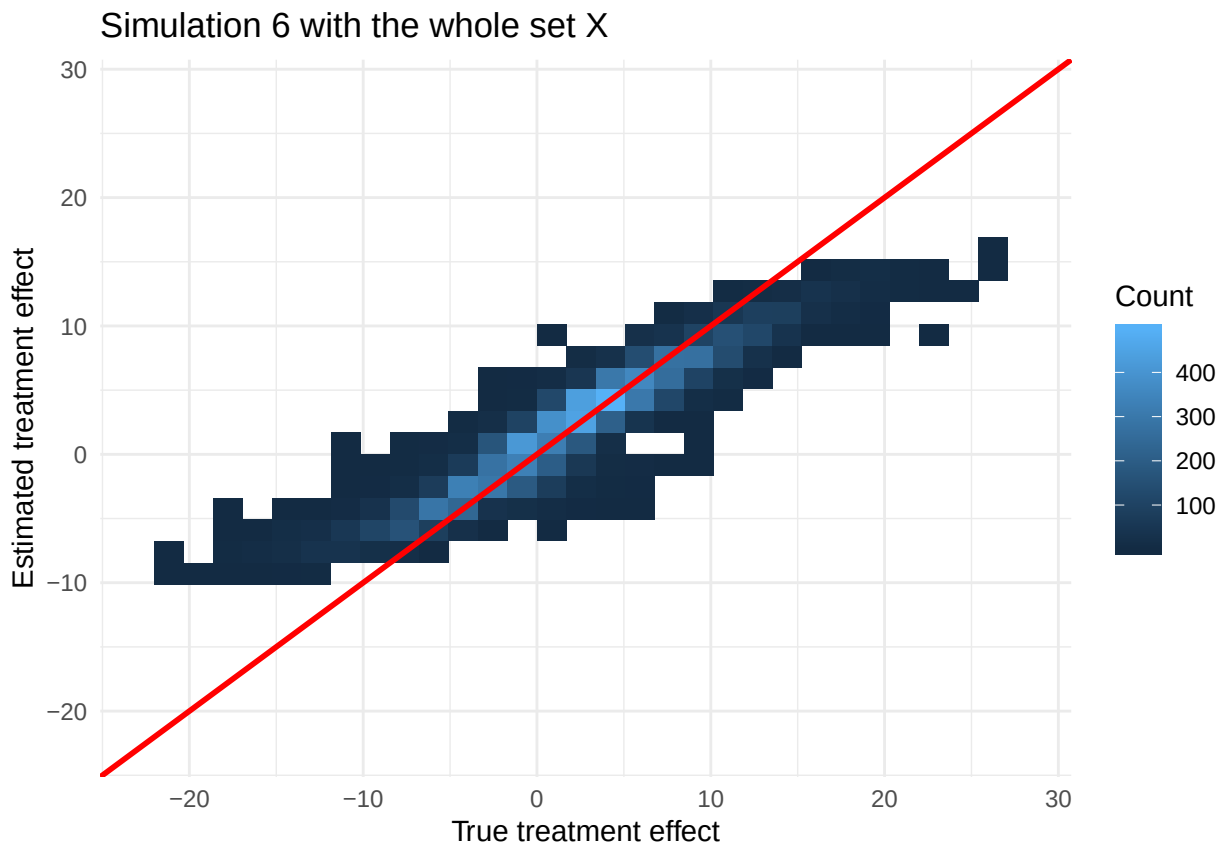
The *proportional difference in the ATE estimates* for the whole population:

```r
fun.diff_ATE(ATE_est = ATE_est_6.full, ATE_true = mean(tau_1))
```

```
## [1] "Proportional mean of the differences:"
##   estimate
## -0.1242247
## [1] "Proportional mean of the differences, 95 % confidence intervals:"
##    estimate    estimate
## -0.22897195 -0.01947736
```

```r
true_vs_pred_6.full <- as.data.frame(cbind(tau_1, pred_tau_6.full$predictions))

colnames(true_vs_pred_6.full) <- c("tau", "pred_tau")

plot12 <- ggplot(data = true_vs_pred_6.full, aes(x = tau, y = pred_tau)) +
  geom_bin2d() +
  geom_abline(slope = 1, intercept = 0, colour = "red", size = 1) +
  theme_minimal() +
  labs(title = "Simulation 6 with the whole set X",
       x = "True treatment effect", y = "Estimated treatment effect", fill = "Count") +
  lims(x = c(min(true_vs_pred_6.full), max(true_vs_pred_6.full)),
       y = c(min(true_vs_pred_6.full), max(true_vs_pred_6.full)))

plot12
```

```
## Warning: Removed 1 rows containing missing values (geom_tile).
```

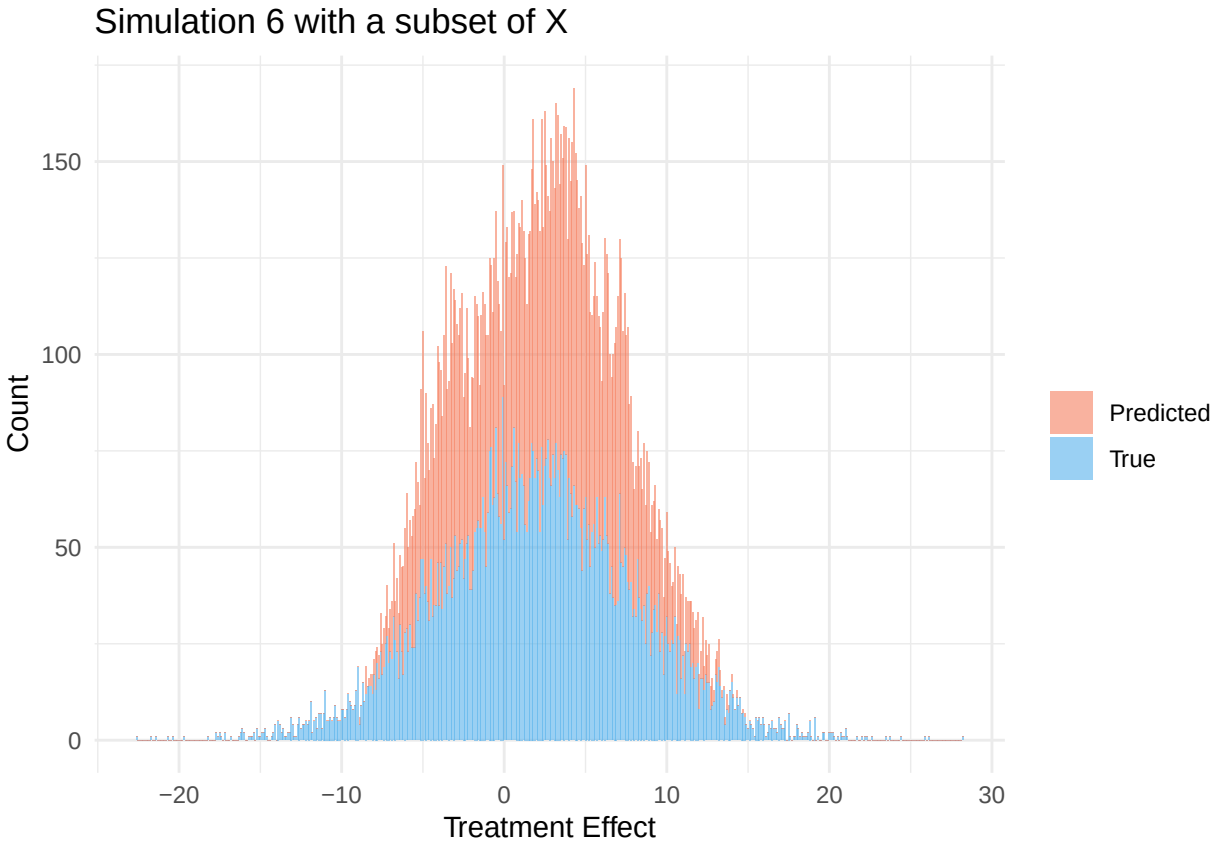


Let's use only the most important variables and see how it affects in the predictions:

```
important_var_6 <- which(variable_importance(cf6.full) >= median(variable_importance(cf6.full))) #Varia
```

```
# Estimate causal forest
start_time_13 <- Sys.time() #Recording the running time

#Fitting the model
cf6.important <- grf::causal_forest(X[, important_var_6], y_6, w_1, orthog.boosting = FALSE)


end_time_13 <- Sys.time()

#Predicted values

pred_tau_6.important <- predict(cf6.important, estimate.variance = TRUE)

plot_pred_tau_13 <- cbind(pred_tau_6.important$predictions, tau_1) %>%
  as.data.frame() %>%
  dplyr::rename(Predicted = V1) %>%
  dplyr::rename(True = tau_1) %>%
  gather(key = "key", value = "value") %>%
  ggplot(aes(x = value, fill = key)) +
  geom_histogram(binwidth = 0.1, alpha=.6) +
  scale_fill_manual(name = "", values=c("#f68060", "#57b2eb")) +
  labs(title ="Simulation 6 with a subset of X",
```

```
       x = "Treatment Effect", y = "Count") +
  theme_minimal()

plot_pred_tau_13
```



Runing time:

```
start_time_13 - end_time_13
```

```
## Time difference of -35.88995 secs
```

RMSE:

```
rmse_6.important <- fun.rmse(predicted = pred_tau_6.important$predictions, true = tau_1)
```

Coverage:

```
coverage_6.important <- fun.coverage(pred_tau_6.important, tau_1)

coverage_6.important
```

```
## [1] 0.6238
```

Estimated ATE:

```r
ATE_est_6.important <- average_treatment_effect(cf6.important)

ATE_est_6.important
```

```
##  estimate   std.err
## 2.2190782 0.2102267
```

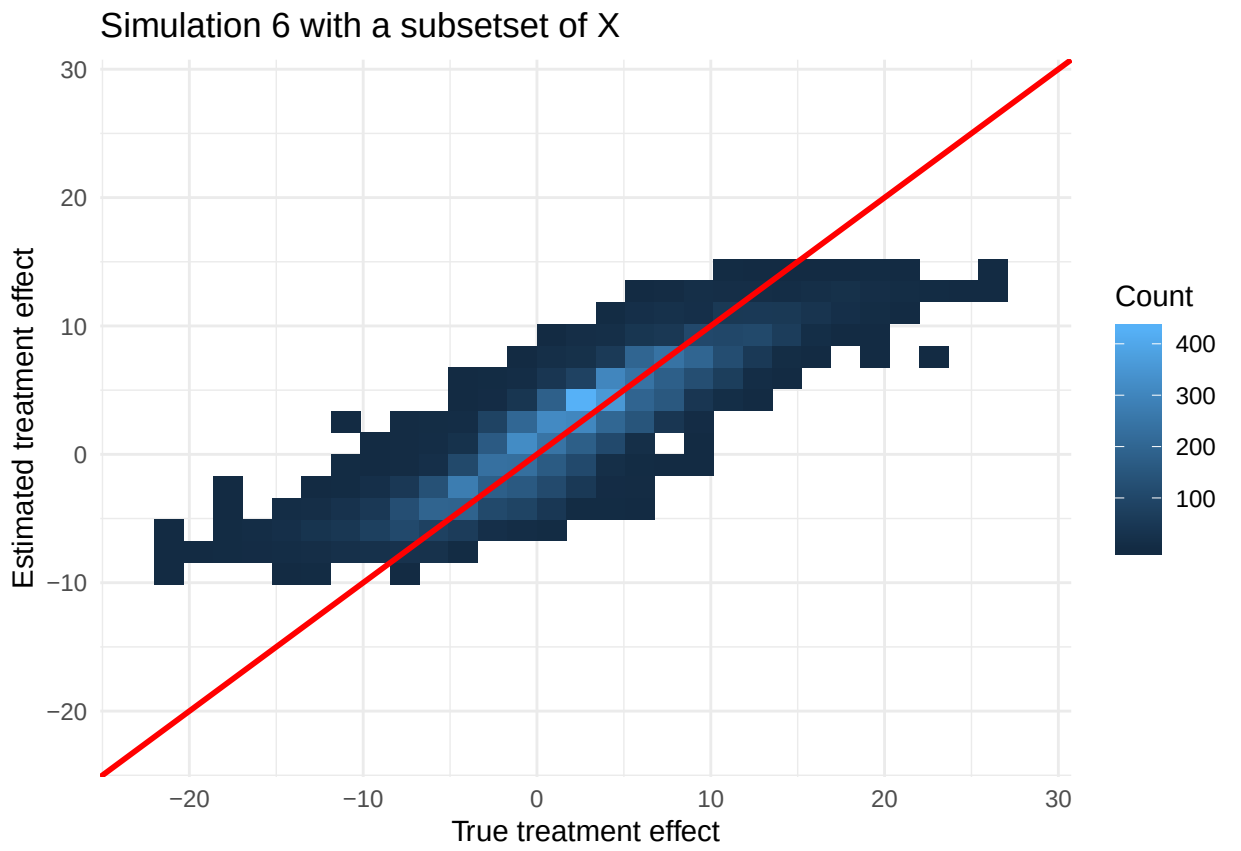
The *proportional difference in the ATE estimates* for the whole population:

```r
fun.diff_ATE(ATE_est = ATE_est_6.important, ATE_true = mean(tau_1))
```

```
## [1] "Proportional mean of the differences:"
##   estimate
## -0.1233143
## [1] "Proportional mean of the differences, 95 % confidence intervals:"
##    estimate    estimate
## -0.22973267 -0.01689592
```

```r
true_vs_pred_6.important <- as.data.frame(cbind(tau_1, pred_tau_6.important$predictions))

colnames(true_vs_pred_6.important) <- c("tau", "pred_tau")

plot13 <- ggplot(data = true_vs_pred_6.important, aes(x = tau, y = pred_tau)) +
  geom_bin2d() +
  geom_abline(slope = 1, intercept = 0, colour = "red", size = 1) +
  theme_minimal() +
  labs(title = "Simulation 6 with a subsetset of X",
       x = "True treatment effect", y = "Estimated treatment effect", fill = "Count") +
  lims(x = c(min(true_vs_pred_6.important), max(true_vs_pred_6.important)),
       y = c(min(true_vs_pred_6.important), max(true_vs_pred_6.important)))

plot13
```

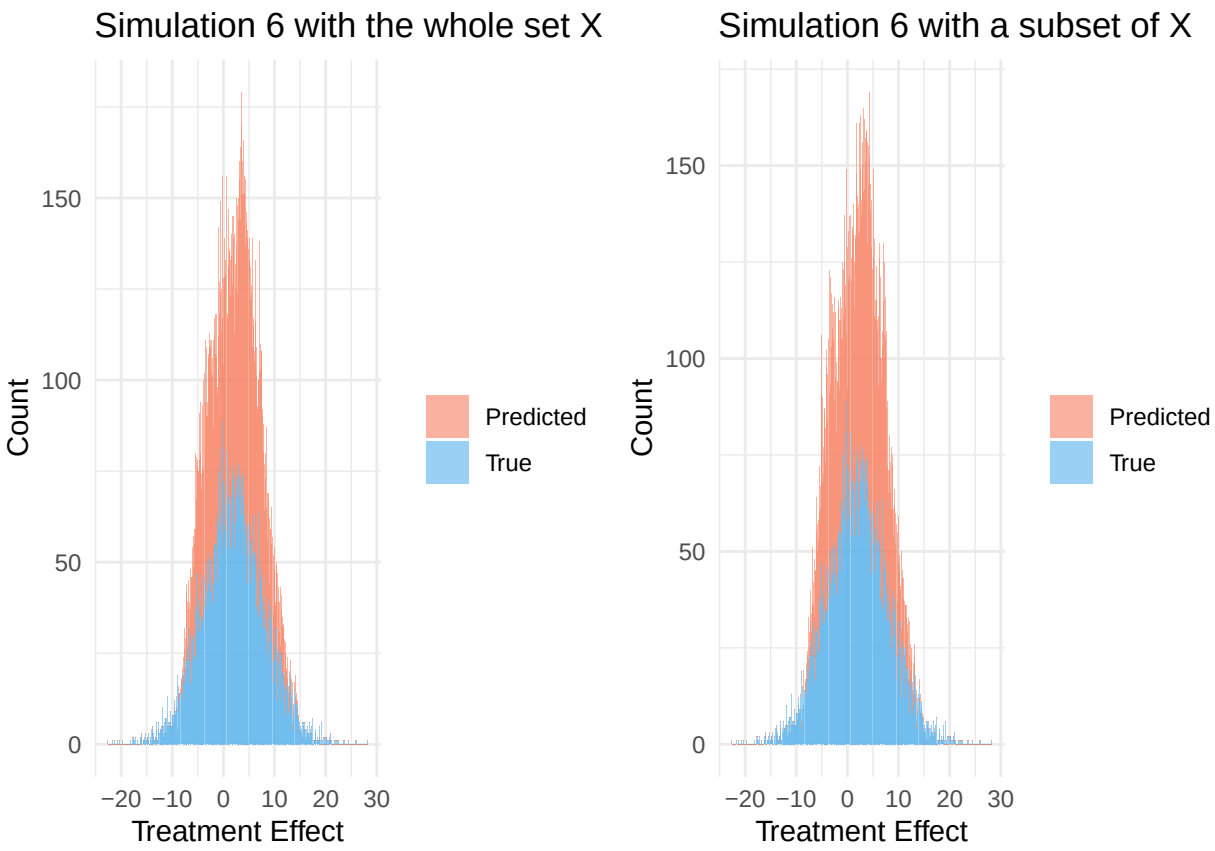
```
## Warning: Removed 1 rows containing missing values (geom_tile).
```



Summary

Predicted $\hat{\tau}s$:

```
grid.arrange(plot_pred_tau_12, plot_pred_tau_13, nrow = 1)
```

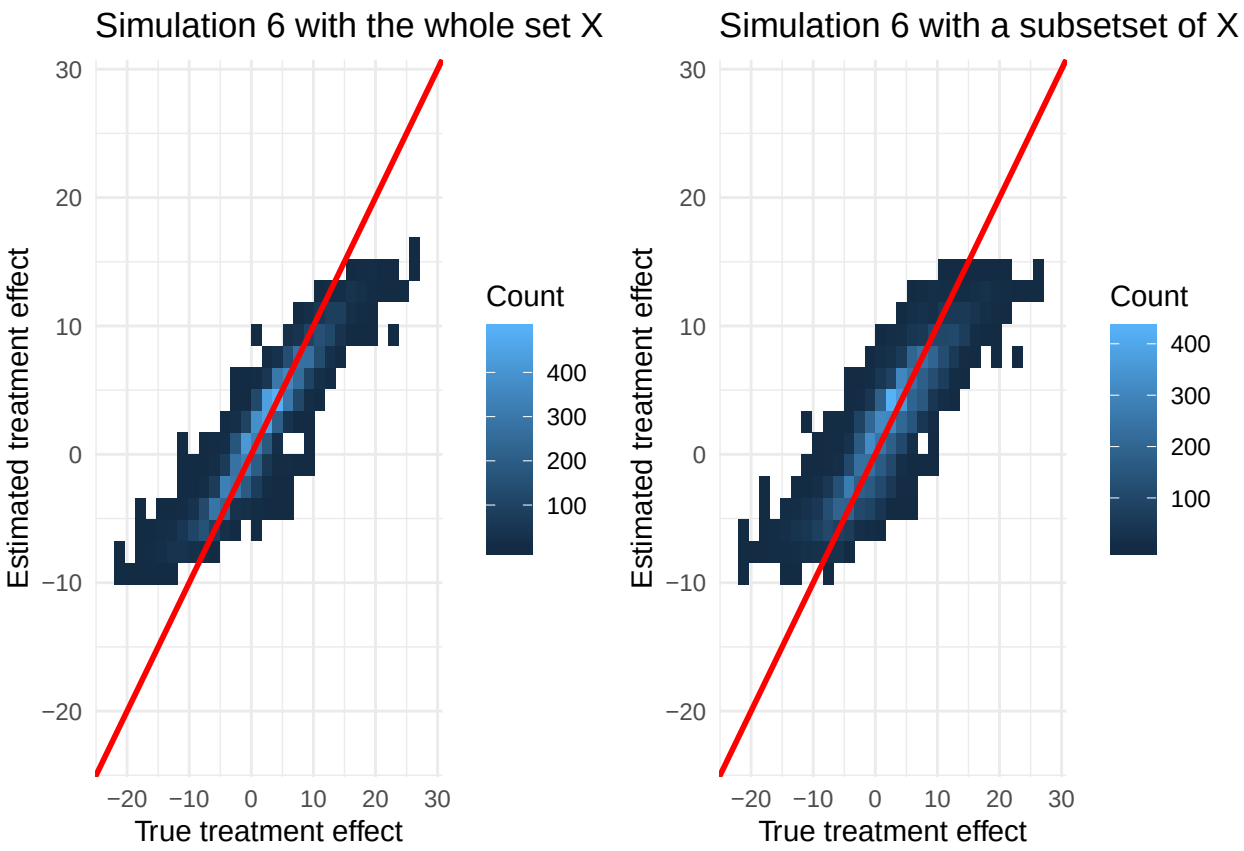


True τ vs. predicted $\hat{\tau}$:

```
grid.arrange(plot12, plot13, nrow = 1)
```

```
## Warning: Removed 1 rows containing missing values (geom_tile).
```

```
## Warning: Removed 1 rows containing missing values (geom_tile).
```



RMSEs:

```
rmse_6 <- as.data.frame(c(rmse_6.full, rmse_6.important))

rownames(rmse_6) <- c("Full set $X$", "Subset of $X$")

colnames(rmse_6) <- c("RMSE")

knitr::kable(rmse_6, escape = FALSE)
```

	RMSE
Full set X	2.274664
Subset of X	2.877371

Coverages:

```
coverage_6 <- as.data.frame(c(coverage_6.full, coverage_6.important))

rownames(coverage_6) <-  c("Full set $X$", "Subset of $X$")

colnames(coverage_6) <- c("Coverage")

knitr::kable(coverage_6, escape = FALSE)
```

	Coverage
Full set X	0.7734
Subset of X	0.6238

LaTeX:

```
knitr::kable(rmse_6, format = "latex",escape = FALSE)
```

	RMSE
Full set X	2.274664
Subset of X	2.877371

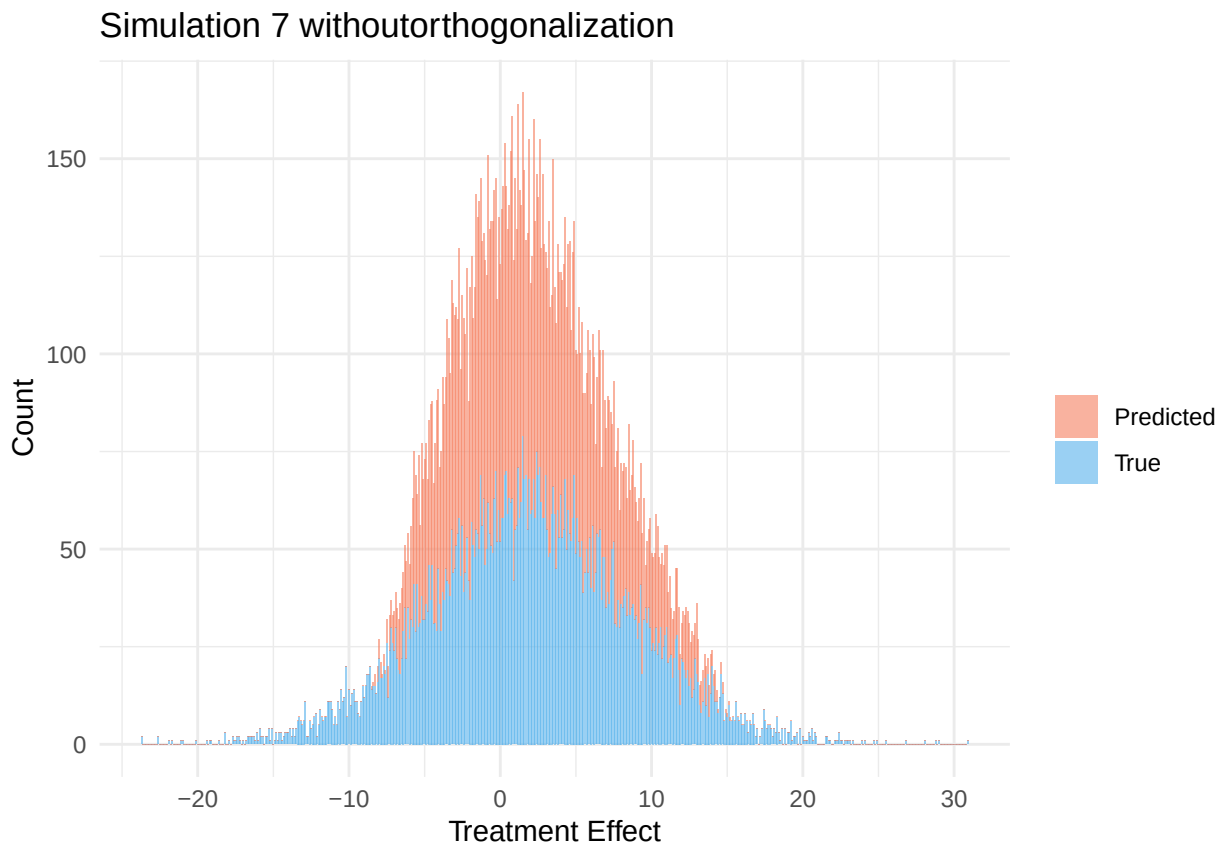
```
knitr::kable(coverage_6, format = "latex",escape = FALSE)
```

	Coverage
Full set X	0.7734
Subset of X	0.6238

3.7 Heterogeneous Treatment Effect with the Confounded Assignment, Including Covariates Affecting to Output

Without orthogonalization:

```
# Estimate causal forest
start_time_14 <- Sys.time() #Recording the running time

#Fitting the model
cf7.no_ort <- grf::causal_forest(cbind(X, Z, C), y_7, w_3, orthog.boosting = FALSE)


end_time_14 <- Sys.time()

#Predicted values

pred_tau_7.no_ort <- predict(cf7.no_ort, estimate.variance = TRUE)

plot_pred_tau_14 <- cbind(pred_tau_7.no_ort$predictions, tau_2) %>%
  as.data.frame() %>%
  dplyr::rename(Predicted = V1) %>%
  dplyr::rename(True = tau_2) %>%
  gather(key = "key", value = "value") %>%
  ggplot(aes(x = value, fill = key)) +
  geom_histogram(binwidth = 0.1, alpha=.6) +
  scale_fill_manual(name = "", values=c("#f68060", "#57b2eb")) +
  labs(title ="Simulation 7 withoutorthogonalization", x = "Treatment Effect", y = "Count") +
  theme_minimal()

plot_pred_tau_14
```



Runing time:

```
start_time_14 - end_time_14
```

```
## Time difference of -56.95369 secs
```

RMSE:

```
rmse_7.no_ort <- fun.rmse(predicted = pred_tau_7.no_ort$predictions, true = tau_2)

rmse_7.no_ort
```

```
## [1] 3.157229
```

Coverage:

```
coverage_7.no_ort <- fun.coverage(pred_tau_7.no_ort, tau_2)

coverage_7.no_ort
```

```
## [1] 0.6832
```

Estimated ATE:

```r
ATE_est_7.no_ort <- average_treatment_effect(cf7.no_ort)

ATE_est_7.no_ort
```

```
##  estimate   std.err
## 2.0263906 0.2233486
```

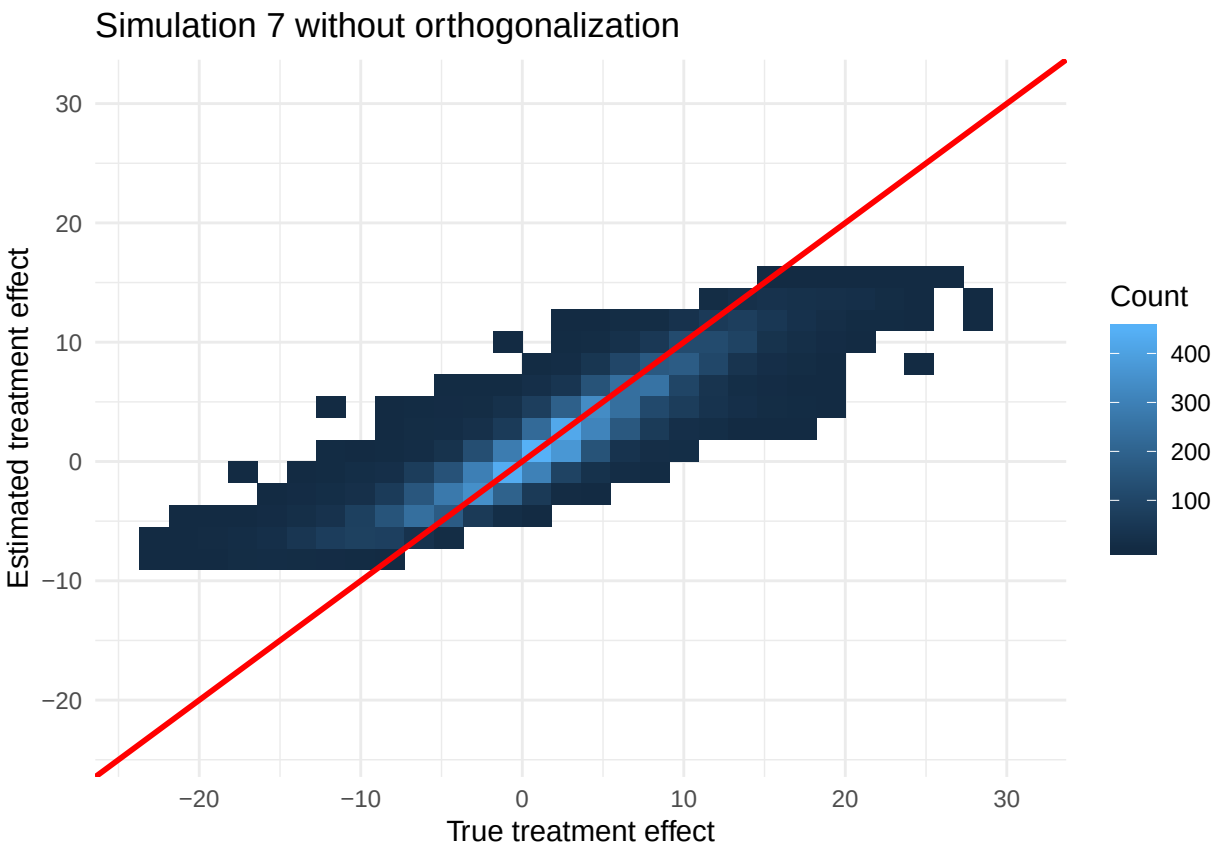
The *proportional difference in the ATE estimates* for the whole population:

```r
fun.diff_ATE(ATE_est = ATE_est_7.no_ort, ATE_true = mean(tau_2))
```

```
## [1] "Proportional mean of the differences:"
##    estimate
## -0.01723146
## [1] "Proportional mean of the differences, 95 % confidence intervals:"
##   estimate   estimate
## -0.1293506  0.0948877
```

```r
true_vs_pred_7.no_ort <- as.data.frame(cbind(tau_2, pred_tau_7.no_ort$predictions))

colnames(true_vs_pred_7.no_ort) <- c("tau", "pred_tau")

plot14 <- ggplot(data = true_vs_pred_7.no_ort, aes(x = tau, y = pred_tau)) +
  geom_bin2d() +
  geom_abline(slope = 1, intercept = 0, colour = "red", size = 1) +
  theme_minimal() +
  labs(title = "Simulation 7 without orthogonalization",
       x = "True treatment effect", y = "Estimated treatment effect", fill = "Count") +
  lims(x = c(min(true_vs_pred_7.no_ort), max(true_vs_pred_7.no_ort)),
       y = c(min(true_vs_pred_7.no_ort), max(true_vs_pred_7.no_ort)))

plot14
```

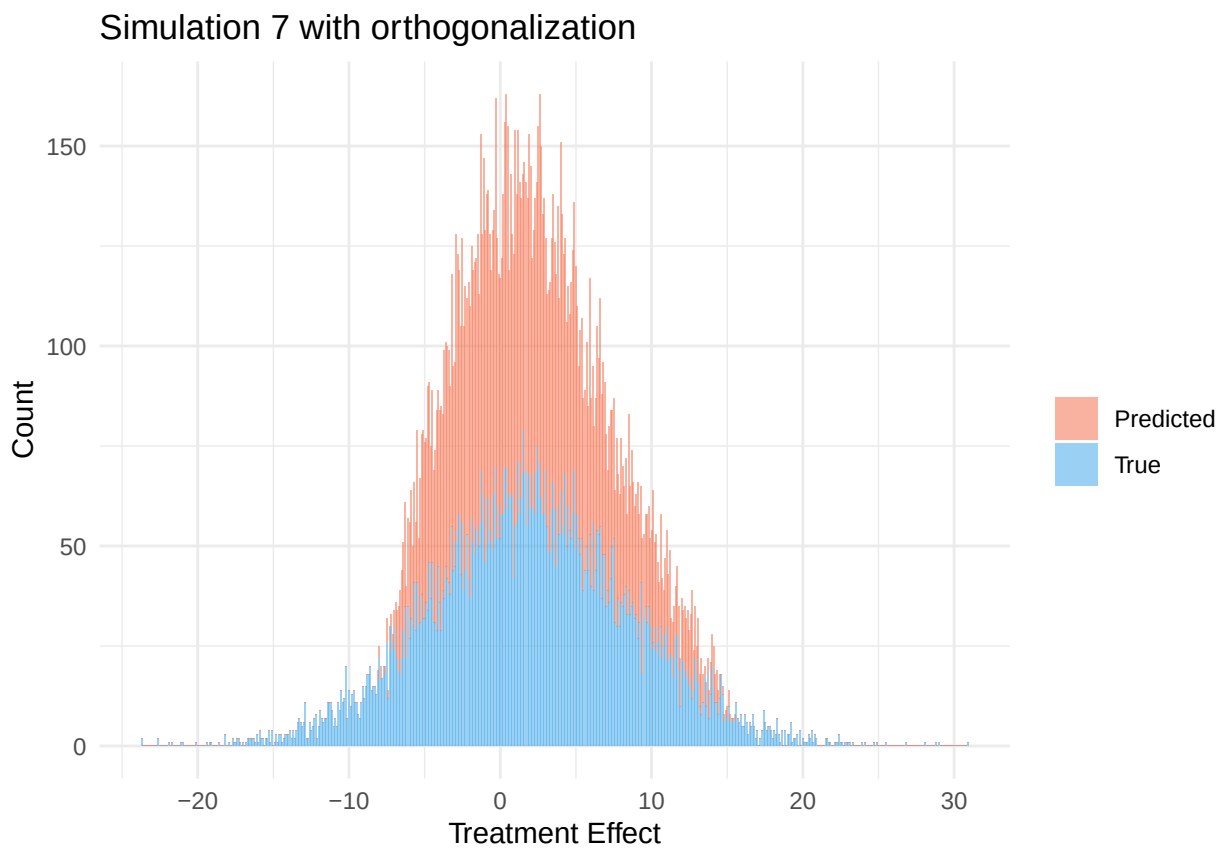
```
## Warning: Removed 2 rows containing missing values (geom_tile).
```



With orthogonalization:

```
# Estimate causal forest
start_time_15 <- Sys.time() #Recording the running time

#Fitting the model
cf7.with_ort <- grf::causal_forest(cbind(X,Z, C), y_7, w_3, orthog.boosting = TRUE)


end_time_15 <- Sys.time()

#Predicted values

pred_tau_7.with_ort <- predict(cf7.with_ort, estimate.variance = TRUE)

plot_pred_tau_15 <- cbind(pred_tau_7.with_ort$predictions, tau_2) %>%
  as.data.frame() %>%
  dplyr::rename(Predicted = V1) %>%
  dplyr::rename(True = tau_2) %>%
  gather(key = "key", value = "value") %>%
  ggplot(aes(x = value, fill = key)) +
  geom_histogram(binwidth = 0.1, alpha=.6) +
  scale_fill_manual(name = "", values=c("#f68060", "#57b2eb")) +
  labs(title ="Simulation 7 with orthogonalization",
       x = "Treatment Effect", y = "Count") +
  theme_minimal()
```

```
plot_pred_tau_15
```



Runing time:

```
start_time_15 - end_time_15
```

```
## Time difference of -1.205449 mins
```

RMSE:

```
rmse_7.with_ort <- fun.rmse(predicted = pred_tau_7.with_ort$predictions, true = tau_2)

rmse_7.with_ort
```

```
## [1] 3.101125
```

Coverage:

```
coverage_7.with_ort <- fun.coverage(pred_tau_7.with_ort, tau_2)

coverage_7.with_ort
```

```
## [1] 0.6911
```

Estimated ATE:

```
ATE_est_7.with_ort <- average_treatment_effect(cf7.with_ort)

ATE_est_7.with_ort
```

```
##  estimate   std.err
## 2.0533831 0.2162684
```

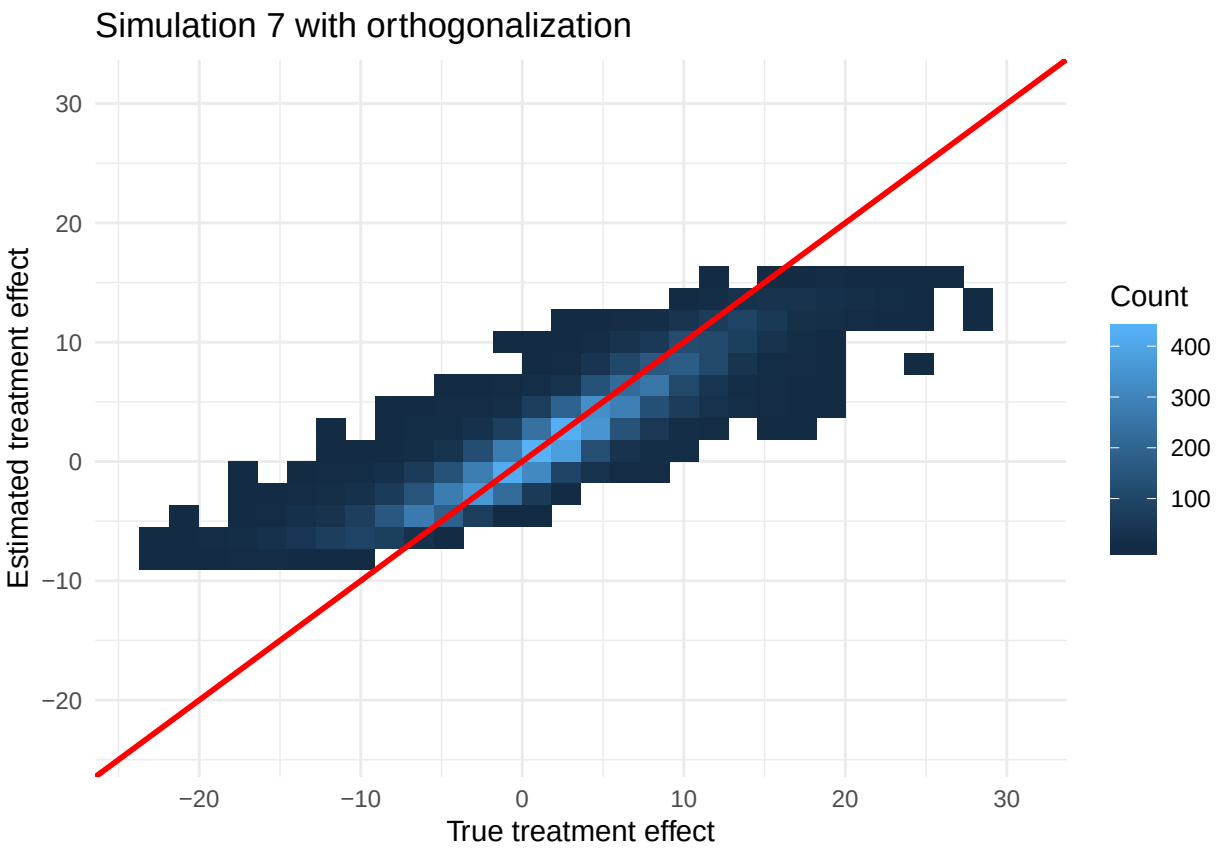
The *proportional difference in the ATE estimates* for the whole population:

```
fun.diff_ATE(ATE_est = ATE_est_7.with_ort, ATE_true = mean(tau_2))
```

```
## [1] "Proportional mean of the differences:"
##    estimate
## -0.03078146
## [1] "Proportional mean of the differences, 95 % confidence intervals:"
##    estimate    estimate
## -0.13934645  0.07778353
```

```
true_vs_pred_7.with_ort <- as.data.frame(cbind(tau_2, pred_tau_7.with_ort$predictions))

colnames(true_vs_pred_7.with_ort) <- c("tau", "pred_tau")

plot15 <- ggplot(data = true_vs_pred_7.with_ort, aes(x = tau, y = pred_tau)) +
  geom_bin2d() +
  geom_abline(slope = 1, intercept = 0, colour = "red", size = 1) +
  theme_minimal() +
  labs(title = "Simulation 7 with orthogonalization",
       x = "True treatment effect", y = "Estimated treatment effect", fill = "Count") +
  lims(x = c(min(true_vs_pred_7.with_ort), max(true_vs_pred_7.with_ort)),
       y = c(min(true_vs_pred_7.with_ort), max(true_vs_pred_7.with_ort)))

plot15
```

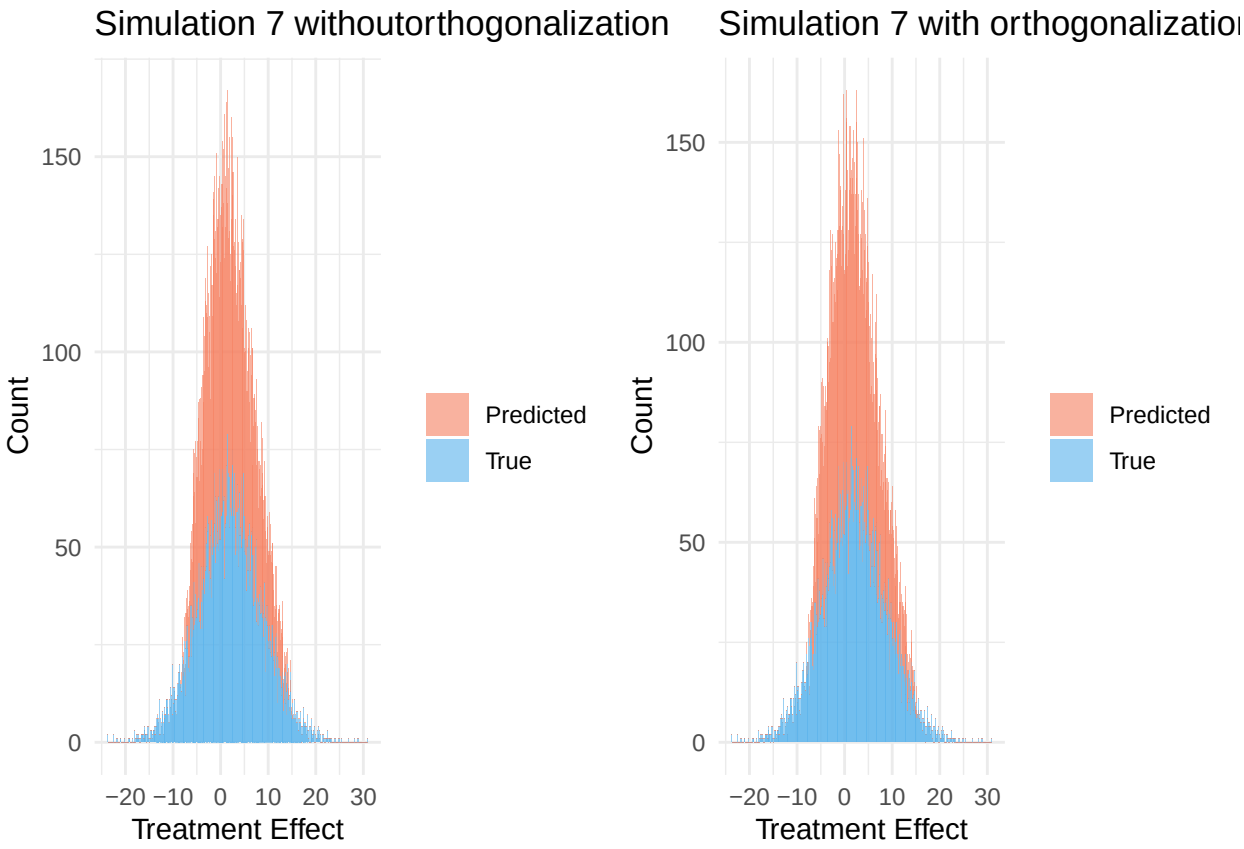
```
## Warning: Removed 2 rows containing missing values (geom_tile).
```



Summary

Predicted $\hat{\tau}s$:

```
grid.arrange(plot_pred_tau_14, plot_pred_tau_15, nrow = 1)
```

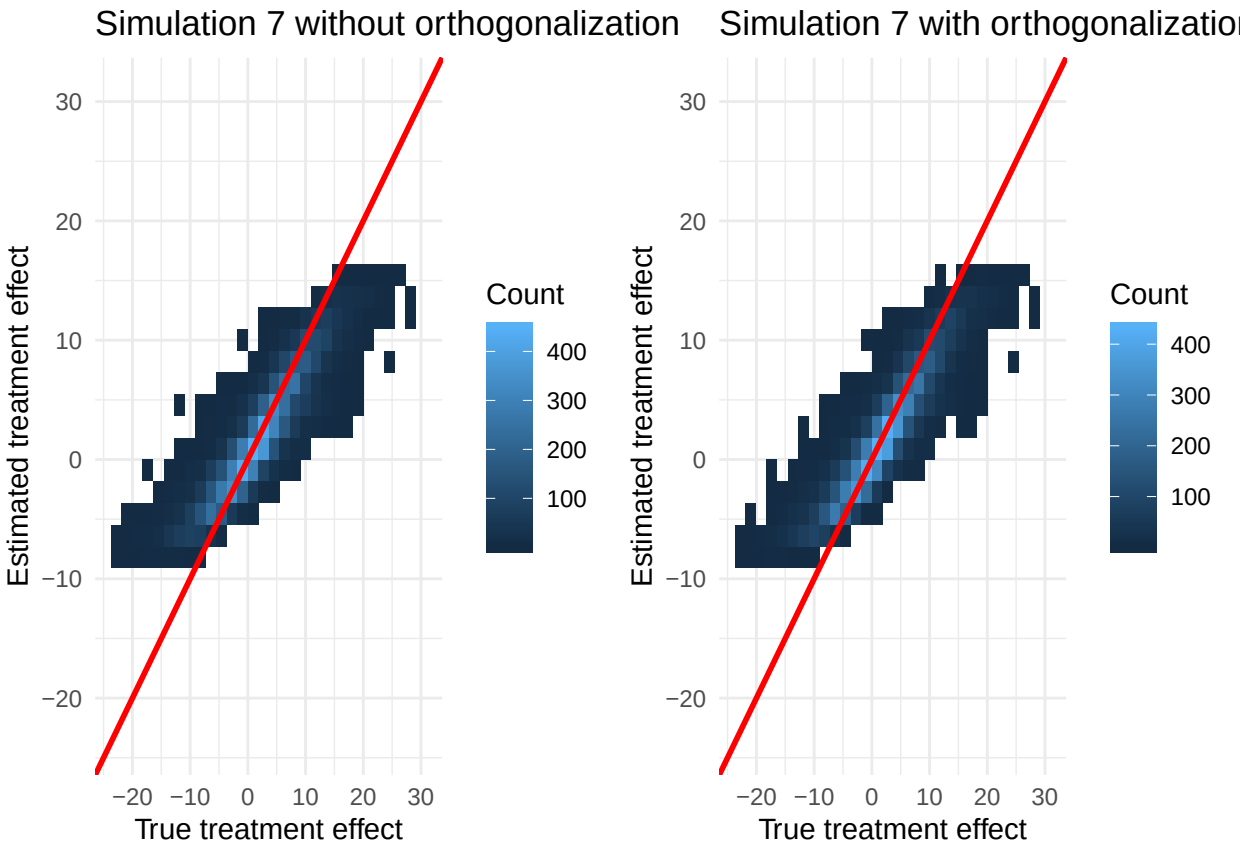


True τ vs. predicted $\hat{\tau}$:

```
grid.arrange(plot14, plot15, nrow = 1)
```

```
## Warning: Removed 2 rows containing missing values (geom_tile).
```

```
## Warning: Removed 2 rows containing missing values (geom_tile).
```



RMSEs:

```
rmse_7 <- as.data.frame(c(rmse_7.no_ort, rmse_7.with_ort))

rownames(rmse_7) <- c("With orthogonalization", "Without orthogonalization")

colnames(rmse_7) <- c("RMSE")

knitr::kable(rmse_7, escape = FALSE)
```

	RMSE
With orthogonalization	3.157229
Without orthogonalization	3.101125

Coverages:

```
coverage_7 <- as.data.frame(c(coverage_7.no_ort, coverage_7.with_ort))

rownames(coverage_7) <-  c("With orthogonalization", "Without orthogonalization")

colnames(coverage_7) <- c("Coverage")

knitr::kable(coverage_7, escape = FALSE)
```

	Coverage
With orthogonalization	0.6832
Without orthogonalization	0.6911

LaTeX:

```
knitr::kable(rmse_7, format = "latex",escape = FALSE)
```

	RMSE
With orthogonalization	3.157229
Without orthogonalization	3.101125

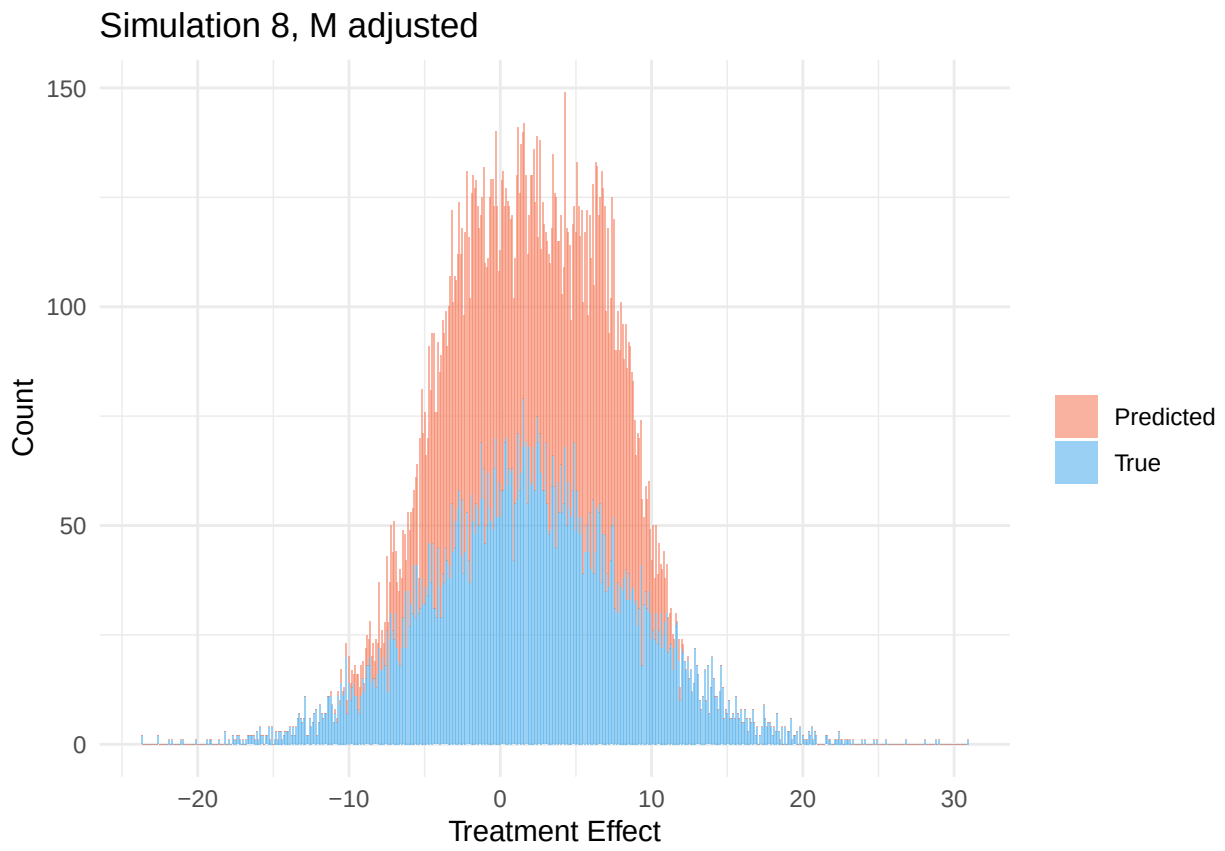
```
knitr::kable(coverage_7, format = "latex",escape = FALSE)
```

	Coverage
With orthogonalization	0.6832
Without orthogonalization	0.6911

3.8 Heterogeneous Treatment Effect with Confounded Assignment, Including Covariates Affecting to Output and a Pure Local M-Structure

With M ($X \cup Z \cup C \cup M \in \mathbb{S}$):

```
# Estimate causal forest
start_time_16 <- Sys.time() #Recording the running time

#Fitting the model
cf8.with_M <- grf::causal_forest(cbind(X, Z, C, m), y_8, w_4, orthog.boosting = TRUE)


end_time_16 <- Sys.time()

#Predicted values

pred_tau_8.with_M <- predict(cf8.with_M, estimate.variance = TRUE)

plot_pred_tau_16 <- cbind(pred_tau_8.with_M$predictions, tau_2) %>%
  as.data.frame() %>%
  dplyr::rename(Predicted = V1) %>%
  dplyr::rename(True = tau_2) %>%
  gather(key = "key", value = "value") %>%
  ggplot(aes(x = value, fill = key)) +
  geom_histogram(binwidth = 0.1, alpha=.6) +
  scale_fill_manual(name = "", values=c("#f68060", "#57b2eb")) +
  labs(title ="Simulation 8, M adjusted",
       x = "Treatment Effect", y = "Count") +
  theme_minimal()

plot_pred_tau_16
```



Runing time:

```
start_time_16 - end_time_16
```

```
## Time difference of -1.733859 mins
```

RMSE:

```
rmse_8.with_M <- fun.rmse(predicted = pred_tau_8.with_M$predictions, true = tau_2)

rmse_8.with_M
```

```
## [1] 3.201408
```

Coverage:

```
coverage_8.with_M <- fun.coverage(pred_tau_8.with_M, tau_2)

coverage_8.with_M
```

```
## [1] 0.67
```

Estimated ATE:

```r
ATE_est_8.with_M <- average_treatment_effect(cf8.with_M)

ATE_est_8.with_M
```

```
##  estimate   std.err
## 1.8739371 0.2197631
```

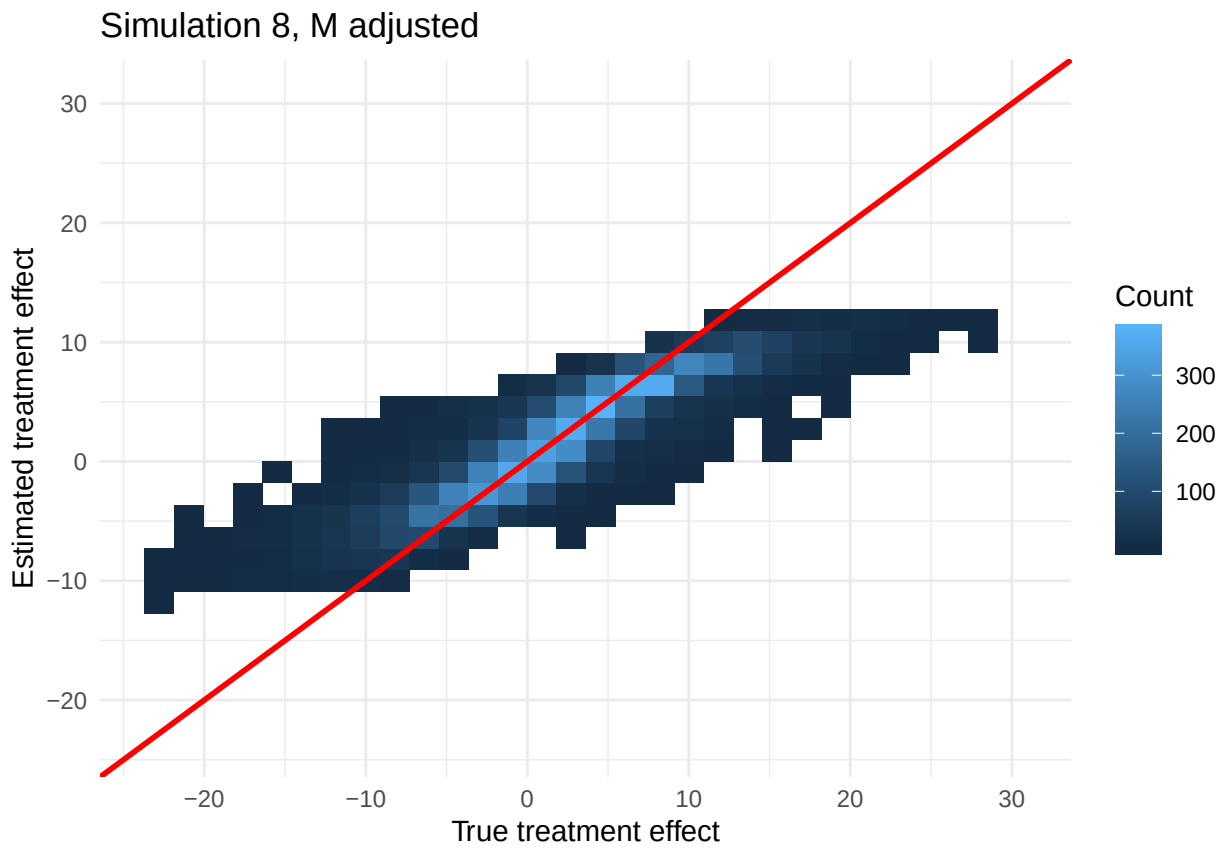
The *proportional difference in the ATE estimates* for the whole population:

```r
fun.diff_ATE(ATE_est = ATE_est_8.with_M, ATE_true = mean(tau_2))
```

```
## [1] "Proportional mean of the differences:"
##   estimate
## 0.05929896
## [1] "Proportional mean of the differences, 95 % confidence intervals:"
##    estimate    estimate
## -0.05102032  0.16961823
```

```r
true_vs_pred_8.with_M <- as.data.frame(cbind(tau_2, pred_tau_8.with_M$predictions))

colnames(true_vs_pred_8.with_M) <- c("tau", "pred_tau")

plot16 <- ggplot(data = true_vs_pred_8.with_M, aes(x = tau, y = pred_tau)) +
  geom_bin2d() +
  geom_abline(slope = 1, intercept = 0, colour = "red", size = 1) +
  theme_minimal() +
  labs(title = "Simulation 8, M adjusted",
       x = "True treatment effect", y = "Estimated treatment effect", fill = "Count") +
  lims(x = c(min(true_vs_pred_8.with_M), max(true_vs_pred_8.with_M)),
       y = c(min(true_vs_pred_8.with_M), max(true_vs_pred_8.with_M)))

plot16
```

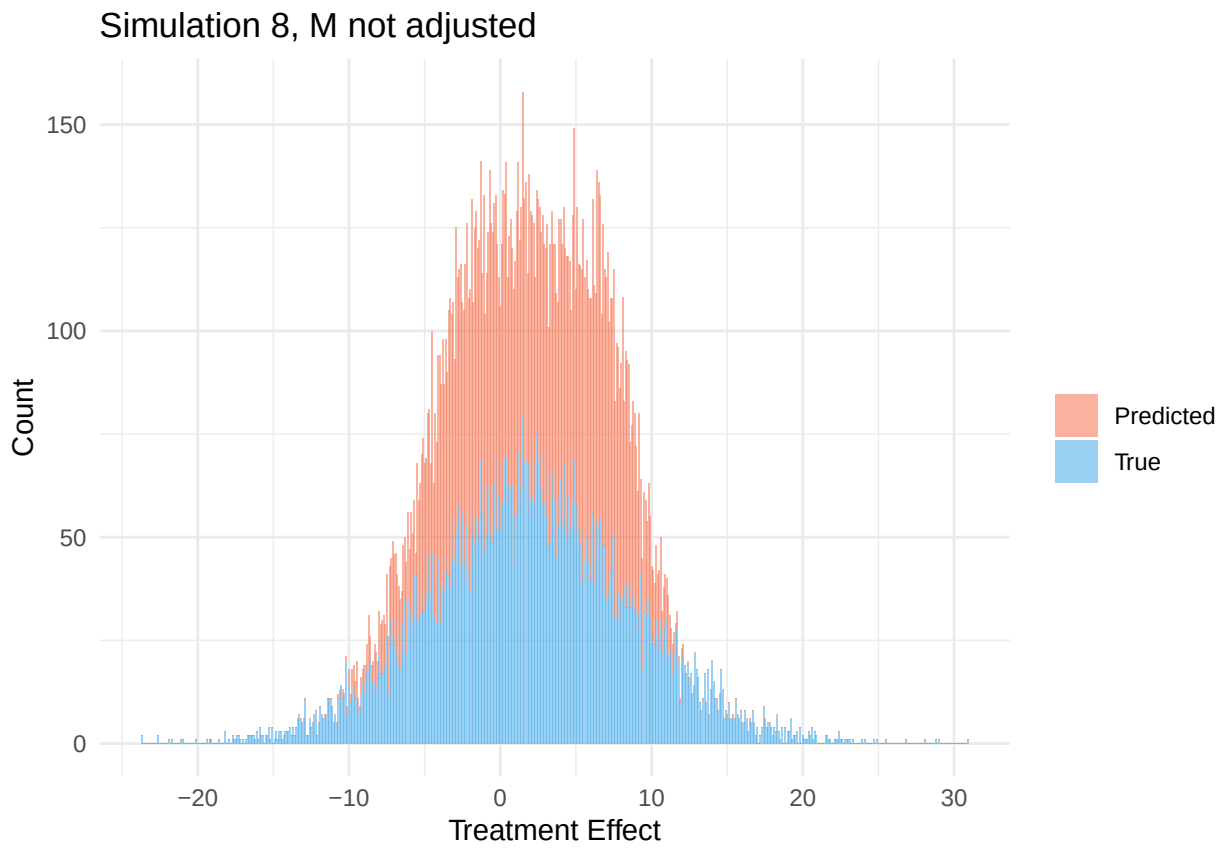
```
## Warning: Removed 2 rows containing missing values (geom_tile).
```



$M \notin \mathbb{S}$ $(X \cup Z \cup C \in \mathbb{S})$:

```
# Estimate causal forest
start_time_17 <- Sys.time() #Recording the running time

#Fitting the model
cf8.no_M <- grf::causal_forest(cbind(X, Z, C), y_8, w_4, orthog.boosting = TRUE)


end_time_17 <- Sys.time()

#Predicted values

pred_tau_8.no_M <- predict(cf8.no_M, estimate.variance = TRUE)

plot_pred_tau_17 <- cbind(pred_tau_8.no_M$predictions, tau_2) %>%
  as.data.frame() %>%
  dplyr::rename(Predicted = V1) %>%
  dplyr::rename(True = tau_2) %>%
  gather(key = "key", value = "value") %>%
  ggplot(aes(x = value, fill = key)) +
  geom_histogram(binwidth = 0.1, alpha=.6) +
  scale_fill_manual(name = "", values=c("#f68060", "#57b2eb")) +
  labs(title ="Simulation 8, M not adjusted",
       x = "Treatment Effect", y = "Count") +
  theme_minimal()
```

```
plot_pred_tau_17
```



Runing time:

```
start_time_17 - end_time_17
```

```
## Time difference of -1.204982 mins
```

RMSE:

```
rmse_8.no_M <- fun.rmse(predicted = pred_tau_8.no_M$predictions, true = tau_2)
```

Coverage:

```
coverage_8.no_M <- fun.coverage(pred_tau_8.no_M, tau_2)

coverage_8.no_M
```

```
## [1] 0.6671
```

Estimated ATE:

```r
ATE_est_8.no_M <- average_treatment_effect(cf8.no_M)

ATE_est_8.no_M
```

```
##  estimate   std.err
## 1.8671429 0.2218439
```

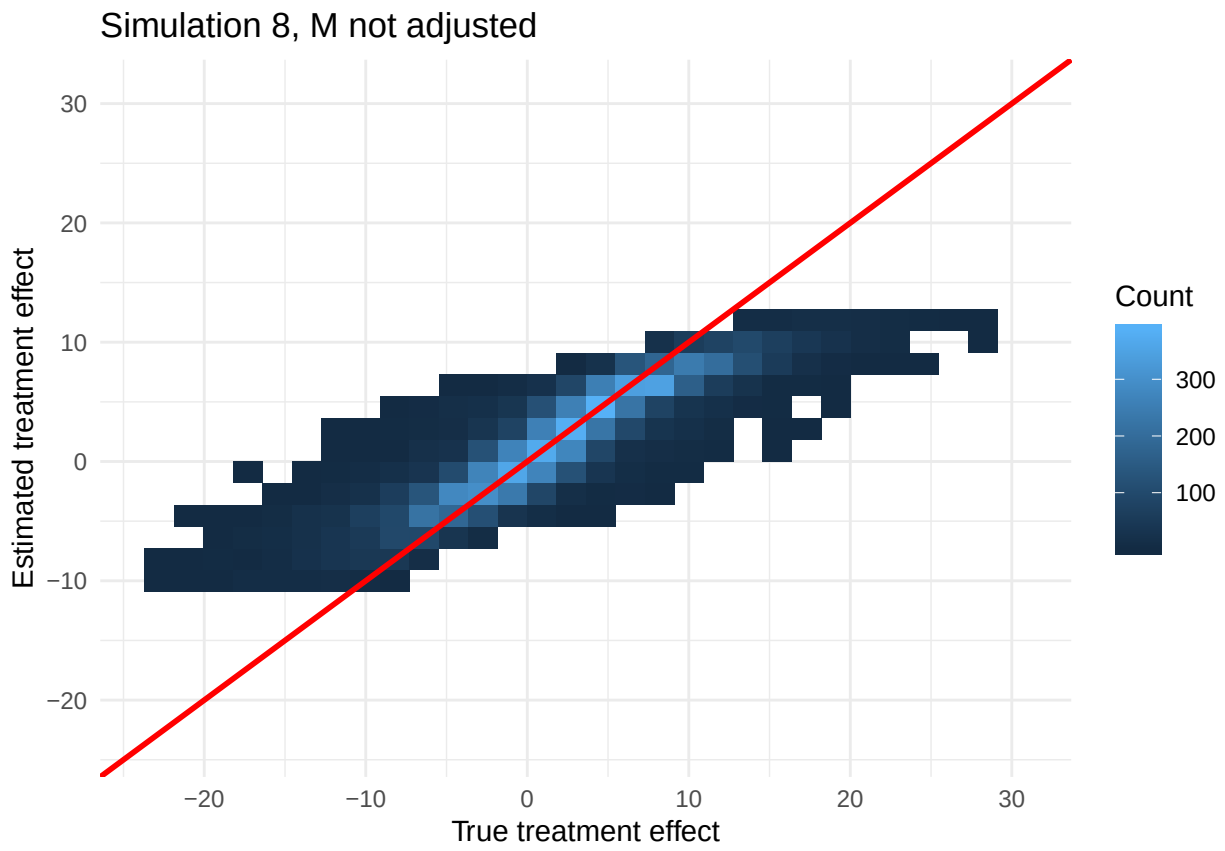
The *proportional difference in the ATE estimates* for the whole population:

```r
fun.diff_ATE(ATE_est = ATE_est_8.no_M, ATE_true = mean(tau_2))
```

```
## [1] "Proportional mean of the differences:"
##   estimate
## 0.06270957
## [1] "Proportional mean of the differences, 95 % confidence intervals:"
##    estimate    estimate
## -0.04865424  0.17407338
```

```r
true_vs_pred_8.no_M <- as.data.frame(cbind(tau_2, pred_tau_8.no_M$predictions))

colnames(true_vs_pred_8.no_M) <- c("tau", "pred_tau")

plot17 <- ggplot(data = true_vs_pred_8.no_M, aes(x = tau, y = pred_tau)) +
  geom_bin2d() +
  geom_abline(slope = 1, intercept = 0, colour = "red", size = 1) +
  theme_minimal() +
  labs(title = "Simulation 8, M not adjusted",
       x = "True treatment effect", y = "Estimated treatment effect", fill = "Count") +
  lims(x = c(min(true_vs_pred_8.no_M), max(true_vs_pred_8.no_M)),
       y = c(min(true_vs_pred_8.no_M), max(true_vs_pred_8.no_M)))

plot17
```

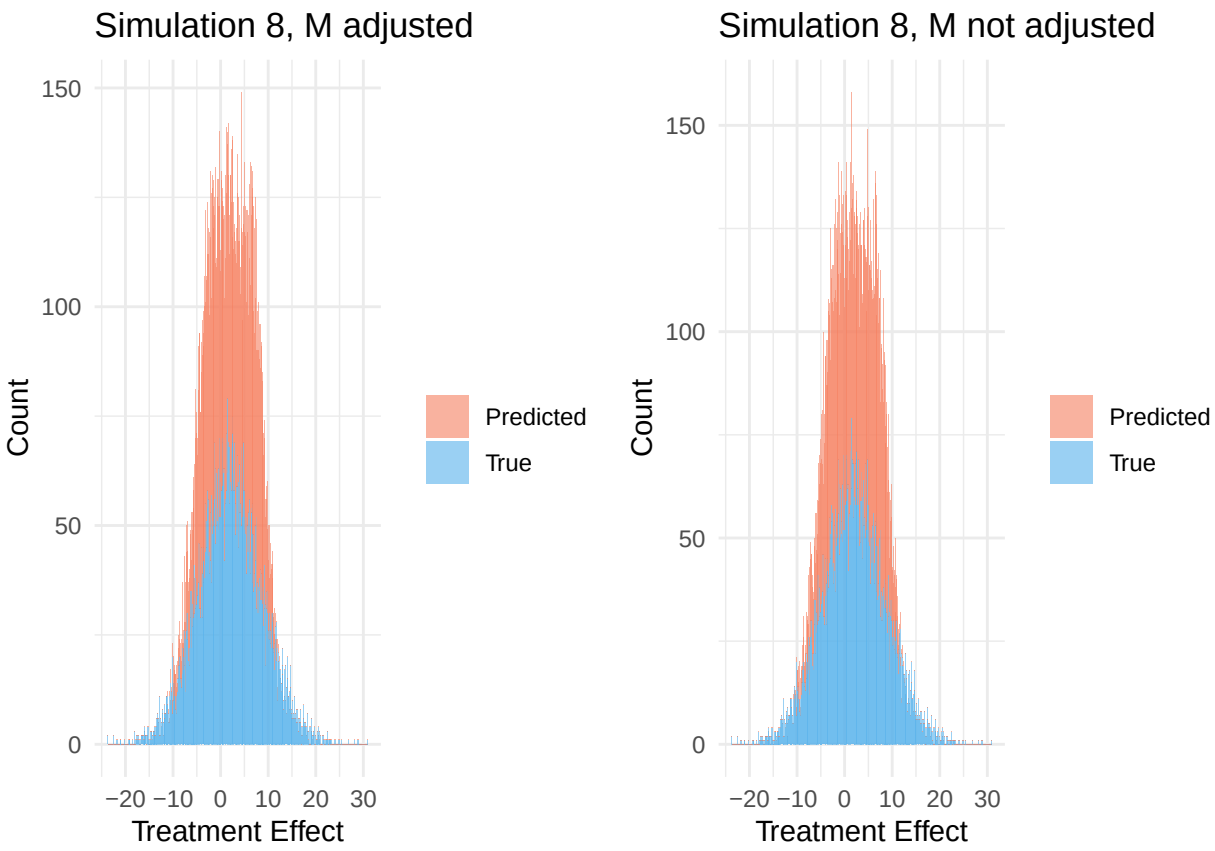
```
## Warning: Removed 2 rows containing missing values (geom_tile).
```



Summary

Predicted $\hat{\tau}s$:

```
grid.arrange(plot_pred_tau_16, plot_pred_tau_17, nrow = 1)
```

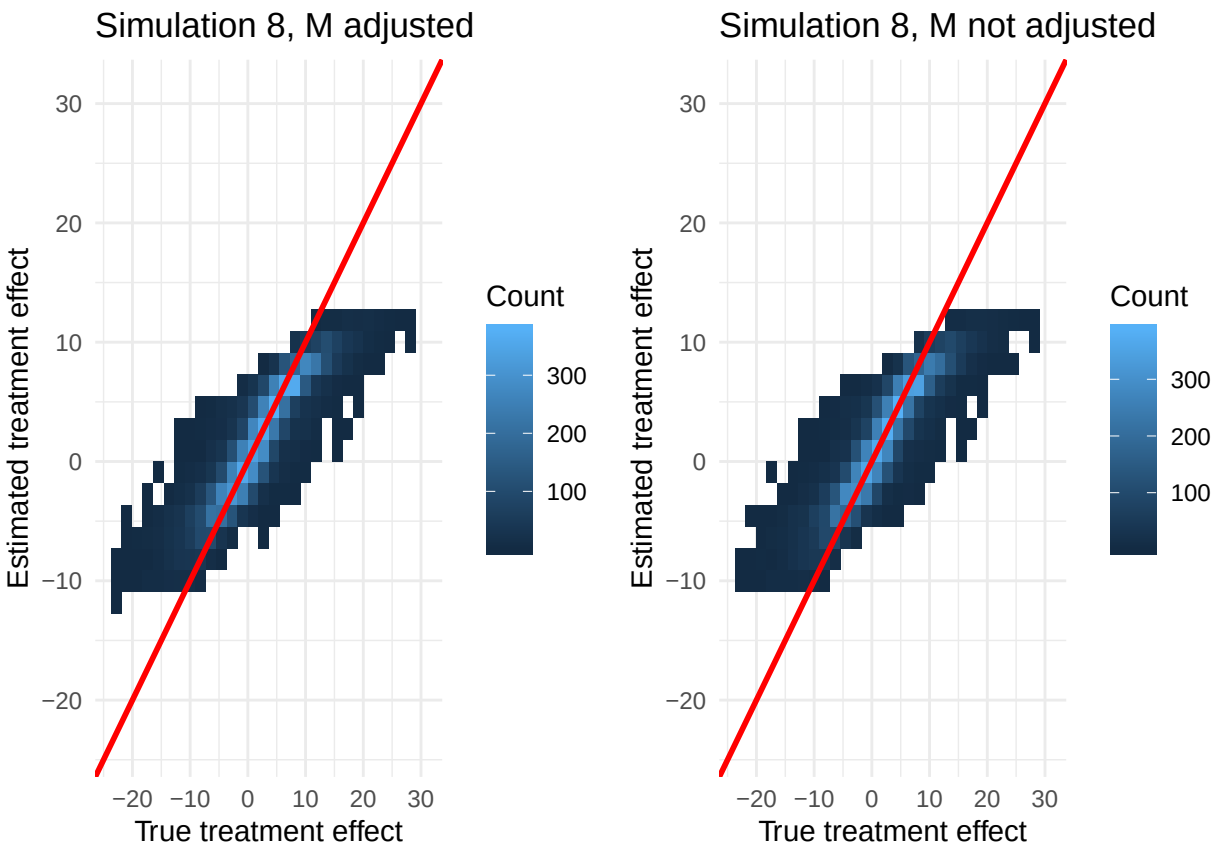


True τ vs. predicted $\hat{\tau}$:

```
grid.arrange(plot16, plot17, nrow = 1)
```

```
## Warning: Removed 2 rows containing missing values (geom_tile).
```

```
## Warning: Removed 2 rows containing missing values (geom_tile).
```



RMSEs:

```
rmse_8 <- as.data.frame(c(rmse_8.with_M, rmse_8.no_M))

rownames(rmse_8) <- c("$M$ adjusted", "$M$ not adjusted")

colnames(rmse_8) <- c("RMSE")

knitr::kable(rmse_8, escape = FALSE)
```

	RMSE
M adjusted	3.201408
M not adjusted	3.218593

Coverages:

```
coverage_8 <- as.data.frame(c(coverage_8.with_M, coverage_8.no_M))

rownames(coverage_8) <-  c("$M$ adjusted", "$M$ not adjusted")

colnames(coverage_8) <- c("Coverage")

knitr::kable(coverage_8, escape = FALSE)
```

	Coverage
M adjusted	0.6700
M not adjusted	0.6671

LaTeX:

```
knitr::kable(rmse_8, format = "latex",escape = FALSE)
```

	RMSE
M adjusted	3.201408
M not adjusted	3.218593

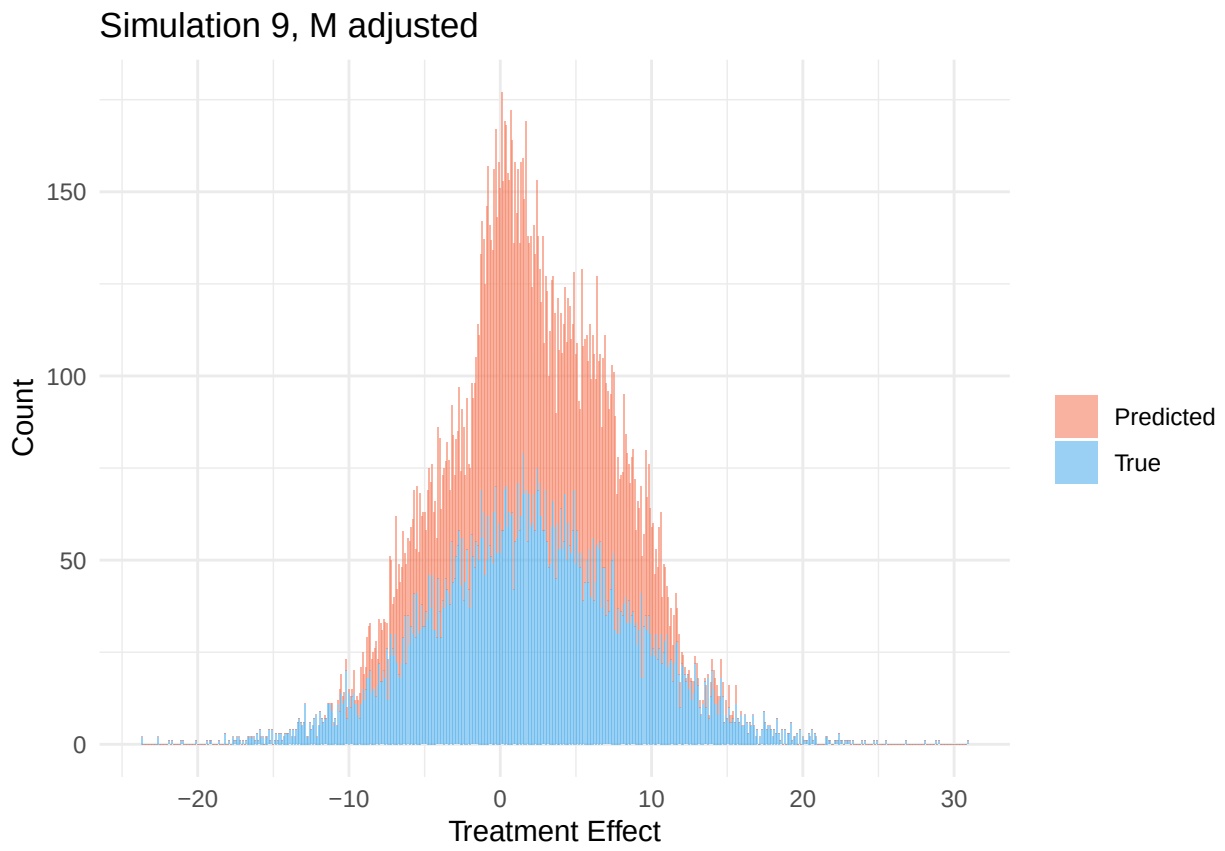
```
knitr::kable(coverage_8, format = "latex",escape = FALSE)
```

	Coverage
M adjusted	0.6700
M not adjusted	0.6671

3.9 Heterogeneous Treatment Effect with Confounded Assignment, Including Covariates Affecting to Output and a Impure Local M-Structure

With M ($X \cup Z \cup C \cup M \in \mathbb{S}$):

```
# Estimate causal forest
start_time_18 <- Sys.time() #Recording the running time

#Fitting the model
cf9.with_M <- grf::causal_forest(cbind(X, Z, C, m), y_9, w_5, orthog.boosting = TRUE)


end_time_18 <- Sys.time()

#Predicted values

pred_tau_9.with_M <- predict(cf9.with_M, estimate.variance = TRUE)

plot_pred_tau_18 <- cbind(pred_tau_9.with_M$predictions, tau_2) %>%
  as.data.frame() %>%
  dplyr::rename(Predicted = V1) %>%
  dplyr::rename(True = tau_2) %>%
  gather(key = "key", value = "value") %>%
  ggplot(aes(x = value, fill = key)) +
  geom_histogram(binwidth = 0.1, alpha=.6) +
  scale_fill_manual(name = "", values=c("#f68060", "#57b2eb")) +
  labs(title ="Simulation 9, M adjusted",
       x = "Treatment Effect", y = "Count") +
  theme_minimal()

plot_pred_tau_18
```



Runing time:

```
start_time_18 - end_time_18
```

```
## Time difference of -1.2066 mins
```

RMSE:

```
rmse_9.with_M <- fun.rmse(predicted = pred_tau_9.with_M$predictions, true = tau_2)

rmse_9.with_M
```

```
## [1] 3.291505
```

Coverage:

```
coverage_9.with_M <- fun.coverage(pred_tau_9.with_M, tau_2)

coverage_9.with_M
```

```
## [1] 0.6589
```

Estimated ATE:

```r
ATE_est_9.with_M <- average_treatment_effect(cf9.with_M)

ATE_est_9.with_M
```

```
##  estimate   std.err
## 2.0312799 0.2196883
```

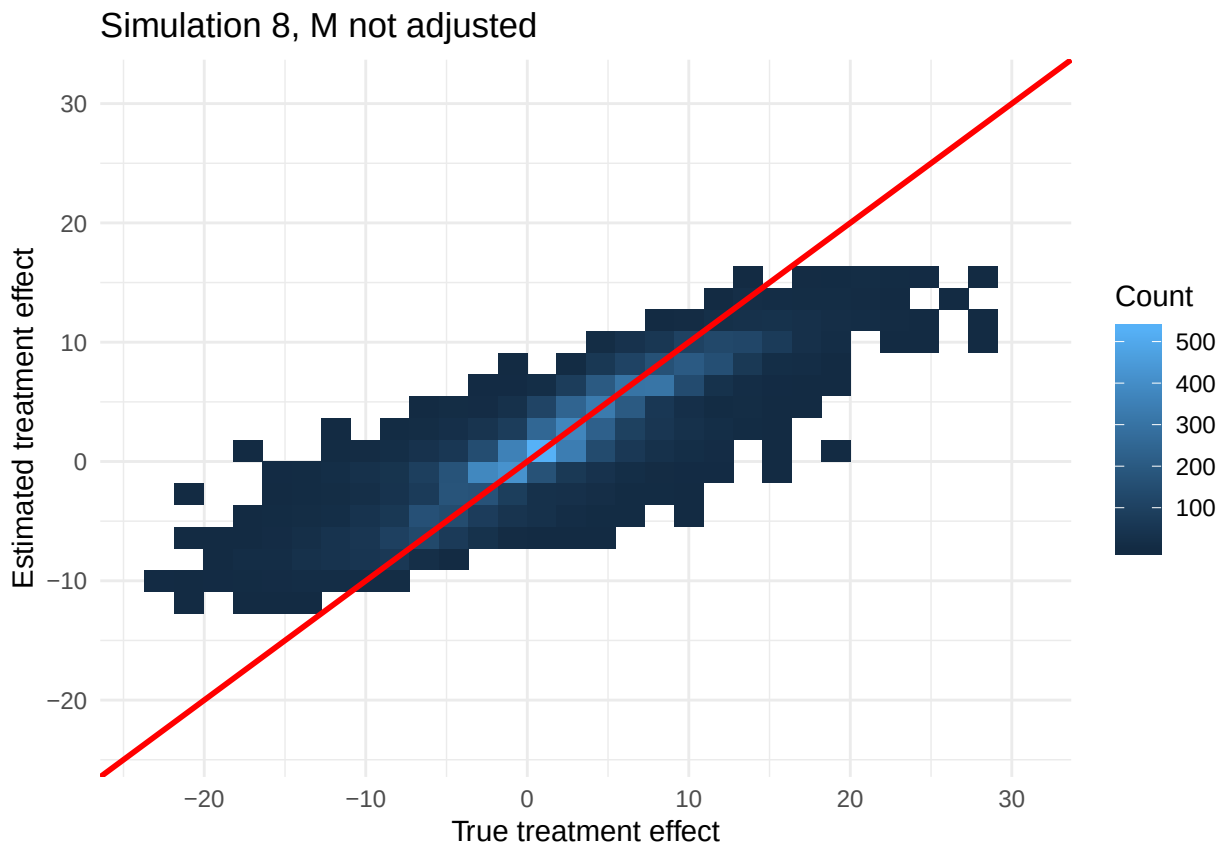
The *proportional difference in the ATE estimates* for the whole population:

```r
fun.diff_ATE(ATE_est = ATE_est_9.with_M, ATE_true = mean(tau_2))
```

```
## [1] "Proportional mean of the differences:"
##    estimate
## -0.01968584
## [1] "Proportional mean of the differences, 95 % confidence intervals:"
##   estimate   estimate
## -0.1299676  0.0905959
```

```r
true_vs_pred_9.with_M <- as.data.frame(cbind(tau_2, pred_tau_9.with_M$predictions))

colnames(true_vs_pred_9.with_M) <- c("tau", "pred_tau")

plot18 <- ggplot(data = true_vs_pred_9.with_M, aes(x = tau, y = pred_tau)) +
  geom_bin2d() +
  geom_abline(slope = 1, intercept = 0, colour = "red", size = 1) +
  theme_minimal() +
  labs(title = "Simulation 8, M not adjusted",
       x = "True treatment effect", y = "Estimated treatment effect", fill = "Count") +
  lims(x = c(min(true_vs_pred_9.with_M), max(true_vs_pred_9.with_M)),
       y = c(min(true_vs_pred_9.with_M), max(true_vs_pred_9.with_M)))

plot18
```

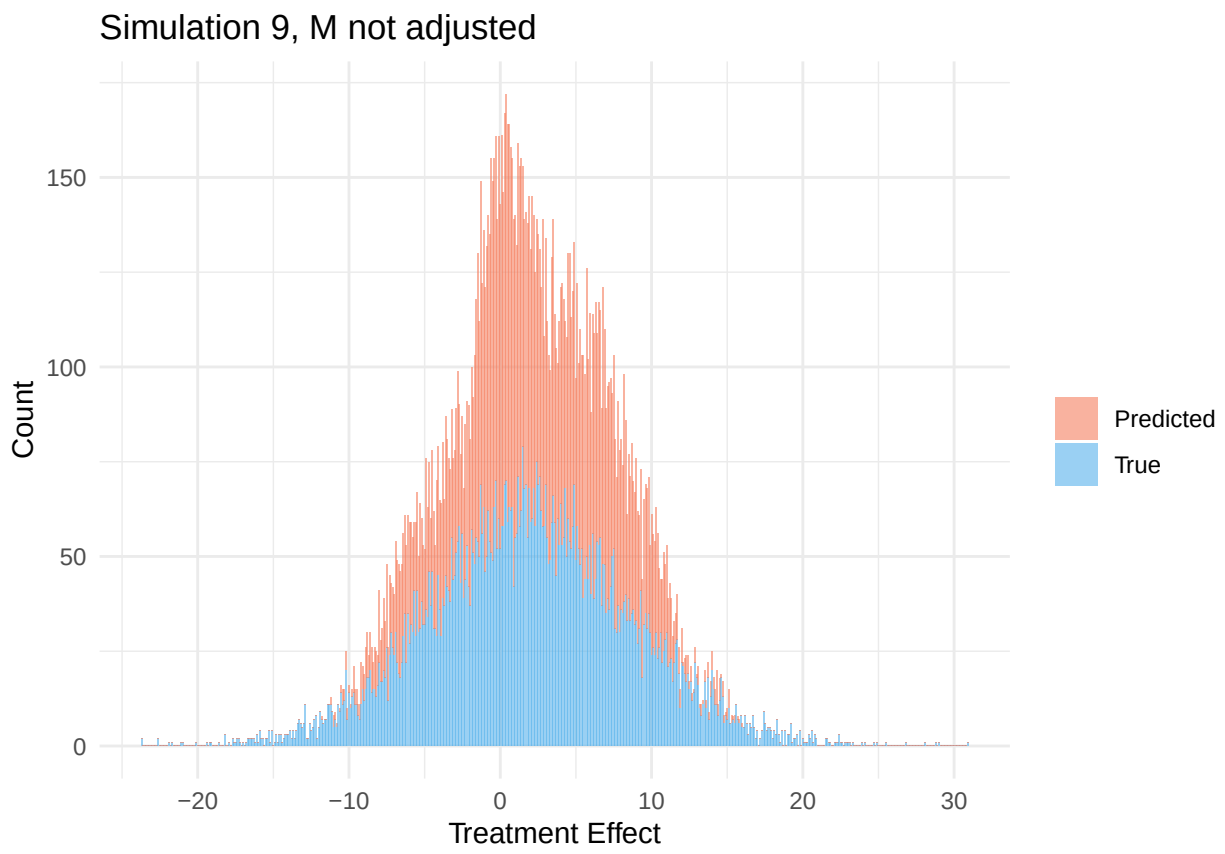
```
## Warning: Removed 2 rows containing missing values (geom_tile).
```



$M \notin \mathbb{S}$ $(X \cup Z \cup C \in \mathbb{S})$:

```
# Estimate causal forest
start_time_19 <- Sys.time() #Recording the running time

#Fitting the model
cf9.no_M <- grf::causal_forest(cbind(X, Z, C), y_9, w_5, orthog.boosting = TRUE)


end_time_19 <- Sys.time()

#Predicted values

pred_tau_9.no_M <- predict(cf9.no_M, estimate.variance = TRUE)

plot_pred_tau_19 <- cbind(pred_tau_9.no_M$predictions, tau_2) %>%
  as.data.frame() %>%
  dplyr::rename(Predicted = V1) %>%
  dplyr::rename(True = tau_2) %>%
  gather(key = "key", value = "value") %>%
  ggplot(aes(x = value, fill = key)) +
  geom_histogram(binwidth = 0.1, alpha=.6) +
  scale_fill_manual(name = "", values=c("#f68060", "#57b2eb")) +
  labs(title ="Simulation 9, M not adjusted",
       x = "Treatment Effect", y = "Count") +
  theme_minimal()
```

```
plot_pred_tau_19
```



Runing time:

```
start_time_19 - end_time_19
```

```
## Time difference of -1.202041 mins
```

RMSE:

```
rmse_9.no_M <- fun.rmse(predicted = pred_tau_9.no_M$predictions, true = tau_2)

rmse_9.no_M
```

```
## [1] 3.279869
```

Coverage:

```
coverage_9.no_M <- fun.coverage(pred_tau_9.no_M, tau_2)

coverage_9.no_M
```

```
## [1] 0.6664
```

Estimated ATE:

```
ATE_est_9.no_M <- average_treatment_effect(cf9.no_M)

ATE_est_9.no_M
```

```
##  estimate   std.err
## 2.0584609 0.2217999
```

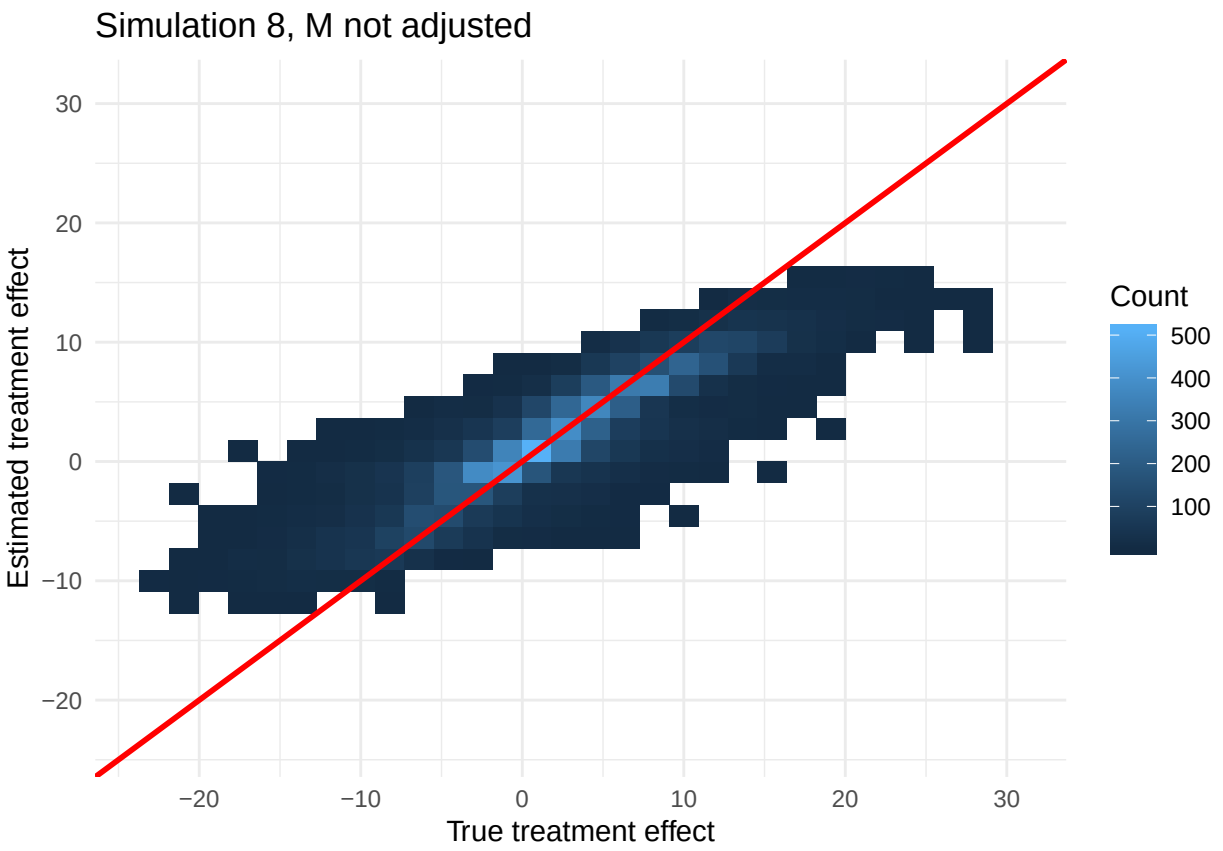
The *proportional difference in the ATE estimates* for the whole population:

```
fun.diff_ATE(ATE_est = ATE_est_9.no_M, ATE_true = mean(tau_2))
```

```
## [1] "Proportional mean of the differences:"
##   estimate
## -0.0333305
## [1] "Proportional mean of the differences, 95 % confidence intervals:"
##    estimate    estimate
## -0.14467221  0.07801121
```

```
true_vs_pred_9.no_M <- as.data.frame(cbind(tau_2, pred_tau_9.no_M$predictions))

colnames(true_vs_pred_9.no_M) <- c("tau", "pred_tau")

plot19<- ggplot(data = true_vs_pred_9.no_M, aes(x = tau, y = pred_tau)) +
  geom_bin2d() +
  geom_abline(slope = 1, intercept = 0, colour = "red", size = 1) +
  theme_minimal() +
  labs(title = "Simulation 8, M not adjusted",
       x = "True treatment effect", y = "Estimated treatment effect", fill = "Count") +
  lims(x = c(min(true_vs_pred_9.no_M), max(true_vs_pred_9.no_M)),
       y = c(min(true_vs_pred_9.no_M), max(true_vs_pred_9.no_M)))

plot19
```

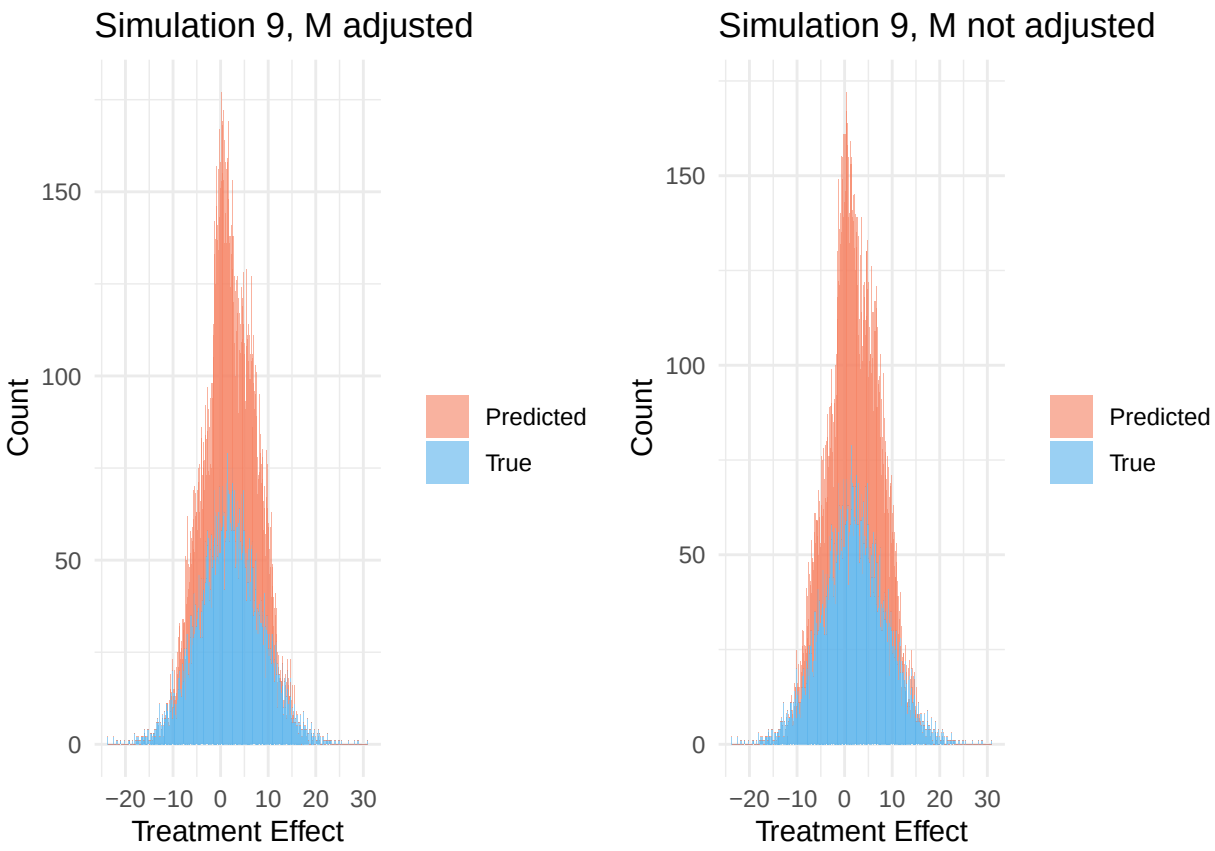
```
## Warning: Removed 2 rows containing missing values (geom_tile).
```



Summary

Predicted $\hat{\tau}s$:

```
grid.arrange(plot_pred_tau_18, plot_pred_tau_19, nrow = 1)
```

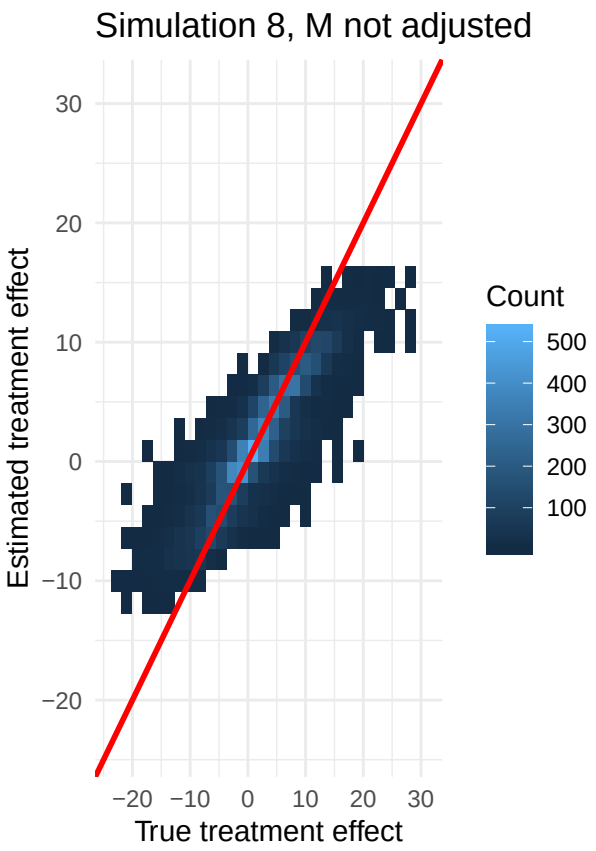


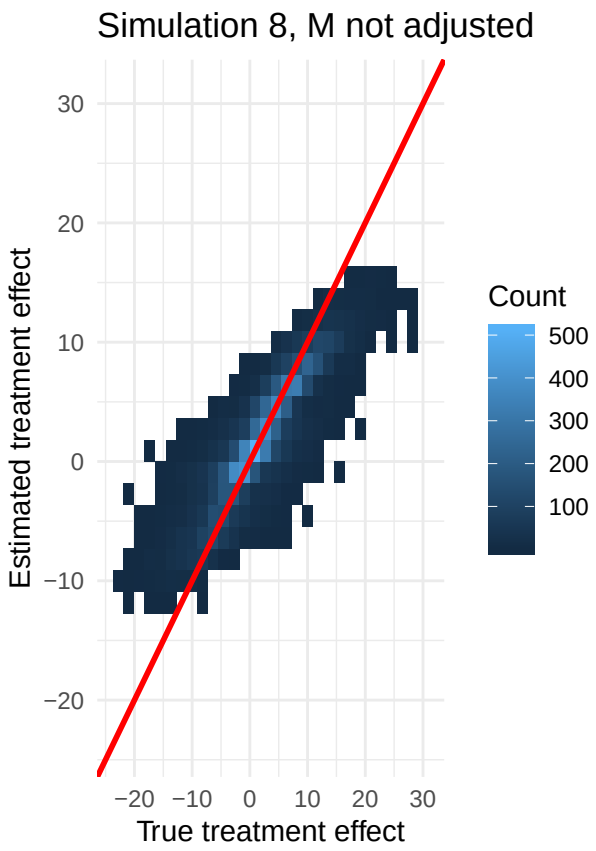
True τ vs. predicted $\hat{\tau}$:

```
grid.arrange(plot18, plot19, nrow = 1)
```

```
## Warning: Removed 2 rows containing missing values (geom_tile).
```

```
## Warning: Removed 2 rows containing missing values (geom_tile).
```





RMSEs:

```
rmse_9 <- as.data.frame(c(rmse_9.with_M, rmse_9.no_M))

rownames(rmse_9) <- c("$M$ adjusted", "$M$ not adjusted")

colnames(rmse_9) <- c("RMSE")

knitr::kable(rmse_9, escape = FALSE)
```

	RMSE
M adjusted	3.291505
M not adjusted	3.279869

Coverages:

```
coverage_9 <- as.data.frame(c(coverage_9.with_M, coverage_9.no_M))

rownames(coverage_9) <-  c("$M$ adjusted", "$M$ not adjusted")

colnames(coverage_9) <- c("Coverage")

knitr::kable(coverage_9, escape = FALSE)
```

	Coverage
M adjusted	0.6589
M not adjusted	0.6664

LaTeX:

```
knitr::kable(rmse_9, format = "latex",escape = FALSE)
```

	RMSE
M adjusted	3.291505
M not adjusted	3.279869

```
knitr::kable(coverage_9, format = "latex",escape = FALSE)
```

	Coverage
M adjusted	0.6589
M not adjusted	0.6664