Reproduce simulations and plots

Bénédicte Colnet[*] *and Imke Mayer*[†]
Other contributors[‡]

October 2020

Abstract

This notebook allows to reproduce the simulations and plots presented in the review **Causal inference methods for combining randomizedtrials and observational studies: a review**. Note that this notebook does not reproduce the Calibration Weighting results as the paper and package (coming, and called genRCT) from Lin Dong is under review.

Contents


```r
knitr::opts_chunk$set(echo = TRUE, verbose = FALSE, warning = FALSE, message=FALSE, cache = TRUE)

# Clear any existing variables
rm(list = ls())

# Set seed for reproducibility
set.seed(1234)

# Load implemented functions
source('./estimators_and_simulations-wo-cw.R')
```

```
## Warning: package 'reshape2' was built under R version 3.6.2
```

```r
# Libraries
library(ggplot2) # plots
library(dplyr) # data frame re arrangment
library(table1) # table for baseline
library(wesanderson) # colors

# number of repetitions in simulation
repetitions = 100 #(Choose 100 to reproduce exact plots of the publication, except for CW)
```

[*]Inria, benedicte.colnet@inria.fr

[†]EHESS, imke.mayer@ehess.fr

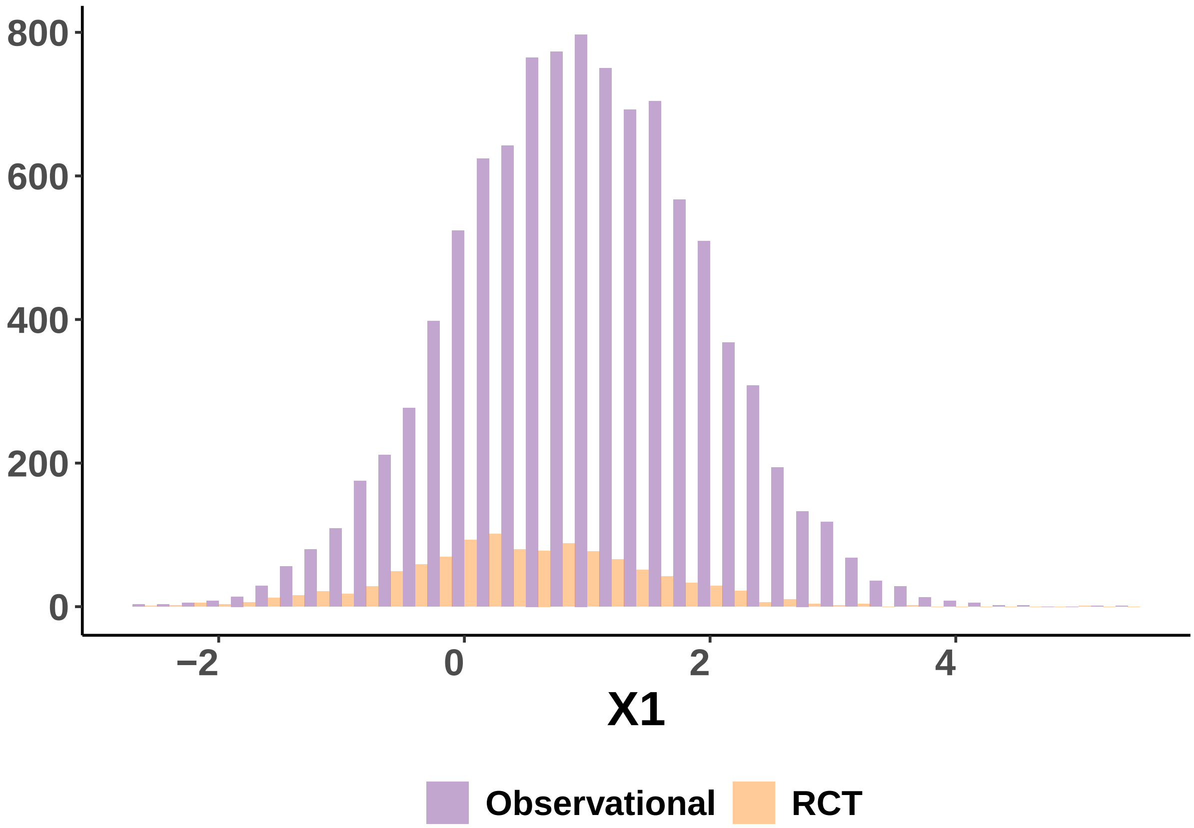
[‡]Others contributors to this notebook through reviewing or active discussions: Julie Josse, Gael Varoquaux, Jean-Philippe Vert, Shu Yang.

Distributional shift

```
one_simulation <- simulate_continuous(n = 1000, m = 10000)

one_simulation$sample <- ifelse(one_simulation$V == 1, "RCT", "Observational")
baseline <- table1(~ X1 + X2 + X3 + X4 | sample, data = one_simulation, overall="Total")
baseline
```

```
## [1] "<table class=\"Rtable1\">\n<thead>\n<tr>\n<th class='rowlabel firstrow lastrow'></th>\n<th clas
```

```
ggplot(one_simulation, aes(x = X1, group = sample, fill = sample)) +
    geom_histogram(binwidth = 0.2, alpha=0.4, position="dodge") +
        scale_fill_manual(values=c("darkorchid4", "darkorange1")) +
    theme_classic() +
    theme(legend.title = element_blank(), legend.position = "bottom",
          legend.box = "horizontal", legend.text = element_text(size=13,
                                                     face="bold")) +
  ylab("") +  # no title in legend
     theme(axis.text = element_text(vjust = 0.5, hjust=1, size=14, face="bold"),
           axis.title.x = element_text(size=18, face="bold"))
```

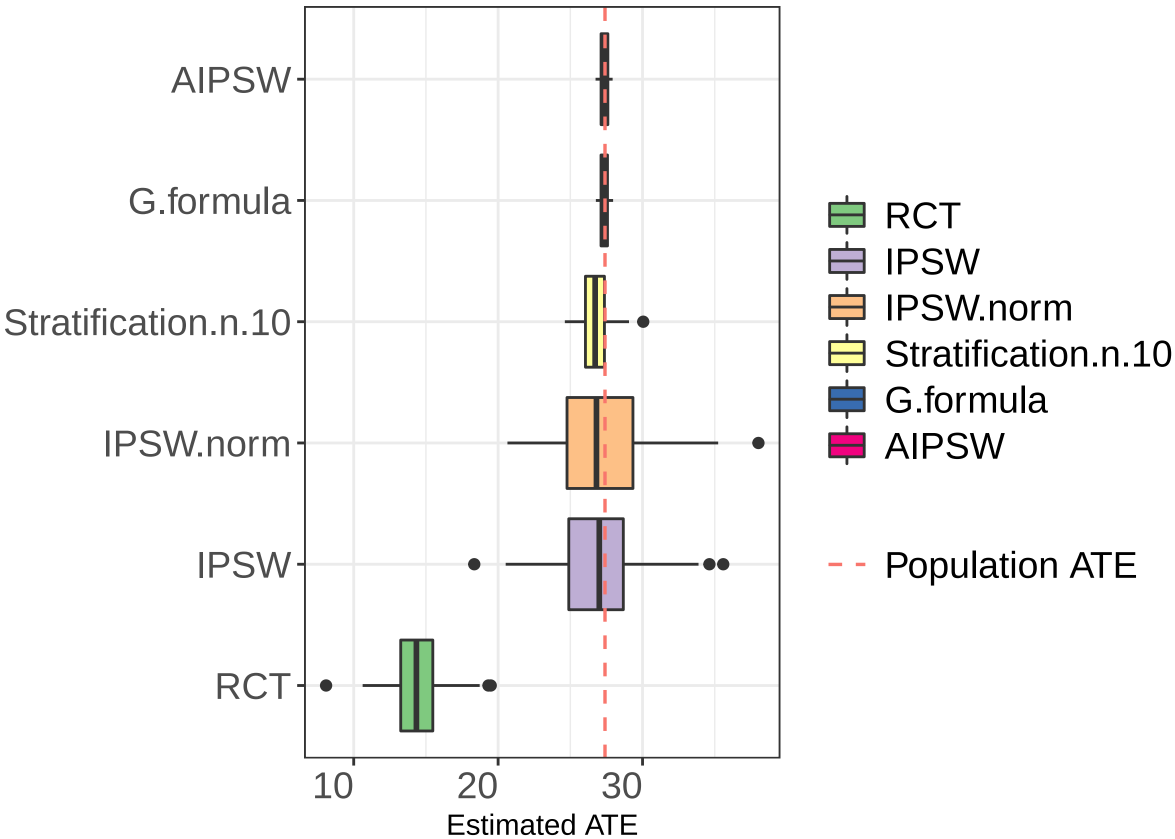


Standard simulation

```
results <- compute_estimators_and_store(rep = repetitions, n = 1000, m = 10000)
```

```
ggplot(data = melt(results), aes(x = variable, y = value)) +
    geom_boxplot(aes(fill=variable)) +
```

```
    theme_bw() +
    geom_hline(aes(yintercept = 27.4, color = "Population ATE"),
               size = 0.6, linetype="dashed") +
    xlab("") +
    ylab("Estimated ATE")  +
    theme(legend.title = element_blank(),
          legend.text = element_text(size=14)) +
    theme(axis.text = element_text(angle = 0, vjust = 0.5,
                                   hjust=1, size=14)) +
    scale_fill_brewer(palette = "Accent") +
    coord_flip()
```



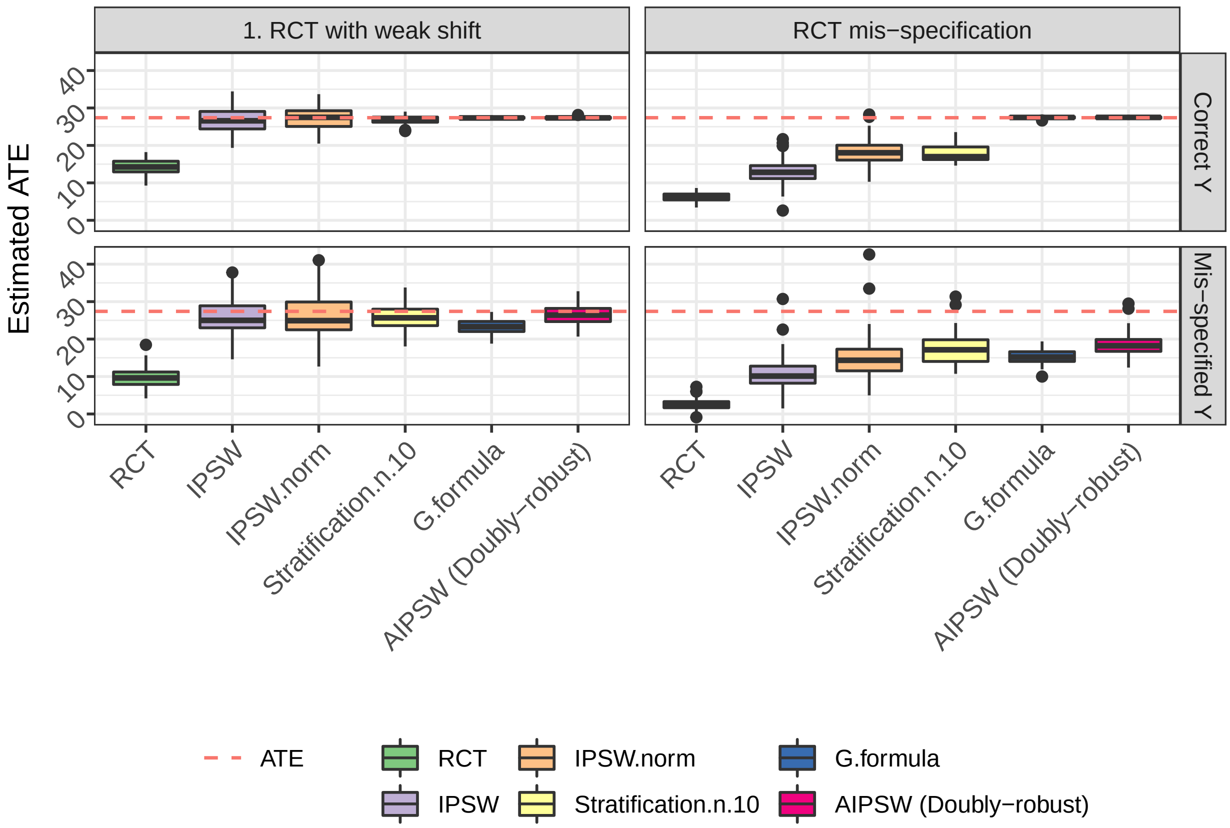
Systematic analysis

```
RCT_param <- c("correct", "strongbias", "exponential")
Outcome_param <- c("correct", "wrong")

total_results <- compute_estimators_and_store(rep = repetitions)
total_results$n = 1000
total_results$m = 49000
total_results$param_RCT = "correct"
total_results$outcome = "correct"

for (m in c(10000)){
  for (rct_param in RCT_param){
```

```r
      for (outcome_param in Outcome_param){
        results <- compute_estimators_and_store(rep = repetitions,
                                                n = 1000, m = m,
                                                misRCT = rct_param,
                                                misoutcome = outcome_param)
        results$n <- rep(1000, nrow(results))
        results$m <- rep(m, nrow(results))
        results$param_RCT <- rep(rct_param, nrow(results))
        results$outcome <- rep(outcome_param, nrow(results))
        total_results <- rbind(total_results, results)
      }
  }
}

data <- total_results[2:nrow(total_results),]
```

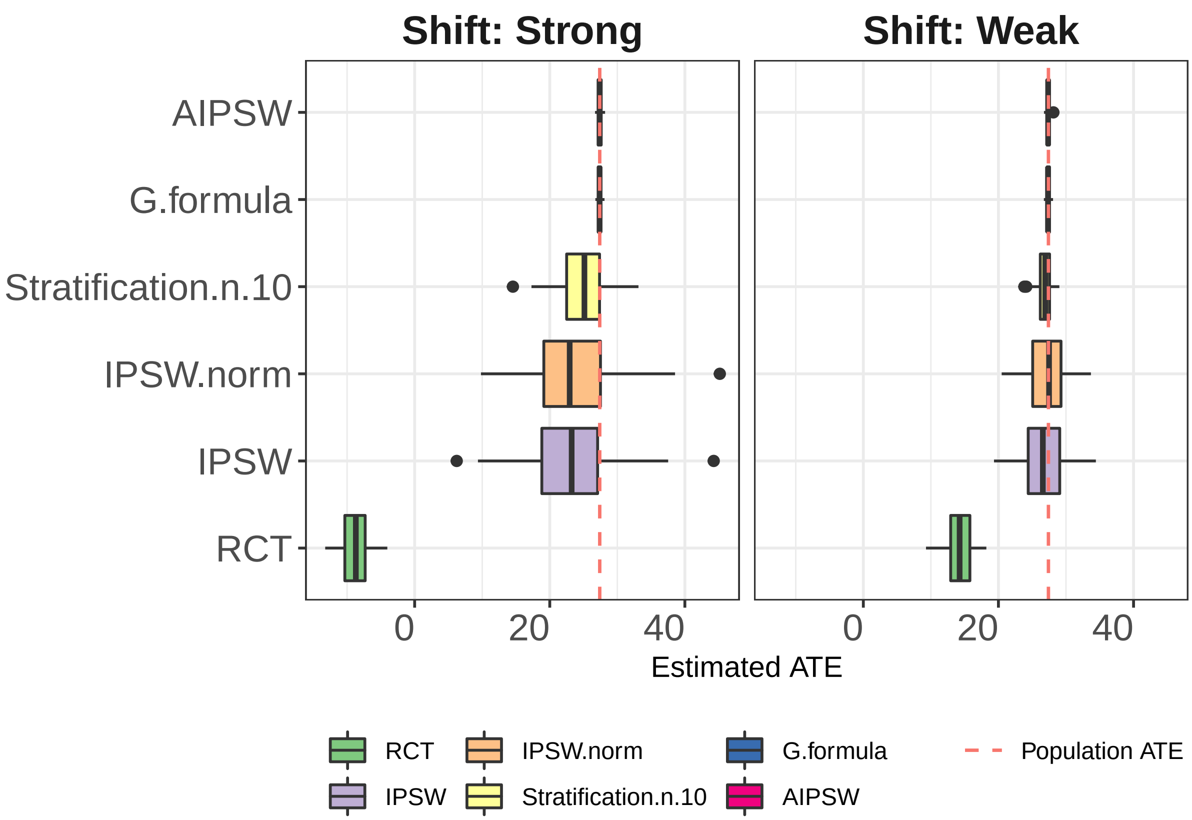
```r
data$relative.size <- ifelse(data$m == 10000, "10%", "other")

data_bis <- data
colnames(data_bis)[colnames(data_bis) == 'AIPSW'] <- 'AIPSW (Doubly-robust)'

DF <- melt(data_bis ,
           id.vars = c("param_RCT", "outcome", "relative.size"),
           measure.vars = c("RCT", "IPSW", "IPSW.norm",
                            "Stratification.n.10", "G.formula",
                            "AIPSW (Doubly-robust)"))

DF$param_RCT <- ifelse(DF$param_RCT == "correct",
                       "1. RCT with weak shift",
                       ifelse(DF$param_RCT == "exponential",
                              "RCT mis-specification",
                              "2. RCT with strong shift"))

DF$outcome <- ifelse(DF$outcome == "correct", "Correct Y", "Mis-specified Y")
```

```r
ggplot(data = DF[DF$relative.size == "10%" & DF$param_RCT != "2. RCT with strong shift",],
       aes(x = variable, y = value)) +
    geom_boxplot(aes(fill=variable)) +
    facet_grid(outcome~param_RCT) +
    theme_bw() +
    geom_hline(aes(yintercept = 27.4, color = "ATE"), size = 0.6, linetype="dashed") +
    xlab("") +
    ylab("Estimated ATE")  +
    theme(legend.title = element_blank(),
          legend.position="bottom", legend.box = "horizontal") +  # no title in legend
    theme(axis.text = element_text(angle = 45, vjust = 0.5, hjust=1, size=10)) +
  scale_fill_brewer(palette = "Accent")
```




```
data$relative.size <- ifelse(data$m == 10000, "10%", "other")

DF <- melt(data, id.vars = c("param_RCT", "outcome", "relative.size"),
           measure.vars = c("RCT", "IPSW", "IPSW.norm",
                            "Stratification.n.10", "G.formula", "AIPSW"))

DF$param_RCT <- ifelse(DF$param_RCT == "correct",
                       "Shift: Weak",
                       ifelse(DF$param_RCT == "exponential",
                              "RCT mis-specification",
                              "Shift: Strong"))
DF$outcome <- ifelse(DF$outcome == "correct", "Correct Y", "Mis-specified Y")



ggplot(data = DF[DF$relative.size == "10%" &
                   DF$param_RCT !="RCT mis-specification"&
                   DF$outcome == "Correct Y",],
       aes(x = variable, y = value)) +
      geom_boxplot(aes(fill=variable)) +
      facet_wrap(~param_RCT) +
#     #geom_jitter(alpha = 0.2, size = 0.2, width = 0.2)  +
    theme_bw() +
    geom_hline(aes(yintercept = 27.4, color = "Population ATE"), size = 0.6, linetype="dashed") +
    xlab("") +
    ylab("Estimated ATE")  +
    theme(legend.title = element_blank(),
```

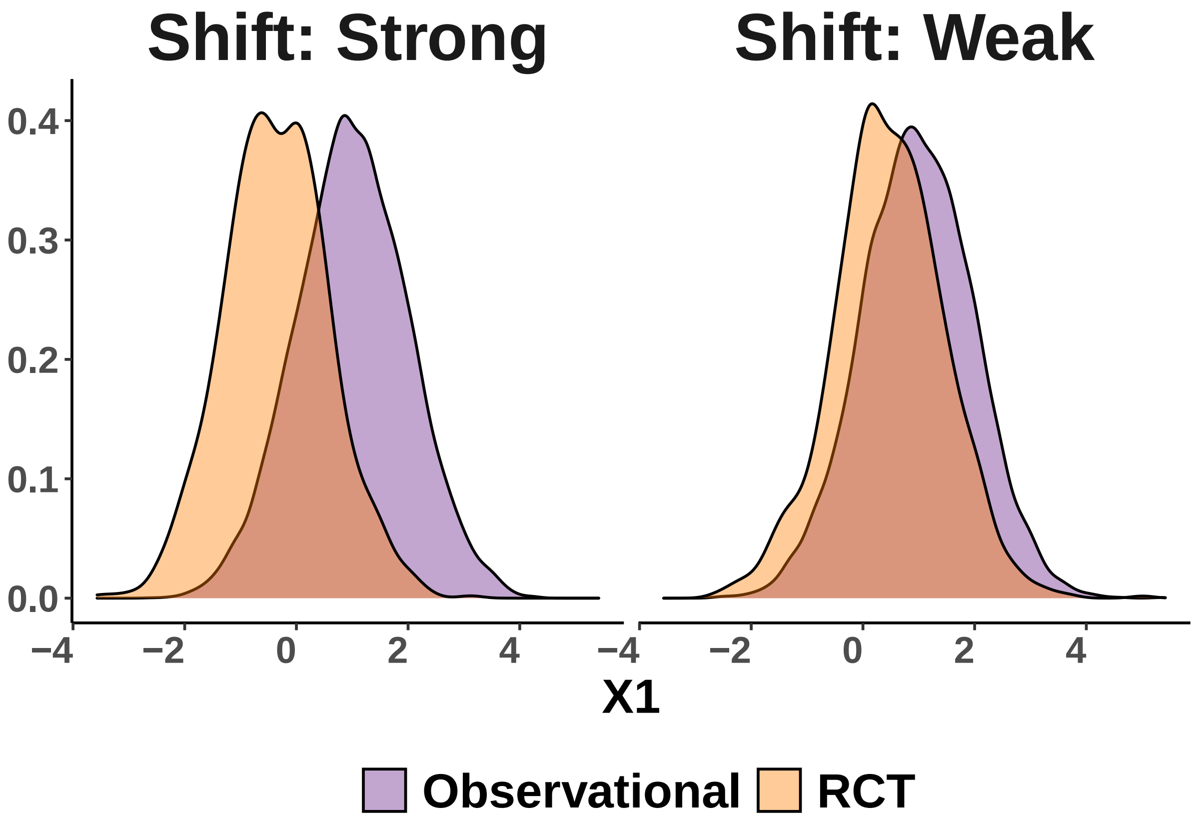
```
        legend.text = element_text(size=9), legend.position="bottom") +
    theme(axis.text = element_text(angle = 0, vjust = 0.5, hjust=1, size=14)) +
          scale_fill_brewer(palette = "Accent") +
   coord_flip() +
  theme(strip.background=element_rect(fill=NA, color=NA),
        strip.text=element_text(size=15, face = "bold"))
```




```
one_shifted_simulation <- simulate_continuous(n = 1000, m = 10000, misRCT = "strongbias")
one_shifted_simulation $sample <- ifelse(one_shifted_simulation $V == 1, "RCT", "Observational")
```

```
one_shifted_simulation$Shift <- rep("Shift: Strong", nrow(one_shifted_simulation))
one_simulation$Shift <- rep("Shift: Weak", nrow(one_simulation))

shift_comparison <- rbind(one_simulation, one_shifted_simulation)

ggplot(shift_comparison, aes(x = X1, group = sample, fill = sample)) +
    #geom_histogram(binwidth = 0.2, alpha=0.4, position="dodge") +
    geom_density(alpha=0.4, position="dodge") +
    scale_fill_manual(values=c("darkorchid4", "darkorange1")) +
    theme_classic() +
    theme(legend.title = element_blank(),
          legend.position = "bottom",
          legend.box = "horizontal",
          legend.text = element_text(size=18, face="bold")) +
     ylab("") +
     theme(axis.text = element_text(vjust = 0.5, hjust=1, size=14, face="bold"), axis.title.x = element_
  facet_grid(~Shift)  +
  theme(strip.background=element_rect(fill=NA, color=NA),
```

```
      strip.text=element_text(size=25, face = "bold"))
```

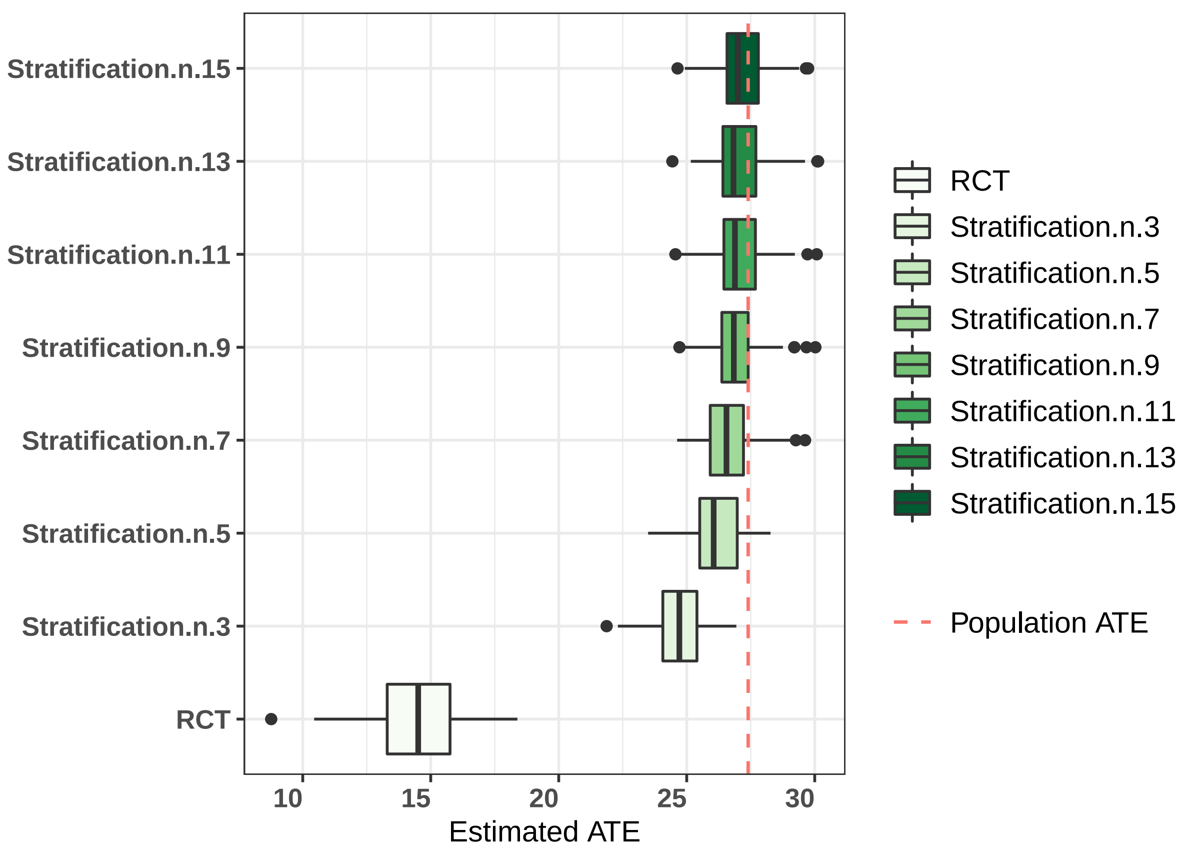


Strata effect

```
RCT <- c()
Stratification.n.3 <- c()
Stratification.n.5 <- c()
Stratification.n.7 <- c()
Stratification.n.9 <- c()
Stratification.n.11 <- c()
Stratification.n.13 <- c()
Stratification.n.15 <- c()
for (i in 1:repetitions){
  DF <- simulate_continuous(n = 1000, m = 10000)
  RCT <- c(RCT, compute_mean_diff_RCT(DF))
  Stratification.n.3 <- c(Stratification.n.3, compute_stratification(DF, nb_strat = 3))
  Stratification.n.5 <- c(Stratification.n.5, compute_stratification(DF, nb_strat = 5))
  Stratification.n.7 <- c(Stratification.n.7, compute_stratification(DF, nb_strat = 7))
  Stratification.n.9 <- c(Stratification.n.9, compute_stratification(DF, nb_strat = 9))
  Stratification.n.11 <- c(Stratification.n.11, compute_stratification(DF, nb_strat = 11))
  Stratification.n.13 <- c(Stratification.n.13, compute_stratification(DF, nb_strat = 13))
  Stratification.n.15 <- c(Stratification.n.15, compute_stratification(DF, nb_strat = 15))
}

results_strata <- data.frame(RCT, Stratification.n.3,
                             Stratification.n.5, Stratification.n.7,
```

```
                                        Stratification.n.9,Stratification.n.11,
                                        Stratification.n.13, Stratification.n.15)
```

```
DF <- melt(results_strata,
           measure.vars = c("RCT", "Stratification.n.3",
                            "Stratification.n.5","Stratification.n.7",
                            "Stratification.n.9", "Stratification.n.11",
                            "Stratification.n.13", "Stratification.n.15"))

ggplot(data = DF, aes(x = variable, y = value)) +
    geom_boxplot(aes(fill=variable)) +
    theme_bw() +
    geom_hline(aes(yintercept = 27.4, color = "Population ATE"), size = 0.6, linetype="dashed") +
    xlab("") +
    ylab("Estimated ATE")  +
    theme(legend.title = element_blank(),
          legend.text = element_text(size=11)) +  # no title in legend
     theme(axis.text = element_text(angle = 0, vjust = 0.5, hjust=1, size=10, face="bold")) +
            scale_fill_brewer(palette = "viridix") +
  coord_flip()
```



Focus on X1 and IPSW

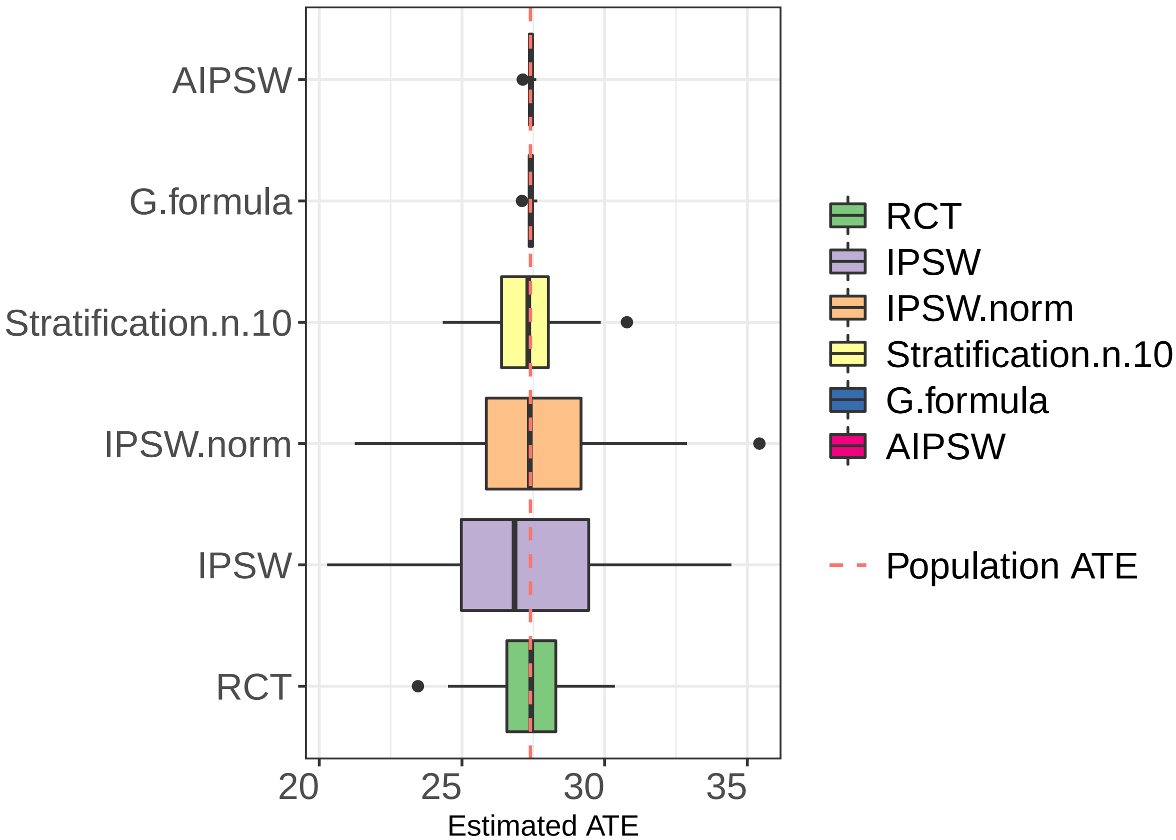
```
rct_ate <- c()
ipsw <- c()
```

```r
ipsw_x1_only <- c()
ipsw_wo_x1 <- c()
gformula <- c()

for (i in 1:repetitions){
  DF <- simulate_continuous(n = 1000, m = 10000)

  # naive estimator
  rct_ate <- c(rct_ate,
               mean(DF[DF$A == 1 & DF$V == 1, "Y"]) -
              mean(DF[DF$A == 0  & DF$V == 1, "Y"]))

  #ipsw
  ipsw  <- c(ipsw, compute_ipsw(DF, normalized = FALSE))

  #ipsw with X1 only
  ipsw_x1_only <- c(ipsw_x1_only, compute_ipsw(DF, normalized = FALSE, covariates = "X1"))

  #ipsw without X1
  ipsw_wo_x1 <- c(ipsw_wo_x1, compute_ipsw(DF, normalized = FALSE, covariates = "-X1"))

  #gformula
  gformula <- c(gformula, compute_gformula(DF))

}

results_ipsw <- data.frame("RCT" = rct_ate,
                      "IPSW" = ipsw,
                      "IPSW-X1" = ipsw_x1_only,
                      "IPSW-without-X1" = ipsw_wo_x1,
                      "G.formula" = gformula)
```

Homogeneous treatment effect

```r
results_simple <- compute_estimators_and_store(rep = repetitions, misoutcome = "+a")
```

```r
ggplot(data = melt(results_simple), aes(x = variable, y = value)) +
    geom_boxplot(aes(fill=variable)) +
    theme_bw() +
    geom_hline(aes(yintercept = 27.4, color = "Population ATE"),
               size = 0.6, linetype="dashed") +
    xlab("") +
    ylab("Estimated ATE")  +
    theme(legend.title = element_blank(), legend.text = element_text(size=14)) +
    theme(axis.text = element_text(angle = 0, vjust = 0.5, hjust=1, size=14)) +
    scale_fill_brewer(palette = "Accent") +
    coord_flip()
```



X1 effect

```
rct_ate <- c()
ipsw <- c()
ipsw_x1_only <- c()
ipsw_wo_x1 <- c()
gformula <- c()

for (i in 1:repetitions){
  DF <- simulate_continuous(n = 1000, m = 10000)

  # naive estimator
  rct_ate <- c(rct_ate,
               mean(DF[DF$A == 1 & DF$V == 1, "Y"]) -
              mean(DF[DF$A == 0  & DF$V == 1, "Y"]))

  #ipsw
  ipsw  <- c(ipsw, compute_ipsw(DF, normalized = FALSE))

  #ipsw with X1 only
  ipsw_x1_only <- c(ipsw_x1_only, compute_ipsw(DF, normalized = FALSE, covariates = "X1"))

  #ipsw without X1
  ipsw_wo_x1 <- c(ipsw_wo_x1, compute_ipsw(DF, normalized = FALSE, covariates = "-X1"))
```

```r
  #gformula
  gformula <- c(gformula, compute_gformula(DF))

}

results_ipsw <- data.frame("RCT" = rct_ate,
                           "IPSW" = ipsw,
                           "IPSW-X1" = ipsw_x1_only,
                           "IPSW-without-X1" = ipsw_wo_x1,
                           "G.formula" = gformula)
```

```r
ggplot(data = melt(results_ipsw), aes(x = variable, y = value)) +
    geom_boxplot(aes(fill=variable)) +
    theme_bw() +
    geom_hline(aes(yintercept = 27.4, color = "Population ATE"),
               size = 0.6, linetype="dashed") +
    xlab("") +
    ylab("Estimated ATE")  +
    theme(legend.title = element_blank(), legend.text = element_text(size=14)) +
    theme(axis.text = element_text(angle = 0, vjust = 0.5, hjust=1, size=14)) +
    coord_flip()
```