Mediation Workshop

Benoît Lepage

2023-02-17

Contents

Chapter 1

Introduction

The objective of this document is to provide practical examples for the Expanse report *"Mediation Analysis: a Starting Guide for Epidemiologists"* with R scripts corresponding to the different estimation methods presented in the report.

Chapter 2

Software

The examples given in this workshop have been elaborated for R (version 4.2.2).

Depending on the estimator, some R packages might be necessary:

- COMPLETE ONCE ALL THE EXAMPLES ARE DONE

Chapter 3

Data sets

3.1 General presentation of the data used in our examples

Four data sets have been simulated, each containing 7 variables:

- 2 baseline confounders (denoted $L(0)$ in the DAGs):
    - `L0_male`, a binary variable indicating the sex of the participant (1 for men, 0 for women);
    - `L0_parent_low_educ_lv`, a binary variable indicated if the parents of the participants had a low level of education (1 for a low educational level, 0 for a high educational level);
- 1 exposure of interest (denoted A in the DAGs):
    - `A0_ace`, a binary variable indicating if the participants had been exposed to a high level of "Adverse childhood experience";
- 1 confounder of the mediator-outcome relationship (denoted $L(1)$ in the DAGs):
    - `L1`, a binary variable indicating if the participant has a low educational level (1 for a low educational level, 0 for a high educational level);
- 1 mediator of interest (denoted M in the DAGs):
    - `M_smoking`, a binary variable indicating if the participant is a smoker (1 for smokers, 0 for non-smokers);
- 2 outcomes (denoted Y in the DAGs):

- **Y2_death**, a binary variable indicating the occurrence of death before 60 years of age (1 if dead, 0 if alive);
- **Y2_qol**, a quantitative variable corresponding to a quality of life measurement.

3.2   Data generating mechanisms

The 4 data generating mechanisms used to simulate the data sets are described in chapter 4 of the *Expanse "Mediation analysis" report*:

- The first two data sets are simulated from a causal model where confounders of the mediator-outcome relationship ($L(1)$) are not affected by the exposure A (Figure 3.1),

    - The data set **df1.csv** is simulated from the statistical model $\mathcal{M}_1$, which does not contain any $A * M$ interaction effect on the outcome Y.
    - The data set **df1_int.csv** is simulated from the statistical model $\mathcal{M}_{1*}$, which contains an $A * M$ interaction effect on the outcome Y.

DAG of models $\mathcal{M}_1$ and $\mathcal{M}_{1*}$

Figure 3.1: Causal model 1

- The next two data set are simulated from a causal model where confounders of the mediator-outcome relationship ($L(1)$) are affected by the exposure A (Figure 3.2),

    - The data set **df2.csv** is simulated from the statistical model $\mathcal{M}_2$, which does not contain any $A * M$ interaction effect on the outcome Y.
    - The data set **df2_int.csv** is simulated from the statistical model $\mathcal{M}_{2*}$, which contains an $A * M$ interaction effect on the outcome Y.

The R functions used to simulate these 4 data sets are given in the Appendix A.
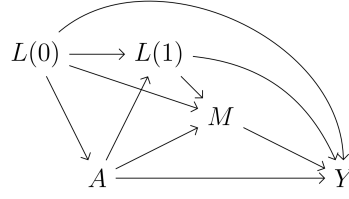
DAG of models $\mathcal{M}_2$ and $\mathcal{M}_{2*}$

Figure 3.2: Causal model 2

The Appendix B describes how the true values for the estimands of the causal quantities of interest given in Table 2 of the *Expanse "Mediation analysis" report* were calculated. Those true values are the theoretical values expected under the causal and statistical models $\mathcal{M}_1$, $\mathcal{M}_{1*}$, $\mathcal{M}_2$ and $\mathcal{M}_{2*}$. Estimations that will be obtained from the data sets `df1.csv`, `df1_int.csv`, `df2.csv`, and `df2_int.csv` will be slightly different from the true values because of sample variability.

Chapter 4

Baron and Kenny, structural equation models

Chapter 5

Traditional regression models

Traditional regression models can be applied in the absence of an intermediate confounder $L(1)$ of the $M - Y$ relationship affected by the exposure A (Causal model 1). They can be used for two-way, three-way and four-way decomposition of the average total effect.

In the following examples, we use the `df1_int.csv` data set with a $A \star M$ interaction effect on the outcome.

```
df1_int <- read.csv(file = "df1_int.csv")
```

If we assumed that there was no $A \star M$ interaction, then the `A0_ace:M_smoking` interaction terms should be removed from the models below (applicable if we use the `df1.csv` data set).

5.1 Estimation of the Average Total Effect (ATE)

The average total effect is the difference between the mean outcome had the whole population been exposed to adverse childhood experience (ACE), compared to the mean outcome had the whole population been unexposed to ACE: $\text{ATE} = \mathbb{E}(Y_{A=1}) - \mathbb{E}(Y_{A=0})$.

For the quantitative outcome, the ATE of the adverse childhood experience A on the quality of life score Y can be estimated using a traditional linear regression of `Y_qol` on `A0_ace`, adjusted for the baseline confounders `L0_male` and `L0_parent_low_educ_lv`.

For the binary outcome (death), we can estimate a risk difference applying a Generalized Linear Model with a Gaussian distribution and identity link, as suggested by Naimi *et al* (Naimi and Whitcomb 2020).

The regression coefficient of the exposure variable A is used to estimate the risk difference or the average difference.

$$\mathbb{E}(Y \mid A, L(0)) = \alpha_0 + \alpha_A A + \alpha_{L(0)} L(0) \tag{5.1}$$

$$\hat{\Psi}_{\text{trad}}^{\text{ATE}} = \hat{\alpha}_A$$

```r
# For quantitative outcomes, apply a linear regression of Y on A (A0_ace),
# adjusted for the baseline confounders L(0):
trad_ATE_qol <- lm(Y_qol ~ A0_ace + L0_male + L0_parent_low_educ_lv,
                   data = df1_int)

# For binary outcomes, apply a GLM of Y on A with a Gaussian distribution and
# identity link, adjusted for the baseline confounders:
trad_ATE_death <- glm(Y_death ~ A0_ace + L0_male + L0_parent_low_educ_lv,
                      family = gaussian("identity"),
                      data = df1_int)

# Use the regression coefficient of the exposure (A0_ace) to estimate the ATE
ATE_trad_qol <- coefficients(trad_ATE_qol)["A0_ace"]
# -7.210089

ATE_trad_death <- coefficients(trad_ATE_death)["A0_ace"]
# 0.07720726
```

The estimation of 95% confidence intervals could be obtained directly from the linear regression with quantitative outcomes (equation (5.1)). However, using a robust (sandwich) variance estimator or applying a bootstrap procedure is recommended (Naimi and Whitcomb 2020).

```r
library(sandwich)

# for the Quality of Life outcome
ATE_trad_qol <- list(ATE = coef(trad_ATE_qol)["A0_ace"],
                     lo = coef(trad_ATE_qol)["A0_ace"] - qnorm(0.975) *
                       sqrt(sandwich(trad_ATE_qol)["A0_ace","A0_ace"]),
                     hi = coef(trad_ATE_qol)["A0_ace"] + qnorm(0.975) *
                       sqrt(sandwich(trad_ATE_qol)["A0_ace","A0_ace"]))

ATE_trad_qol
# ATE = -7.210089 , IC95% = [-7.978234 ; -6.441944]
```

```
# for death outcome
ATE_trad_death <- list(ATE = coef(trad_ATE_death)["A0_ace"],
                       lo = coef(trad_ATE_death)["A0_ace"] - qnorm(0.975) *
                         sqrt(sandwich(trad_ATE_death)["A0_ace","A0_ace"]),
                       hi = coef(trad_ATE_death)["A0_ace"] + qnorm(0.975) *
                         sqrt(sandwich(trad_ATE_death)["A0_ace","A0_ace"]))
ATE_trad_death
# ATE = 0.07720726 , IC95% = [0.04945859 ; 0.1049559]

# 95% CI calculation applying a bootstrap procedure
library(boot)
bootfunc <- function(data,index){
  boot_dat <- data[index,]
  mod.qol <- lm(Y_qol ~ A0_ace + L0_male + L0_parent_low_educ_lv,
                data = boot_dat)
  mod.death <- glm(Y_death ~ A0_ace + L0_male + L0_parent_low_educ_lv,
                   family = gaussian("identity"),
                   data = boot_dat)
  est <- c(coef(mod.qol)["A0_ace"],
           coef(mod.death)["A0_ace"])
  return(est)
}

set.seed(1234)
boot_est <- boot(df1_int,bootfunc,R=2000)

# the 95% CI for the estimation of the ATE of ACE on QoL is:
boot.ci(boot_est, index = 1, type = "norm")
# (-7.978, -6.444 )

# the 95% CI for the estimation of the ATE of ACE on death is:
boot.ci(boot_est, index = 2, type = "norm")
# ( 0.0502,  0.1040 )
```

Alternatively for binary outcomes, the total effect conditional on baseline confounders can be expressed on an Odds Ratio scale OR^{TE}, using the logistic regression (5.2).

$$\text{logit}\,P(Y = 1 \mid A, L(0)) = \alpha_0 + \alpha_A A + \alpha'_{L(0)} L(0) \tag{5.2}$$

$$\text{OR}^{\text{TE}} \mid L(0) = \exp \hat{\alpha}_A$$

```r
TE_death_model <- glm(Y_death ~ A0_ace + L0_male + L0_parent_low_educ_lv,
                      family = "binomial",
                      data = df1_int)
res_TE_death <- summary(TE_death_model)
tot.effect.death.OR <- list(OR = exp(coef(res_TE_death)["A0_ace","Estimate"]),
                            lo = exp(coef(res_TE_death)["A0_ace","Estimate"] -
                                      qnorm(0.975) *
                                      coef(res_TE_death)["A0_ace","Std. Error"]),
                            hi = exp(coef(res_TE_death)["A0_ace","Estimate"] +
                                      qnorm(0.975) *
                                      coef(res_TE_death)["A0_ace","Std. Error"]))
tot.effect.death.OR
# OR = 1.523254 , 95% CI = [ 1.323317 ; 1.753398]
```

5.2   Two-way decomposition

In order to carry-out two-way decomposition mediation analyses, with a binary mediator and a continuous outcome, Valeri and VanderWeele suggest using the following linear regression of the outcome and logistic regression of the mediator:(Valeri and VanderWeele 2013)

$$\mathbb{E}(Y \mid A, M, L(0), L(1)) = \gamma_0 + \gamma_A A + \gamma_M M + \gamma_{A*M}(A*M) + \gamma'_{L(0)}L(0) + \gamma'_{L(1)}L(1) \tag{5.3}$$

$$\mathrm{logit}P(M = 1 \mid A, L(0), L(1)) = \beta_0 + \beta_A A + \beta'_{L(0)}L(0) + \beta'_{L(1)}L(1) \tag{5.4}$$

If the outcome is binary, they suggest using the following logistic regression of the outcome instead of the previous linear regression:

$$\mathrm{logit}P(Y \mid A, M, L(0), L(1)) = \gamma_0 + \gamma_A A + \gamma_M M + \gamma_{A*M}(A*M) + \gamma'_{L(0)}L(0) + \gamma'_{L(1)}L(1) \tag{5.5}$$

```r
trad_qol_am <- lm(Y_qol ~ A0_ace + M_smoking + A0_ace:M_smoking +
                    L0_male + L0_parent_low_educ_lv + L1,
                  data = df1_int)
gamma.A.q <- coef(trad_qol_am)["A0_ace"]
gamma.M.q <- coef(trad_qol_am)["M_smoking"]
gamma.AM.q <- coef(trad_qol_am)["A0_ace:M_smoking"]

trad_m <- glm(M_smoking ~ A0_ace + L0_male + L0_parent_low_educ_lv + L1,
              family = "binomial",
              data = df1_int)
```

```r
beta.0 <- coef(trad_m)["(Intercept)"]
beta.A <- coef(trad_m)["A0_ace"]

trad_death_am <- glm(Y_death ~ A0_ace + M_smoking + A0_ace:M_smoking +
                        L0_male + L0_parent_low_educ_lv + L1,
                     family = "binomial",
                     data = df1_int)
gamma.A.d <- coef(trad_death_am)["A0_ace"]
gamma.M.d <- coef(trad_death_am)["M_smoking"]
gamma.AM.d <- coef(trad_death_am)["A0_ace:M_smoking"]
```

5.2.1 Controlled Direct Effect

The Controlled Direct Effect is defined as $\text{CDE}_m = \mathbb{E}(Y_{A=1,M=m}) - \mathbb{E}(Y_{A=0,M=m})$:

For continuous outcome, using parameters from equation (5.3), it can be estimated by:

$$\text{CDE}_m = \hat{\gamma}_A + \hat{\gamma}_{A*M} \times m$$

```r
### For a continuous outcome
# setting the mediator to M=0
trad_CDE_qol_m0 <- gamma.A.q + gamma.AM.q * 0
trad_CDE_qol_m0
# -3.715265
# setting the mediator to M=1
trad_CDE_qol_m1 <- gamma.A.q + gamma.AM.q * 1
trad_CDE_qol_m1
# -9.330657
```

For binary outcomes, using parameters from equation (5.5), it can be estimated on the OR scale by:

$$OR^{\text{CDE}_m} = \exp\left(\hat{\gamma}_A + \hat{\gamma}_{A*M} \times m\right)$$

```r
### For a binary outcome
## setting the mediator to M=0
trad_OD_CDE_death_m0 <- exp(gamma.A.d + gamma.AM.d * 0)
trad_OD_CDE_death_m0
# OR_CDE_{M=0} = 1.442942

## setting the mediator to M=1
trad_OD_CDE_death_m1 <- exp(gamma.A.d + gamma.AM.d * 1)
trad_OD_CDE_death_m1
# OR_CDE_{M=1} = 1.461464
```

5.2.2 Natural Direct and Indirect effects

The Pure Natural Direct Effect (PNDE) and the Total Natural Indirect Effect (TNIE) are defined as:

- PNDE $= \mathbb{E}\left(Y_{A=1,M_{A=0}}\right) - \mathbb{E}(Y_{A=0,M_{A=0}})$,
- TNIE $= \mathbb{E}\left(Y_{A=1,M_{A=1}}\right) - \mathbb{E}(Y_{A=1,M_{A=0}})$.

Alternatively, one can use the Total Natural Direct Effect (TNDE) and the Pure Natural Indirect Effect (PNIE):

- TNDE $= \mathbb{E}\left(Y_{A=1,M_{A=1}}\right) - \mathbb{E}(Y_{A=0,M_{A=1}})$,
- PNIE $= \mathbb{E}\left(Y_{A=0,M_{A=1}}\right) - \mathbb{E}(Y_{A=0,M_{A=0}})$.

With a continuous outcome and a binary mediator, the PNDE and TNDE can be estimated using the linear regression of the outcome (equation (5.3)) and the logistic regression of the mediator (equation (5.4)):

$$\text{PNDE} \mid L(0), L(1) = \hat{\gamma}_A + \hat{\gamma}_{A*M} \frac{exp(\hat{\beta}_0 + \hat{\beta}_A \times 0 + \hat{\beta}'_{L(0)}L(0) + \hat{\beta}'_{L(1)}L(1))}{1 + exp(\hat{\beta}_0 + \hat{\beta}_A \times 0 + \hat{\beta}'_{L(0)}L(0) + \hat{\beta}'_{L(1)}L(1))}$$

$$\text{TNIE} \mid L(0), L(1) = (\hat{\gamma}_M + \hat{\gamma}_{A*M} \times 1) \left[\frac{exp(\hat{\beta}_0 + \hat{\beta}_A \times 1 + \hat{\beta}'_{L(0)}L(0) + \hat{\beta}'_{L(1)}L(1))}{1 + exp(\hat{\beta}_0 + \hat{\beta}_A \times 1 + \hat{\beta}'_{L(0)}L(0) + \hat{\beta}'_{L(1)}L(1))} - \frac{exp(\hat{\beta}_0 + \hat{\beta}_A \times}{1 + exp(\hat{\beta}_0 + \hat{\beta}_A}\right.$$

The alternative TNDE ad PNIE can be estimated by:

$$\text{TNDE} \mid L(0), L(1) = \hat{\gamma}_A + \hat{\gamma}_{A*M} \frac{exp(\hat{\beta}_0 + \hat{\beta}_A \times 1 + \hat{\beta}'_{L(0)}L(0) + \hat{\beta}'_{L(1)}L(1))}{1 + exp(\hat{\beta}_0 + \hat{\beta}_A \times 1 + \hat{\beta}'_{L(0)}L(0) + \hat{\beta}'_{L(1)}L(1))}$$

$$\text{PNIE} \mid L(0), L(1) = (\hat{\gamma}_M + \hat{\gamma}_{A*M} \times 0) \left[\frac{exp(\hat{\beta}_0 + \hat{\beta}_A \times 1 + \hat{\beta}'_{L(0)}L(0) + \hat{\beta}'_{L(1)}L(1))}{1 + exp(\hat{\beta}_0 + \hat{\beta}_A \times 1 + \hat{\beta}'_{L(0)}L(0) + \hat{\beta}'_{L(1)}L(1))} - \frac{exp(\hat{\beta}_0 + \hat{\beta}_A \times}{1 + exp(\hat{\beta}_0 + \hat{\beta}_A}\right.$$

Conditional on participants for which $L(0) = 0$ and $L(1) = 0$, these expressions are simplified:

$$\text{PNDE}\Big|(L(0) = 0, L(1) = 0) = \hat{\gamma}_A + \hat{\gamma}_{A*M} \frac{exp(\hat{\beta}_0)}{1 + exp(\hat{\beta}_0)}$$

$$\text{TNIE}\Big|(L(0) = 0, L(1) = 0) = (\hat{\gamma}_M + \hat{\gamma}_{A*M}) \left[\frac{exp(\hat{\beta}_0 + \hat{\beta}_A)}{1 + exp(\hat{\beta}_0 + \hat{\beta}_A)} - \frac{exp(\hat{\beta}_0)}{1 + exp(\hat{\beta}_0)} \right]$$

and

$$\text{TNDE}\Big|(L(0) = 0, L(1) = 0) = \hat{\gamma}_A + \hat{\gamma}_{A*M}\frac{exp(\hat{\beta}_0 + \hat{\beta}_A)}{1 + exp(\hat{\beta}_0 + \hat{\beta}_A)}$$

$$\text{PNIE}\Big|(L(0) = 0, L(1) = 0) = \hat{\gamma}_M \left[\frac{exp(\hat{\beta}_0 + \hat{\beta}_A)}{1 + exp(\hat{\beta}_0 + \hat{\beta}_A)} - \frac{exp(\hat{\beta}_0)}{1 + exp(\hat{\beta}_0)}\right]$$

```
### For a continuous outcome, in the subgroup with L(0)=0 and L(1)=0
## The PNDE and TNIE are:
trad_PNDE_qol <- gamma.A.q + gamma.AM.q * (exp(beta.0)) / (1 + exp(beta.0))
trad_PNDE_qol
# -4.845089

trad_TNIE_qol <- (gamma.M.q + gamma.AM.q) *
  (exp(beta.0 + beta.A) / (1 + exp(beta.0 + beta.A)) -
     exp(beta.0) / (1 + exp(beta.0)))
trad_TNIE_qol
# -1.50119

## The TNDE and PNIE are:
trad_TNDE_qol <- gamma.A.q +
  gamma.AM.q * exp(beta.0 + beta.A) / (1 + exp(beta.0 + beta.A))
trad_TNDE_qol
# -5.436773

trad_PNIE_qol <- gamma.M.q *
  (exp(beta.0 + beta.A) /
     (1 + exp(beta.0 + beta.A)) - exp(beta.0) / (1 + exp(beta.0)))
trad_PNIE_qol
# -0.9095061
```

For binary outcomes, total, direct and indirect effects can be expressed on relative risk or odds ratio scales:

The total effect risk ratio is equal to :

$$\text{RR}^{\text{TE}} = \frac{\mathbb{E}(Y_1)}{\mathbb{E}(Y_0)} = \frac{\mathbb{E}(Y_{1,M_1})}{\mathbb{E}(Y_{0,M_0})}$$

The total effect risk ratio can be decomposed as the product of the PNDE risk ratio and the TNIE risk ratio:

$$\text{RR}^{\text{TE}} = \frac{\mathbb{E}(Y_{1,M_1})}{\mathbb{E}(Y_{0,M_0})} = \frac{\mathbb{E}(Y_{1,M_0})}{\mathbb{E}(Y_{0,M_0})} \times \frac{\mathbb{E}(Y_{1,M_1})}{\mathbb{E}(Y_{1,M_0})} = \text{RR}^{\text{PNDE}} \times \text{RR}^{\text{TNIE}}$$

Similarly, the total effect risk ratio can be decomposed as the product of the TNDE risk ratio and the PNIE risk ratio:

$$\mathrm{RR}^{\mathrm{TE}} = \frac{\mathbb{E}(Y_{1,M_1})}{\mathbb{E}(Y_{0,M_0})} = \frac{\mathbb{E}(Y_{1,M_1})}{\mathbb{E}(Y_{0,M_1})} \times \frac{\mathbb{E}(Y_{0,M_1})}{\mathbb{E}(Y_{0,M_0})} = \mathrm{RR}^{\mathrm{TNDE}} \times \mathrm{RR}^{\mathrm{PNIE}}$$

PNDE, TNIE, TNDE and PNIE can also be given on the OR scale,

$$\mathrm{OR}^{\mathrm{PNDE}} = \frac{\frac{P\left(Y_{A=1,M_{A=0}}=1\right)}{1-P\left(Y_{A=1,M_{A=0}}=1\right)}}{\frac{P\left(Y_{A=0,M_{A=0}}=1\right)}{1-P\left(Y_{A=0,M_{A=0}}=1\right)}} \quad , \quad \mathrm{OR}^{\mathrm{TNIE}} = \frac{\frac{P\left(Y_{A=1,M_{A=1}}=1\right)}{1-P\left(Y_{A=1,M_{A=1}}=1\right)}}{\frac{P\left(Y_{A=1,M_{A=0}}=1\right)}{1-P\left(Y_{A=1,M_{A=0}}=1\right)}}$$

and

$$\mathrm{OR}^{\mathrm{TNDE}} = \frac{\frac{P\left(Y_{A=1,M_{A=0}}=1\right)}{1-P\left(Y_{A=1,M_{A=0}}=1\right)}}{\frac{P\left(Y_{A=0,M_{A=0}}=1\right)}{1-P\left(Y_{A=0,M_{A=0}}=1\right)}} \quad \text{and} \quad \mathrm{OR}^{\mathrm{PNIE}} = \frac{\frac{P\left(Y_{A=1,M_{A=1}}=1\right)}{1-P\left(Y_{A=1,M_{A=1}}=1\right)}}{\frac{P\left(Y_{A=1,M_{A=0}}=1\right)}{1-P\left(Y_{A=1,M_{A=0}}=1\right)}}.$$

If the outcome is rare, we have $P(Y=1) \approx \frac{P(Y=1)}{1-P(Y=1)}$ so that, $\mathrm{OR}^{\mathrm{PNDE}}$ AND $\mathrm{OR}^{\mathrm{TNIE}}$ can be estimated using the logistic model of the outcome (equation (5.5)) and the logistic model of the mediator (equation (5.4)):

$$\mathrm{OR}^{\mathrm{PNDE}} \mid L(0), L(1) \approx \frac{\exp(\hat{\gamma}_A \times 1)\left[1 + \exp(\hat{\gamma}_M + \hat{\gamma}_{A*M} \times 1 + \hat{\beta}_0 + \hat{\beta}_A \times 0 + \hat{\beta}'_{L(0)}L(0) + \hat{\beta}'_{L(1)}L(1))\right]}{\exp(\hat{\gamma}_A \times 0)\left[1 + \exp(\hat{\gamma}_M + \hat{\gamma}_{A*M} \times 0 + \hat{\beta}_0 + \hat{\beta}_A \times 0 + \hat{\beta}'_{L(0)}L(0) + \hat{\beta}'_{L(1)}L(1))\right]}$$

and

$$\mathrm{OR}^{\mathrm{TNIE}} \mid L(0), L(1) \approx \frac{\left[1 + \exp(\hat{\beta}_0 + \hat{\beta}_A \times 0 + \hat{\beta}'_{L(0)}L(0) + \hat{\beta}'_{L(1)}L(1))\right]\left[1 + \exp(\hat{\gamma}_M + \hat{\gamma}_{A*M} \times 1 + \hat{\beta}_0 + \hat{\beta}_A \right.}{\left[1 + \exp(\hat{\beta}_0 + \hat{\beta}_A \times 1 + \hat{\beta}'_{L(0)}L(0) + \hat{\beta}'_{L(1)}L(1))\right]\left[1 + \exp(\hat{\gamma}_M + \hat{\gamma}_{A*M} \times 1 + \hat{\beta}_0 + \hat{\beta}_A \right.}$$

Similarly, if the outcome is rare, $\mathrm{OR}^{\mathrm{TNDE}}$ AND $\mathrm{OR}^{\mathrm{PNIE}}$ can be estimated using the logistic regression models for the outcome and the mediator (equations (5.5) and (5.4)):

$$\mathrm{OR}^{\mathrm{TNDE}} \mid L(0), L(1) \approx \frac{\exp\left(\hat{\gamma}_A \times 1\right)\left[1 + \exp\left(\hat{\gamma}_M + \hat{\gamma}_{A*M} \times 1 + \hat{\beta}_0 + \hat{\beta}_A \times 1 + \hat{\beta}'_{L(0)}L(0) + \hat{\beta}'_{L(1)}L(1)\right)\right]}{\exp\left(\hat{\gamma}_A \times 0\right)\left[1 + \exp\left(\hat{\gamma}_M + \hat{\gamma}_{A*M} \times 0 + \hat{\beta}_0 + \hat{\beta}_A \times 1 + \hat{\beta}'_{L(0)}L(0) + \hat{\beta}'_{L(1)}L(1)\right)\right]}$$

and

$$\mathrm{OR}^{\mathrm{PNIE}} \mid L(0), L(1) \approx \frac{\left[1 + \exp\left(\hat{\beta}_0 + \hat{\beta}_A \times 0 + \hat{\beta}'_{L(0)}L(0) + \hat{\beta}'_{L(1)}L(1)\right)\right]\left[1 + \exp\left(\hat{\gamma}_M + \hat{\gamma}_{A*M} \times 0 + \hat{\beta}_0 + \hat{\beta}\right.\right.}{\left[1 + \exp\left(\hat{\beta}_0 + \hat{\beta}_A \times 1 + \hat{\beta}'_{L(0)}L(0) + \hat{\beta}'_{L(1)}L(1)\right)\right]\left[1 + \exp\left(\hat{\gamma}_M + \hat{\gamma}_{A*M} \times 0 + \hat{\beta}_0 + \hat{\beta}\right.\right.}$$

Conditional on participants for which $L(0) = 0$ and $L(1) = 0$, these expressions are simplified:

$$\mathrm{OR}^{\mathrm{PNDE}}\bigg|(L(0) = 0, L(1) = 0) \approx \frac{\exp(\hat{\gamma}_A)\left[1 + \exp\left(\hat{\gamma}_M + \hat{\gamma}_{A*M} + \hat{\beta}_0\right)\right]}{\left[1 + \exp\left(\hat{\gamma}_M + \hat{\beta}_0\right)\right]}$$

and

$$\mathrm{OR}^{\mathrm{TNIE}}\bigg|(L(0) = 0, L(1) = 0) \approx \frac{\left[1 + \exp\left(\hat{\beta}_0\right)\right] \times \left[1 + \exp\left(\hat{\gamma}_M + \hat{\gamma}_{A*M} + \hat{\beta}_0 + \hat{\beta}_A\right)\right]}{\left[1 + \exp\left(\hat{\beta}_0 + \hat{\beta}_A\right)\right] \times \left[1 + \exp\left(\hat{\gamma}_M + \hat{\gamma}_{A*M} + \hat{\beta}_0\right)\right]}$$

Similarly, conditional on $L(0) = 0$ and $L(1) = 0$,

$$\mathrm{OR}^{\mathrm{TNDE}}\bigg|(L(0) = 0, L(1) = 0) \approx \frac{\exp(\hat{\gamma}_A)\left[1 + \exp\left(\hat{\gamma}_M + \hat{\gamma}_{A*M} + \hat{\beta}_0 + \hat{\beta}_A\right)\right]}{\left[1 + \exp\left(\hat{\gamma}_M + \hat{\beta}_0 + \hat{\beta}_A\right)\right]}$$

$$\mathrm{OR}^{\mathrm{PNIE}}\bigg|(L(0) = 0, L(1) = 0) \approx \frac{\left[1 + \exp\left(\hat{\beta}_0\right)\right] \times \left[1 + \exp\left(\hat{\gamma}_M + \hat{\beta}_0 + \hat{\beta}_A\right)\right]}{\left[1 + \exp\left(\hat{\beta}_0 + \hat{\beta}_A\right)\right] \times \left[1 + \exp\left(\hat{\gamma}_M + \hat{\beta}_0\right)\right]}$$

```
### For a binary outcome, in the subgroup with L(0)=0 and L(1)=0
## The PNDE and TNIE are (on the OR scale):
trad_OR_PNDE_death <- exp(gamma.A.d) *
  (1 + exp(gamma.M.d + gamma.AM.d + beta.0 )) /
  (1 + exp(gamma.M.d + beta.0))
trad_OR_PNDE_death
# 1.448035

trad_OR_TNIE_death <- (1 + exp(beta.0)) *
  (1 + exp(gamma.M.d + gamma.AM.d + beta.0 + beta.A)) /
  ((1 + exp(beta.0 + beta.A)) * (1 + exp(gamma.M.d + gamma.AM.d + beta.0)))
trad_OR_TNIE_death
# 1.050029

## The TNDE and PNIE are (on the OR scale):
trad_OR_TNDE_death <- exp(gamma.A.d) *
  (1 + exp(gamma.M.d + gamma.AM.d + beta.0 + beta.A)) /
  (1 + exp(gamma.M.d + beta.0 + beta.A))
trad_OR_TNDE_death
# 1.450344

trad_OR_PNIE_death <- (1 + exp(beta.0)) *
  (1 + exp(gamma.M.d + beta.0 + beta.A)) /
  ((1 + exp(beta.0 + beta.A)) * (1 + exp(gamma.M.d + beta.0)))
trad_OR_PNIE_death
# 1.048358
```

The `regmedint` package (Regression-Based Causal Mediation Analysis with Interaction and Effect Modification Terms) can be used for two-way decomposition. Estimations of the CDE, PNDE, TNIE, TNDE and PNIE presented above can be obtained as we show in the following example.

For continuous outcomes:

```r
library(regmedint)
regmedint_cont <- regmedint(data = df1_int,
                            ## Variables
                            yvar = "Y_qol",                 # outcome variable
                            avar = "A0_ace",                # exposure
                            mvar = "M_smoking",             # mediator
                            cvar = c("L0_male",             # confounders
                                     "L0_parent_low_educ_lv",
                                     "L1"),
                            #eventvar = "event",     # only for survival outcome
                            ## Values at which effects are evaluated
                            a0 = 0,
                            a1 = 1,
                            m_cde = 0,               # mediator level for the CDE
                            c_cond = c(0,0,0),              # covariate level
                            ## Model types
                            mreg = "logistic",
                            yreg = "linear",
                            ## Additional specification
                            interaction = TRUE,
                            casecontrol = FALSE)
summary(regmedint_cont)
#### Mediation analysis
#              est         se          Z            p        lower       upper
# cde   -3.7152652 0.41600219  -8.930879 0.000000e+00 -4.5306145 -2.8999159
# pnde  -4.8450888 0.35052810 -13.822255 0.000000e+00 -5.5321113 -4.1580663
# tnie  -1.5011902 0.20821830  -7.209694 5.608847e-13 -1.9092905 -1.0930898
# tnde  -5.4367728 0.34049175 -15.967414 0.000000e+00 -6.1041244 -4.7694213
# pnie  -0.9095061 0.12266064  -7.414817 1.219025e-13 -1.1499166 -0.6690957
# te    -6.3462790 0.38788368 -16.361294 0.000000e+00 -7.1065170 -5.5860409
# pm     0.2365465 0.02947624   8.024991 1.110223e-15  0.1787742  0.2943189

# note: te = total effect = (pnde + tnie) = (tnde + pnie)
#       pm = proportion mediated = tnie / te
```

For binary outcomes:

```r
regmedint_bin <- regmedint(data = df1_int,
                           ## Variables
                           yvar = "Y_death",              # outcome variable
                           avar = "A0_ace",               # exposure
                           mvar = "M_smoking",            # mediator
                           cvar = c("L0_male",            # confounders
                                    "L0_parent_low_educ_lv",
                                    "L1"),
                           #eventvar = "event",      # only for survival outcome
                           ## Values at which effects are evaluated
                           a0 = 0,
                           a1 = 1,
                           m_cde = 0,                 # mediator level for the CDE
                           c_cond = c(0,0,0),             # covariate level
                           ## Model types
                           mreg = "logistic",
                           yreg = "logistic",
                           ## Additional specification
                           interaction = TRUE,
                           casecontrol = FALSE)
results.binary <- summary(regmedint_bin)

# taking the exponential of the estimations
exp(results.binary$summary_myreg[c("cde","pnde","tnie","tnde","pnie","te"),
                                 c("est","lower","upper")])
#### Mediation analysis
#            est     lower     upper
# cde   1.442942 1.191195 1.747893
# pnde  1.448035 1.245470 1.683545
# tnie  1.050029 1.013842 1.087509
# tnde  1.450344 1.257042 1.673371
# pnie  1.048358 1.029285 1.067783
# te    1.520479 1.316954 1.755457
```

5.3 Three-way decomposition

In order to carry-out a three-way decomposition with standard regressions, we will use the same models as for the two-way decomposition (equations (5.3), (5.5) and (5.4)).

(VanderWeele 2013) defines:

- the PNDE $= \mathbb{E}\left(Y_{A=1,M_{A=0}}\right) - \mathbb{E}(Y_{A=0,M_{A=0}})$,

- the PNIE $= \mathbb{E}\left(Y_{A=0,M_{A=1}}\right) - \mathbb{E}(Y_{A=0,M_{A=0}})$,
- and the mediated interactive effect MIE $= \mathbb{E}\left(\left[Y_{1,1} - Y_{1,0} - Y_{0,1} - Y_{0,0}\right] \times [M_1 - M_0]\right)$.

The sum of these 3 components is equal to the Average total effect (ATE).

With a continuous outcome and a binary mediator, the PNDE and PNIE can be estimated as for the two-way decomposition (section 5.2.2) using the linear regression of the outcome (equation (5.3)) and the logistic regression of the mediator (equation (5.4)).

The mediated interactive effect can be estimated using the same equations (5.3) and (5.4), by:

$$\text{MIE} \mid (L(0), L(1)) = \hat{\gamma}_{A*M} \left[\frac{\exp(\hat{\beta}_0 + \hat{\beta}_A \times 1 + \hat{\beta}'_{L(0)}L(0) + \hat{\beta}'_{L(1)}L(1))}{1 + \exp(\hat{\beta}_0 + \hat{\beta}_A \times 1 + \hat{\beta}'_{L(0)}L(0) + \hat{\beta}'_{L(1)}L(1))} - \frac{\exp(\hat{\beta}_0 + \hat{\beta}_A \times 0 + \hat{\beta}'_{L(0)}L}{1 + \exp(\hat{\beta}_0 + \hat{\beta}_A \times 0 + \hat{\beta}'_{L(0)}} \right.$$

Conditional on participants for which $L(0) = 0$ and $L(1) = 0$, the expression is simplified:

$$\text{MIE}\Big|(L(0) = 0, L(1) = 0) = \hat{\gamma}_{A*M} \left[\frac{\exp(\hat{\beta}_0 + \hat{\beta}_A)}{1 + \exp(\hat{\beta}_0 + \hat{\beta}_A)} - \frac{\exp(\hat{\beta}_0)}{1 + \exp(\hat{\beta}_0)} \right]$$

```
### For a continuous outcome, in the subgroup with L(0)=0 and L(1)=0
## The PNDE is:
trad_PNDE_qol <- gamma.A.q + gamma.AM.q * (exp(beta.0)) / (1 + exp(beta.0))
trad_PNDE_qol
# -4.845089

## The PNIE is:
trad_PNIE_qol <- gamma.M.q *
  (exp(beta.0 + beta.A) /
    (1 + exp(beta.0 + beta.A)) - exp(beta.0) / (1 + exp(beta.0)))
trad_PNIE_qol
# -0.9095061

## The MIE is:
trad_MIE_qol <- gamma.AM.q *
  (exp(beta.0 + beta.A) / (1 + exp(beta.0 + beta.A)) -
    exp(beta.0) / (1 + exp(beta.0)))
trad_MIE_qol
# -0.591684
```

With a binary outcome, the total effect, the direct and indirect effects can be expressed using risk ratios or odds ratios. In order to express the mediated interactive effect, (VanderWeele 2013) suggested decomposing the *excess relative*

risk of the total effect $(\mathrm{RR}^{\mathrm{TE}} - 1)$, which enables the expression of the mediated interactive effect on an additive scale.

On the difference scale, the total effect can be decomposed as the sum of the PNDE, the PNIE and the MIE:

$$
\begin{aligned}
\mathbb{E}(Y_1) - \mathbb{E}(Y_0) = \ & \left[\mathbb{E}\left(Y_{1M_0}\right) - \mathbb{E}\left(Y_{0M_0}\right) \right] + \left[\mathbb{E}\left(Y_{0M_1}\right) - \mathbb{E}\left(Y_{0M_0}\right) \right] \\
& + \left[\left[\mathbb{E}\left(Y_{1M_1}\right) - \mathbb{E}\left(Y_{1M_0}\right) \right] - \left[\mathbb{E}\left(Y_{0M_1}\right) - \mathbb{E}\left(Y_{0M_0}\right) \right] \right] \\
= \ & \mathrm{PNDE} + \mathrm{PNIE} \\
& + \mathrm{MIE}
\end{aligned}
$$

Dividing by $\mathbb{E}(Y_0) = \mathbb{E}\left(Y_{0M_0}\right)$, we obtain the excess relative risk of the total effect decomposition:

$$
\begin{aligned}
\frac{\mathbb{E}(Y_1)}{\mathbb{E}(Y_0)} - 1 = \ & \left[\frac{\mathbb{E}\left(Y_{1M_0}\right)}{\mathbb{E}\left(Y_{0M_0}\right)} - 1 \right] + \left[\frac{\mathbb{E}\left(Y_{0M_1}\right)}{\mathbb{E}\left(Y_{0M_0}\right)} - 1 \right] \\
& + \left[\frac{\mathbb{E}\left(Y_{1M_1}\right)}{\mathbb{E}\left(Y_{0M_0}\right)} - \frac{\mathbb{E}\left(Y_{1M_0}\right)}{\mathbb{E}\left(Y_{0M_0}\right)} - \frac{\mathbb{E}\left(Y_{0M_1}\right)}{\mathbb{E}\left(Y_{0M_0}\right)} + 1 \right]
\end{aligned}
$$

where the first component is the *excess relative risk due to the PNDE*, the second component is the *excess relative risk due to the PNIE* and the third component is the *mediated excess relative risk due to interaction*.

If the outcome is rare, relative risks are approximately equal to odds ratios, and the 3 components of the excess relative risk can be estimated using the logistic regression of the outcome (equation (5.5)) and the logistic regression of the mediator (equation (5.4)).

The component of the excess relative risk due to the PNDE is approximately equal to:

$$
\mathrm{RR}^{\mathrm{PNDE}} - 1 \approx \frac{\exp\left[\hat{\gamma}_A(1-0)\right]\left[1 + \exp\left(\hat{\beta}_0 + \hat{\beta}_A \times 0 + \hat{\beta}'_{L(0)}L(0) + \hat{\beta}'_{L(1)}L(1) + \hat{\gamma}_M + \hat{\gamma}_{A*M} \times 1\right)\right]}{\left[1 + \exp\left(\hat{\beta}_0 + \hat{\beta}_A \times 0 + \hat{\beta}'_{L(0)}L(0) + \hat{\beta}'_{L(1)}L(1) + \hat{\gamma}_M + \hat{\gamma}_{A*M} \times 0\right)\right]} - 1
$$

The component of the excess relative risk due to the PNIE is approximately equal to:

$$
\mathrm{RR}^{\mathrm{PNIE}} - 1 \approx \frac{\left[1 + \exp\left(\hat{\beta}_0 + \hat{\beta}_A \times 0 + \hat{\beta}'_{L(0)}L(0) + \hat{\beta}'_{L(1)}L(1)\right)\right]\left[1 + \exp\left(\hat{\beta}_0 + \hat{\beta}_A \times 1 + \hat{\beta}'_{L(0)}L(0) + \hat{\beta}'_{L(1)}L(1) + \hat{\gamma}_M + \right.\right.}{\left[1 + \exp\left(\hat{\beta}_0 + \hat{\beta}_A \times 1 + \hat{\beta}'_{L(0)}L(0) + \hat{\beta}'_{L(1)}L(1)\right)\right]\left[1 + \exp\left(\hat{\beta}_0 + \hat{\beta}_A \times 0 + \hat{\beta}'_{L(0)}L(0) + \hat{\beta}'_{L(1)}L(1) + \hat{\gamma}_M + \right.\right.}
$$

The component of the excess relative risk due to the mediated interactive effect is approximately equal to:

$$
\begin{aligned}
\mathrm{RERI}_{mediated} \approx \ & \frac{\exp[\hat{\gamma}_A(1-0)]\left[1 + \exp\left(\hat{\beta}_0 + \hat{\beta}_A \times 1 + \hat{\beta}'_{L(0)}L(0) + \hat{\beta}'_{L(1)}L(1) + \hat{\gamma}_M + \hat{\gamma}_{A*M} \times 1\right)\right]\left[1 + \exp\left(\hat{\beta}_0 + \hat{\beta}_A \times 0 + \hat{\beta}'_{L(0)}L(0) + \hat{\beta}'_{L(1)}L(1)\right)\right]}{\left[1 + \exp\left(\hat{\beta}_0 + \hat{\beta}_A \times 0 + \hat{\beta}'_{L(0)}L(0) + \hat{\beta}'_{L(1)}L(1) + \hat{\gamma}_M + \hat{\gamma}_{A*M} \times 0\right)\right]\left[1 + \exp\left(\hat{\beta}_0 + \hat{\beta}_A \times 1 + \hat{\beta}'_{L(0)}L(0) + \hat{\beta}'_{L(1)}L(1)\right)\right]} \\
& - \frac{\left[1 + \exp\left(\hat{\beta}_0 + \hat{\beta}_A \times 1 + \hat{\beta}'_{L(0)}L(0) + \hat{\beta}'_{L(1)}L(1) + \hat{\gamma}_M + \hat{\gamma}_{A*M} \times 0\right)\right]\left[1 + \exp\left(\hat{\beta}_0 + \hat{\beta}_A \times 0 + \hat{\beta}'_{L(0)}L(0) + \hat{\beta}'_{L(1)}L(1)\right)\right]}{\left[1 + \exp\left(\hat{\beta}_0 + \hat{\beta}_A \times 0 + \hat{\beta}'_{L(0)}L(0) + \hat{\beta}'_{L(1)}L(1) + \hat{\gamma}_M + \hat{\gamma}_{A*M} \times 0\right)\right]\left[1 + \exp\left(\hat{\beta}_0 + \hat{\beta}_A \times 1 + \hat{\beta}'_{L(0)}L(0) + \hat{\beta}'_{L(1)}L(1)\right)\right]} \\
& - \frac{\exp[\hat{\gamma}_A(1-0)]\left[1 + \exp\left(\hat{\beta}_0 + \hat{\beta}_A \times 0 + \hat{\beta}'_{L(0)}L(0) + \hat{\beta}'_{L(1)}L(1) + \hat{\gamma}_M + \hat{\gamma}_{A*M} \times 1\right)\right]}{\left[1 + \exp\left(\hat{\beta}_0 + \hat{\beta}_A \times 0 + \hat{\beta}'_{L(0)}L(0) + \hat{\beta}'_{L(1)}L(1) + \hat{\gamma}_M + \hat{\gamma}_{A*M} \times 0\right)\right]} + 1
\end{aligned}
$$

Conditional on participants for which $L(0) = 0$ and $L(1) = 0$, these expressions are simplified:

$$\mathrm{RR}^{\mathrm{PNDE}} - 1 \approx \frac{\exp\left(\hat{\gamma}_A\right)\left[1 + \exp\left(\hat{\beta}_0 + \hat{\gamma}_M + \hat{\gamma}_{A*M}\right)\right]}{\left[1 + \exp\left(\hat{\beta}_0 + \hat{\gamma}_M\right)\right]} - 1$$

$$\mathrm{RR}^{\mathrm{PNIE}} - 1 \approx \frac{\left[1 + \exp\left(\hat{\beta}_0\right)\right]\left[1 + \exp\left(\hat{\beta}_0 + \hat{\beta}_A + \hat{\gamma}_M\right)\right]}{\left[1 + \exp\left(\hat{\beta}_0 + \hat{\beta}_A\right)\right]\left[1 + \exp\left(\hat{\beta}_0 + \hat{\gamma}_M\right)\right]} - 1$$

and

$$\mathrm{RERI}_{\mathrm{mediated}} \approx \frac{\exp(\hat{\gamma}_A)\left[1+\exp\left(\hat{\beta}_0+\hat{\beta}_A+\hat{\gamma}_M+\hat{\gamma}_{A*M}\right)\right]\left[1+\exp\left(\hat{\beta}_0\right)\right]}{\left[1+\exp\left(\hat{\beta}_0+\hat{\gamma}_M\right)\right]\left[1+\exp\left(\hat{\beta}_0+\hat{\beta}_A\right)\right]} - \frac{\left[1+\exp\left(\hat{\beta}_0+\hat{\beta}_A+\hat{\gamma}_M\right)\right]\left[1+\exp\left(\hat{\beta}_0\right)\right]}{\left[1+\exp\left(\hat{\beta}_0+\hat{\gamma}_M\right)\right]\left[1+\exp\left(\hat{\beta}_0+\hat{\beta}_A\right)\right]}$$
$$- \frac{\exp(\hat{\gamma}_A)\left[1+\exp\left(\hat{\beta}_0+\hat{\gamma}_M+\hat{\gamma}_{A*M}\right)\right]}{\left[1+\exp\left(\hat{\beta}_0+\hat{\gamma}_M\right)\right]} + 1$$

```
### For a binary outcome, in the subgroup with L(0)=0 and L(1)=0
## The excess relative risk is 52.3% (calculated from the OR of the total effect)
1.523254 - 1
# 0.523254


## The excess relative risk is decomposed into 3 components:
## The component of the excess relative risk due to PNDE is:
comp_PNDE_death <- exp(gamma.A.d) * (1 + exp(beta.0 + gamma.M.d + gamma.AM.d)) /
  (1 + exp(beta.0 + gamma.M.d)) - 1
comp_PNDE_death
# 0.4480347


## The component of the excess relative risk due to PNIE is:
comp_PNIE_death <- (1 + exp(beta.0)) * (1 + exp(beta.0 + beta.A + gamma.M.d)) /
  ((1 + exp(beta.0 + beta.A)) * (1 + exp(beta.0 + gamma.M.d))) - 1
comp_PNIE_death
# 0.04835753


## The component of the excess relative risk due to the mediated interactive
## effect is:
comp_MIE_qol <- exp(gamma.A.d) *
  (1 + exp(beta.0 + beta.A + gamma.M.d + gamma.AM.d)) * (1 + exp(beta.0)) /
  ((1 + exp(beta.0 + gamma.M.d)) * (1 + exp(beta.0 + beta.A))) -
  (1 + exp(beta.0 + beta.A + gamma.M.d)) * (1 + exp(beta.0)) /
  ((1 + exp(beta.0 + gamma.M.d)) * (1 + exp(beta.0 + beta.A))) -
  exp(gamma.A.d) * (1 + exp(beta.0 + gamma.M.d + gamma.AM.d)) /
  (1 + exp(beta.0 + gamma.M.d)) + 1
comp_MIE_qol
# 0.02408674
```

In this example, the excess relative risk of the exposure to adverse childhood (ACE) exposure is $\approx 52.3\%$, and of this excess relative risk...:

- $\approx 44.8\%$ is attributable to the PNDE of ACE,
- $\approx 4.8\%$ is attributable to the PNIE of ACE through smoking,
- $\approx 2.4\%$ is attributable to the mediated interactive effect between ACE and smoking.

Note: in this simulated data, the probability of death is around 20%, so that the requirement of a rare outcome is not really fulfilled (usually, we would consider $< 10\%$ to be acceptable).

5.4 Four-way decomposition

The same models as for the two-way and three-way decomposition (equations (5.3), (5.5) and (5.4)) will be used in order to apply the four-way decomposition.

(VanderWeele 2014) defines:

- the $\text{CDE}_{M=0} = \mathbb{E}\left(Y_{1,0}\right) - \mathbb{E}\left(Y_{0,0}\right)$,
- the mediated interaction effect $\text{MIE} = \mathbb{E}\left(\left[Y_{1,1} - Y_{1,0} - Y_{0,1} - Y_{0,0}\right] \times [M_1 - M_0]\right)$,
- the reference interaction effect $\text{RIE} = \mathbb{E}\left(\left[Y_{1,1} - Y_{1,0} - Y_{0,1} - Y_{0,0}\right] \times M_0\right)$,
- and the $\text{PNIE} = \mathbb{E}\left(Y_{A=0,M_{A=1}}\right) - \mathbb{E}(Y_{A=0,M_{A=0}})$.

The sum of these 4 components is equal to the Average total effect (ATE), and if the exposure affects the outcome, then at least one of these 4 components should be non-null.

With a continuous outcome and a binary mediator, the $\text{CDE}_{M=0}$ and PNIE can be estimated as for the two-way decomposition (sections 5.2.1 and 5.2.2), and the MIE can be estimated as for the three-way decomposition (section 5.3), using the linear regression of the outcome (equation (5.3)) and the logistic regression of the mediator (equation (5.4)).

$$\text{CDE}_{M=0} \mid (L(0), L(1)) = \hat{\gamma}_A + \hat{\gamma}_{A*M} \times 0$$

$$\text{PNIE} \mid L(0), L(1) = (\hat{\gamma}_M + \hat{\gamma}_{A*M} \times 0) \left[\frac{exp(\hat{\beta}_0 + \hat{\beta}_A \times 1 + \hat{\beta}'_{L(0)}L(0) + \hat{\beta}'_{L(1)}L(1))}{1 + exp(\hat{\beta}_0 + \hat{\beta}_A \times 1 + \hat{\beta}'_{L(0)}L(0) + \hat{\beta}'_{L(1)}L(1))} - \frac{exp(\hat{\beta}_0 + \hat{\beta}_A \times 0 + \hat{\beta}'_{L(0)}L(}{1 + exp(\hat{\beta}_0 + \hat{\beta}_A \times 0 + \hat{\beta}'_{L(0)}L}\right.$$

and

$$\text{MIE} \mid (L(0), L(1)) = \hat{\gamma}_{A*M} \left[\frac{\exp(\hat{\beta}_0 + \hat{\beta}_A \times 1 + \hat{\beta}'_{L(0)} L(0) + \hat{\beta}'_{L(1)} L(1))}{1 + \exp(\hat{\beta}_0 + \hat{\beta}_A \times 1 + \hat{\beta}'_{L(0)} L(0) + \hat{\beta}'_{L(1)} L(1))} - \frac{\exp(\hat{\beta}_0 + \hat{\beta}_A \times 0 + \hat{\beta}'_{L(0)} L}{1 + \exp(\hat{\beta}_0 + \hat{\beta}_A \times 0 + \hat{\beta}'_{L(0)}} \right.$$

The RIE can be estimated by:

$$\text{RIE} \mid (L(0), L(1)) = \hat{\gamma}_{A*M} \left[\frac{\exp(\hat{\beta}_0 + \hat{\beta}_A \times 0 + \hat{\beta}'_{L(0)} L(0) + \hat{\beta}'_{L(1)} L(1))}{1 + \exp(\hat{\beta}_0 + \hat{\beta}_A \times 0 + \hat{\beta}'_{L(0)} L(0) + \hat{\beta}'_{L(1)} L(1))} - 0 \right]$$

```r
### For a continuous outcome, in the subgroup with L(0)=0 and L(1)=0
## The CDE_(M=0) is:
trad_CDE_qol_m0 <- gamma.A.q + gamma.AM.q * 0
trad_CDE_qol_m0
# -3.715265

## The PNIE is:
trad_PNIE_qol <- gamma.M.q *
  (exp(beta.0 + beta.A) /
     (1 + exp(beta.0 + beta.A)) - exp(beta.0) / (1 + exp(beta.0)))
trad_PNIE_qol
# -0.9095061

## The MIE is:
trad_MIE_qol <- gamma.AM.q *
  (exp(beta.0 + beta.A) / (1 + exp(beta.0 + beta.A)) -
     exp(beta.0) / (1 + exp(beta.0)))
trad_MIE_qol
# -0.591684

## The RIE is:
trad_RIE_qol <- gamma.AM.q * (exp(beta.0)) / (1 + exp(beta.0))
trad_RIE_qol
# -1.129824
```

With a binary outcome and a binary mediator, (VanderWeele 2014) suggested decomposing the *excess relative risk of the total effect* $(\text{RR}^{\text{TE}} - 1)$ (as for the 3-way decomposition), which enables the expression of the MIE and the RIE on an additive scale.

If the outcome is rare,

The component of the excess relative risk due to the CDE is approximately equal to:

$$\frac{\mathbb{E}(Y_{0,0}|L(0),L(1))}{\mathbb{E}(Y_0|L(0),L(1))}\left(\frac{\mathbb{E}(Y_{1,0}|L(0),L(1))}{\mathbb{E}(Y_{0,0}|L(0),L(1))}-1\right)\approx \frac{\exp(\hat\gamma_A(1-0)+\hat\gamma_M\times0+\hat\gamma_{A*M}\times1\times0)\left[1+\exp\left(\hat\beta_0+\hat\beta_A\times0+\hat\beta'_{L(0)}L(0)+\hat\beta'_{L(0)}L(1)\right)\right]}{1+\exp\left(\hat\beta_0+\hat\beta_A\times0+\hat\beta'_{L(0)}L(0)+\hat\beta'_{L(0)}L(1)+\hat\gamma_M+\hat\gamma_{A*M}\times0\right)}$$

$$-\frac{\exp(\hat\gamma_M\times0+\hat\gamma_{A*M}\times0\times0)\left[1+\exp\left(\hat\beta_0+\hat\beta_A\times0+\hat\beta'_{L(0)}L(0)+\hat\beta'_{L(0)}L(1)\right)\right]}{1+\exp\left(\hat\beta_0+\hat\beta_A\times0+\hat\beta'_{L(0)}L(0)+\hat\beta'_{L(0)}L(1)+\hat\gamma_M+\hat\gamma_{A*M}\times0\right)}$$

The component of the excess relative risk due to the PNIE is approximately equal to:

$$\mathrm{RR}^{\mathrm{PNIE}}-1\approx\frac{\left[1+\exp\left(\hat\beta_0+\hat\beta_A\times0+\hat\beta'_{L(0)}L(0)+\hat\beta'_{L(1)}L(1)\right)\right]\left[1+\exp\left(\hat\beta_0+\hat\beta_A\times1+\hat\beta'_{L(0)}L(0)+\hat\beta'_{L(1)}L(1)+\hat\gamma_M+\right.\right.}{\left[1+\exp\left(\hat\beta_0+\hat\beta_A\times1+\hat\beta'_{L(0)}L(0)+\hat\beta'_{L(1)}L(1)\right)\right]\left[1+\exp\left(\hat\beta_0+\hat\beta_A\times0+\hat\beta'_{L(0)}L(0)+\hat\beta'_{L(1)}L(1)+\hat\gamma_M+\right.\right.}$$

The component of the excess relative risk due to the mediated interactive effect is approximately equal to:

$$\mathrm{RERI}_{mediated}\approx\frac{\exp[\hat\gamma_A(1-0)]\left[1+\exp\left(\hat\beta_0+\hat\beta_A\times1+\hat\beta'_{L(0)}L(0)+\hat\beta'_{L(1)}L(1)+\hat\gamma_M+\hat\gamma_{A*M}\times1\right)\right]\left[1+\exp\left(\hat\beta_0+\hat\beta_A\times0+\hat\beta'_{L(0)}L(0)+\hat\beta'_{L(1)}L(1)\right)\right]}{\left[1+\exp\left(\hat\beta_0+\hat\beta_A\times0+\hat\beta'_{L(0)}L(0)+\hat\beta'_{L(1)}L(1)+\hat\gamma_M+\hat\gamma_{A*M}\times0\right)\right]\left[1+\exp\left(\hat\beta_0+\hat\beta_A\times1+\hat\beta'_{L(0)}L(0)+\hat\beta'_{L(1)}L(1)\right)\right]}$$

$$-\frac{\left[1+\exp\left(\hat\beta_0+\hat\beta_A\times1+\hat\beta'_{L(0)}L(0)+\hat\beta'_{L(1)}L(1)+\hat\gamma_M+\hat\gamma_{A*M}\times0\right)\right]\left[1+\exp\left(\hat\beta_0+\hat\beta_A\times0+\hat\beta'_{L(0)}L(0)+\hat\beta'_{L(1)}L(1)\right)\right]}{\left[1+\exp\left(\hat\beta_0+\hat\beta_A\times0+\hat\beta'_{L(0)}L(0)+\hat\beta'_{L(1)}L(1)+\hat\gamma_M+\hat\gamma_{A*M}\times0\right)\right]\left[1+\exp\left(\hat\beta_0+\hat\beta_A\times1+\hat\beta'_{L(0)}L(0)+\hat\beta'_{L(1)}L(1)\right)\right]}$$

$$-\frac{\exp[\hat\gamma_A(1-0)]\left[1+\exp\left(\hat\beta_0+\hat\beta_A\times0+\hat\beta'_{L(0)}L(0)+\hat\beta'_{L(1)}L(1)+\hat\gamma_M+\hat\gamma_{A*M}\times1\right)\right]}{\left[1+\exp\left(\hat\beta_0+\hat\beta_A\times0+\hat\beta'_{L(0)}L(0)+\hat\beta'_{L(1)}L(1)+\hat\gamma_M+\hat\gamma_{A*M}\times0\right)\right]}+1$$

and the component of the excess relative risk due to the reference interaction effect is approximately equal to:

$$\mathrm{RERI}_{\mathrm{ref}}\approx\frac{\exp[\hat\gamma_A(1-0)]\left[1+\exp\left(\hat\beta_0+\hat\beta_A\times0+\hat\beta'_{L(0)}L(0)+\hat\beta'_{L(1)}L(1)+\hat\gamma_M+\hat\gamma_{A*M}\times1\right)\right]}{\left[1+\exp\left(\hat\beta_0+\hat\beta_A\times0+\hat\beta'_{L(0)}L(0)+\hat\beta'_{L(1)}L(1)+\hat\gamma_M+\hat\gamma_{A*M}\times0\right)\right]}-1$$

$$-\frac{\exp[\hat\gamma_A(1-0)+\hat\gamma_M\times0+\hat\gamma_{A*M}1\times0]\left[1+\exp\left(\hat\beta_0+\hat\beta_A\times0+\hat\beta'_{L(0)}L(0)+\hat\beta'_{L(1)}L(1)\right)\right]}{1+\exp\left(\hat\beta_0+\hat\beta_A\times0+\hat\beta'_{L(0)}L(0)+\hat\beta'_{L(1)}L(1)+\hat\gamma_M+\hat\gamma_{A*M}\times0\right)}$$

$$+\frac{\exp(\hat\gamma_M\times0+\hat\gamma_{A*M}\times0\times0)\left[1+\exp\left(\hat\beta_0+\hat\beta_A\times0+\hat\beta'_{L(0)}L(0)+\hat\beta'_{L(1)}L(1)\right)\right]}{1+\exp\left(\hat\beta_0+\hat\beta_A\times0+\hat\beta'_{L(0)}L(0)+\hat\beta'_{L(1)}L(1)+\hat\gamma_M+\hat\gamma_{A*M}\times0\right)}$$

Conditional on participants for which $L(0)=0$ and $L(1)=0$, these expressions are simplified:

The component of the excess relative risk due to the CDE is approximately equal to:

$$\approx\frac{\exp(\hat\gamma_A)\left[1+\exp\left(\hat\beta_0\right)\right]}{1+\exp\left(\hat\beta_0+\hat\gamma_M\right)}-\frac{\left[1+\exp\left(\hat\beta_0\right)\right]}{1+\exp\left(\hat\beta_0+\hat\gamma_M\right)}$$

The component of the excess relative risk due to the PNIE is approximately equal to:

$$\mathrm{RR}^{\mathrm{PNIE}}-1\approx\frac{\left[1+\exp\left(\hat\beta_0\right)\right]\left[1+\exp\left(\hat\beta_0+\hat\beta_A+\hat\gamma_M\right)\right]}{\left[1+\exp\left(\hat\beta_0+\hat\beta_A\right)\right]\left[1+\exp\left(\hat\beta_0+\hat\gamma_M\right)\right]}-1$$

The component of the excess relative risk due to the mediated interactive effect is approximately equal to:

$$\text{RERI}_{mediated} \approx \frac{\exp(\hat{\gamma}_A)\left[1+\exp\left(\hat{\beta}_0+\hat{\beta}_A+\hat{\gamma}_M+\hat{\gamma}_{A*M}\right)\right]\left[1+\exp(\hat{\beta}_0)\right]}{\left[1+\exp\left(\hat{\beta}_0+\hat{\gamma}_M\right)\right]\left[1+\exp\left(\hat{\beta}_0+\hat{\beta}_A\right)\right]} - \frac{\left[1+\exp\left(\hat{\beta}_0+\hat{\beta}_A+\hat{\gamma}_M\right)\right]\left[1+\exp\left(\hat{\beta}_0\right)\right]}{\left[1+\exp\left(\hat{\beta}_0+\hat{\gamma}_M\right)\right]\left[1+\exp\left(\hat{\beta}_0+\hat{\beta}_A\right)\right]}$$
$$-\frac{\exp(\hat{\gamma}_A)\left[1+\exp\left(\hat{\beta}_0+\hat{\gamma}_M+\hat{\gamma}_{A*M}\right)\right]}{\left[1+\exp\left(\hat{\beta}_0+\hat{\gamma}_M\right)\right]} + 1$$

and the component of the excess relative risk due to the reference interaction effect is approximately equal to:

$$\text{RERI}_{\text{ref}} \approx \frac{\exp\left(\hat{\gamma}_A\right)\left[1+\exp\left(\hat{\beta}_0+\hat{\gamma}_M+\hat{\gamma}_{A*M}\right)\right]}{\left[1+\exp\left(\hat{\beta}_0+\hat{\gamma}_M\right)\right]} - 1 - \frac{\exp\left(\hat{\gamma}_A\right)\left[1+\exp\left(\hat{\beta}_0\right)\right]}{1+\exp\left(\hat{\beta}_0+\hat{\gamma}_M\right)} + \frac{\left[1+\exp\left(\hat{\beta}_0\right)\right]}{1+\exp\left(\hat{\beta}_0+\hat{\gamma}_M\right)}$$

```
### For a binary outcome, in the subgroup with L(0)=0 and L(1)=0
## The excess relative risk is 52.3% (calculated from the OR of the total effect)
1.523254 - 1
# 0.523254

## The excess relative risk is decomposed into 4 components:
## The component of the excess relative risk due to the CDE(M=0) is:
comp_CDE_death_m0 <- exp(gamma.A.d) * (1 + exp(beta.0)) /
  (1 + exp(beta.0 + gamma.M.d)) -
  (1 + exp(beta.0)) / (1 + exp(beta.0 + gamma.M.d))
comp_CDE_death_m0
# 0.402041

## The component of the excess relative risk due to the PNIE is:
comp_PNIE_death <- (1 + exp(beta.0)) * (1 + exp(beta.0 + beta.A + gamma.M.d)) /
  ((1 + exp(beta.0 + beta.A)) * (1 + exp(beta.0 + gamma.M.d))) - 1
comp_PNIE_death
# 0.04835753

## The component of the excess relative risk due to the MIE is:
comp_MIE_death <- exp(gamma.A.d) * (1 + exp(beta.0 + beta.A + gamma.M.d +
                                       gamma.AM.d)) * (1 + exp(beta.0)) /
  ((1 + exp(beta.0 + gamma.M.d)) * (1 + exp(beta.0 + beta.A))) -
  (1 + exp(beta.0 + beta.A + gamma.M.d)) * (1 + exp(beta.0)) /
  ((1 + exp(beta.0 + gamma.M.d)) * (1 + exp(beta.0 + beta.A))) -
  exp(gamma.A.d) * (1 + exp(beta.0 + gamma.M.d + gamma.AM.d)) /
  (1 + exp(beta.0 + gamma.M.d)) + 1
comp_MIE_death
# 0.02408674

## The component of the excess relative risk due to the RIE is:
comp_RIE_death <- exp(gamma.A.d) * (1 + exp(beta.0 + gamma.M.d + gamma.AM.d)) /
```

```
  (1 + exp(beta.0 + gamma.M.d)) - 1 -
  exp(gamma.A.d) * (1 + exp(beta.0)) / (1 + exp(beta.0 + gamma.M.d)) +
  (1 + exp(beta.0)) / (1 + exp(beta.0 + gamma.M.d))
comp_RIE_death
# 0.04599376
```

In this example, the excess relative risk of the exposure to adverse childhood (ACE) exposure is $\approx 52.3\%$, and of this excess relative risk...:

- $\approx 40.2\%$ is attributable to the CDE of ACE,
- $\approx 4.8\%$ is attributable to the PNIE of ACE through smoking,
- $\approx 2.4\%$ is attributable to the mediated interactive effect between ACE and smoking.
- and $\approx 4.6\%$ is attributable to the (ACE $*$ smoking) reference interactive effect.

Note: in this simulated data, the probability of death is around 20%, so that the requirement of a rare outcome is not really fulfilled (usually, we would consider $< 10\%$ to be acceptable).

R package for 3-way and 4-way decomposition

The `CMAverse` R package (a suite of functions for causal mediation analysis) can be used for 3-way and 4-way decomposition. Estimations of the CDE(M=0), PNIE, MIE and INTref presented above can be obtained as we show in the following example.

For continuous outcomes:

```
library(CMAverse)

### For the continuous outcome
## Closed-form parameter function estimation and delta method inferece
res_rb_param_delta <- cmest(data = df1_int,
                            model = "rb", # for "regression based" (rb) approach
                            outcome = "Y_qol",        # outcome variable
                            exposure = "A0_ace",      # exposure variable
                            mediator = "M_smoking",   # mediator
                            basec = c("L0_male",      # confounders
                                      "L0_parent_low_educ_lv",
                                      "L1"),
                            EMint = TRUE, # exposures*mediator interaction
                            mreg = list("logistic"), # model of the mediator
                            yreg = "linear",        # model of the outcome
                            astar = 0,
                            a = 1,
```

```
                        mval = list(0),
                        basecval = list(0,0,0),      # covariate level
                        estimation = "paramfunc", #  closed-form parameter
                                                  # function estimation
                        inference = "delta") # IC95% : delta method
summary(res_rb_param_delta)
# Closed-form parameter function estimation with
# delta method standard errors, confidence intervals and p-values
#
#             Estimate Std.error  95% CIL 95% CIU    P.val
#   cde        -3.71527   0.41600 -4.53061  -2.900  < 2e-16 ***    CDE(M=0)
#   pnde       -4.84509   0.35053 -5.53211  -4.158  < 2e-16 ***
#   tnde       -5.43677   0.34049 -6.10412  -4.769  < 2e-16 ***
#   pnie       -0.90951   0.12266 -1.14992  -0.669 1.22e-13 ***    PNIE
#   tnie       -1.50119   0.20822 -1.90929  -1.093 5.61e-13 ***
#   te         -6.34628   0.38788 -7.10652  -5.586  < 2e-16 ***
#   intref     -1.12982   0.13824 -1.40076  -0.859 2.22e-16 ***    INTref
#   intmed     -0.59168   0.10398 -0.79547  -0.388 1.27e-08 ***    MIE
#   cde(prop)   0.58542   0.04505  0.49712   0.674  < 2e-16 ***
#   intref(prop) 0.17803  0.02560  0.12786   0.228 3.51e-12 ***
#   intmed(prop) 0.09323  0.01586  0.06216   0.124 4.11e-09 ***
#   pnie(prop)  0.14331   0.01655  0.11087   0.176  < 2e-16 ***
#   pm          0.23655   0.02948  0.17877   0.294 1.11e-15 ***
#   int         0.27126   0.03780  0.19717   0.345 7.20e-13 ***
#   pe          0.41458   0.04505  0.32627   0.503  < 2e-16 ***
#   ---
#   Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

# for the 3-way decomposition, the PNDE, PNIE, and MIE are given by:
data.frame("Estimate" = res_rb_param_delta$effect.pe,
           "lower95CI" = res_rb_param_delta$effect.ci.low,
           "upper95CI" = res_rb_param_delta$effect.ci.high,
           "P.value" = res_rb_param_delta$effect.pval)[c("pnde","pnie", "intmed"),]
#          Estimate   lower95CI   upper95CI       P.value
# pnde   -4.8450888 -5.5321113 -4.1580663 0.000000e+00
# pnie   -0.9095061 -1.1499166 -0.6690957 1.219025e-13
# intmed -0.5916840 -0.7954739 -0.3878941 1.266206e-08

# for the 4-way decomposition, the CDE(M=0), Intref, MIE and PNIE are given by:
data.frame("Estimate" = res_rb_param_delta$effect.pe,
           "lower95CI" = res_rb_param_delta$effect.ci.low,
           "upper95CI" = res_rb_param_delta$effect.ci.high,
           "P.value" = res_rb_param_delta$effect.pval)[c("cde","intref","intmed","pnie"
#          Estimate   lower95CI   upper95CI       P.value
# cde    -3.7152652 -4.5306145 -2.8999159 0.000000e+00
```

```
# intref -1.1298236 -1.4007600 -0.8588871 2.220446e-16
# intmed -0.5916840 -0.7954739 -0.3878941 1.266206e-08
# pnie   -0.9095061 -1.1499166 -0.6690957 1.219025e-13
```

For binary outcomes:

```
### For the binary outcome
## Closed-form parameter function estimation and delta method inferece
res_rb_param_delta <- cmest(data = df1_int,
                        model = "rb", # for "regression based" (rb) approach
                        outcome = "Y_death",        # outcome variable
                        exposure = "A0_ace",      # exposure variable
                        mediator = "M_smoking",   # mediator
                        basec = c("L0_male",      # confounders
                                  "L0_parent_low_educ_lv",
                                  "L1"),
                        EMint = TRUE, # exposures*mediator interaction
                        mreg = list("logistic"), # model of the mediator
                        yreg = "logistic",      # model of the outcome
                        astar = 0,
                        a = 1,
                        mval = list(0),
                        basecval = list(0,0,0),     # covariate level
                        estimation = "paramfunc", #  closed-form parameter
                        # function estimation
                        inference = "delta") # IC95% : delta method
summary(res_rb_param_delta)
# Closed-form parameter function estimation with
# delta method standard errors, confidence intervals and p-values
#
# Estimate Std.error  95% CIL 95% CIU     P.val
#   Rcde             1.44294   0.14115  1.19120   1.748 0.000178 ***
#   Rpnde            1.44803   0.11133  1.24547   1.684 1.47e-06 ***
#   Rtnde            1.45034   0.10585  1.25704   1.673 3.50e-07 ***
#   Rpnie            1.04836   0.00982  1.02929   1.068 4.62e-07 ***
#   Rtnie            1.05003   0.01879  1.01384   1.088 0.006368 **
#   Rte              1.52048   0.11148  1.31695   1.755 1.10e-08 ***
#   ERcde            0.40204   0.12699  0.15315   0.651 0.001546 **   CDE(M=0)
#   ERintref         0.04599   0.04821 -0.04850   0.140 0.340093      INTref
#   ERintmed         0.02409   0.02543 -0.02576   0.074 0.343607      MIE
#   ERpnie           0.04836   0.00982  0.02911   0.068 8.47e-07 *** PNIE
#   ERcde(prop)      0.77244   0.14282  0.49252   1.052 6.36e-08 ***
#   ERintref(prop)   0.08837   0.09311 -0.09413   0.271 0.342600
#   ERintmed(prop)   0.04628   0.04901 -0.04978   0.142 0.345058
#   ERpnie(prop)     0.09291   0.02602  0.04192   0.144 0.000356 ***
```

```
#   pm              0.13919   0.05498  0.03142   0.247 0.011359 *
#   int             0.13465   0.14186 -0.14340   0.413 0.342562
#   pe              0.22756   0.14282 -0.05237   0.507 0.111092
#   ---
#   Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1


# for the 3-way decomposition, the excess relative risk due to
# the PNDE, PNIE and MIE are given by:
data.frame("Estimate" = res_rb_param_delta$effect.pe - 1,
           "lower95CI" = res_rb_param_delta$effect.ci.low - 1,
           "upper95CI" = res_rb_param_delta$effect.ci.high - 1)[c("Rpnde"),]
#        Estimate lower95CI upper95CI
# Rpnde 0.4480347 0.2454699 0.6835449
# and
data.frame("Estimate" = res_rb_param_delta$effect.pe,
           "lower95CI" = res_rb_param_delta$effect.ci.low,
           "upper95CI" = res_rb_param_delta$effect.ci.high,
           "P.value" = res_rb_param_delta$effect.pval)[c("ERpnie", "ERintmed"),]
#           Estimate    lower95CI  upper95CI      P.value
# ERpnie   0.04835753  0.02910970 0.06760535 8.473169e-07
# ERintmed 0.02408674 -0.02576124 0.07393472 3.436070e-01




# for the 4-way decomposition, the CDE(M=0), the excess relative risk due to
# CDE(M=0), Intref, MIE and PNIE are given by:
data.frame("Estimate" = res_rb_param_delta$effect.pe,
           "lower95CI" = res_rb_param_delta$effect.ci.low,
           "upper95CI" = res_rb_param_delta$effect.ci.high,
           "P.value" = res_rb_param_delta$effect.pval)[c("ERcde","ERintref","ERintmed"
#           Estimate    lower95CI  upper95CI      P.value
# ERcde    0.40204098  0.15315072 0.65093124 1.545523e-03
# ERintref 0.04599376 -0.04850084 0.14048835 3.400930e-01
# ERintmed 0.02408674 -0.02576124 0.07393472 3.436070e-01
# ERpnie   0.04835753  0.02910970 0.06760535 8.473169e-07
```

Chapter 6

G-computation

If we make the assumption that the intermediate confounder $L(1)$ of the $M-Y$ relationship is affected by the exposure A (Causal model 2), it is necessary to use other methods than traditional regressions models. To illustrate g-computation estimators, we will use the `df2_int.csv` data set, which was generated from a system corresponding to this assumption. Moreover, we will assume that their is an $A \star M$ interaction effect on the outcome.

G-computation can be used for the estimation of the total effect and two-way decomposition (CDE, marginal and conditional randomized direct and indirect effects).

6.1 Estimation of the Average Total Effect (ATE)

The following steps describe the implementation of the g-computation estimator of the average total effect $ATE = \mathbb{E}(Y_{A=1}) - \mathbb{E}(Y_{A=0})$:

1. Fit a logistic or a linear regression to estimate $\overline{Q} = \mathbb{E}(Y \mid A, L(0))$

2. Use this estimate to predict an outcome for each subject $\hat{\overline{Q}}(A = 0)_i$ and $\hat{\overline{Q}}(A = 1)_i$, by evaluating the regression fit $\overline{Q}$ at $A = 0$ and $A = 1$ respectively

3. Plug the predicted outcomes in the g-formula and use the sample mean to estimate Ψ_{ATE}

$$\hat{\Psi}_{\text{gcomp}}^{\text{ATE}} = \frac{1}{n} \sum_{i=1}^{n} \left[\hat{\overline{Q}}(A = 1)_i - \hat{\overline{Q}}(A = 0)_i \right] \tag{6.1}$$

For continuous outcomes, $\overline{Q}(A = a)$ functions can be estimated using linear regressions. For binary outcomes, they can be estimated using logistic regressions.

```r
# 1. Estimate Qbar
Q.tot.death <- glm(Y_death ~ A0_ace + L0_male + L0_parent_low_educ_lv,
                   family = "binomial", data = df2_int)
Q.tot.qol <- glm(Y_qol ~ A0_ace + L0_male + L0_parent_low_educ_lv,
                 family = "gaussian", data = df2_int)

# 2. Predict an outcome for each subject, setting A=0 and A=1
# prepare data sets used to predict the outcome under the counterfactual
# scenarios setting A=0 and A=1
data.A1 <- data.A0 <- df2_int
data.A1$A0_ace <- 1
data.A0$A0_ace <- 0

# predict values
Y1.death.pred <- predict(Q.tot.death, newdata = data.A1, type = "response")
Y0.death.pred <- predict(Q.tot.death, newdata = data.A0, type = "response")

Y1.qol.pred <- predict(Q.tot.qol, newdata = data.A1, type = "response")
Y0.qol.pred <- predict(Q.tot.qol, newdata = data.A0, type = "response")

# 3. Plug the predicted outcome in the gformula and use the sample mean
#    to estimate the ATE
ATE.death.gcomp <- mean(Y1.death.pred - Y0.death.pred)
ATE.death.gcomp
# [1] 0.08270821

ATE.qol.gcomp <- mean(Y1.qol.pred - Y0.qol.pred)
ATE.qol.gcomp
# [1] -8.360691
```

A 95% confidence interval can be estimated applying a bootstrap procedure. An example is given in the following code.

```r
set.seed(1234)
B <- 2000
bootstrap.estimates <- data.frame(matrix(NA, nrow = B, ncol = 2))
colnames(bootstrap.estimates) <- c("boot.death.est", "boot.qol.est")
for (b in 1:B){
  # sample the indices 1 to n with replacement
  bootIndices <- sample(1:nrow(df2_int), replace=T)
  bootData <- df2_int[bootIndices,]
```

```
  if ( round(b/100, 0) == b/100 ) print(paste0("bootstrap number ",b))

  Q.tot.death <- glm(Y_death ~ A0_ace + L0_male + L0_parent_low_educ_lv,
                     family = "binomial", data = bootData)
  Q.tot.qol <- glm(Y_qol ~ A0_ace + L0_male + L0_parent_low_educ_lv,
                   family = "gaussian", data = bootData)

  boot.A.1 <- boot.A.0 <- bootData
  boot.A.1$A0_ace <- 1
  boot.A.0$A0_ace <- 0

  Y1.death.boot <- predict(Q.tot.death, newdata = boot.A.1, type = "response")
  Y0.death.boot <- predict(Q.tot.death, newdata = boot.A.0, type = "response")

  Y1.qol.boot <- predict(Q.tot.qol, newdata = boot.A.1, type = "response")
  Y0.qol.boot <- predict(Q.tot.qol, newdata = boot.A.0, type = "response")

  bootstrap.estimates[b,"boot.death.est"] <- mean(Y1.death.boot - Y0.death.boot)
  bootstrap.estimates[b,"boot.qol.est"] <- mean(Y1.qol.boot - Y0.qol.boot)
}

IC95.ATE.death <- c(ATE.death.gcomp -
                        qnorm(0.975)*sd(bootstrap.estimates[,"boot.death.est"]),
                    ATE.death.gcomp +
                        qnorm(0.975)*sd(bootstrap.estimates[,"boot.death.est"]) )
IC95.ATE.death
# [1] 0.05571017 0.10970624

IC95.ATE.qol <- c(ATE.qol.gcomp -
                      qnorm(0.975)*sd(bootstrap.estimates[,"boot.qol.est"]),
                  ATE.qol.gcomp +
                      qnorm(0.975)*sd(bootstrap.estimates[,"boot.qol.est"]) )
IC95.ATE.qol
# [1] -9.156051 -7.565331
```

6.2 Estimation of Controlled Direct Effects (CDE)

The controlled direct effect $\Psi^{\mathrm{CDE}_m} = \mathbb{E}(Y_{A=1,M=m}) - \mathbb{E}(Y_{A=0,M=m})$ is the difference between the mean outcome had the whole population been exposed to ACE (setting $A = 1$), compared to the mean outcome had the whole population been unexposed (setting $A = 0$), while keeping the mediator equal to a constant

given value $(M = m)$ in both scenarios.

The g-formula for a CDE ($\mathbb{E}(Y_{A=a', M=m})$) is more complex than for the average total effect, and the simple substitution approach described previously is less convenient to apply:

$\mathbb{E}(Y_{A=a', M=m}) = \sum_{l(0), l(1)} [\mathbb{E}(Y \mid m, l(1), a', l(0)) \times P(L(1) = l(1)|a', l(0))] \times P(L(0) = l(0))$

In our simple example with a binary exposure A, a binary mediator M and a binary intermediate confounder $L(1)$, it is still possible to apply the substitution approach (corresponding to a non-parametric g-computation estimation) by estimating the following components of the g-formula:

- $\overline{Q}_Y(A, L(1), M) = \mathbb{E}(Y \mid L(0), A, L(1), M)$,
- and $\overline{Q}_{L(1)}(A) = P(L(1) = 1) \mid A, l(0))$

We can then generate predicted outcomes from these 3 models for each subject in the data set, and obtain a *non-parametric maximum likelihood estimator (NPMLE)* of the CDE using the empirical mean:

$$\Psi_{\text{NPMLE}}^{\text{CDE}m} = \frac{1}{n} \sum \quad \left[\hat{\overline{Q}}_Y(A=1, L(1)=1, M=m) \times \hat{\overline{Q}}_{L(1)}(A=1) + \hat{\overline{Q}}_Y(A=1, L(1)=0, M=m) \times (1 - \hat{\overline{Q}}_{L(1)}(A=1)) \right]$$
$$- \left[\hat{\overline{Q}}_Y(A=0, L(1)=1, M=m) \times \hat{\overline{Q}}_{L(1)}(A=0) + \hat{\overline{Q}}_Y(A=0, L(1)=0, M=m) \times (1 - \hat{\overline{Q}}_{L(1)}(A=0)) \right]$$

However NPMLE is tedious with high-dimensional intermediate confounders $L(1)$ or if mediators is repeated over time. In that case, parametric g-computation using a Monte Carlo algorithm, or g-computation by iterative conditional expectation are easier to apply.

Below, we describe three g-computation procedures for the estimation of a CDE:

- parametric g-computation, using Monte Carlo simulation
- g-computation by iterative conditional expectation
- sequential g-estimator

6.2.1   Parametric g-computation

Parametric g-computation by Monte Carlo simulation have been described by Robins (Robins 1986), Taubman *et al.* (Taubman et al. 2009), or Daniel *et al.* (Daniel et al. 2013).

1. Fit a parametric model to estimate the density of the intermediate confounder $L(1)$ conditional on its parents. If $L(1)$ is a set of several variables, it is necessary to fit a model for each variable conditional on its parents.

$$Q_{L(1)}(A) = P(L(1) = 1 \mid L(0), A) \tag{6.2}$$

2. Fit a model of the outcome Y conditional on its parents:

$$\overline{Q}_Y(A, L(1), M) = \mathbb{E}\left(Y \mid L(0), A, L(1), M\right) \tag{6.3}$$

3. Simulate individual values of $L(1)_a$ using the estimated density $\hat{Q}_{L(1)}(A = a)$ under the counterfactual scenarios setting $A = 0$ or $A = 1$

4. Estimate mean values of the outcome under the counterfactual scenarios setting $A = 0$ (or $A = 1$), $L(1) = l(1)_{A=0}$ (or $L(1) = l(1)_{A=1}$) and $M = m$, using $\hat{\overline{Q}}_Y(A = a, L(1) = l(1)_a, M = m)$

5. Estimate the controlled direct effect Ψ_{CDE_m} by the sample mean:

$$\hat{\Psi}_{\mathrm{param.gcomp}}^{\mathrm{CDE}_m} = \frac{1}{n} \sum_{i=1}^{n} \left[\hat{\overline{Q}}_Y(A = 1, L(1) = l(1)_{A=1}, M = m)_i - \hat{\overline{Q}}_Y(A = 0, L(1) = l(1)_{A=0}, M = m)_i \right]$$

$$\tag{6.4}$$

For continuous outcomes, $\overline{Q}_Y(A, L(1), M)$ functions can be estimated using linear regressions. For binary outcomes, they can be estimated using logistic regressions.

```
# 1. Fit parametric models to estimate the density of intermediate confounders,
#    conditional on the parents of the intermediate confounders
L1.model <- glm(L1 ~ L0_male + L0_parent_low_educ_lv + A0_ace,
                family = "binomial", data = df2_int)

# 2. Fit parametric models for the outcome conditional on past
Y.death.model <- glm(Y_death ~ L0_male + L0_parent_low_educ_lv + A0_ace + L1 +
                              M_smoking + A0_ace:M_smoking,
                     family = "binomial", data = df2_int)
Y.qol.model <- glm(Y_qol ~ L0_male + L0_parent_low_educ_lv + A0_ace + L1 +
                          M_smoking + A0_ace:M_smoking,
                   family = "gaussian", data = df2_int)

# 3. Simulate individual L1 values under the counterfactual scenarios setting A0=0 or A0=1
set.seed(54321)
data.A0  <- data.A1 <- df2_int
data.A0$A0_ace <- 0
data.A1$A0_ace <- 1
p.L1.A0 <- predict(L1.model, newdata = data.A0, type="response")
p.L1.A1 <- predict(L1.model, newdata = data.A1, type="response")
sim.L1.A0 <- rbinom(n = nrow(df2_int), size = 1, prob = p.L1.A0)
sim.L1.A1 <- rbinom(n = nrow(df2_int), size = 1, prob = p.L1.A1)

# 4. Estimate mean outcomes under the counterfactual scenarios setting different
#    levels of exposures for A and M:
```

```r
#     {A=0, M=0} or {A=1, M=0} or {A=0, M=1} or {A=1, M=1}

data.A0.M0 <- data.A0.M1 <- data.A0
data.A1.M0 <- data.A1.M1 <- data.A1

# L1 variable is replaced by the simulated values in step 3)
data.A0.M0$L1 <- sim.L1.A0
data.A0.M1$L1 <- sim.L1.A0
data.A1.M0$L1 <- sim.L1.A1
data.A1.M1$L1 <- sim.L1.A1

# set M to 0 or 1
data.A0.M0$M_smoking <- 0
data.A0.M1$M_smoking <- 1
data.A1.M0$M_smoking <- 0
data.A1.M1$M_smoking <- 1

# predict the probability of death
p.death.A0.M0 <- predict(Y.death.model, newdata = data.A0.M0, type="response")
p.death.A1.M0 <- predict(Y.death.model, newdata = data.A1.M0, type="response")
p.death.A0.M1 <- predict(Y.death.model, newdata = data.A0.M1, type="response")
p.death.A1.M1 <- predict(Y.death.model, newdata = data.A1.M1, type="response")

# predict the mean value of QoL
m.qol.A0.M0 <- predict(Y.qol.model, newdata = data.A0.M0, type="response")
m.qol.A1.M0 <- predict(Y.qol.model, newdata = data.A1.M0, type="response")
m.qol.A0.M1 <- predict(Y.qol.model, newdata = data.A0.M1, type="response")
m.qol.A1.M1 <- predict(Y.qol.model, newdata = data.A1.M1, type="response")

# 5. Estimate the CDE
# CDE setting M=0
CDE.death.m0.gcomp.param <- mean(p.death.A1.M0) - mean(p.death.A0.M0)
CDE.death.m0.gcomp.param
# [1] 0.06289087

CDE.qol.m0.gcomp.param <- mean(m.qol.A1.M0) - mean(m.qol.A0.M0)
CDE.qol.m0.gcomp.param
# [1] -4.838654

# CDE setting M=1
CDE.death.m1.gcomp.param <- mean(p.death.A1.M1) - mean(p.death.A0.M1)
CDE.death.m1.gcomp.param
# [1] 0.08751016

CDE.qol.m1.gcomp.param <- mean(m.qol.A1.M1) - mean(m.qol.A0.M1)
```

```
CDE.qol.m1.gcomp.param
# [1] -10.35059
```

6.2.2   G-computation by iterative conditional expectation

The following steps describe the implementation of the g-computation estimator by iterative conditional expectation for the component $\mathbb{E}(Y_{A=a',M=m})$ used in the definition of CDE $\Psi^{\mathrm{CDE}_m} = \mathbb{E}(Y_{A=1,M=m}) - \mathbb{E}(Y_{A=0,M=m})$. Interestingly, there is no need to estimate or simulate $L(1)$ density with this method.

1. Fit a logistic or a linear regression of the final outcome, conditional on the exposure A, the mediator M and all the parents of Y preceeding M, to estimate $\overline{Q}_Y = \mathbb{E}(Y \mid L(0), A, L(1), M)$;

2. Use this estimate to predict an outcome for each subject $\hat{\overline{Q}}_Y(M = m)_i$, by evaluating the regression fit $\overline{Q}_Y$ at the chosen value for the mediator $M = m$;

3. Fit a quasibinomial or a linear regression of the predicted values $\hat{\overline{Q}}_Y(M = m)_i$ conditional on the exposure A and baseline confounders $L(0)$ to estimate $\overline{Q}_{L(1)} = \mathbb{E}\left(\hat{\overline{Q}}_Y(M = m) \Big| L(0), A\right)$;

4. Use this estimate to predict the outcome $\hat{\overline{Q}}_{L(1)}(A = a')_i$ for each subject, by evaluating the regression fit $\overline{Q}_{L(1)}$ at $A = a'$;

5. Use the sample mean to estimate $\Psi^{\mathrm{CDE}_m}_{\mathrm{gcomp}}$

$$\hat{\Psi}^{\mathrm{CDE}_m}_{\mathrm{gcomp}} = \frac{1}{n} \sum_{i=1}^{n} \left[\hat{\overline{Q}}_{L(1)}(A = 1)_i - \hat{\overline{Q}}_{L(1)}(A = 0)_i \right] \qquad (6.5)$$

```
# 1) Regress the outcome on L0, A, L1 and M (and the A*M interaction if appropriate)
Y.death.model <- glm(Y_death ~ L0_male + L0_parent_low_educ_lv + A0_ace + L1 +
                                M_smoking + A0_ace:M_smoking,
                     family = "binomial", data = df2_int)
Y.qol.model <- glm(Y_qol ~ L0_male + L0_parent_low_educ_lv + A0_ace + L1 +
                             M_smoking + A0_ace:M_smoking,
                  family = "gaussian", data = df2_int)

# 2) Generate predicted values by evaluating the regression setting the mediator
#    value to M=0 or to M=1
#    (Note: it is also possible to set A=0 or A=1 to evaluate the regression at
#     exposure history of interest: {A0=1,M=0},{A0=0,M=0},{A0=1,M=1},{A0=0,M=1})
data.Mis0 <- data.Mis1 <- df2_int
```

```r
data.Mis0$M_smoking <- 0
data.Mis1$M_smoking <- 1

Q.Y.death.Mis0 <- predict(Y.death.model, newdata = data.Mis0, type="response")
Q.Y.death.Mis1 <- predict(Y.death.model, newdata = data.Mis1, type="response")

Q.Y.qol.Mis0 <- predict(Y.qol.model, newdata = data.Mis0, type="response")
Q.Y.qol.Mis1 <- predict(Y.qol.model, newdata = data.Mis1, type="response")


# 3) Regress the predicted values conditional on the exposure A
#    and baseline confounders L(0)
L1.death.Mis0.model <- glm(Q.Y.death.Mis0 ~ L0_male + L0_parent_low_educ_lv + A0_ace,
                        family = "quasibinomial", data = df2_int)
L1.death.Mis1.model <- glm(Q.Y.death.Mis1 ~ L0_male + L0_parent_low_educ_lv + A0_ace,
                        family = "quasibinomial", data = df2_int)

L1.qol.Mis0.model <- glm(Q.Y.qol.Mis0 ~ L0_male + L0_parent_low_educ_lv + A0_ace,
                        family = "gaussian", data = df2_int)
L1.qol.Mis1.model <- glm(Q.Y.qol.Mis1 ~ L0_male + L0_parent_low_educ_lv + A0_ace,
                        family = "gaussian", data = df2_int)

# 4) generate predicted values by evaluating the regression at exposure
#    of interest: {A=1} & {A=0}
data.Ais0 <- data.Ais1 <- df2_int
data.Ais0$A0_ace <- 0
data.Ais1$A0_ace <- 1

Q.L1.death.Ais0.Mis0 <- predict(L1.death.Mis0.model, newdata = data.Ais0, type="respons
Q.L1.death.Ais1.Mis0 <- predict(L1.death.Mis0.model, newdata = data.Ais1, type="respons
Q.L1.death.Ais0.Mis1 <- predict(L1.death.Mis1.model, newdata = data.Ais0, type="respons
Q.L1.death.Ais1.Mis1 <- predict(L1.death.Mis1.model, newdata = data.Ais1, type="respons

Q.L1.qol.Ais0.Mis0 <- predict(L1.qol.Mis0.model, newdata = data.Ais0, type="response")
Q.L1.qol.Ais1.Mis0 <- predict(L1.qol.Mis0.model, newdata = data.Ais1, type="response")
Q.L1.qol.Ais0.Mis1 <- predict(L1.qol.Mis1.model, newdata = data.Ais0, type="response")
Q.L1.qol.Ais1.Mis1 <- predict(L1.qol.Mis1.model, newdata = data.Ais1, type="response")

# 5) Take empirical mean of final predicted outcomes to estimate CDE
# CDE setting M=0
CDE.death.m0.gcomp.ice <- mean(Q.L1.death.Ais1.Mis0) - mean(Q.L1.death.Ais0.Mis0)
CDE.death.m0.gcomp.ice
# [1] 0.06341297

CDE.qol.m0.gcomp.ice <- mean(Q.L1.qol.Ais1.Mis0) - mean(Q.L1.qol.Ais0.Mis0)
```

```
CDE.qol.m0.gcomp.ice
# [1] -4.869509

# CDE setting M=1
CDE.death.m1.gcomp.ice <- mean(Q.L1.death.Ais1.Mis1) - mean(Q.L1.death.Ais0.Mis1)
CDE.death.m1.gcomp.ice
# [1] 0.08810508

CDE.qol.m1.gcomp.ice <- mean(Q.L1.qol.Ais1.Mis1) - mean(Q.L1.qol.Ais0.Mis1)
CDE.qol.m1.gcomp.ice
# [1] -10.38144
```

6.2.3 Sequential g-estimator

For quantitative outcomes, Vansteelandt et al. (Epidemiology 20(6);2009) described a sequential g-estimator for CDE. An extension for binary outcomes in case-control studies is also described using OR.

The following 2 steps are applied:

1. Fit a regression model for the outcome conditional on the exposure A, the mediator M, baseline and intermediate confounders $L(0)$ and $L(1)$, in order to estimate the regression coefficients $\hat{\gamma}_M$ and $\hat{\gamma}_{A*M}$ (in case of $(A*M)$ interaction effect).

$$\mathbb{E}(Y \mid L(0), A, L(1), M) = \gamma_0 + \gamma_A A + \gamma_M M + \psi_{A*M}(A*M) + \gamma_{L(0)} L(0) + \gamma_{L(1)} L(1) \tag{6.6}$$

2. Remove the effect of mediator on the outcome, by evaluating the residual outcome:

$$Y_{res} = Y - \hat{\gamma}_M M - \hat{\psi}_{A*M} \times A \times M \tag{6.7}$$

and regress the residual outcome on the exposure A and baseline confounders $L(0)$:

$$\mathbb{E}(Y_{res} \mid A, L(0)) = \alpha_0 + \psi_A A + \beta_{L(0)} L(0) \tag{6.8}$$

The controlled direct effect CDE_m can then be estimated by:

$$\hat{\Psi}_{\text{seq.g.est}}^{\text{CDE}_m} = \hat{\psi}_A + \hat{\psi}_{A*M} \times m \tag{6.9}$$

```
# 1) Regress the outcome on past
Y.qol.model <- glm(Y_qol ~ L0_male + L0_parent_low_educ_lv + A0_ace + L1 +
                          M_smoking + A0_ace:M_smoking,
                 family = "gaussian", data = df2_int)
```

```r
# 2) Calculate a residual outcome Y - (coef.M * M_smoking) - (coef.A0:M * A0:M)
Y.res <- (df2_int$Y_qol -
            (Y2.qol.model$coefficients["M_smoking"] *  df2_int$M_smoking) -
            (Y2.qol.model$coefficients["A0_ace:M_smoking"] * df2_int$A0_ace
              * data.inter1$M_smoking) )

# 3) Regress the residual outcome on the exposure A and baseline confounders L(0)
Y.res.model <- glm(Y.res ~ L0_male + L0_parent_low_educ_lv + A0_ace,
                    family = "gaussian", data = df2_int)

# 4) Use coefficients estimated from the 1st and 2nd regression to estimate CDE:
CDE.qol.m0.seq <- (Y.res.model$coefficients["A0_ace"] +
                    0*Y.qol.model$coefficients["A0_ace:M_smoking"])
CDE.qol.m0.seq
# -4.869509

CDE.qol.m1.seq <- (Y.res.model$coefficients["A0_ace"] +
                    1*Y.qol.model$coefficients["A0_ace:M_smoking"])
CDE.qol.m1.seq
# -10.38144
```

Chapter 7

Inverse Probability of Treatment Weighting (IPTW)

7.1 Estimation of the Average total effect

7.1.1 IPTW for the ATE

If the average total effect (ATE) is identifiable, $\Psi_{ATE} = \mathbb{E}(Y_{A=1}) - \mathbb{E}(Y_{A=0})$ can be expressed using Inverse probability of treatment weighting (IPTW), denoting $\mathbb{P}(A = a \mid L(0)) = g(A = a \mid L(0))$:

$$\Psi_{ATE} = \mathbb{E}\left(\frac{\mathbb{I}(A = 1)}{g(A = 1 \mid L(0))} Y \right) - \mathbb{E}\left(\frac{\mathbb{I}(A = 0)}{g(A = 0 \mid L(0))} Y \right) \qquad (7.1)$$

The following steps describe the implementation of the IPTW estimator

1. Estimate the treatment mechanism $g(A = 1 \mid L(0))$

2. Predict each individual's probability of being exposed to her own exposure

3. Apply weights corresponding to the inverse of the predicted probability $w_i = \frac{1}{\hat{g}(A_i = a \mid L(0)_i)}$

4. Use the empirical mean of the weighted outcome Y: $\hat{\mathbb{E}}(Y_a) = \frac{1}{n} \sum_{i=1}^{n} \frac{\mathbb{I}(A_i = a)}{\hat{g}(A_i = a \mid L(0)_i)} Y_i$

```
# 1. Estimate g
g.L <- glm(A0_ace ~ L0_male + L0_parent_low_educ_lv,
           family = "binomial", data = df2_int)

# 2. Predict each individual's probability of being exposed to her own exposure
# predict the probabilities P(A0_ace=1|L(0)) & P(A0_ace=0|L(0))
pred.g1.L <- predict(g.L, type="response")
pred.g0.L <- 1 - pred.g1.L
# the predicted probability of the observed treatment A_i=a is :
gA.L <- rep(NA, nrow(df2_int))
gA.L[df2_int$A0_ace==1] <- pred.g1.L[df2_int$A0_ace==1]
gA.L[df2_int$A0_ace==0] <- pred.g0.L[df2_int$A0_ace==0]

# 3. Apply weights corresponding to the inverse of the predicted probability
wt <- 1 / gA.L

# 4. Use the empirical mean of the weighted outcome
# point estimates:
IPTW.death <- mean(wt * as.numeric(df2_int$A0_ace==1) * df2_int$Y_death) -
  mean(wt * as.numeric(df2_int$A0_ace==0) * df2_int$Y_death)
IPTW.death
# [1] 0.08224947

IPTW.qol <- mean(wt * as.numeric(df2_int$A0_ace==1) * df2_int$Y_qol) -
  mean(wt * as.numeric(df2_int$A0_ace==0) * df2_int$Y_qol)
IPTW.qol
# [1] -8.436797
```

The ATE estimates using IPTW for death probability and mean quality of life
are respectively $+8.2\%$ and -8.44.

7.1.2   Stabilized IPTW for the ATE

If the average total effect (ATE) is identifiable, Ψ_{ATE} can be estimated using a
stabilized IPTW estimator:

$$\hat{\mathbb{E}}(Y_1) - \hat{\mathbb{E}}(Y_0) = \frac{\frac{1}{n}\sum_{i=1}^{n}\frac{\mathbb{I}(A_i=1)\hat{g}^*(A_i=1)}{\hat{g}(A_i=1|L(0)_i)}Y_i}{\frac{1}{n}\sum_{i=1}^{n}\frac{\mathbb{I}(A_i=1)\hat{g}^*(A_i=1)}{\hat{g}(A_i=1|L(0)_i)}} - \frac{\frac{1}{n}\sum_{i=1}^{n}\frac{\mathbb{I}(A_i=0)\hat{g}^*(A_i=0)}{\hat{g}(A_i=0|L(0)_i)}Y_i}{\frac{1}{n}\sum_{i=1}^{n}\frac{\mathbb{I}(A_i=0)\hat{g}^*(A_i=0)}{\hat{g}(A_i=0|L(0)_i)}} \quad (7.2)$$

The estimation algorithm is the same as for IPTW, but taking into account any
non-null function of A $(g^*(A_i = a))$ in the denominator of the weight in step 3,
and applying the stabilized estimator in step 4.

```
# 3. For example, applying g^*(A) = 1
# 4. Applying the stabilized estimator
# point estimates:
s.IPTW.death <- (mean(wt * as.numeric(df2_int$A0_ace==1) * df2_int$Y_death) /
                   mean(wt * as.numeric(df2_int$A0_ace==1))) -
  (mean(wt * as.numeric(df2_int$A0_ace==0) * df2_int$Y_death) /
     mean(wt * as.numeric(df2_int$A0_ace==0)))
s.IPTW.death
# [1] 0.08294185

s.IPTW.qol <- (mean(wt * as.numeric(df2_int$A0_ace==1) * df2_int$Y_qol) /
                 mean(wt * as.numeric(df2_int$A0_ace==1))) -
  (mean(wt * as.numeric(df2_int$A0_ace==0) * df2_int$Y_qol) /
     mean(wt * as.numeric(df2_int$A0_ace==0)))
s.IPTW.qol
# [1] -8.291992
```

The ATE estimates using stabilized IPTW for death probability and mean quality of life are respectively +8.3% and -8.29.

Chapter 8

Targeted Maximum Likelihood Estimation (TMLE)

Chapter 9

Appendix A: Data generating mechanisms

The data generating mechanisms are characterized by a causal model and a statistical model that generate data given in example.

In the first causal model, the mediator-outcome confounder $L(1)$ is not affected by the exposure. In the second causal model, the mediator-outcome confounder $L(1)$ is affected by the exposure.

9.1 First causal model: Data generating mechanism without mediator-outcome confounder affected by the exposure

This data generating mechanism is defined by the following set of structural equations:

$$
\begin{aligned}
P(L(0)_{male} = 1) &= p_{L(0)_{male}} \\
P(L(0)_{parent} = 1) &= p_{L(0)_{parent}} \\
P(A_{ACE} = 1) &= \beta_A + \beta^A_{male} \times L(0)_{male} + \beta^A_{parent} \times L(0)_{parent} \\
P(L(1) = 1) &= p_{L(1)} \\
P(M_{smoking} = 1) &= \beta_M + \beta^M_{male} \times L(0)_{male} + \beta^M_{parent} \times L(0)_{parent} + \beta^M_{L(1)} \times L(1) + \beta^M_A \times A_{ACE} \\
P(Y_{death} = 1) &= \beta_Y + \beta^Y_{male} \times L(0)_{male} + \beta^Y_{parent} \times L(0)_{parent} + \beta^Y_{L(1)} \times L(1) \\
&\quad + \beta^Y_A \times A_{ACE} + \beta^Y_M \times M_{smoking} + \beta^Y_{A*M} \times A_{ACE} \times M_{smoking} \\
\mathbb{E}(Y_{Qol} = 1) &= \gamma_Y + \gamma^Y_{male} \times L(0)_{male} + \gamma^Y_{parent} \times L(0)_{parent} + \gamma^Y_{L(1)} \times L(1) \\
&\quad + \gamma^Y_A \times A_{ACE} + \gamma^Y_M \times M_{smoking} + \gamma^Y_{A*M} \times A_{ACE} \times M_{smoking} + \varepsilon_Y
\end{aligned}
$$

where $\varepsilon_Y \sim \mathcal{N}(0, \sigma_Y = 10)$.

One can set the parameters of these structural equations using the following
function `param.causal.model.1()`:

```r
param.causal.model.1 <- function(A.M.interaction = NULL) {
# L0
p_L0_male <- 0.5
p_L0_parent_low_educ_lv <- 0.65

# A: A0_ace <- rbinom( 0.05 + 0.04 * L0_male + 0.06 * L0_parent_low_educ_lv )
b_A <- 0.05    # reference prevalence is 5%
b_male_A <- 0.04   # + 0.04 for the effect of L0_male -> A0_ace
b_parent_educ_A <- 0.06   # +0.06 for the effect of L0_parent_low_educ_lv -> A0_ace

# L1: intermediate confounder between M and Y, not influenced by A
p_L1 <- 0.3

# M: M_smoking <- rbinom( 0.2 + 0.05 * L0_male + 0.06 * L0_parent_low_educ_lv + 0.07 *
#                         0.1 * A0_ace )
b_M <- 0.2 # reference prevalence is 20%
b_male_M <- 0.05 # +0.05 for the effect of L0_male -> M_smoking
b_parent_educ_M <- 0.06 # +0.06 for the effect of L0_parent_low_educ_lv -> M_smoking
b_L1_M <- 0.07 # +0.07 for the effect of L1 -> M_smoking
b_A_M <- 0.1 # +0.10 for the effect of A0_ace -> M_smoking

# Y binary: rbinom( 0.10 + 0.06 * L0_male + 0.04 * L0_parent_low_educ_lv + 0.05 * A0_a
#                  0.07 * L1 + 0.08 * M_smoking +
#                  0.03 * A0_ace * M_smoking * A.M.inter )
b_Y <- 0.1 # reference prevalence is 10%
b_male_Y <- 0.06 # +0.06 for the effect of L0_male -> Y
b_parent_educ_Y <- 0.04 # +0.04 for the effect of L0_parent_low_educ_lv -> Y
b_A_Y <- 0.05 # 0.05 for the effect of A0_ace -> Y
b_L1_Y <- 0.07 # +0.07 for the effect of L1 -> Y
b_M_Y <- 0.08 # 0.08 for the effect of M_smoking -> Y
b_AM_Y <- 0.03 # 0.03 for the interaction effect A0_ace * M_smoking -> Y

# Y continuous: (75 - 1 * L0_male - 3 * L0_parent_low_educ_lv - 4 * A0_ace -3.5 * L1 -
#                9 * M_smoking -5 * A0_ace * M_smoking * A.M.inter ) +
#                rnorm(N, mean = 0, sd = 10)
mu_Y <- 75 # reference mean for QoL
c_male_Y <- -1 # -1 for the effect of L0_male -> Y
c_parent_educ_Y <- -3 # -3 for the effect of L0_parent_low_educ_lv -> Y
c_A_Y <- -4 # -4 for the effect of A0_ace -> Y
c_L1_Y <- -3.5 # -3.5 for the effect of L1 -> Y
c_M_Y <- -9 # -9 for the effect of M_smoking -> Y
c_AM_Y <- -5   # - 5 for the interaction effect A0_ace * M_smoking  -> Y
sd_Y <- 10 # standard deviation of the residuals
```

```
# A*M interaction ?
A.M.inter <- A.M.interaction

coef <- c( p_L0_male = p_L0_male, p_L0_parent_low_educ_lv = p_L0_parent_low_educ_lv,
           b_A = b_A, b_male_A = b_male_A, b_parent_educ_A = b_parent_educ_A,
           p_L1 = p_L1,
           b_M = b_M, b_male_M = b_male_M, b_parent_educ_M = b_parent_educ_M,
           b_L1_M = b_L1_M, b_A_M = b_A_M,
           b_Y = b_Y, b_male_Y = b_male_Y, b_parent_educ_Y = b_parent_educ_Y,
           b_A_Y = b_A_Y, b_L1_Y = b_L1_Y, b_M_Y = b_M_Y, b_AM_Y = b_AM_Y,
           mu_Y = mu_Y, c_male_Y = c_male_Y, c_parent_educ_Y = c_parent_educ_Y,
           c_A_Y = c_A_Y, c_L1_Y = c_L1_Y, c_M_Y = c_M_Y, c_AM_Y = c_AM_Y,
           sd_Y = sd_Y, A.M.inter = A.M.inter)

return(coef)
}
```

9.2 Second causal model: Data generating mechanism with mediator-outcome confounder affected by the exposure

This data generating mechanism is defined by the following set of structural equations:

$$
\begin{aligned}
P(L(0)_{male} = 1) &= p_{L(0)_{male}} \\
P(L(0)_{parent} = 1) &= p_{L(0)_{parent}} \\
P(A_{ACE} = 1) &= \beta_A + \beta_{male}^A \times L(0)_{male} + \beta_{parent}^A \times L(0)_{parent} \\
P(L(1) = 1) &= \beta_{L(1)} + \beta_{male}^{L(1)} \times L(0)_{male} + \beta_{parent}^{L(1)} \times L(0)_{parent} + \beta_A^{L(1)} \times A_{ACE} \\
P(M_{smoking} = 1) &= \beta_M + \beta_{male}^M \times L(0)_{male} + \beta_{parent}^M \times L(0)_{parent} + \beta_{L(1)}^M \times L(1) + \beta_A^M \times A_{ACE} \\
P(Y_{death} = 1) &= \beta_Y + \beta_{male}^Y \times L(0)_{male} + \beta_{parent}^Y \times L(0)_{parent} + \beta_{L(1)}^Y \times L(1) \\
&\quad + \beta_A^Y \times A_{ACE} + \beta_M^Y \times M_{smoking} + \beta_{A*M}^Y \times A_{ACE} \times M_{smoking} \\
\mathbb{E}(Y_{Qol} = 1) &= \gamma_Y + \gamma_{male}^Y \times L(0)_{male} + \gamma_{parent}^Y \times L(0)_{parent} + \gamma_{L(1)}^Y \times L(1) \\
&\quad + \gamma_A^Y \times A_{ACE} + \gamma_M^Y \times M_{smoking} + \gamma_{A*M}^Y \times A_{ACE} \times M_{smoking} + \varepsilon_Y
\end{aligned}
$$

where $\varepsilon_Y \sim \mathcal{N}(0, \sigma_Y = 10)$.

One can set the parameters of these structural equations using the following function `param.causal.model.2()`:

```
param.causal.model.2 <- function(A.M.interaction = NULL) {
# L0
p_L0_male <- 0.5
```

```r
p_L0_parent_low_educ_lv <- 0.65

# A: A0_ace <- rbinom( 0.05 + 0.04 * L0_male + 0.06 * L0_parent_low_educ_lv )
b_A <- 0.05    # reference prevalence is 5%
b_male_A <- 0.04   # + 0.04 for the effect of L0_male -> A0_ace
b_parent_educ_A <- 0.06   # +0.06 for the effect of L0_parent_low_educ_lv -> A0_ace

# L1: L1 <- rbinom( 0.30 - 0.05 * L0_male + 0.08 * L0_parent_low_educ_lv +
#                   0.2 * A0_ace )
b_L1 <- 0.30    # reference prevalence is 30%
b_male_L1 <- -0.05   # - 0.05 for the effect of L0_male -> L1
b_parent_L1 <- +0.08 # + 0.08 for the effect of L0_parent_low_educ_lv -> L1
b_A_L1 <- +0.2 # +0.2 for the effect of A0_ace -> L1

# M: M_smoking <- rbinom( 0.2 + 0.05 * L0_male + 0.06 * L0_parent_low_educ_lv +
#                          0.2 * L1 + 0.1 * A0_ace )
b_M <- 0.2 # reference prevalence is 20%
b_male_M <- 0.05 # +0.05 for the effect of L0_male -> M_smoking
b_parent_educ_M <- 0.06 # +0.06 for the effect of L0_parent_low_educ_lv -> M_smoking
b_A_M <- 0.1 # +0.10 for the effect of A0_ace -> M_smoking
b_L1_M <- 0.2 # +0.2 for the effect of L1 -> M_smoking

# Y binary: rbinom( 0.10 + 0.06 * L0_male + 0.04 * L0_parent_low_educ_lv +
#                   0.05 * A0_ace + 0.07 * L1 + 0.08 * M_smoking +
#                   0.03 * A0_ace * M_smoking * A.M.inter )
b_Y <- 0.1 # reference prevalence is 10%
b_male_Y <- 0.06 # +0.06 for the effect of L0_male -> Y
b_parent_educ_Y <- 0.04 # +0.04 for the effect of L0_parent_low_educ_lv -> Y
b_A_Y <- 0.05 # 0.05 for the effect of A0_ace -> Y
b_L1_Y <- 0.07 # +0.07 for the effect of L1 -> Y
b_M_Y <- 0.08 # 0.08 for the effect of M_smoking -> Y
b_AM_Y <- 0.03 # 0.03 for the interaction effect A0_ace * M_smoking -> Y

# Y continuous: (75 - 1 * L0_male - 3 * L0_parent_low_educ_lv - 4 * A0_ace +
#                -3.5 * L1 - 9 * M_smoking +
#                -5 * A0_ace * M_smoking * A.M.inter ) + rnorm(N, mean = 0, sd = 10)
mu_Y <- 75 # reference mean for QoL
c_male_Y <- -1 # -1 for the effect of L0_male -> Y
c_parent_educ_Y <- -3 # -3 for the effect of L0_parent_low_educ_lv -> Y
c_A_Y <- -4 # -4 for the effect of A0_ace -> Y
c_L1_Y <- -5 # -5 for the effect of L1 -> Y
c_M_Y <- -9 # -9 for the effect of M_smoking -> Y
c_AM_Y <- -5   # - 5 for the interaction effect A0_ace * M_smoking  -> Y
sd_Y <- 10 # standard deviation of the residuals
```

```r
# A*M interaction ?
A.M.inter <- A.M.interaction

coef <- c( p_L0_male = p_L0_male, p_L0_parent_low_educ_lv = p_L0_parent_low_educ_lv,
          b_A = b_A, b_male_A = b_male_A, b_parent_educ_A = b_parent_educ_A,
          b_L1 = b_L1, b_male_L1 = b_male_L1, b_parent_L1 = b_parent_L1,
           b_A_L1 = b_A_L1,
          b_M = b_M, b_male_M = b_male_M, b_parent_educ_M = b_parent_educ_M,
           b_L1_M = b_L1_M, b_A_M = b_A_M,
          b_Y = b_Y, b_male_Y = b_male_Y, b_parent_educ_Y = b_parent_educ_Y,
           b_A_Y = b_A_Y, b_L1_Y = b_L1_Y, b_M_Y = b_M_Y, b_AM_Y = b_AM_Y,
          mu_Y = mu_Y, c_male_Y = c_male_Y, c_parent_educ_Y = c_parent_educ_Y,
           c_A_Y = c_A_Y, c_L1_Y = c_L1_Y, c_M_Y = c_M_Y, c_AM_Y = c_AM_Y,
             sd_Y = sd_Y, A.M.inter = A.M.inter)


  return(coef)
}
```

9.3 Simulation of the four data sets used in examples

9.3.1 Data sets generated from the causal model 1

The following function `gen.data.causal.model.1` can be used to simulate data sets using the parameters defined previously in the `param.causal.model.1` function.

```r
gen.data.causal.model.1 <- function(N, A.M.inter) { # input parameters are the
  #   sample size N and the presence of A*M interaction with A.M.inter = 0 or 1

  b <- param.causal.model.1(A.M.interaction = A.M.inter)

  # baseline confounders: parent's educational level=L0_parent_low_educ_lv & sex=L0_male
  L0_male <- rbinom(N, size = 1, prob = b["p_L0_male"])
  L0_parent_low_educ_lv <- rbinom(N, size = 1, prob = b["p_L0_parent_low_educ_lv"])

  # exposure: A0_ace
  A0_ace <- rbinom(N, size = 1, prob =  b["b_A"] +
                      b["b_male_A"] * L0_male +
                      b["b_parent_educ_A"] * L0_parent_low_educ_lv )

  # intermediate confounder between M_smoking and Y, not affected by A0 L1
  L1 <- rbinom(N, size = 1, prob = b["p_L1"])
```

```r
  # mediator: M_smoking
  M_smoking <- rbinom(N, size = 1, prob = b["b_M"] +
                            b["b_male_M"] * L0_male +
                            b["b_parent_educ_M"] * L0_parent_low_educ_lv +
                            b["b_A_M"] * A0_ace +
                            b["b_L1_M"] * L1)

  # Y_death
  Y_death <- rbinom(N, size = 1, prob = b["b_Y"] +
                            b["b_male_Y"] * L0_male +
                            b["b_parent_educ_Y"] * L0_parent_low_educ_lv +
                            b["b_A_Y"] * A0_ace +
                            b["b_L1_Y"] * L1 +
                            b["b_M_Y"] * M_smoking +
                            b["b_AM_Y"] * A0_ace * M_smoking * A.M.inter )

  # Y_qol
  Y_qol <- ( b["mu_Y"] +
                b["c_male_Y"] * L0_male +
                b["c_parent_educ_Y"] * L0_parent_low_educ_lv +
                b["c_A_Y"] * A0_ace +
                b["c_L1_Y"] * L1 +
                b["c_M_Y"] * M_smoking +
                b["c_AM_Y"] * A0_ace * M_smoking * A.M.inter ) +
     rnorm(N, mean = 0, sd = b["sd_Y"])

  # data.frame
  data.sim <- data.frame(L0_male, L0_parent_low_educ_lv, A0_ace, L1, M_smoking,
                            Y_death, Y_qol)

  return( data.sim )
}
```

Applying a sample size N=10000, we generate the `df1.csv` and `df1_int.csv` data sets.

```r
set.seed(1234)
df1 <- gen.data.causal.model.1(N=10000, A.M.inter=0)
write.csv(df1, file = "df1.csv", row.names = FALSE)

set.seed(1234)
df1_int <- gen.data.causal.model.1(N=10000, A.M.inter=1)
write.csv(df1_int, file = "df1_int.csv", row.names = FALSE)
```

```
head(df1)
```

```
##   L0_male L0_parent_low_educ_lv A0_ace L1 M_smoking Y_death    Y_qol
## 1       0                     1      0  1         0       0 93.41819
## 2       1                     1      1  1         0       1 64.03221
## 3       1                     1      0  0         0       0 75.56249
## 4       1                     0      0  0         0       0 89.77055
## 5       1                     1      0  0         0       0 77.22353
## 6       1                     1      0  0         1       0 73.87975
```

```
head(df1_int)
```

```
##   L0_male L0_parent_low_educ_lv A0_ace L1 M_smoking Y_death    Y_qol
## 1       0                     1      0  1         0       0 93.41819
## 2       1                     1      1  1         0       1 64.03221
## 3       1                     1      0  0         0       0 75.56249
## 4       1                     0      0  0         0       0 89.77055
## 5       1                     1      0  0         0       0 77.22353
## 6       1                     1      0  0         1       0 73.87975
```

9.3.2   Data sets generated from the causal model 2

The following function gen.data.causal.model.2 can be used to simulate data sets using the parameters defined previously in the param.causal.model.2 function.

```
gen.data.causal.model.2 <- function(N, A.M.inter) { # input parameters are the
  #   sample size N and the presence of A*M interaction with A.M.inter = 0 or 1

  b <- param.causal.model.2(A.M.interaction = A.M.inter)

  # baseline confounders: parent's educational level=L0_parent_low_educ_lv & sex=L0_male
  L0_male <- rbinom(N, size = 1, prob = b["p_L0_male"])
  L0_parent_low_educ_lv <- rbinom(N, size = 1, prob = b["p_L0_parent_low_educ_lv"])

  # exposure: A0_ace
  A0_ace <- rbinom(N, size = 1, prob =  b["b_A"] +
                  b["b_male_A"] * L0_male +
                  b["b_parent_educ_A"] * L0_parent_low_educ_lv )

  # intermediate confounder between M_smoking and Y,
  L1 <- rbinom(N, size = 1, prob = b["b_L1"] +
              b["b_male_L1"] * L0_male +
```

```
                          b["b_parent_L1"] * L0_parent_low_educ_lv +
                          b["b_A_L1"]* A0_ace)

  # mediator: M_smoking
  M_smoking <- rbinom(N, size = 1, prob = b["b_M"] +
                          b["b_male_M"] * L0_male +
                          b["b_parent_educ_M"] * L0_parent_low_educ_lv +
                          b["b_A_M"] * A0_ace +
                          b["b_L1_M"] * L1)

  # Y_death
  Y_death <- rbinom(N, size = 1, prob = b["b_Y"] +
                          b["b_male_Y"] * L0_male +
                          b["b_parent_educ_Y"] * L0_parent_low_educ_lv +
                          b["b_A_Y"] * A0_ace +
                          b["b_L1_Y"] * L1 +
                          b["b_M_Y"] * M_smoking +
                          b["b_AM_Y"] * A0_ace * M_smoking * A.M.inter )

  # Y_qol
  Y_qol <- ( b["mu_Y"] +
                  b["c_male_Y"] * L0_male +
                  b["c_parent_educ_Y"] * L0_parent_low_educ_lv +
                  b["c_A_Y"] * A0_ace +
                  b["c_L1_Y"] * L1 +
                  b["c_M_Y"] * M_smoking +
                  b["c_AM_Y"] * A0_ace * M_smoking * A.M.inter ) +
    rnorm(N, mean = 0, sd = b["sd_Y"])

  # data.frame
  data.sim <- data.frame(L0_male, L0_parent_low_educ_lv, A0_ace, L1, M_smoking,
                          Y_death, Y_qol)

  return( data.sim )
}
```

Applying a sample size N=10000, we generate the `df2.csv` and `df2_int.csv` data sets.

```
set.seed(1234)
df2 <- gen.data.causal.model.2(N=10000, A.M.inter=0)
write.csv(df2, file = "df2.csv", row.names = FALSE)

set.seed(1234)
df2_int <- gen.data.causal.model.2(N=10000, A.M.inter=1)
```

```
write.csv(df2_int, file = "df2_int.csv", row.names = FALSE)
```

```
tail(df2)
```

```
##       L0_male L0_parent_low_educ_lv A0_ace L1 M_smoking Y_death    Y_qol
## 9995        0                     1      0  1         1       0 53.25115
## 9996        0                     1      0  0         1       0 66.36484
## 9997        0                     1      1  1         1       0 74.20579
## 9998        1                     1      0  0         1       0 41.30248
## 9999        0                     0      0  1         0       0 85.60169
## 10000       1                     1      0  0         0       0 61.56969
```

```
tail(df2_int)
```

```
##       L0_male L0_parent_low_educ_lv A0_ace L1 M_smoking Y_death    Y_qol
## 9995        0                     1      0  1         1       0 53.25115
## 9996        0                     1      0  0         1       0 66.36484
## 9997        0                     1      1  1         1       0 69.20579
## 9998        1                     1      0  0         1       0 41.30248
## 9999        0                     0      0  1         0       0 85.60169
## 10000       1                     1      0  0         0       0 61.56969
```

Chapter 10

Appendix B: Calculation of the true causal quantities

10.1 True causal quantities without mediator-ouctome confounder affected by the exposure

10.1.1 Average total effects (ATE)

The following function `true.ATE1` can be used to run the calculation for the average total effects (ATE).

```
true.ATE1 <- function(interaction = NULL) {
  b <- param.causal.model.1(A.M.interaction = interaction)

  # binary outcome (death)
  S <- cbind(expand.grid(c(0,1),c(0,1),c(0,1), c(0,1)), rep(NA,n=2^4))
  colnames(S) <- list("male","parent_educ","L1","M","sum")
  for (n in 1:16) {
    S[n,"sum"] <- ( ( ( b["b_Y"] +
                      b["b_male_Y"] * S[n,"male"] +
                      b["b_parent_educ_Y"] * S[n,"parent_educ"] +
                      b["b_A_Y"] * 1 +
                      b["b_L1_Y"] * S[n,"L1"] +
                      b["b_M_Y"] * S[n,"M"] +
                      b["b_AM_Y"] * 1 * S[n,"M"] * b["A.M.inter"] ) *
                    (( b["b_M"] +
                          b["b_male_M"] * S[n,"male"] +
```

```r
                            b["b_parent_educ_M"] * S[n,"parent_educ"] +
                            b["b_L1_M"] * S[n,"L1"] +
                            b["b_A_M"] * 1 )^( S[n,"M"] )) *
                    (( 1 - (b["b_M"] +
                            b["b_male_M"] * S[n,"male"] +
                            b["b_parent_educ_M"] * S[n,"parent_educ"] +
                            b["b_L1_M"] * S[n,"L1"] +
                            b["b_A_M"] * 1) )^( 1 - S[n,"M"] )) ) -
                    ( ( b["b_Y"] +
                            b["b_male_Y"] * S[n,"male"] +
                            b["b_parent_educ_Y"] * S[n,"parent_educ"] +
                            b["b_A_Y"] * 0 +
                            b["b_L1_Y"] * S[n,"L1"] +
                            b["b_M_Y"] * S[n,"M"] +
                            b["b_AM_Y"] * 0 * S[n,"M"] * b["A.M.inter"] ) *
                    (( b["b_M"] +
                            b["b_male_M"] * S[n,"male"] +
                            b["b_parent_educ_M"] * S[n,"parent_educ"] +
                            b["b_L1_M"] * S[n,"L1"] +
                            b["b_A_M"] * 0 )^( S[n,"M"] )) *
                    (( 1 - (b["b_M"] +
                                b["b_male_M"] * S[n,"male"] +
                                b["b_parent_educ_M"] * S[n,"parent_educ"] +
                                b["b_L1_M"] * S[n,"L1"] +
                                b["b_A_M"] * 0) )^( 1 - S[n,"M"] )) ) ) *
    ((b["p_L1"])^(S[n,"L1"])) *
    ((1 - b["p_L1"])^(1 - S[n,"L1"])) *
    ((b["p_L0_male"])^(S[n,"male"])) *
    ((1 - b["p_L0_male"])^(1 - S[n,"male"])) *
    ((b["p_L0_parent_low_educ_lv"])^(S[n,"parent_educ"])) *
    ((1 - b["p_L0_parent_low_educ_lv"])^(1 - S[n,"parent_educ"]))
}

ATE.death <- sum(S[,"sum"])


# quantitative outcome (QoL)
S <- cbind(expand.grid(c(0,1),c(0,1),c(0,1), c(0,1)), rep(NA,n=2^4))
colnames(S) <- list("male","parent_educ","L1","M","sum")
for (n in 1:16) {
  S[n,"sum"] <- ( ( ( b["mu_Y"] +
                        b["c_male_Y"] * S[n,"male"] +
                        b["c_parent_educ_Y"] * S[n,"parent_educ"] +
                        b["c_A_Y"] * 1 +
                        b["c_L1_Y"] * S[n,"L1"] +
```

```r
                            b["c_M_Y"] * S[n,"M"] +
                            b["c_AM_Y"] * 1 * S[n,"M"] * b["A.M.inter"] ) *
                          (( b["b_M"] +
                                b["b_male_M"] * S[n,"male"] +
                                b["b_parent_educ_M"] * S[n,"parent_educ"] +
                                b["b_L1_M"] * S[n,"L1"] +
                                b["b_A_M"] * 1 )^( S[n,"M"] )) *
                          (( 1 - (b["b_M"] +
                                    b["b_male_M"] * S[n,"male"] +
                                    b["b_parent_educ_M"] * S[n,"parent_educ"] +
                                    b["b_L1_M"] * S[n,"L1"] +
                                    b["b_A_M"] * 1) )^( 1 - S[n,"M"] )) ) -
                      ( ( b["mu_Y"] +
                            b["c_male_Y"] * S[n,"male"] +
                            b["c_parent_educ_Y"] * S[n,"parent_educ"] +
                            b["c_A_Y"] * 0 +
                            b["c_L1_Y"] * S[n,"L1"] +
                            b["c_M_Y"] * S[n,"M"] +
                            b["c_AM_Y"] * 0 * S[n,"M"] * b["A.M.inter"] ) *
                          (( b["b_M"] +
                                b["b_male_M"] * S[n,"male"] +
                                b["b_parent_educ_M"] * S[n,"parent_educ"] +
                                b["b_L1_M"] * S[n,"L1"] +
                                b["b_A_M"] * 0 )^( S[n,"M"] )) *
                          (( 1 - (b["b_M"] +
                                    b["b_male_M"] * S[n,"male"] +
                                    b["b_parent_educ_M"] * S[n,"parent_educ"] +
                                    b["b_L1_M"] * S[n,"L1"] +
                                    b["b_A_M"] * 0) )^( 1 - S[n,"M"] )) ) ) *
      ((b["p_L1"])^(S[n,"L1"])) *
      ((1 - b["p_L1"])^(1 - S[n,"L1"])) *
      ((b["p_L0_male"])^(S[n,"male"])) *
      ((1 - b["p_L0_male"])^(1 - S[n,"male"])) *
      ((b["p_L0_parent_low_educ_lv"])^(S[n,"parent_educ"])) *
      ((1 - b["p_L0_parent_low_educ_lv"])^(1 - S[n,"parent_educ"]))
}

ATE.qol <- sum(S[,"sum"])

return(list(ATE.death = ATE.death, ATE.qol = ATE.qol))

}


true.ATE1.no.inter <- true.ATE1(interaction = 0)
```

```
true.ATE1.with.inter <- true.ATE1(interaction = 1)
```

The average total effects $\text{ATE} = \mathbb{E}(Y_1) - \mathbb{E}(Y_0)$ are:

- 0.058 for death and -4.9 for quality of life without interaction;
- 0.06955 for death and -6.825 for quality of life with interaction.

10.1.2 Controlled direct effects (CDE)

The following function `true.CDE1` can be used to run the calculation for controlled direct effects (CDE).

```
true.CDE1 <- function(interaction = NULL) {
  b <- param.causal.model.1(A.M.interaction = interaction)

  # binary outcome (death)
  # we estimate both CDE, fixing do(M) = 0 et do(M) = 1 and
  # using the corresponding lines in the S matrix
  S <- cbind(expand.grid(c(0,1),c(0,1),c(0,1),c(0,1)), rep(NA,n=2^3))
  colnames(S) <- list("male","parent_educ","L1","M","sum")
  for (n in 1:16) {
    S[n,"sum"] <- ( ( b["b_Y"] +
                        b["b_male_Y"] * S[n,"male"] +
                        b["b_parent_educ_Y"] * S[n,"parent_educ"] +
                        b["b_A_Y"] * 1 +
                        b["b_L1_Y"] * S[n,"L1"] +
                        b["b_M_Y"] * S[n,"M"] +
                        b["b_AM_Y"] * 1 * S[n,"M"] * b["A.M.inter"] ) -
                      ( b["b_Y"] +
                          b["b_male_Y"] * S[n,"male"] +
                          b["b_parent_educ_Y"] * S[n,"parent_educ"] +
                          b["b_A_Y"] * 0 +
                          b["b_L1_Y"] * S[n,"L1"] +
                          b["b_M_Y"] * S[n,"M"] +
                          b["b_AM_Y"] * 0 * S[n,"M"] * b["A.M.inter"] ) ) *
      ((b["p_L1"])^(S[n,"L1"])) *
      ((1 - b["p_L1"])^(1 - S[n,"L1"])) *
      ((b["p_L0_male"])^(S[n,"male"])) *
      ((1 - b["p_L0_male"])^(1 - S[n,"male"])) *
      ((b["p_L0_parent_low_educ_lv"])^(S[n,"parent_educ"])) *
      ((1 - b["p_L0_parent_low_educ_lv"])^(1 - S[n,"parent_educ"]))
  }

  CDE.M0.death <- sum(S[1:8,"sum"])
```

```
CDE.M1.death <- sum(S[9:16,"sum"])

# quantitative outcome (QoL)
# we estimate both CDE, fixing do(M) = 0 et do(M) = 1 and using
# the corresponding lines in the S matrix
for (n in 1:16) {
  S[n,"sum"] <- ( ( b["mu_Y"] +
                        b["c_male_Y"] * S[n,"male"] +
                        b["c_parent_educ_Y"] * S[n,"parent_educ"] +
                        b["c_A_Y"] * 1 +
                        b["c_L1_Y"] * S[n,"L1"] +
                        b["c_M_Y"] * S[n,"M"] +
                        b["c_AM_Y"] * 1 * S[n,"M"] * b["A.M.inter"] ) -
                      ( b["mu_Y"] +
                          b["c_male_Y"] * S[n,"male"] +
                          b["c_parent_educ_Y"] * S[n,"parent_educ"] +
                          b["c_A_Y"] * 0 +
                          b["c_L1_Y"] * S[n,"L1"] +
                          b["c_M_Y"] * S[n,"M"] +
                          b["c_AM_Y"] * 0 * S[n,"M"] * b["A.M.inter"] ) ) *
    ((b["p_L1"])^(S[n,"L1"])) *
    ((1 - b["p_L1"])^(1 - S[n,"L1"])) *
    ((b["p_L0_male"])^(S[n,"male"])) *
    ((1 - b["p_L0_male"])^(1 - S[n,"male"])) *
    ((b["p_L0_parent_low_educ_lv"])^(S[n,"parent_educ"])) *
    ((1 - b["p_L0_parent_low_educ_lv"])^(1 - S[n,"parent_educ"]))
  }

CDE.M0.qol <- sum(S[1:8,"sum"])
CDE.M1.qol <- sum(S[9:16,"sum"])

return(list(CDE.M0.death = CDE.M0.death, CDE.M1.death = CDE.M1.death,
            CDE.M0.qol = CDE.M0.qol, CDE.M1.qol = CDE.M1.qol))
}
```

```
true.CDE1.no.inter <- true.CDE1(interaction = 0)

true.CDE1.with.inter <- true.CDE1(interaction = 1)
```

Setting $do(M = 0)$, the controlled direct effects $\text{CDE}_{M=0} = \mathbb{E}\left(Y_{1,0}\right) - \mathbb{E}\left(Y_{0,0}\right)$ are:

- 0.05 for death and -4 for quality of life without interaction,
- 0.05 for death and -4 for quality of life with interaction.

Setting $do(M = 1)$, the controlled direct effects $\text{CDE}_{M=1} = \mathbb{E}\left(Y_{1,1}\right) - \mathbb{E}\left(Y_{0,1}\right)$ are:

- 0.05 for death and -4 for quality of life without interaction,
- 0.08 for death and -9 for quality of life with interaction.

10.1.3 Pure natural direct effect and Total natural indirect effect

The following function `true.PNDE.TNIE1` can be used to run the calculation for pure natural direct effects (PNDE) and total natural indirect effects (TNIE).

```
true.PNDE.TNIE1 <- function(interaction = NULL) {
  b <- param.causal.model.1(A.M.interaction = interaction)

  # binary outcome (death)
  S <- cbind(expand.grid(c(0,1),c(0,1),c(0,1),c(0,1)), rep(NA,n=2^4), rep(NA,n=2^4))
  colnames(S) <- list("male","parent_educ","L1","M","sum.pnde", "sum.tnie")
  for (n in 1:16) {
    # PNDE
    S[n,"sum.pnde"] <- ( ( b["b_Y"] +
                             b["b_male_Y"] * S[n,"male"] +
                             b["b_parent_educ_Y"] * S[n,"parent_educ"] +
                             b["b_A_Y"] * 1 +
                             b["b_L1_Y"] * S[n,"L1"] +
                             b["b_M_Y"] * S[n,"M"] +
                             b["b_AM_Y"] * 1 * S[n,"M"] * b["A.M.inter"] ) -
                           ( b["b_Y"] +
                               b["b_male_Y"] * S[n,"male"] +
                               b["b_parent_educ_Y"] * S[n,"parent_educ"] +
                               b["b_A_Y"] * 0 +
                               b["b_L1_Y"] * S[n,"L1"] +
                               b["b_M_Y"] * S[n,"M"] +
                               b["b_AM_Y"] * 0 * S[n,"M"] * b["A.M.inter"] ) ) *
        (( b["b_M"] +
             b["b_male_M"] * S[n,"male"] +
             b["b_parent_educ_M"] * S[n,"parent_educ"] +
             b["b_L1_M"] * S[n,"L1"] +
             b["b_A_M"] * 0 )^( S[n,"M"] )) *
        (( 1 - (b["b_M"] +
                  b["b_male_M"] * S[n,"male"] +
                  b["b_parent_educ_M"] * S[n,"parent_educ"] +
                  b["b_L1_M"] * S[n,"L1"] +
                  b["b_A_M"] * 0) )^( 1 - S[n,"M"] ))  *
```

```r
    ((b["p_L1"])^(S[n,"L1"])) *
    ((1 - b["p_L1"])^(1 - S[n,"L1"])) *
    ((b["p_L0_male"])^(S[n,"male"])) *
    ((1 - b["p_L0_male"])^(1 - S[n,"male"])) *
    ((b["p_L0_parent_low_educ_lv"])^(S[n,"parent_educ"])) *
    ((1 - b["p_L0_parent_low_educ_lv"])^(1 - S[n,"parent_educ"]))

  # TNIE
  S[n,"sum.tnie"] <- ( b["b_Y"] +
                         b["b_male_Y"] * S[n,"male"] +
                         b["b_parent_educ_Y"] * S[n,"parent_educ"] +
                         b["b_A_Y"] * 1 +
                         b["b_L1_Y"] * S[n,"L1"] +
                         b["b_M_Y"] * S[n,"M"] +
                         b["b_AM_Y"] * 1 * S[n,"M"] * b["A.M.inter"] ) *
    ( ((( b["b_M"] +
          b["b_male_M"] * S[n,"male"] +
          b["b_parent_educ_M"] * S[n,"parent_educ"] +
          b["b_L1_M"] * S[n,"L1"] +
          b["b_A_M"] * 1 )^( S[n,"M"] )) +
       (( 1 - (b["b_M"] +
              b["b_male_M"] * S[n,"male"] +
              b["b_parent_educ_M"] * S[n,"parent_educ"] +
              b["b_L1_M"] * S[n,"L1"] +
              b["b_A_M"] * 1) )^( 1 - S[n,"M"] )) -
       (( b["b_M"] +
          b["b_male_M"] * S[n,"male"] +
          b["b_parent_educ_M"] * S[n,"parent_educ"] +
          b["b_L1_M"] * S[n,"L1"] +
          b["b_A_M"] * 0 )^( S[n,"M"] )) -
       (( 1 - (b["b_M"] +
              b["b_male_M"] * S[n,"male"] +
              b["b_parent_educ_M"] * S[n,"parent_educ"] +
              b["b_L1_M"] * S[n,"L1"] +
              b["b_A_M"] * 0) )^( 1 - S[n,"M"] )) ) *
    ((b["p_L1"])^(S[n,"L1"])) *
    ((1 - b["p_L1"])^(1 - S[n,"L1"])) *
    ((b["p_L0_male"])^(S[n,"male"])) *
    ((1 - b["p_L0_male"])^(1 - S[n,"male"])) *
    ((b["p_L0_parent_low_educ_lv"])^(S[n,"parent_educ"])) *
    ((1 - b["p_L0_parent_low_educ_lv"])^(1 - S[n,"parent_educ"]))
  }

PNDE.death <- sum(S[,"sum.pnde"])
TNIE.death <- sum(S[,"sum.tnie"])
```

```r
# quantitative outcome (QoL)
S <- cbind(expand.grid(c(0,1),c(0,1),c(0,1),c(0,1)), rep(NA,n=2^4), rep(NA,n=2^4))
colnames(S) <- list("male","parent_educ","L1","M","sum.pnde", "sum.tnie")
for (n in 1:16) {
  # PNDE
  S[n,"sum.pnde"] <- ( ( b["mu_Y"] +
                             b["c_male_Y"] * S[n,"male"] +
                             b["c_parent_educ_Y"] * S[n,"parent_educ"] +
                             b["c_A_Y"] * 1 +
                             b["c_L1_Y"] * S[n,"L1"] +
                             b["c_M_Y"] * S[n,"M"] +
                             b["c_AM_Y"] * 1 * S[n,"M"] * b["A.M.inter"] ) -
                         ( b["mu_Y"] +
                             b["c_male_Y"] * S[n,"male"] +
                             b["c_parent_educ_Y"] * S[n,"parent_educ"] +
                             b["c_A_Y"] * 0 +
                             b["c_L1_Y"] * S[n,"L1"] +
                             b["c_M_Y"] * S[n,"M"] +
                             b["c_AM_Y"] * 0 * S[n,"M"] * b["A.M.inter"] ) ) *
    (( b["b_M"] +
         b["b_male_M"] * S[n,"male"] +
         b["b_parent_educ_M"] * S[n,"parent_educ"] +
         b["b_L1_M"] * S[n,"L1"] +
         b["b_A_M"] * 0 )^( S[n,"M"] )) *
    (( 1 - (b["b_M"] +
            b["b_male_M"] * S[n,"male"] +
            b["b_parent_educ_M"] * S[n,"parent_educ"] +
            b["b_L1_M"] * S[n,"L1"] +
            b["b_A_M"] * 0) )^( 1 - S[n,"M"] ))  *
    ((b["p_L1"])^(S[n,"L1"])) *
    ((1 - b["p_L1"])^(1 - S[n,"L1"])) *
    ((b["p_L0_male"])^(S[n,"male"])) *
    ((1 - b["p_L0_male"])^(1 - S[n,"male"])) *
    ((b["p_L0_parent_low_educ_lv"])^(S[n,"parent_educ"])) *
    ((1 - b["p_L0_parent_low_educ_lv"])^(1 - S[n,"parent_educ"]))

  # TNIE
  S[n,"sum.tnie"] <- ( b["mu_Y"] +
                             b["c_male_Y"] * S[n,"male"] +
                             b["c_parent_educ_Y"] * S[n,"parent_educ"] +
                             b["c_A_Y"] * 1 +
                             b["c_L1_Y"] * S[n,"L1"] +
                             b["c_M_Y"] * S[n,"M"] +
                             b["c_AM_Y"] * 1 * S[n,"M"] * b["A.M.inter"] ) *
```

```r
    ( ((  b["b_M"] +
            b["b_male_M"] * S[n,"male"] +
            b["b_parent_educ_M"] * S[n,"parent_educ"] +
            b["b_L1_M"] * S[n,"L1"] +
            b["b_A_M"] * 1 )^( S[n,"M"] )) +
        (( 1 - (b["b_M"] +
                b["b_male_M"] * S[n,"male"] +
                b["b_parent_educ_M"] * S[n,"parent_educ"] +
                b["b_L1_M"] * S[n,"L1"] +
                b["b_A_M"] * 1) )^( 1 - S[n,"M"] )) -
        (( b["b_M"] +
            b["b_male_M"] * S[n,"male"] +
            b["b_parent_educ_M"] * S[n,"parent_educ"] +
            b["b_L1_M"] * S[n,"L1"] +
            b["b_A_M"] * 0 )^( S[n,"M"] )) -
        (( 1 - (b["b_M"] +
                b["b_male_M"] * S[n,"male"] +
                b["b_parent_educ_M"] * S[n,"parent_educ"] +
                b["b_L1_M"] * S[n,"L1"] +
                b["b_A_M"] * 0) )^( 1 - S[n,"M"] )) ) *
    ((b["p_L1"])^(S[n,"L1"])) *
    ((1 - b["p_L1"])^(1 - S[n,"L1"])) *
    ((b["p_L0_male"])^(S[n,"male"])) *
    ((1 - b["p_L0_male"])^(1 - S[n,"male"])) *
    ((b["p_L0_parent_low_educ_lv"])^(S[n,"parent_educ"])) *
    ((1 - b["p_L0_parent_low_educ_lv"])^(1 - S[n,"parent_educ"]))
  }

  PNDE.qol <- sum(S[,"sum.pnde"])
  TNIE.qol <- sum(S[,"sum.tnie"])

  return(list(PNDE.death = PNDE.death, TNIE.death = TNIE.death,
              PNDE.qol = PNDE.qol, TNIE.qol = TNIE.qol))
}
```

```r
true.PNDE.TNIE.no.inter <- true.PNDE.TNIE1(interaction = 0)

true.PNDE.TNIE.with.inter <- true.PNDE.TNIE1(interaction = 1)
```

The PNDE $= \mathbb{E}\left(Y_{1,M_0}\right) - \mathbb{E}\left(Y_{0,M_0}\right)$ and TNIE $= \mathbb{E}\left(Y_{1,M_1}\right) - \mathbb{E}\left(Y_{1,M_0}\right)$ are respectively:

- 0.05 and 0.00800000000000001 for death without interaction,
- 0.05855 and 0.011 for death with interaction,
- -4 and -0.9 for quality of life without interaction,

- -5.425 and -1.4 for quality of life with interaction.

10.1.4 Total natural direct effect and Pure natural indirect effect

The following function `true.TNDE.PNIE1` can be used to run the calculation for total natural direct effects (TNDE) and pure natural indirect effects (PNIE).

```r
true.TNDE.PNIE1 <- function(interaction = NULL) {
  b <- param.causal.model.1(A.M.interaction = interaction)

  # binary outcome (death)
  S <- cbind(expand.grid(c(0,1),c(0,1),c(0,1),c(0,1)), rep(NA,n=2^4), rep(NA,n=2^4))
  colnames(S) <- list("male","parent_educ","L1","M","sum.tnde", "sum.pnie")

  for (n in 1:16) {
    # TNDE
    S[n,"sum.tnde"] <- ( ( b["b_Y"] +
                              b["b_male_Y"] * S[n,"male"] +
                              b["b_parent_educ_Y"] * S[n,"parent_educ"] +
                              b["b_A_Y"] * 1 +
                              b["b_L1_Y"] * S[n,"L1"] +
                              b["b_M_Y"] * S[n,"M"] +
                              b["b_AM_Y"] * 1 * S[n,"M"] * b["A.M.inter"] ) -
                          ( b["b_Y"] +
                              b["b_male_Y"] * S[n,"male"] +
                              b["b_parent_educ_Y"] * S[n,"parent_educ"] +
                              b["b_A_Y"] * 0 +
                              b["b_L1_Y"] * S[n,"L1"] +
                              b["b_M_Y"] * S[n,"M"] +
                              b["b_AM_Y"] * 0 * S[n,"M"] * b["A.M.inter"] ) ) *
      (( b["b_M"] +
           b["b_male_M"] * S[n,"male"] +
           b["b_parent_educ_M"] * S[n,"parent_educ"] +
           b["b_L1_M"] * S[n,"L1"] +
           b["b_A_M"] * 1 )^( S[n,"M"] )) *
      (( 1 - (b["b_M"] +
                b["b_male_M"] * S[n,"male"] +
                b["b_parent_educ_M"] * S[n,"parent_educ"] +
                b["b_L1_M"] * S[n,"L1"] +
                b["b_A_M"] * 1) )^( 1 - S[n,"M"] ))  *
      ((b["p_L1"])^(S[n,"L1"])) *
      ((1 - b["p_L1"])^(1 - S[n,"L1"])) *
      ((b["p_L0_male"])^(S[n,"male"])) *
      ((1 - b["p_L0_male"])^(1 - S[n,"male"])) *
```

```r
      ((b["p_L0_parent_low_educ_lv"])^(S[n,"parent_educ"])) *
      ((1 - b["p_L0_parent_low_educ_lv"])^(1 - S[n,"parent_educ"]))

  # PNIE
  S[n,"sum.pnie"] <- ( b["b_Y"] +
                          b["b_male_Y"] * S[n,"male"] +
                          b["b_parent_educ_Y"] * S[n,"parent_educ"] +
                          b["b_A_Y"] * 0 +
                          b["b_L1_Y"] * S[n,"L1"] +
                          b["b_M_Y"] * S[n,"M"] +
                          b["b_AM_Y"] * 0 * S[n,"M"] * b["A.M.inter"] ) *
    ( ((  b["b_M"] +
            b["b_male_M"] * S[n,"male"] +
            b["b_parent_educ_M"] * S[n,"parent_educ"] +
            b["b_L1_M"] * S[n,"L1"] +
            b["b_A_M"] * 1 )^( S[n,"M"] )) +
        (( 1 - (b["b_M"] +
                b["b_male_M"] * S[n,"male"] +
                b["b_parent_educ_M"] * S[n,"parent_educ"] +
                b["b_L1_M"] * S[n,"L1"] +
                b["b_A_M"] * 1) )^( 1 - S[n,"M"] )) -
        (( b["b_M"] +
            b["b_male_M"] * S[n,"male"] +
            b["b_parent_educ_M"] * S[n,"parent_educ"] +
            b["b_L1_M"] * S[n,"L1"] +
            b["b_A_M"] * 0 )^( S[n,"M"] )) -
        (( 1 - (b["b_M"] +
                b["b_male_M"] * S[n,"male"] +
                b["b_parent_educ_M"] * S[n,"parent_educ"] +
                b["b_L1_M"] * S[n,"L1"] +
                b["b_A_M"] * 0) )^( 1 - S[n,"M"] )) ) *
    ((b["p_L1"])^(S[n,"L1"])) *
    ((1 - b["p_L1"])^(1 - S[n,"L1"])) *
    ((b["p_L0_male"])^(S[n,"male"])) *
    ((1 - b["p_L0_male"])^(1 - S[n,"male"])) *
    ((b["p_L0_parent_low_educ_lv"])^(S[n,"parent_educ"])) *
    ((1 - b["p_L0_parent_low_educ_lv"])^(1 - S[n,"parent_educ"]))
  }

TNDE.death <- sum(S[,"sum.tnde"])
PNIE.death <- sum(S[,"sum.pnie"])

# quantitative outcome (QoL)
S <- cbind(expand.grid(c(0,1),c(0,1),c(0,1),c(0,1)), rep(NA,n=2^4), rep(NA,n=2^4))
colnames(S) <- list("male","parent_educ","L1","M","sum.tnde", "sum.pnie")
```

```r
for (n in 1:16) {
  # TNDE
  S[n,"sum.tnde"] <- ( ( b["mu_Y"] +
                           b["c_male_Y"] * S[n,"male"] +
                           b["c_parent_educ_Y"] * S[n,"parent_educ"] +
                           b["c_A_Y"] * 1 +
                           b["c_L1_Y"] * S[n,"L1"] +
                           b["c_M_Y"] * S[n,"M"] +
                           b["c_AM_Y"] * 1 * S[n,"M"] * b["A.M.inter"] ) -
                         ( b["mu_Y"] +
                             b["c_male_Y"] * S[n,"male"] +
                             b["c_parent_educ_Y"] * S[n,"parent_educ"] +
                             b["c_A_Y"] * 0 +
                             b["c_L1_Y"] * S[n,"L1"] +
                             b["c_M_Y"] * S[n,"M"] +
                             b["c_AM_Y"] * 0 * S[n,"M"] * b["A.M.inter"] ) ) *
    (( b["b_M"] +
         b["b_male_M"] * S[n,"male"] +
         b["b_parent_educ_M"] * S[n,"parent_educ"] +
         b["b_L1_M"] * S[n,"L1"] +
         b["b_A_M"] * 1 )^( S[n,"M"] )) *
    (( 1 - (b["b_M"] +
              b["b_male_M"] * S[n,"male"] +
              b["b_parent_educ_M"] * S[n,"parent_educ"] +
              b["b_L1_M"] * S[n,"L1"] +
              b["b_A_M"] * 1) )^( 1 - S[n,"M"] ))  *
    ((b["p_L1"])^(S[n,"L1"])) *
    ((1 - b["p_L1"])^(1 - S[n,"L1"])) *
    ((b["p_L0_male"])^(S[n,"male"])) *
    ((1 - b["p_L0_male"])^(1 - S[n,"male"])) *
    ((b["p_L0_parent_low_educ_lv"])^(S[n,"parent_educ"])) *
    ((1 - b["p_L0_parent_low_educ_lv"])^(1 - S[n,"parent_educ"]))

  # PNIE
  S[n,"sum.pnie"] <- ( b["mu_Y"] +
                         b["c_male_Y"] * S[n,"male"] +
                         b["c_parent_educ_Y"] * S[n,"parent_educ"] +
                         b["c_A_Y"] * 0 +
                         b["c_L1_Y"] * S[n,"L1"] +
                         b["c_M_Y"] * S[n,"M"] +
                         b["c_AM_Y"] * 0 * S[n,"M"] * b["A.M.inter"] ) *
    ( (( b["b_M"] +
           b["b_male_M"] * S[n,"male"] +
           b["b_parent_educ_M"] * S[n,"parent_educ"] +
           b["b_L1_M"] * S[n,"L1"] +
```

```
                b["b_A_M"] * 1 )^( S[n,"M"] )) +
            (( 1 - (b["b_M"] +
                    b["b_male_M"] * S[n,"male"] +
                    b["b_parent_educ_M"] * S[n,"parent_educ"] +
                    b["b_L1_M"] * S[n,"L1"] +
                    b["b_A_M"] * 1) )^( 1 - S[n,"M"] )) -
            (( b["b_M"] +
                b["b_male_M"] * S[n,"male"] +
                b["b_parent_educ_M"] * S[n,"parent_educ"] +
                b["b_L1_M"] * S[n,"L1"] +
                b["b_A_M"] * 0 )^( S[n,"M"] )) -
            (( 1 - (b["b_M"] +
                    b["b_male_M"] * S[n,"male"] +
                    b["b_parent_educ_M"] * S[n,"parent_educ"] +
                    b["b_L1_M"] * S[n,"L1"] +
                    b["b_A_M"] * 0) )^( 1 - S[n,"M"] )) ) *
        ((b["p_L1"])^(S[n,"L1"])) *
        ((1 - b["p_L1"])^(1 - S[n,"L1"])) *
        ((b["p_L0_male"])^(S[n,"male"])) *
        ((1 - b["p_L0_male"])^(1 - S[n,"male"])) *
        ((b["p_L0_parent_low_educ_lv"])^(S[n,"parent_educ"])) *
        ((1 - b["p_L0_parent_low_educ_lv"])^(1 - S[n,"parent_educ"]))
    }

  TNDE.qol <- sum(S[,"sum.tnde"])
  PNIE.qol <- sum(S[,"sum.pnie"])

  return(list(TNDE.death = TNDE.death, PNIE.death = PNIE.death,
              TNDE.qol = TNDE.qol, PNIE.qol = PNIE.qol))
}
```

```
true.TNDE.PNIE.no.inter <- true.TNDE.PNIE1(interaction = 0)

true.TNDE.PNIE.with.inter <- true.TNDE.PNIE1(interaction = 1)
```

The TNDE $= \mathbb{E}\left(Y_{1,M_1}\right) - \mathbb{E}\left(Y_{0,M_1}\right)$ and PNIE $= \mathbb{E}\left(Y_{0,M_1}\right) - \mathbb{E}\left(Y_{0,M_0}\right)$ are respectively:

- 0.05 and 0.00800000000000001 for death without interaction,
- 0.06155 and 0.00800000000000001 for death with interaction,
- -4 and -0.899999999999999 for quality of life without interaction,
- -5.925 and -0.899999999999999 for quality of life with interaction.

10.1.5   Vanderweele's 3-way decomposition

The following function `true.3way.decomp` can be used to run the calculation
for the 3-way decomposition of the total effect into a "pure natural direct effect"
(PNDE), a "pure natural indirect effect" (PNIE) and a "mediated interactive
effect" (MIE).

```r
true.3way.decomp <- function(interaction = NULL) {
  b <- param.causal.model.1(A.M.interaction = interaction)

  # binary outcome (death)
  S <- cbind(expand.grid(c(0,1),c(0,1),c(0,1),c(0,1)), rep(NA,n=2^4), rep(NA,n=2^4))
  colnames(S) <- list("male","parent_educ","L1","M","sum.pde", "sum.pie")
  for (n in 1:16) {
    # PDE
    S[n,"sum.pde"] <- ( ( b["b_Y"] +
                              b["b_male_Y"] * S[n,"male"] +
                              b["b_parent_educ_Y"] * S[n,"parent_educ"] +
                              b["b_A_Y"] * 1 +
                              b["b_L1_Y"] * S[n,"L1"] +
                              b["b_M_Y"] * S[n,"M"] +
                              b["b_AM_Y"] * 1 * S[n,"M"] * b["A.M.inter"] ) -
                          ( b["b_Y"] +
                                b["b_male_Y"] * S[n,"male"] +
                                b["b_parent_educ_Y"] * S[n,"parent_educ"] +
                                b["b_A_Y"] * 0 +
                                b["b_L1_Y"] * S[n,"L1"] +
                                b["b_M_Y"] * S[n,"M"] +
                                b["b_AM_Y"] * 0 * S[n,"M"] * b["A.M.inter"] ) ) *
      (( b["b_M"] +
           b["b_male_M"] * S[n,"male"] +
           b["b_parent_educ_M"] * S[n,"parent_educ"] +
           b["b_L1_M"] * S[n,"L1"] +
           b["b_A_M"] * 0 )^( S[n,"M"] )) *
      (( 1 - (b["b_M"] +
              b["b_male_M"] * S[n,"male"] +
              b["b_parent_educ_M"] * S[n,"parent_educ"] +
              b["b_L1_M"] * S[n,"L1"] +
              b["b_A_M"] * 0) )^( 1 - S[n,"M"] ))  *
      ((b["p_L1"])^(S[n,"L1"])) *
      ((1 - b["p_L1"])^(1 - S[n,"L1"])) *
      ((b["p_L0_male"])^(S[n,"male"])) *
      ((1 - b["p_L0_male"])^(1 - S[n,"male"])) *
      ((b["p_L0_parent_low_educ_lv"])^(S[n,"parent_educ"])) *
      ((1 - b["p_L0_parent_low_educ_lv"])^(1 - S[n,"parent_educ"]))
```

```r
  # PIE
  S[n,"sum.pie"] <- ( b["b_Y"] +
                      b["b_male_Y"] * S[n,"male"] +
                      b["b_parent_educ_Y"] * S[n,"parent_educ"] +
                      b["b_A_Y"] * 0 +
                      b["b_L1_Y"] * S[n,"L1"] +
                      b["b_M_Y"] * S[n,"M"] +
                      b["b_AM_Y"] * 0 * S[n,"M"] * b["A.M.inter"] ) *
    ( ((( b["b_M"] +
          b["b_male_M"] * S[n,"male"] +
          b["b_parent_educ_M"] * S[n,"parent_educ"] +
          b["b_L1_M"] * S[n,"L1"] +
          b["b_A_M"] * 1 )^( S[n,"M"] )) +
        (( 1 - (b["b_M"] +
                b["b_male_M"] * S[n,"male"] +
                b["b_parent_educ_M"] * S[n,"parent_educ"] +
                b["b_L1_M"] * S[n,"L1"] +
                b["b_A_M"] * 1) )^( 1 - S[n,"M"] )) -
        (( b["b_M"] +
           b["b_male_M"] * S[n,"male"] +
           b["b_parent_educ_M"] * S[n,"parent_educ"] +
           b["b_L1_M"] * S[n,"L1"] +
           b["b_A_M"] * 0 )^( S[n,"M"] )) -
        (( 1 - (b["b_M"] +
                b["b_male_M"] * S[n,"male"] +
                b["b_parent_educ_M"] * S[n,"parent_educ"] +
                b["b_L1_M"] * S[n,"L1"] +
                b["b_A_M"] * 0) )^( 1 - S[n,"M"] )) ) *
    ((b["p_L1"])^(S[n,"L1"])) *
    ((1 - b["p_L1"])^(1 - S[n,"L1"])) *
    ((b["p_L0_male"])^(S[n,"male"])) *
    ((1 - b["p_L0_male"])^(1 - S[n,"male"])) *
    ((b["p_L0_parent_low_educ_lv"])^(S[n,"parent_educ"])) *
    ((1 - b["p_L0_parent_low_educ_lv"])^(1 - S[n,"parent_educ"]))
  }

# MI
S.MI <- cbind(expand.grid(c(0,1),c(0,1),c(0,1)), rep(NA,n=2^4))
colnames(S.MI) <- list("male","parent_educ","L1", "sum.mi")
for (n in 1:8) {
  S.MI[n,"sum.mi"] <- ( ( b["b_Y"] +
                          b["b_male_Y"] * S.MI[n,"male"] +
                          b["b_parent_educ_Y"] * S.MI[n,"parent_educ"] +
                          b["b_A_Y"] * 1 +
                          b["b_L1_Y"] * S.MI[n,"L1"] +
```

```r
                          b["b_M_Y"] * 1 +
                          b["b_AM_Y"] * 1 * 1 * b["A.M.inter"] ) -
                        ( b["b_Y"] +
                          b["b_male_Y"] * S.MI[n,"male"] +
                          b["b_parent_educ_Y"] * S.MI[n,"parent_educ"] +
                          b["b_A_Y"] * 1 +
                          b["b_L1_Y"] * S.MI[n,"L1"] +
                          b["b_M_Y"] * 0 +
                          b["b_AM_Y"] * 1 * 0 * b["A.M.inter"] ) -
                        ( b["b_Y"] +
                          b["b_male_Y"] * S.MI[n,"male"] +
                          b["b_parent_educ_Y"] * S.MI[n,"parent_educ"] +
                          b["b_A_Y"] * 0 +
                          b["b_L1_Y"] * S.MI[n,"L1"] +
                          b["b_M_Y"] * 1 +
                          b["b_AM_Y"] * 0 * 1 * b["A.M.inter"] ) +
                        ( b["b_Y"] +
                          b["b_male_Y"] * S.MI[n,"male"] +
                          b["b_parent_educ_Y"] * S.MI[n,"parent_educ"] +
                          b["b_A_Y"] * 0 +
                          b["b_L1_Y"] * S.MI[n,"L1"] +
                          b["b_M_Y"] * 0 +
                          b["b_AM_Y"] * 0 * 0 * b["A.M.inter"] )) *
      ( ( b["b_M"] +
            b["b_male_M"] * S.MI[n,"male"] +
            b["b_parent_educ_M"] * S.MI[n,"parent_educ"] +
            b["b_L1_M"] * S[n,"L1"] +
            b["b_A_M"] * 1 ) -
          ( b["b_M"] +
              b["b_male_M"] * S.MI[n,"male"] +
              b["b_parent_educ_M"] * S.MI[n,"parent_educ"] +
              b["b_L1_M"] * S[n,"L1"] +
              b["b_A_M"] * 0 )) *
      ((b["p_L1"])^(S[n,"L1"])) *
      ((1 - b["p_L1"])^(1 - S[n,"L1"])) *
      ((b["p_L0_male"])^(S.MI[n,"male"])) *
      ((1 - b["p_L0_male"])^(1 - S.MI[n,"male"])) *
      ((b["p_L0_parent_low_educ_lv"])^(S.MI[n,"parent_educ"])) *
      ((1 - b["p_L0_parent_low_educ_lv"])^(1 - S.MI[n,"parent_educ"]))
    }

  PDE.death <- sum(S[,"sum.pde"])
  PIE.death <- sum(S[,"sum.pie"])
  MI.death <- sum(S.MI[,"sum.mi"])
```

```r
# quantitative outcome (QoL)
S <- cbind(expand.grid(c(0,1),c(0,1),c(0,1),c(0,1)), rep(NA,n=2^4), rep(NA,n=2^4))
colnames(S) <- list("male","parent_educ","L1","M","sum.pde", "sum.pie")
for (n in 1:16) {
  # PDE
  S[n,"sum.pde"] <- ( ( b["mu_Y"] +
                          b["c_male_Y"] * S[n,"male"] +
                          b["c_parent_educ_Y"] * S[n,"parent_educ"] +
                          b["c_A_Y"] * 1 +
                          b["c_L1_Y"] * S[n,"L1"] +
                          b["c_M_Y"] * S[n,"M"] +
                          b["c_AM_Y"] * 1 * S[n,"M"] * b["A.M.inter"] ) -
                        ( b["mu_Y"] +
                            b["c_male_Y"] * S[n,"male"] +
                            b["c_parent_educ_Y"] * S[n,"parent_educ"] +
                            b["c_A_Y"] * 0 +
                            b["c_L1_Y"] * S[n,"L1"] +
                            b["c_M_Y"] * S[n,"M"] +
                            b["c_AM_Y"] * 0 * S[n,"M"] * b["A.M.inter"] ) ) *
    (( b["b_M"] +
         b["b_male_M"] * S[n,"male"] +
         b["b_parent_educ_M"] * S[n,"parent_educ"] +
         b["b_L1_M"] * S[n,"L1"] +
         b["b_A_M"] * 0 )^( S[n,"M"] )) *
    (( 1 - (b["b_M"] +
              b["b_male_M"] * S[n,"male"] +
              b["b_parent_educ_M"] * S[n,"parent_educ"] +
              b["b_L1_M"] * S[n,"L1"] +
              b["b_A_M"] * 0) )^( 1 - S[n,"M"] ))  *
    ((b["p_L1"])^(S[n,"L1"])) *
    ((1 - b["p_L1"])^(1 - S[n,"L1"])) *
    ((b["p_L0_male"])^(S[n,"male"])) *
    ((1 - b["p_L0_male"])^(1 - S[n,"male"])) *
    ((b["p_L0_parent_low_educ_lv"])^(S[n,"parent_educ"])) *
    ((1 - b["p_L0_parent_low_educ_lv"])^(1 - S[n,"parent_educ"]))

  # PIE
  S[n,"sum.pie"] <- ( b["mu_Y"] +
                        b["c_male_Y"] * S[n,"male"] +
                        b["c_parent_educ_Y"] * S[n,"parent_educ"] +
                        b["c_A_Y"] * 0 +
                        b["c_L1_Y"] * S[n,"L1"] +
                        b["c_M_Y"] * S[n,"M"] +
                        b["c_AM_Y"] * 0 * S[n,"M"] * b["A.M.inter"] ) *
    ( ((  b["b_M"] +
```

```
              b["b_male_M"] * S[n,"male"] +
              b["b_parent_educ_M"] * S[n,"parent_educ"] +
              b["b_L1_M"] * S[n,"L1"] +
              b["b_A_M"] * 1 )^( S[n,"M"] )) +
        (( 1 - (b["b_M"] +
                b["b_male_M"] * S[n,"male"] +
                b["b_parent_educ_M"] * S[n,"parent_educ"] +
                b["b_L1_M"] * S[n,"L1"] +
                b["b_A_M"] * 1) )^( 1 - S[n,"M"] )) -
        (( b["b_M"] +
             b["b_male_M"] * S[n,"male"] +
             b["b_parent_educ_M"] * S[n,"parent_educ"] +
             b["b_L1_M"] * S[n,"L1"] +
             b["b_A_M"] * 0 )^( S[n,"M"] )) -
        (( 1 - (b["b_M"] +
                b["b_male_M"] * S[n,"male"] +
                b["b_parent_educ_M"] * S[n,"parent_educ"] +
                b["b_L1_M"] * S[n,"L1"] +
                b["b_A_M"] * 0) )^( 1 - S[n,"M"] )) ) *
    ((b["p_L1"])^(S[n,"L1"])) *
    ((1 - b["p_L1"])^(1 - S[n,"L1"])) *
    ((b["p_L0_male"])^(S[n,"male"])) *
    ((1 - b["p_L0_male"])^(1 - S[n,"male"])) *
    ((b["p_L0_parent_low_educ_lv"])^(S[n,"parent_educ"])) *
    ((1 - b["p_L0_parent_low_educ_lv"])^(1 - S[n,"parent_educ"]))
  }

# MI
S.MI <- cbind(expand.grid(c(0,1),c(0,1),c(0,1)), rep(NA,n=2^4))
colnames(S.MI) <- list("male","parent_educ","L1","sum.mi")
for (n in 1:8) {
  S.MI[n,"sum.mi"] <- ( ( b["mu_Y"] +
                          b["c_male_Y"] * S.MI[n,"male"] +
                          b["c_parent_educ_Y"] * S.MI[n,"parent_educ"] +
                          b["c_A_Y"] * 1 +
                          b["c_L1_Y"] * S.MI[n,"L1"] +
                          b["c_M_Y"] * 1 +
                          b["c_AM_Y"] * 1 * 1 * b["A.M.inter"] ) -
                        ( b["mu_Y"] +
                            b["c_male_Y"] * S.MI[n,"male"] +
                            b["c_parent_educ_Y"] * S.MI[n,"parent_educ"] +
                            b["c_A_Y"] * 1 +
                            b["c_L1_Y"] * S.MI[n,"L1"] +
                            b["c_M_Y"] * 0 +
                            b["c_AM_Y"] * 1 * 0 * b["A.M.inter"] ) -
```

```
                            ( b["mu_Y"] +
                                b["c_male_Y"] * S.MI[n,"male"] +
                                b["c_parent_educ_Y"] * S.MI[n,"parent_educ"] +
                                b["c_A_Y"] * 0 +
                                b["c_L1_Y"] * S.MI[n,"L1"] +
                                b["c_M_Y"] * 1 +
                                b["c_AM_Y"] * 0 * 1 * b["A.M.inter"] ) +
                            ( b["mu_Y"] +
                                b["c_male_Y"] * S.MI[n,"male"] +
                                b["c_parent_educ_Y"] * S.MI[n,"parent_educ"] +
                                b["c_A_Y"] * 0 +
                                b["c_L1_Y"] * S.MI[n,"L1"] +
                                b["c_M_Y"] * 0 +
                                b["c_AM_Y"] * 0 * 0 * b["A.M.inter"] )) *
        ( ( b["b_M"] +
                b["b_male_M"] * S.MI[n,"male"] +
                b["b_parent_educ_M"] * S.MI[n,"parent_educ"] +
                b["b_L1_M"] * S[n,"L1"] +
                b["b_A_M"] * 1 ) -
            ( b["b_M"] +
                b["b_male_M"] * S.MI[n,"male"] +
                b["b_parent_educ_M"] * S.MI[n,"parent_educ"] +
                b["b_L1_M"] * S[n,"L1"] +
                b["b_A_M"] * 0 )) *
        ((b["p_L1"])^(S[n,"L1"])) *
        ((1 - b["p_L1"])^(1 - S[n,"L1"])) *
        ((b["p_L0_male"])^(S.MI[n,"male"])) *
        ((1 - b["p_L0_male"])^(1 - S.MI[n,"male"])) *
        ((b["p_L0_parent_low_educ_lv"])^(S.MI[n,"parent_educ"])) *
        ((1 - b["p_L0_parent_low_educ_lv"])^(1 - S.MI[n,"parent_educ"]))
    }

  PDE.qol <- sum(S[,"sum.pde"])
  PIE.qol <- sum(S[,"sum.pie"])
  MI.qol <- sum(S.MI[,"sum.mi"])

  return(list(PDE.death = PDE.death, PIE.death = PIE.death, MI.death = MI.death,
            PDE.qol = PDE.qol, PIE.qol = PIE.qol, MI.qol = MI.qol))
}
```

```
true.3way.no.inter <- true.3way.decomp(interaction = 0)

true.3way.with.inter <- true.3way.decomp(interaction = 1)
```

The PNDE $= \mathbb{E}\left(Y_{1,M_0}\right) - \mathbb{E}\left(Y_{0,M_0}\right)$, the PNIE $= \mathbb{E}\left(Y_{0,M_1}\right) - \mathbb{E}\left(Y_{0,M_0}\right)$ and

the MIE $= \mathbb{E}\left((Y_{1,1} - Y_{1,0} - Y_{0,1} + Y_{0,0}) \times (M_1 - M_0)\right)$ are respectively:

- 0.05, 0.00800000000000001 and 0.000 for death without interaction,
- 0.05855, 0.00800000000000001 and 0.003 for death with interaction,
- -4, -0.899999999999999 and 0 for quality of life without interaction,
- -5.425, -0.899999999999999 and -0.5 for quality of life with interaction.

10.1.6 Vanderweele's 4-way decomposition

The following function `true.4way.decomp` can be used to run the calculation for the 4-way decomposition of the total effect into a "controlled direct effect" (CDE), a "reference interaction effect" (RIE), a "mediated interaction effect" (MIE) and a "pure natural indirect effect" (PNIE).

```
true.4way.decomp <- function(interaction = NULL) {
  b <- param.causal.model.1(A.M.interaction = interaction)

  # binary outcome (death)
  S <- cbind(expand.grid(c(0,1),c(0,1),c(0,1)), rep(NA,n=2^3), rep(NA,n=2^3),
              rep(NA,n=2^3), rep(NA,n=2^3))
  colnames(S) <- list("male","parent_educ", "L1", "sum.cde", "sum.intref",
                        "sum.intmed", "sum.pie")
  for (n in 1:8) {
    # CDE
    S[n,"sum.cde"] <- ( ( b["b_Y"] +
                                b["b_male_Y"] * S[n,"male"] +
                                b["b_parent_educ_Y"] * S[n,"parent_educ"] +
                                b["b_A_Y"] * 1 +
                                b["b_L1_Y"] * S[n,"L1"] +
                                b["b_M_Y"] * 0 +
                                b["b_AM_Y"] * 1 * 0 * b["A.M.inter"] ) -
                            ( b["b_Y"] +
                                b["b_male_Y"] * S[n,"male"] +
                                b["b_parent_educ_Y"] * S[n,"parent_educ"] +
                                b["b_A_Y"] * 0 +
                                b["b_L1_Y"] * S[n,"L1"] +
                                b["b_M_Y"] * 0 +
                                b["b_AM_Y"] * 0 * 0 * b["A.M.inter"] ) ) *
      ((b["p_L1"])^(S[n,"L1"])) *
      ((1 - b["p_L1"])^(1 - S[n,"L1"])) *
      ((b["p_L0_male"])^(S[n,"male"])) *
      ((1 - b["p_L0_male"])^(1 - S[n,"male"])) *
      ((b["p_L0_parent_low_educ_lv"])^(S[n,"parent_educ"])) *
      ((1 - b["p_L0_parent_low_educ_lv"])^(1 - S[n,"parent_educ"]))
```

```r
# INTref
S[n,"sum.intref"] <- ( ( b["b_Y"] +
                          b["b_male_Y"] * S[n,"male"] +
                          b["b_parent_educ_Y"] * S[n,"parent_educ"] +
                          b["b_A_Y"] * 1 +
                          b["b_L1_Y"] * S[n,"L1"] +
                          b["b_M_Y"] * 1 +
                          b["b_AM_Y"] * 1 * 1 * b["A.M.inter"] ) -
                        ( b["b_Y"] +
                            b["b_male_Y"] * S[n,"male"] +
                            b["b_parent_educ_Y"] * S[n,"parent_educ"] +
                            b["b_A_Y"] * 1 +
                            b["b_L1_Y"] * S[n,"L1"] +
                            b["b_M_Y"] * 0 +
                            b["b_AM_Y"] * 1 * 0 * b["A.M.inter"] ) -
                        ( b["b_Y"] +
                            b["b_male_Y"] * S[n,"male"] +
                            b["b_parent_educ_Y"] * S[n,"parent_educ"] +
                            b["b_A_Y"] * 0 +
                            b["b_L1_Y"] * S[n,"L1"] +
                            b["b_M_Y"] * 1 +
                            b["b_AM_Y"] * 0 * 1 * b["A.M.inter"] ) +
                        ( b["b_Y"] +
                            b["b_male_Y"] * S[n,"male"] +
                            b["b_parent_educ_Y"] * S[n,"parent_educ"] +
                            b["b_A_Y"] * 0 +
                            b["b_L1_Y"] * S[n,"L1"] +
                            b["b_M_Y"] * 0 +
                            b["b_AM_Y"] * 0 * 0 * b["A.M.inter"] )) *
  ( b["b_M"] +
      b["b_male_M"] * S[n,"male"] +
      b["b_parent_educ_M"] * S[n,"parent_educ"] +
      b["b_L1_M"] * S[n,"L1"] +
      b["b_A_M"] * 0 ) *
  ((b["p_L1"])^(S[n,"L1"])) *
  ((1 - b["p_L1"])^(1 - S[n,"L1"])) *
  ((b["p_L0_male"])^(S[n,"male"])) *
  ((1 - b["p_L0_male"])^(1 - S[n,"male"])) *
  ((b["p_L0_parent_low_educ_lv"])^(S[n,"parent_educ"])) *
  ((1 - b["p_L0_parent_low_educ_lv"])^(1 - S[n,"parent_educ"]))

# INTmed
S[n,"sum.intmed"] <- ( ( b["b_Y"] +
                          b["b_male_Y"] * S[n,"male"] +
                          b["b_parent_educ_Y"] * S[n,"parent_educ"] +
```

```
                                    b["b_A_Y"] * 1 +
                                    b["b_L1_Y"] * S[n,"L1"] +
                                    b["b_M_Y"] * 1 +
                                    b["b_AM_Y"] * 1 * 1 * b["A.M.inter"] ) -
                                 ( b["b_Y"] +
                                      b["b_male_Y"] * S[n,"male"] +
                                      b["b_parent_educ_Y"] * S[n,"parent_educ"] +
                                      b["b_A_Y"] * 1 +
                                      b["b_L1_Y"] * S[n,"L1"] +
                                      b["b_M_Y"] * 0 +
                                      b["b_AM_Y"] * 1 * 0 * b["A.M.inter"] ) -
                                 ( b["b_Y"] +
                                      b["b_male_Y"] * S[n,"male"] +
                                      b["b_parent_educ_Y"] * S[n,"parent_educ"] +
                                      b["b_A_Y"] * 0 +
                                      b["b_L1_Y"] * S[n,"L1"] +
                                      b["b_M_Y"] * 1 +
                                      b["b_AM_Y"] * 0 * 1 * b["A.M.inter"] ) +
                                 ( b["b_Y"] +
                                      b["b_male_Y"] * S[n,"male"] +
                                      b["b_parent_educ_Y"] * S[n,"parent_educ"] +
                                      b["b_A_Y"] * 0 +
                                      b["b_L1_Y"] * S[n,"L1"] +
                                      b["b_M_Y"] * 0 +
                                      b["b_AM_Y"] * 0 * 0 * b["A.M.inter"] )) *
     ( ( b["b_M"] +
           b["b_male_M"] * S[n,"male"] +
           b["b_parent_educ_M"] * S[n,"parent_educ"] +
           b["b_L1_M"] * S[n,"L1"] +
           b["b_A_M"] * 1 ) -
       ( b["b_M"] +
            b["b_male_M"] * S[n,"male"] +
            b["b_parent_educ_M"] * S[n,"parent_educ"] +
            b["b_L1_M"] * S[n,"L1"] +
            b["b_A_M"] * 0 )) *
     ((b["p_L1"])^(S[n,"L1"])) *
     ((1 - b["p_L1"])^(1 - S[n,"L1"])) *
     ((b["p_L0_male"])^(S[n,"male"])) *
     ((1 - b["p_L0_male"])^(1 - S[n,"male"])) *
     ((b["p_L0_parent_low_educ_lv"])^(S[n,"parent_educ"])) *
     ((1 - b["p_L0_parent_low_educ_lv"])^(1 - S[n,"parent_educ"]))

     # PIE
     S[n,"sum.pie"] <- ( ( b["b_Y"] +
                            b["b_male_Y"] * S[n,"male"] +
```

```
                            b["b_parent_educ_Y"] * S[n,"parent_educ"] +
                            b["b_A_Y"] * 0 +
                            b["b_L1_Y"] * S[n,"L1"] +
                            b["b_M_Y"] * 1 +
                            b["b_AM_Y"] * 0 * 1 * b["A.M.inter"] ) -
                          ( b["b_Y"] +
                              b["b_male_Y"] * S[n,"male"] +
                              b["b_parent_educ_Y"] * S[n,"parent_educ"] +
                              b["b_A_Y"] * 0 +
                              b["b_L1_Y"] * S[n,"L1"] +
                              b["b_M_Y"] * 0 +
                              b["b_AM_Y"] * 0 * 0 * b["A.M.inter"] ) ) *
      ( ( b["b_M"] +
            b["b_male_M"] * S[n,"male"] +
            b["b_parent_educ_M"] * S[n,"parent_educ"] +
            b["b_L1_M"] * S[n,"L1"] +
            b["b_A_M"] * 1 ) -
          ( b["b_M"] +
              b["b_male_M"] * S[n,"male"] +
              b["b_parent_educ_M"] * S[n,"parent_educ"] +
              b["b_L1_M"] * S[n,"L1"] +
              b["b_A_M"] * 0 )) *
      ((b["p_L1"])^(S[n,"L1"])) *
      ((1 - b["p_L1"])^(1 - S[n,"L1"])) *
      ((b["p_L0_male"])^(S[n,"male"])) *
      ((1 - b["p_L0_male"])^(1 - S[n,"male"])) *
      ((b["p_L0_parent_low_educ_lv"])^(S[n,"parent_educ"])) *
      ((1 - b["p_L0_parent_low_educ_lv"])^(1 - S[n,"parent_educ"]))
    }

CDE.death <- sum(S[,"sum.cde"])
INTref.death <- sum(S[,"sum.intref"])
INTmed.death <- sum(S[,"sum.intmed"])
PIE.death <- sum(S[,"sum.pie"])

# quantitative outcome (QoL)
S <- cbind(expand.grid(c(0,1),c(0,1),c(0,1)), rep(NA,n=2^3), rep(NA,n=2^3),
           rep(NA,n=2^3), rep(NA,n=2^3))
colnames(S) <- list("male","parent_educ", "L1", "sum.cde", "sum.intref",
                    "sum.intmed", "sum.pie")
for (n in 1:8) {
  # CDE
  S[n,"sum.cde"] <- ( ( b["mu_Y"] +
                            b["c_male_Y"] * S[n,"male"] +
                            b["c_parent_educ_Y"] * S[n,"parent_educ"] +
```

```
                            b["c_A_Y"] * 1 +
                            b["c_L1_Y"] * S[n,"L1"] +
                            b["c_M_Y"] * 0 +
                            b["c_AM_Y"] * 1 * 0 * b["A.M.inter"] ) -
                          ( b["mu_Y"] +
                              b["c_male_Y"] * S[n,"male"] +
                              b["c_parent_educ_Y"] * S[n,"parent_educ"] +
                              b["c_A_Y"] * 0 +
                              b["c_L1_Y"] * S[n,"L1"] +
                              b["c_M_Y"] * 0 +
                              b["c_AM_Y"] * 0 * 0 * b["A.M.inter"] ) ) *
  ((b["p_L1"])^(S[n,"L1"])) *
  ((1 - b["p_L1"])^(1 - S[n,"L1"])) *
  ((b["p_L0_male"])^(S[n,"male"])) *
  ((1 - b["p_L0_male"])^(1 - S[n,"male"])) *
  ((b["p_L0_parent_low_educ_lv"])^(S[n,"parent_educ"])) *
  ((1 - b["p_L0_parent_low_educ_lv"])^(1 - S[n,"parent_educ"]))

  # INTref
  S[n,"sum.intref"] <- ( ( b["mu_Y"] +
                              b["c_male_Y"] * S[n,"male"] +
                              b["c_parent_educ_Y"] * S[n,"parent_educ"] +
                              b["c_A_Y"] * 1 +
                              b["c_L1_Y"] * S[n,"L1"] +
                              b["c_M_Y"] * 1 +
                              b["c_AM_Y"] * 1 * 1 * b["A.M.inter"] ) -
                          ( b["mu_Y"] +
                              b["c_male_Y"] * S[n,"male"] +
                              b["c_parent_educ_Y"] * S[n,"parent_educ"] +
                              b["c_A_Y"] * 1 +
                              b["c_L1_Y"] * S[n,"L1"] +
                              b["c_M_Y"] * 0 +
                              b["c_AM_Y"] * 1 * 0 * b["A.M.inter"] ) -
                          ( b["mu_Y"] +
                              b["c_male_Y"] * S[n,"male"] +
                              b["c_parent_educ_Y"] * S[n,"parent_educ"] +
                              b["c_A_Y"] * 0 +
                              b["c_L1_Y"] * S[n,"L1"] +
                              b["c_M_Y"] * 1 +
                              b["c_AM_Y"] * 0 * 1 * b["A.M.inter"] ) +
                          ( b["mu_Y"] +
                              b["c_male_Y"] * S[n,"male"] +
                              b["c_parent_educ_Y"] * S[n,"parent_educ"] +
                              b["c_A_Y"] * 0 +
                              b["c_L1_Y"] * S[n,"L1"] +
```

```
                                b["c_M_Y"] * 0 +
                                b["c_AM_Y"] * 0 * 0 * b["A.M.inter"] )) *
    ( b["b_M"] +
        b["b_male_M"] * S[n,"male"] +
        b["b_parent_educ_M"] * S[n,"parent_educ"] +
        b["b_L1_M"] * S[n,"L1"] +
        b["b_A_M"] * 0 ) *
    ((b["p_L1"])^(S[n,"L1"])) *
    ((1 - b["p_L1"])^(1 - S[n,"L1"])) *
    ((b["p_L0_male"])^(S[n,"male"])) *
    ((1 - b["p_L0_male"])^(1 - S[n,"male"])) *
    ((b["p_L0_parent_low_educ_lv"])^(S[n,"parent_educ"])) *
    ((1 - b["p_L0_parent_low_educ_lv"])^(1 - S[n,"parent_educ"]))

# INTmed
S[n,"sum.intmed"] <- ( ( b["mu_Y"] +
                            b["c_male_Y"] * S[n,"male"] +
                            b["c_parent_educ_Y"] * S[n,"parent_educ"] +
                            b["c_A_Y"] * 1 +
                            b["c_L1_Y"] * S[n,"L1"] +
                            b["c_M_Y"] * 1 +
                            b["c_AM_Y"] * 1 * 1 * b["A.M.inter"] ) -
                        ( b["mu_Y"] +
                            b["c_male_Y"] * S[n,"male"] +
                            b["c_parent_educ_Y"] * S[n,"parent_educ"] +
                            b["c_A_Y"] * 1 +
                            b["c_L1_Y"] * S[n,"L1"] +
                            b["c_M_Y"] * 0 +
                            b["c_AM_Y"] * 1 * 0 * b["A.M.inter"] ) -
                        ( b["mu_Y"] +
                            b["c_male_Y"] * S[n,"male"] +
                            b["c_parent_educ_Y"] * S[n,"parent_educ"] +
                            b["c_A_Y"] * 0 +
                            b["c_L1_Y"] * S[n,"L1"] +
                            b["c_M_Y"] * 1 +
                            b["c_AM_Y"] * 0 * 1 * b["A.M.inter"] ) +
                        ( b["mu_Y"] +
                            b["c_male_Y"] * S[n,"male"] +
                            b["c_parent_educ_Y"] * S[n,"parent_educ"] +
                            b["c_A_Y"] * 0 +
                            b["c_L1_Y"] * S[n,"L1"] +
                            b["c_M_Y"] * 0 +
                            b["c_AM_Y"] * 0 * 0 * b["A.M.inter"] )) *
    ( ( b["b_M"] +
        b["b_male_M"] * S[n,"male"] +
```

```r
          b["b_parent_educ_M"] * S[n,"parent_educ"] +
          b["b_L1_M"] * S[n,"L1"] +
          b["b_A_M"] * 1 ) -
        ( b["b_M"] +
            b["b_male_M"] * S[n,"male"] +
            b["b_parent_educ_M"] * S[n,"parent_educ"] +
            b["b_L1_M"] * S[n,"L1"] +
            b["b_A_M"] * 0 )) *
    ((b["p_L1"])^(S[n,"L1"])) *
    ((1 - b["p_L1"])^(1 - S[n,"L1"])) *
    ((b["p_L0_male"])^(S[n,"male"])) *
    ((1 - b["p_L0_male"])^(1 - S[n,"male"])) *
    ((b["p_L0_parent_low_educ_lv"])^(S[n,"parent_educ"])) *
    ((1 - b["p_L0_parent_low_educ_lv"])^(1 - S[n,"parent_educ"]))

  # PIE
  S[n,"sum.pie"] <- ( ( b["mu_Y"] +
                          b["c_male_Y"] * S[n,"male"] +
                          b["c_parent_educ_Y"] * S[n,"parent_educ"] +
                          b["c_A_Y"] * 0 +
                          b["c_L1_Y"] * S[n,"L1"] +
                          b["c_M_Y"] * 1 +
                          b["c_AM_Y"] * 0 * 1 * b["A.M.inter"] ) -
                        ( b["mu_Y"] +
                            b["c_male_Y"] * S[n,"male"] +
                            b["c_parent_educ_Y"] * S[n,"parent_educ"] +
                            b["c_A_Y"] * 0 +
                            b["c_L1_Y"] * S[n,"L1"] +
                            b["c_M_Y"] * 0 +
                            b["c_AM_Y"] * 0 * 0 * b["A.M.inter"] ) ) *
    ( ( b["b_M"] +
          b["b_male_M"] * S[n,"male"] +
          b["b_parent_educ_M"] * S[n,"parent_educ"] +
          b["b_L1_M"] * S[n,"L1"] +
          b["b_A_M"] * 1 ) -
        ( b["b_M"] +
            b["b_male_M"] * S[n,"male"] +
            b["b_parent_educ_M"] * S[n,"parent_educ"] +
            b["b_L1_M"] * S[n,"L1"] +
            b["b_A_M"] * 0 )) *
    ((b["p_L1"])^(S[n,"L1"])) *
    ((1 - b["p_L1"])^(1 - S[n,"L1"])) *
    ((b["p_L0_male"])^(S[n,"male"])) *
    ((1 - b["p_L0_male"])^(1 - S[n,"male"])) *
    ((b["p_L0_parent_low_educ_lv"])^(S[n,"parent_educ"])) *
```

```
        ((1 - b["p_L0_parent_low_educ_lv"])^(1 - S[n,"parent_educ"]))
    }

  CDE.qol <- sum(S[,"sum.cde"])
  INTref.qol <- sum(S[,"sum.intref"])
  INTmed.qol <- sum(S[,"sum.intmed"])
  PIE.qol <- sum(S[,"sum.pie"])

  return(list(CDE.death = CDE.death, INTref.death = INTref.death,
              INTmed.death = INTmed.death, PIE.death = PIE.death,
              CDE.qol = CDE.qol, INTref.qol = INTref.qol,
              INTmed.qol = INTmed.qol, PIE.qol = PIE.qol))
}
```

```
true.4way.no.inter <- true.4way.decomp(interaction = 0)

true.4way.with.inter <- true.4way.decomp(interaction = 1)
```

The CDE $= \mathbb{E}(Y_{1,0}) - \mathbb{E}(Y_{0,0})$, RIE $= ((Y_{1,1} - Y_{1,0} - Y_{0,1} + Y_{0,0}) \times M_0)$, MIE $= \mathbb{E}((Y_{1,1} - Y_{1,0} - Y_{0,1} + Y_{0,0}) \times (M_1 - M_0))$ and PNIE $= \mathbb{E}(Y_{0,M_1}) - \mathbb{E}(Y_{0,M_0})$ are respectively:

- 0.05, 0.000, 0.000 and 0.008 for death without interaction,
- 0.05, 0.00855, 0.003 and 0.008 for death with interaction,
- -4, 0, 0 and -0.9 for quality of life without interaction,
- -4, -1.425, -0.5 and -0.9 for quality of life with interaction.

10.1.7 Marginal randomized direct and indirect effects

The following function `true.marg.random` can be used to run the calculation for the marginal randomized natural direct (marginal MRDE) and indirect effects (marginal MRIE).

```
true.marg.random <- function(interaction = NULL) {
  b <- param.causal.model.1(A.M.interaction = interaction)

  # marginal distribution of M
  M.S <- cbind(expand.grid(c(0,1),c(0,1),c(0,1),c(0,1),c(0,1)), rep(NA,n=2^5))
  colnames(M.S) <- list("male","parent_educ","L1","M","A","sum")

  for (n in 1:32) {
    M.S[n,"sum"] <- (( b["b_M"] +
                        b["b_male_M"] * M.S[n,"male"] +
```

```
                       b["b_parent_educ_M"] * M.S[n,"parent_educ"] +
                       b["b_L1_M"] * M.S[n,"L1"] +
                       b["b_A_M"] * M.S[n,"A"])^( M.S[n,"M"] )) *
    (( 1 - (b["b_M"] +
            b["b_male_M"] * M.S[n,"male"] +
            b["b_parent_educ_M"] * M.S[n,"parent_educ"] +
            b["b_L1_M"] * M.S[n,"L1"] +
            b["b_A_M"] * M.S[n,"A"]) )^( 1 - M.S[n,"M"] ))
  }

 M0.A0.L0_00.L1_0 <- M.S[M.S[,"M"]==0 & M.S[,"A"]==0 & M.S[,"male"]==0 &
                       M.S[,"parent_educ"]==0 & M.S[,"L1"]==0,"sum"]
 M0.A0.L0_01.L1_0 <- M.S[M.S[,"M"]==0 & M.S[,"A"]==0 & M.S[,"male"]==0 &
                       M.S[,"parent_educ"]==1 & M.S[,"L1"]==0,"sum"]
 M0.A0.L0_10.L1_0 <- M.S[M.S[,"M"]==0 & M.S[,"A"]==0 & M.S[,"male"]==1 &
                       M.S[,"parent_educ"]==0 & M.S[,"L1"]==0,"sum"]
 M0.A0.L0_11.L1_0 <- M.S[M.S[,"M"]==0 & M.S[,"A"]==0 & M.S[,"male"]==1 &
                       M.S[,"parent_educ"]==1 & M.S[,"L1"]==0,"sum"]
 M0.A0.L0_00.L1_1 <- M.S[M.S[,"M"]==0 & M.S[,"A"]==0 & M.S[,"male"]==0 &
                       M.S[,"parent_educ"]==0 & M.S[,"L1"]==1,"sum"]
 M0.A0.L0_01.L1_1 <- M.S[M.S[,"M"]==0 & M.S[,"A"]==0 & M.S[,"male"]==0 &
                       M.S[,"parent_educ"]==1 & M.S[,"L1"]==1,"sum"]
 M0.A0.L0_10.L1_1 <- M.S[M.S[,"M"]==0 & M.S[,"A"]==0 & M.S[,"male"]==1 &
                       M.S[,"parent_educ"]==0 & M.S[,"L1"]==1,"sum"]
 M0.A0.L0_11.L1_1 <- M.S[M.S[,"M"]==0 & M.S[,"A"]==0 & M.S[,"male"]==1 &
                       M.S[,"parent_educ"]==1 & M.S[,"L1"]==1,"sum"]


 M1.A0.L0_00.L1_0 <- M.S[M.S[,"M"]==1 & M.S[,"A"]==0 & M.S[,"male"]==0 &
                       M.S[,"parent_educ"]==0 & M.S[,"L1"]==0,"sum"]
 M1.A0.L0_01.L1_0 <- M.S[M.S[,"M"]==1 & M.S[,"A"]==0 & M.S[,"male"]==0 &
                       M.S[,"parent_educ"]==1 & M.S[,"L1"]==0,"sum"]
 M1.A0.L0_10.L1_0 <- M.S[M.S[,"M"]==1 & M.S[,"A"]==0 & M.S[,"male"]==1 &
                       M.S[,"parent_educ"]==0 & M.S[,"L1"]==0,"sum"]
 M1.A0.L0_11.L1_0 <- M.S[M.S[,"M"]==1 & M.S[,"A"]==0 & M.S[,"male"]==1 &
                       M.S[,"parent_educ"]==1 & M.S[,"L1"]==0,"sum"]
 M1.A0.L0_00.L1_1 <- M.S[M.S[,"M"]==1 & M.S[,"A"]==0 & M.S[,"male"]==0 &
                       M.S[,"parent_educ"]==0 & M.S[,"L1"]==1,"sum"]
 M1.A0.L0_01.L1_1 <- M.S[M.S[,"M"]==1 & M.S[,"A"]==0 & M.S[,"male"]==0 &
                       M.S[,"parent_educ"]==1 & M.S[,"L1"]==1,"sum"]
 M1.A0.L0_10.L1_1 <- M.S[M.S[,"M"]==1 & M.S[,"A"]==0 & M.S[,"male"]==1 &
                       M.S[,"parent_educ"]==0 & M.S[,"L1"]==1,"sum"]
 M1.A0.L0_11.L1_1 <- M.S[M.S[,"M"]==1 & M.S[,"A"]==0 & M.S[,"male"]==1 &
                       M.S[,"parent_educ"]==1 & M.S[,"L1"]==1,"sum"]


 M0.A1.L0_00.L1_0 <- M.S[M.S[,"M"]==0 & M.S[,"A"]==1 & M.S[,"male"]==0 &
```

```r
                                      M.S[,"parent_educ"]==0 & M.S[,"L1"]==0,"sum"]
MO.A1.LO_01.L1_0 <- M.S[M.S[,"M"]==0 & M.S[,"A"]==1 & M.S[,"male"]==0 &
                                      M.S[,"parent_educ"]==1 & M.S[,"L1"]==0,"sum"]
MO.A1.LO_10.L1_0 <- M.S[M.S[,"M"]==0 & M.S[,"A"]==1 & M.S[,"male"]==1 &
                                      M.S[,"parent_educ"]==0 & M.S[,"L1"]==0,"sum"]
MO.A1.LO_11.L1_0 <- M.S[M.S[,"M"]==0 & M.S[,"A"]==1 & M.S[,"male"]==1 &
                                      M.S[,"parent_educ"]==1 & M.S[,"L1"]==0,"sum"]
MO.A1.LO_00.L1_1 <- M.S[M.S[,"M"]==0 & M.S[,"A"]==1 & M.S[,"male"]==0 &
                                      M.S[,"parent_educ"]==0 & M.S[,"L1"]==1,"sum"]
MO.A1.LO_01.L1_1 <- M.S[M.S[,"M"]==0 & M.S[,"A"]==1 & M.S[,"male"]==0 &
                                      M.S[,"parent_educ"]==1 & M.S[,"L1"]==1,"sum"]
MO.A1.LO_10.L1_1 <- M.S[M.S[,"M"]==0 & M.S[,"A"]==1 & M.S[,"male"]==1 &
                                      M.S[,"parent_educ"]==0 & M.S[,"L1"]==1,"sum"]
MO.A1.LO_11.L1_1 <- M.S[M.S[,"M"]==0 & M.S[,"A"]==1 & M.S[,"male"]==1 &
                                      M.S[,"parent_educ"]==1 & M.S[,"L1"]==1,"sum"]


M1.A1.LO_00.L1_0 <- M.S[M.S[,"M"]==1 & M.S[,"A"]==1 & M.S[,"male"]==0 &
                                      M.S[,"parent_educ"]==0 & M.S[,"L1"]==0,"sum"]
M1.A1.LO_01.L1_0 <- M.S[M.S[,"M"]==1 & M.S[,"A"]==1 & M.S[,"male"]==0 &
                                      M.S[,"parent_educ"]==1 & M.S[,"L1"]==0,"sum"]
M1.A1.LO_10.L1_0 <- M.S[M.S[,"M"]==1 & M.S[,"A"]==1 & M.S[,"male"]==1 &
                                      M.S[,"parent_educ"]==0 & M.S[,"L1"]==0,"sum"]
M1.A1.LO_11.L1_0 <- M.S[M.S[,"M"]==1 & M.S[,"A"]==1 & M.S[,"male"]==1 &
                                      M.S[,"parent_educ"]==1 & M.S[,"L1"]==0,"sum"]
M1.A1.LO_00.L1_1 <- M.S[M.S[,"M"]==1 & M.S[,"A"]==1 & M.S[,"male"]==0 &
                                      M.S[,"parent_educ"]==0 & M.S[,"L1"]==1,"sum"]
M1.A1.LO_01.L1_1 <- M.S[M.S[,"M"]==1 & M.S[,"A"]==1 & M.S[,"male"]==0 &
                                      M.S[,"parent_educ"]==1 & M.S[,"L1"]==1,"sum"]
M1.A1.LO_10.L1_1 <- M.S[M.S[,"M"]==1 & M.S[,"A"]==1 & M.S[,"male"]==1 &
                                      M.S[,"parent_educ"]==0 & M.S[,"L1"]==1,"sum"]
M1.A1.LO_11.L1_1 <- M.S[M.S[,"M"]==1 & M.S[,"A"]==1 & M.S[,"male"]==1 &
                                      M.S[,"parent_educ"]==1 & M.S[,"L1"]==1,"sum"]


# binary outcome (death)
S <- cbind(expand.grid(c(0,1),c(0,1),c(0,1),c(0,1)), rep(NA,n=2^4), rep(NA,n=2^4),
           rep(NA,n=2^4))
colnames(S) <- list("male","parent_educ","L1","M","sum.psi11", "sum.psi10", "sum.psi00")
for (n in 1:16) {
  S[n,"sum.psi11"] <-  ( b["b_Y"] +                                        # A=1
                         b["b_male_Y"] * S[n,"male"] +
                         b["b_parent_educ_Y"] * S[n,"parent_educ"] +
                         b["b_A_Y"] * 1 +
                         b["b_L1_Y"] * S[n,"L1"] +
                         b["b_M_Y"] * S[n,"M"] +
                         b["b_AM_Y"] * 1 * S[n,"M"] * b["A.M.inter"] ) *
```

```
      ((M1.A1.L0_00.L1_0*(S[n,"male"]==0)*(S[n,"parent_educ"]==0)*(S[n,"L1"]==0) +   # A
        M1.A1.L0_01.L1_0*(S[n,"male"]==0)*(S[n,"parent_educ"]==1)*(S[n,"L1"]==0) +
        M1.A1.L0_10.L1_0*(S[n,"male"]==1)*(S[n,"parent_educ"]==0)*(S[n,"L1"]==0) +
        M1.A1.L0_11.L1_0*(S[n,"male"]==1)*(S[n,"parent_educ"]==1)*(S[n,"L1"]==0) +
        M1.A1.L0_00.L1_1*(S[n,"male"]==0)*(S[n,"parent_educ"]==0)*(S[n,"L1"]==1) +
        M1.A1.L0_01.L1_1*(S[n,"male"]==0)*(S[n,"parent_educ"]==1)*(S[n,"L1"]==1) +
        M1.A1.L0_10.L1_1*(S[n,"male"]==1)*(S[n,"parent_educ"]==0)*(S[n,"L1"]==1) +
        M1.A1.L0_11.L1_1*(S[n,"male"]==1)*(S[n,"parent_educ"]==1)*
        (S[n,"L1"]==1) )^( S[n,"M"] )) *
      ((M0.A1.L0_00.L1_0*(S[n,"male"]==0)*(S[n,"parent_educ"]==0)*(S[n,"L1"]==0) +
        M0.A1.L0_01.L1_0*(S[n,"male"]==0)*(S[n,"parent_educ"]==1)*(S[n,"L1"]==0) +
        M0.A1.L0_10.L1_0*(S[n,"male"]==1)*(S[n,"parent_educ"]==0)*(S[n,"L1"]==0) +
        M0.A1.L0_11.L1_0*(S[n,"male"]==1)*(S[n,"parent_educ"]==1)*(S[n,"L1"]==0) +
        M0.A1.L0_00.L1_1*(S[n,"male"]==0)*(S[n,"parent_educ"]==0)*(S[n,"L1"]==1) +
        M0.A1.L0_01.L1_1*(S[n,"male"]==0)*(S[n,"parent_educ"]==1)*(S[n,"L1"]==1) +
        M0.A1.L0_10.L1_1*(S[n,"male"]==1)*(S[n,"parent_educ"]==0)*(S[n,"L1"]==1) +
        M0.A1.L0_11.L1_1*(S[n,"male"]==1)*(S[n,"parent_educ"]==1)*
        (S[n,"L1"]==1) )^( 1 - S[n,"M"] )) *
      ((b["p_L1"])^(M.S[n,"L1"])) *
      ((1 - b["p_L1"])^(1 - M.S[n,"L1"])) *
      ((b["p_L0_male"])^(S[n,"male"])) *
      ((1 - b["p_L0_male"])^(1 - S[n,"male"])) *
      ((b["p_L0_parent_low_educ_lv"])^(S[n,"parent_educ"])) *
      ((1 - b["p_L0_parent_low_educ_lv"])^(1 - S[n,"parent_educ"]))

  S[n,"sum.psi10"] <-  ( b["b_Y"] +                                             # A=1
                         b["b_male_Y"] * S[n,"male"] +
                         b["b_parent_educ_Y"] * S[n,"parent_educ"] +
                         b["b_A_Y"] * 1 +
                         b["b_L1_Y"] * S[n,"L1"] +
                         b["b_M_Y"] * S[n,"M"] +
                         b["b_AM_Y"] * 1 * S[n,"M"] * b["A.M.inter"] ) *
      ((M1.A0.L0_00.L1_0*(S[n,"male"]==0)*(S[n,"parent_educ"]==0)*(S[n,"L1"]==0) + # A
        M1.A0.L0_01.L1_0*(S[n,"male"]==0)*(S[n,"parent_educ"]==1)*(S[n,"L1"]==0) +
        M1.A0.L0_10.L1_0*(S[n,"male"]==1)*(S[n,"parent_educ"]==0)*(S[n,"L1"]==0) +
        M1.A0.L0_11.L1_0*(S[n,"male"]==1)*(S[n,"parent_educ"]==1)*(S[n,"L1"]==0) +
        M1.A0.L0_00.L1_1*(S[n,"male"]==0)*(S[n,"parent_educ"]==0)*(S[n,"L1"]==1) +
        M1.A0.L0_01.L1_1*(S[n,"male"]==0)*(S[n,"parent_educ"]==1)*(S[n,"L1"]==1) +
        M1.A0.L0_10.L1_1*(S[n,"male"]==1)*(S[n,"parent_educ"]==0)*(S[n,"L1"]==1) +
        M1.A0.L0_11.L1_1*(S[n,"male"]==1)*(S[n,"parent_educ"]==1)*
        (S[n,"L1"]==1) )^( S[n,"M"] )) *
      ((M0.A0.L0_00.L1_0*(S[n,"male"]==0)*(S[n,"parent_educ"]==0)*(S[n,"L1"]==0) +
        M0.A0.L0_01.L1_0*(S[n,"male"]==0)*(S[n,"parent_educ"]==1)*(S[n,"L1"]==0) +
        M0.A0.L0_10.L1_0*(S[n,"male"]==1)*(S[n,"parent_educ"]==0)*(S[n,"L1"]==0) +
        M0.A0.L0_11.L1_0*(S[n,"male"]==1)*(S[n,"parent_educ"]==1)*(S[n,"L1"]==0) +
```

```r
        M0.A0.L0_00.L1_1*(S[n,"male"]==0)*(S[n,"parent_educ"]==0)*(S[n,"L1"]==1) +
        M0.A0.L0_01.L1_1*(S[n,"male"]==0)*(S[n,"parent_educ"]==1)*(S[n,"L1"]==1) +
        M0.A0.L0_10.L1_1*(S[n,"male"]==1)*(S[n,"parent_educ"]==0)*(S[n,"L1"]==1) +
        M0.A0.L0_11.L1_1*(S[n,"male"]==1)*(S[n,"parent_educ"]==1)*
        (S[n,"L1"]==1) )^( 1 - S[n,"M"] )) *
    ((b["p_L1"])^(M.S[n,"L1"])) *
    ((1 - b["p_L1"])^(1 - M.S[n,"L1"])) *
    ((b["p_L0_male"])^(S[n,"male"])) *
    ((1 - b["p_L0_male"])^(1 - S[n,"male"])) *
    ((b["p_L0_parent_low_educ_lv"])^(S[n,"parent_educ"])) *
    ((1 - b["p_L0_parent_low_educ_lv"])^(1 - S[n,"parent_educ"]))

  S[n,"sum.psi00"] <-  ( b["b_Y"] +                                       # A=0
                        b["b_male_Y"] * S[n,"male"] +
                        b["b_parent_educ_Y"] * S[n,"parent_educ"] +
                        b["b_A_Y"] * 0 +
                        b["b_L1_Y"] * S[n,"L1"] +
                        b["b_M_Y"] * S[n,"M"] +
                        b["b_AM_Y"] * 0 * S[n,"M"] * b["A.M.inter"] ) *
    ((M1.A0.L0_00.L1_0*(S[n,"male"]==0)*(S[n,"parent_educ"]==0)*(S[n,"L1"]==0) + # A'=0
        M1.A0.L0_01.L1_0*(S[n,"male"]==0)*(S[n,"parent_educ"]==1)*(S[n,"L1"]==0) +
        M1.A0.L0_10.L1_0*(S[n,"male"]==1)*(S[n,"parent_educ"]==0)*(S[n,"L1"]==0) +
        M1.A0.L0_11.L1_0*(S[n,"male"]==1)*(S[n,"parent_educ"]==1)*(S[n,"L1"]==0) +
        M1.A0.L0_00.L1_1*(S[n,"male"]==0)*(S[n,"parent_educ"]==0)*(S[n,"L1"]==1) +
        M1.A0.L0_01.L1_1*(S[n,"male"]==0)*(S[n,"parent_educ"]==1)*(S[n,"L1"]==1) +
        M1.A0.L0_10.L1_1*(S[n,"male"]==1)*(S[n,"parent_educ"]==0)*(S[n,"L1"]==1) +
        M1.A0.L0_11.L1_1*(S[n,"male"]==1)*(S[n,"parent_educ"]==1)*
        (S[n,"L1"]==1) )^( S[n,"M"] )) *
    ((M0.A0.L0_00.L1_0*(S[n,"male"]==0)*(S[n,"parent_educ"]==0)*(S[n,"L1"]==0) +
        M0.A0.L0_01.L1_0*(S[n,"male"]==0)*(S[n,"parent_educ"]==1)*(S[n,"L1"]==0) +
        M0.A0.L0_10.L1_0*(S[n,"male"]==1)*(S[n,"parent_educ"]==0)*(S[n,"L1"]==0) +
        M0.A0.L0_11.L1_0*(S[n,"male"]==1)*(S[n,"parent_educ"]==1)*(S[n,"L1"]==0) +
        M0.A0.L0_00.L1_1*(S[n,"male"]==0)*(S[n,"parent_educ"]==0)*(S[n,"L1"]==1) +
        M0.A0.L0_01.L1_1*(S[n,"male"]==0)*(S[n,"parent_educ"]==1)*(S[n,"L1"]==1) +
        M0.A0.L0_10.L1_1*(S[n,"male"]==1)*(S[n,"parent_educ"]==0)*(S[n,"L1"]==1) +
        M0.A0.L0_11.L1_1*(S[n,"male"]==1)*(S[n,"parent_educ"]==1)*
        (S[n,"L1"]==1) )^( 1 - S[n,"M"] )) *
    ((b["p_L1"])^(M.S[n,"L1"])) *
    ((1 - b["p_L1"])^(1 - M.S[n,"L1"])) *
    ((b["p_L0_male"])^(S[n,"male"])) *
    ((1 - b["p_L0_male"])^(1 - S[n,"male"])) *
    ((b["p_L0_parent_low_educ_lv"])^(S[n,"parent_educ"])) *
    ((1 - b["p_L0_parent_low_educ_lv"])^(1 - S[n,"parent_educ"]))
  }
```

```r
mrNDE.death <- sum(S[,"sum.psi10"]) - sum(S[,"sum.psi00"])
mrNIE.death <- sum(S[,"sum.psi11"]) - sum(S[,"sum.psi10"])

# quantitative outcome (QoL)
S <- cbind(expand.grid(c(0,1),c(0,1),c(0,1),c(0,1)), rep(NA,n=2^4),
           rep(NA,n=2^4), rep(NA,n=2^4))
colnames(S) <- list("male","parent_educ","L1","M","sum.psi11", "sum.psi10",
                    "sum.psi00")
for (n in 1:16) {
  S[n,"sum.psi11"] <-  ( b["mu_Y"] +                                        # A=1
                         b["c_male_Y"] * S[n,"male"] +
                         b["c_parent_educ_Y"] * S[n,"parent_educ"] +
                         b["c_A_Y"] * 1 +
                         b["c_L1_Y"] * S[n,"L1"] +
                         b["c_M_Y"] * S[n,"M"] +
                         b["c_AM_Y"] * 1 * S[n,"M"] * b["A.M.inter"] ) *
    ((M1.A1.L0_00.L1_0*(S[n,"male"]==0)*(S[n,"parent_educ"]==0)*(S[n,"L1"]==0) + # A
        M1.A1.L0_01.L1_0*(S[n,"male"]==0)*(S[n,"parent_educ"]==1)*(S[n,"L1"]==0) +
        M1.A1.L0_10.L1_0*(S[n,"male"]==1)*(S[n,"parent_educ"]==0)*(S[n,"L1"]==0) +
        M1.A1.L0_11.L1_0*(S[n,"male"]==1)*(S[n,"parent_educ"]==1)*(S[n,"L1"]==0) +
        M1.A1.L0_00.L1_1*(S[n,"male"]==0)*(S[n,"parent_educ"]==0)*(S[n,"L1"]==1) +
        M1.A1.L0_01.L1_1*(S[n,"male"]==0)*(S[n,"parent_educ"]==1)*(S[n,"L1"]==1) +
        M1.A1.L0_10.L1_1*(S[n,"male"]==1)*(S[n,"parent_educ"]==0)*(S[n,"L1"]==1) +
        M1.A1.L0_11.L1_1*(S[n,"male"]==1)*(S[n,"parent_educ"]==1)*
        (S[n,"L1"]==1) )^( S[n,"M"] )) *
    ((M0.A1.L0_00.L1_0*(S[n,"male"]==0)*(S[n,"parent_educ"]==0)*(S[n,"L1"]==0) +
        M0.A1.L0_01.L1_0*(S[n,"male"]==0)*(S[n,"parent_educ"]==1)*(S[n,"L1"]==0) +
        M0.A1.L0_10.L1_0*(S[n,"male"]==1)*(S[n,"parent_educ"]==0)*(S[n,"L1"]==0) +
        M0.A1.L0_11.L1_0*(S[n,"male"]==1)*(S[n,"parent_educ"]==1)*(S[n,"L1"]==0) +
        M0.A1.L0_00.L1_1*(S[n,"male"]==0)*(S[n,"parent_educ"]==0)*(S[n,"L1"]==1) +
        M0.A1.L0_01.L1_1*(S[n,"male"]==0)*(S[n,"parent_educ"]==1)*(S[n,"L1"]==1) +
        M0.A1.L0_10.L1_1*(S[n,"male"]==1)*(S[n,"parent_educ"]==0)*(S[n,"L1"]==1) +
        M0.A1.L0_11.L1_1*(S[n,"male"]==1)*(S[n,"parent_educ"]==1)*
        (S[n,"L1"]==1) )^( 1 - S[n,"M"] )) *
    ((b["p_L1"])^(M.S[n,"L1"])) *
    ((1 - b["p_L1"])^(1 - M.S[n,"L1"])) *
    ((b["p_L0_male"])^(S[n,"male"])) *
    ((1 - b["p_L0_male"])^(1 - S[n,"male"])) *
    ((b["p_L0_parent_low_educ_lv"])^(S[n,"parent_educ"])) *
    ((1 - b["p_L0_parent_low_educ_lv"])^(1 - S[n,"parent_educ"]))

  S[n,"sum.psi10"] <-  ( b["mu_Y"] +                                        # A=1
                         b["c_male_Y"] * S[n,"male"] +
                         b["c_parent_educ_Y"] * S[n,"parent_educ"] +
                         b["c_A_Y"] * 1 +
```

```
                            b["c_L1_Y"] * S[n,"L1"] +
                            b["c_M_Y"] * S[n,"M"] +
                            b["c_AM_Y"] * 1 * S[n,"M"] * b["A.M.inter"] ) *
     ((M1.A0.L0_00.L1_0*(S[n,"male"]==0)*(S[n,"parent_educ"]==0)*(S[n,"L1"]==0) + # A'=0
        M1.A0.L0_01.L1_0*(S[n,"male"]==0)*(S[n,"parent_educ"]==1)*(S[n,"L1"]==0) +
        M1.A0.L0_10.L1_0*(S[n,"male"]==1)*(S[n,"parent_educ"]==0)*(S[n,"L1"]==0) +
        M1.A0.L0_11.L1_0*(S[n,"male"]==1)*(S[n,"parent_educ"]==1)*(S[n,"L1"]==0) +
        M1.A0.L0_00.L1_1*(S[n,"male"]==0)*(S[n,"parent_educ"]==0)*(S[n,"L1"]==1) +
        M1.A0.L0_01.L1_1*(S[n,"male"]==0)*(S[n,"parent_educ"]==1)*(S[n,"L1"]==1) +
        M1.A0.L0_10.L1_1*(S[n,"male"]==1)*(S[n,"parent_educ"]==0)*(S[n,"L1"]==1) +
        M1.A0.L0_11.L1_1*(S[n,"male"]==1)*(S[n,"parent_educ"]==1)*
        (S[n,"L1"]==1) )^( S[n,"M"] )) *
     ((M0.A0.L0_00.L1_0*(S[n,"male"]==0)*(S[n,"parent_educ"]==0)*(S[n,"L1"]==0) +
        M0.A0.L0_01.L1_0*(S[n,"male"]==0)*(S[n,"parent_educ"]==1)*(S[n,"L1"]==0) +
        M0.A0.L0_10.L1_0*(S[n,"male"]==1)*(S[n,"parent_educ"]==0)*(S[n,"L1"]==0) +
        M0.A0.L0_11.L1_0*(S[n,"male"]==1)*(S[n,"parent_educ"]==1)*(S[n,"L1"]==0) +
        M0.A0.L0_00.L1_1*(S[n,"male"]==0)*(S[n,"parent_educ"]==0)*(S[n,"L1"]==1) +
        M0.A0.L0_01.L1_1*(S[n,"male"]==0)*(S[n,"parent_educ"]==1)*(S[n,"L1"]==1) +
        M0.A0.L0_10.L1_1*(S[n,"male"]==1)*(S[n,"parent_educ"]==0)*(S[n,"L1"]==1) +
        M0.A0.L0_11.L1_1*(S[n,"male"]==1)*(S[n,"parent_educ"]==1)*
        (S[n,"L1"]==1) )^( 1 - S[n,"M"] )) *
     ((b["p_L1"])^(M.S[n,"L1"])) *
     ((1 - b["p_L1"])^(1 - M.S[n,"L1"])) *
     ((b["p_L0_male"])^(S[n,"male"])) *
     ((1 - b["p_L0_male"])^(1 - S[n,"male"])) *
     ((b["p_L0_parent_low_educ_lv"])^(S[n,"parent_educ"])) *
     ((1 - b["p_L0_parent_low_educ_lv"])^(1 - S[n,"parent_educ"]))

  S[n,"sum.psi00"] <- ( b["mu_Y"] +                                  # A=0
                            b["c_male_Y"] * S[n,"male"] +
                            b["c_parent_educ_Y"] * S[n,"parent_educ"] +
                            b["c_A_Y"] * 0 +
                            b["c_L1_Y"] * S[n,"L1"] +
                            b["c_M_Y"] * S[n,"M"] +
                            b["c_AM_Y"] * 0 * S[n,"M"] * b["A.M.inter"] ) *
     ((M1.A0.L0_00.L1_0*(S[n,"male"]==0)*(S[n,"parent_educ"]==0)*(S[n,"L1"]==0) + # A'=0
        M1.A0.L0_01.L1_0*(S[n,"male"]==0)*(S[n,"parent_educ"]==1)*(S[n,"L1"]==0) +
        M1.A0.L0_10.L1_0*(S[n,"male"]==1)*(S[n,"parent_educ"]==0)*(S[n,"L1"]==0) +
        M1.A0.L0_11.L1_0*(S[n,"male"]==1)*(S[n,"parent_educ"]==1)*(S[n,"L1"]==0) +
        M1.A0.L0_00.L1_1*(S[n,"male"]==0)*(S[n,"parent_educ"]==0)*(S[n,"L1"]==1) +
        M1.A0.L0_01.L1_1*(S[n,"male"]==0)*(S[n,"parent_educ"]==1)*(S[n,"L1"]==1) +
        M1.A0.L0_10.L1_1*(S[n,"male"]==1)*(S[n,"parent_educ"]==0)*(S[n,"L1"]==1) +
        M1.A0.L0_11.L1_1*(S[n,"male"]==1)*(S[n,"parent_educ"]==1)*
        (S[n,"L1"]==1) )^( S[n,"M"] )) *
     ((M0.A0.L0_00.L1_0*(S[n,"male"]==0)*(S[n,"parent_educ"]==0)*(S[n,"L1"]==0) +
```

```
          MO.AO.LO_01.L1_0*(S[n,"male"]==0)*(S[n,"parent_educ"]==1)*(S[n,"L1"]==0) +
          MO.AO.LO_10.L1_0*(S[n,"male"]==1)*(S[n,"parent_educ"]==0)*(S[n,"L1"]==0) +
          MO.AO.LO_11.L1_0*(S[n,"male"]==1)*(S[n,"parent_educ"]==1)*(S[n,"L1"]==0) +
          MO.AO.LO_00.L1_1*(S[n,"male"]==0)*(S[n,"parent_educ"]==0)*(S[n,"L1"]==1) +
          MO.AO.LO_01.L1_1*(S[n,"male"]==0)*(S[n,"parent_educ"]==1)*(S[n,"L1"]==1) +
          MO.AO.LO_10.L1_1*(S[n,"male"]==1)*(S[n,"parent_educ"]==0)*(S[n,"L1"]==1) +
          MO.AO.LO_11.L1_1*(S[n,"male"]==1)*(S[n,"parent_educ"]==1)*
          (S[n,"L1"]==1) )^( 1 - S[n,"M"] )) *
    ((b["p_L1"])^(M.S[n,"L1"])) *
    ((1 - b["p_L1"])^(1 - M.S[n,"L1"])) *
    ((b["p_L0_male"])^(S[n,"male"])) *
    ((1 - b["p_L0_male"])^(1 - S[n,"male"])) *
    ((b["p_L0_parent_low_educ_lv"])^(S[n,"parent_educ"])) *
    ((1 - b["p_L0_parent_low_educ_lv"])^(1 - S[n,"parent_educ"]))
  }

  mrNDE.qol <- sum(S[,"sum.psi10"]) - sum(S[,"sum.psi00"])
  mrNIE.qol <- sum(S[,"sum.psi11"]) - sum(S[,"sum.psi10"])

  return(list(mrNDE.death = mrNDE.death, mrNIE.death = mrNIE.death,
              mrNDE.qol = mrNDE.qol, mrNIE.qol = mrNIE.qol))
}
```

```
true.marg.random.no.inter <- true.marg.random(interaction = 0)

true.marg.random.with.inter <- true.marg.random(interaction = 1)
```

The marginal randomized direct effect $\text{MRDE} = \mathbb{E}\left(Y_{1,G_{0|L(0)}}\right) - \mathbb{E}\left(Y_{0,G_{0|L(0)}}\right)$ and the marginal randomized indirect effect $\text{MRIE} = \mathbb{E}\left(Y_{1,G_{1|L(0)}}\right) - \mathbb{E}\left(Y_{1,G_{0|L(0)}}\right)$ are respectively:

- 0.05 and 0.00800000000000003 for death without interaction,
- 0.05855 and 0.011 for death with interaction,
- -3.99999999999999 and -0.900000000000006 for quality of life without interaction,
- -5.42499999999999 and -1.40000000000001 for quality of life with interaction.

10.1.8 Conditional randomized direct and indirect effects

The following function `true.cond.random` can be used to run the calculation for the conditional randomized natural direct (CRDE) and indirect effects (CRIE).

```r
true.cond.random <- function(interaction = NULL) {
  b <- param.causal.model.1(A.M.interaction = interaction)

  # binary outcome (death)
  S <- cbind(expand.grid(c(0,1),c(0,1),c(0,1),c(0,1)), rep(NA,n=2^4),
             rep(NA,n=2^4), rep(NA,n=2^4))
  colnames(S) <- list("male","parent_educ","L1","M","sum.psi11", "sum.psi10",
                      "sum.psi00")
  for (n in 1:16) {
    S[n,"sum.psi11"] <-  ( b["b_Y"] +                                      # A=1
                             b["b_male_Y"] * S[n,"male"] +
                             b["b_parent_educ_Y"] * S[n,"parent_educ"] +
                             b["b_A_Y"] * 1 +
                             b["b_L1_Y"] * S[n,"L1"] +
                             b["b_M_Y"] * S[n,"M"] +
                             b["b_AM_Y"] * 1 * S[n,"M"] * b["A.M.inter"] ) *
      (( b["b_M"] +                                                        # A'=1
           b["b_male_M"] * S[n,"male"] +
           b["b_parent_educ_M"] * S[n,"parent_educ"] +
           b["b_L1_M"] * S[n,"L1"] +
           b["b_A_M"] * 1 )^( S[n,"M"] )) *
      (( 1 - (b["b_M"] +
                b["b_male_M"] * S[n,"male"] +
                b["b_parent_educ_M"] * S[n,"parent_educ"] +
                b["b_L1_M"] * S[n,"L1"] +
                b["b_A_M"] * 1 ) )^( 1 - S[n,"M"] )) *
      ((b["p_L1"])^(S[n,"L1"])) *
      ((1 - b["p_L1"])^(1 - S[n,"L1"])) *
      ((b["p_L0_male"])^(S[n,"male"])) *
      ((1 - b["p_L0_male"])^(1 - S[n,"male"])) *
      ((b["p_L0_parent_low_educ_lv"])^(S[n,"parent_educ"])) *
      ((1 - b["p_L0_parent_low_educ_lv"])^(1 - S[n,"parent_educ"]))

    S[n,"sum.psi10"] <-  ( b["b_Y"] +                                      # A=1
                             b["b_male_Y"] * S[n,"male"] +
                             b["b_parent_educ_Y"] * S[n,"parent_educ"] +
                             b["b_A_Y"] * 1 +
                             b["b_L1_Y"] * S[n,"L1"] +
                             b["b_M_Y"] * S[n,"M"] +
                             b["b_AM_Y"] * 1 * S[n,"M"] * b["A.M.inter"] ) *
      (( b["b_M"] +                                                        # A'=0
           b["b_male_M"] * S[n,"male"] +
           b["b_parent_educ_M"] * S[n,"parent_educ"] +
           b["b_L1_M"] * S[n,"L1"] +
           b["b_A_M"] * 0 )^( S[n,"M"] )) *
```

```r
    (( 1 - (b["b_M"] +
            b["b_male_M"] * S[n,"male"] +
            b["b_parent_educ_M"] * S[n,"parent_educ"] +
            b["b_L1_M"] * S[n,"L1"] +
            b["b_A_M"] * 0 ) )^( 1 - S[n,"M"] )) *
   ((b["p_L1"])^(S[n,"L1"])) *
   ((1 - b["p_L1"])^(1 - S[n,"L1"])) *
   ((b["p_L0_male"])^(S[n,"male"])) *
   ((1 - b["p_L0_male"])^(1 - S[n,"male"])) *
   ((b["p_L0_parent_low_educ_lv"])^(S[n,"parent_educ"])) *
   ((1 - b["p_L0_parent_low_educ_lv"])^(1 - S[n,"parent_educ"]))

  S[n,"sum.psi00"] <-  ( b["b_Y"] +                                           # A=0
                         b["b_male_Y"] * S[n,"male"] +
                         b["b_parent_educ_Y"] * S[n,"parent_educ"] +
                         b["b_A_Y"] * 0 +
                         b["b_L1_Y"] * S[n,"L1"] +
                         b["b_M_Y"] * S[n,"M"] +
                         b["b_AM_Y"] * 0 * S[n,"M"] * b["A.M.inter"] ) *
    (( b["b_M"] +                                                             # A'=0
       b["b_male_M"] * S[n,"male"] +
       b["b_parent_educ_M"] * S[n,"parent_educ"] +
       b["b_L1_M"] * S[n,"L1"] +
       b["b_A_M"] * 0 )^( S[n,"M"] )) *
    (( 1 - (b["b_M"] +
            b["b_male_M"] * S[n,"male"] +
            b["b_parent_educ_M"] * S[n,"parent_educ"] +
            b["b_L1_M"] * S[n,"L1"] +
            b["b_A_M"] * 0 ) )^( 1 - S[n,"M"] )) *
   ((b["p_L1"])^(S[n,"L1"])) *
   ((1 - b["p_L1"])^(1 - S[n,"L1"])) *
   ((b["p_L0_male"])^(S[n,"male"])) *
   ((1 - b["p_L0_male"])^(1 - S[n,"male"])) *
   ((b["p_L0_parent_low_educ_lv"])^(S[n,"parent_educ"])) *
   ((1 - b["p_L0_parent_low_educ_lv"])^(1 - S[n,"parent_educ"]))
  }

crNDE.death <- sum(S[,"sum.psi10"]) - sum(S[,"sum.psi00"])
crNIE.death <- sum(S[,"sum.psi11"]) - sum(S[,"sum.psi10"])

# quantitative outcome (QoL)
S <- cbind(expand.grid(c(0,1),c(0,1),c(0,1),c(0,1)), rep(NA,n=2^4),
           rep(NA,n=2^4), rep(NA,n=2^4))
colnames(S) <- list("male","parent_educ","L1","M","sum.psi11", "sum.psi10",
                    "sum.psi00")
```

```r
for (n in 1:16) {
  S[n,"sum.psi11"] <-  ( b["mu_Y"] +                                          # A=1
                            b["c_male_Y"] * S[n,"male"] +
                            b["c_parent_educ_Y"] * S[n,"parent_educ"] +
                            b["c_A_Y"] * 1 +
                            b["c_L1_Y"] * S[n,"L1"] +
                            b["c_M_Y"] * S[n,"M"] +
                            b["c_AM_Y"] * 1 * S[n,"M"] * b["A.M.inter"] ) *
      (( b["b_M"] +                                                           # A'=1
          b["b_male_M"] * S[n,"male"] +
          b["b_parent_educ_M"] * S[n,"parent_educ"] +
          b["b_L1_M"] * S[n,"L1"] +
          b["b_A_M"] * 1 )^( S[n,"M"] )) *
      (( 1 - (b["b_M"] +
              b["b_male_M"] * S[n,"male"] +
              b["b_parent_educ_M"] * S[n,"parent_educ"] +
              b["b_L1_M"] * S[n,"L1"] +
              b["b_A_M"] * 1 ) )^( 1 - S[n,"M"] )) *
      ((b["p_L1"])^(S[n,"L1"])) *
      ((1 - b["p_L1"])^(1 - S[n,"L1"])) *
      ((b["p_L0_male"])^(S[n,"male"])) *
      ((1 - b["p_L0_male"])^(1 - S[n,"male"])) *
      ((b["p_L0_parent_low_educ_lv"])^(S[n,"parent_educ"])) *
      ((1 - b["p_L0_parent_low_educ_lv"])^(1 - S[n,"parent_educ"]))

  S[n,"sum.psi10"] <-  ( b["mu_Y"] +                                          # A=1
                            b["c_male_Y"] * S[n,"male"] +
                            b["c_parent_educ_Y"] * S[n,"parent_educ"] +
                            b["c_A_Y"] * 1 +
                            b["c_L1_Y"] * S[n,"L1"] +
                            b["c_M_Y"] * S[n,"M"] +
                            b["c_AM_Y"] * 1 * S[n,"M"] * b["A.M.inter"] ) *
      (( b["b_M"] +                                                           # A'=0
          b["b_male_M"] * S[n,"male"] +
          b["b_parent_educ_M"] * S[n,"parent_educ"] +
          b["b_L1_M"] * S[n,"L1"] +
          b["b_A_M"] * 0 )^( S[n,"M"] )) *
      (( 1 - (b["b_M"] +
              b["b_male_M"] * S[n,"male"] +
              b["b_parent_educ_M"] * S[n,"parent_educ"] +
              b["b_L1_M"] * S[n,"L1"] +
              b["b_A_M"] * 0 ) )^( 1 - S[n,"M"] )) *
      ((b["p_L1"])^(S[n,"L1"])) *
      ((1 - b["p_L1"])^(1 - S[n,"L1"])) *
      ((b["p_L0_male"])^(S[n,"male"])) *
```

```
          ((1 - b["p_L0_male"])^(1 - S[n,"male"])) *
          ((b["p_L0_parent_low_educ_lv"])^(S[n,"parent_educ"])) *
          ((1 - b["p_L0_parent_low_educ_lv"])^(1 - S[n,"parent_educ"]))

      S[n,"sum.psi00"] <-  ( b["mu_Y"] +                                        # A=0
                              b["c_male_Y"] * S[n,"male"] +
                              b["c_parent_educ_Y"] * S[n,"parent_educ"] +
                              b["c_A_Y"] * 0 +
                              b["c_L1_Y"] * S[n,"L1"] +
                              b["c_M_Y"] * S[n,"M"] +
                              b["c_AM_Y"] * 0 * S[n,"M"] * b["A.M.inter"] ) *
        (( b["b_M"] +                                                          # A'=0
             b["b_male_M"] * S[n,"male"] +
             b["b_parent_educ_M"] * S[n,"parent_educ"] +
             b["b_L1_M"] * S[n,"L1"] +
             b["b_A_M"] * 0 )^( S[n,"M"] )) *
        (( 1 - (b["b_M"] +
                 b["b_male_M"] * S[n,"male"] +
                 b["b_parent_educ_M"] * S[n,"parent_educ"] +
                 b["b_L1_M"] * S[n,"L1"] +
                 b["b_A_M"] * 0 ) )^( 1 - S[n,"M"] )) *
        ((b["p_L1"])^(S[n,"L1"])) *
        ((1 - b["p_L1"])^(1 - S[n,"L1"])) *
        ((b["p_L0_male"])^(S[n,"male"])) *
        ((1 - b["p_L0_male"])^(1 - S[n,"male"])) *
        ((b["p_L0_parent_low_educ_lv"])^(S[n,"parent_educ"])) *
        ((1 - b["p_L0_parent_low_educ_lv"])^(1 - S[n,"parent_educ"]))
      }

    crNDE.qol <- sum(S[,"sum.psi10"]) - sum(S[,"sum.psi00"])
    crNIE.qol <- sum(S[,"sum.psi11"]) - sum(S[,"sum.psi10"])

    return(list(crNDE.death = crNDE.death, crNIE.death = crNIE.death,
                crNDE.qol = crNDE.qol, crNIE.qol = crNIE.qol))
  }
```

```
true.cond.random.no.inter <- true.cond.random(interaction = 0)

true.cond.random.with.inter <- true.cond.random(interaction = 1)
```

The conditional randomized direct effect CRDE $= \mathbb{E}\left(Y_{1,\Gamma_{0|L(0),L(1)}}\right) - \mathbb{E}\left(Y_{0,\Gamma_{0|L(0),L(1)}}\right)$ and conditional randomized indirect effect CRIE $= \mathbb{E}\left(Y_{1,\Gamma_{1|L(0),L(1)}}\right) - \mathbb{E}\left(Y_{1,\Gamma_{0|L(0),L(1)}}\right)$ are respectively:

- 0.05 and 0.00800000000000003 for death without interaction,
- 0.05855 and 0.011 for death with interaction,
- -3.99999999999999 and -0.900000000000006 for quality of life without interaction,
- -5.42499999999999 and -1.40000000000001 for quality of life with interaction.

Table 10.1: True values without time varying confounders

Effects	Without $A * M$ interaction	with $A * M$ interaction
Binary outcome		
Average total effect (ATE)	0.058	0.06955
Controlled direct effect (CDE)		
- CDE, setting do(M=0)	0.05	0.05
- CDE, setting do(M=1)	0.05	0.08
Pure NDE and Total NIE		
- PNDE	0.05	0.05855
- TNIE	0.00800000000000001	0.011
Total NDE and Pure NIE		
- TNDE	0.05	0.06155
- PNIE	0.00800000000000001	0.00800000000000001
3-way decomposition		
- PDE	0.05	0.05855
- PIE	0.00800000000000001	0.00800000000000001
- MI	0.000	0.003
4-way decomposition		
- CDE	0.05	0.05
- INTref	0.000	0.00855
- INTmed	0.000	0.003

Effects	Without $A * M$ interaction	with $A * M$ interaction
- PIE	0.008	0.008
Marginal randomized		
- marginal rNDE	0.05	0.05855
- marginal rNIE	0.00800000000000003	0.011
Conditional randomized		
- conditional rNDE	0.05	0.05855
- conditional rNIE	0.00800000000000003	0.011
Quantitative outcome		
Average total effect (ATE)	-4.9	-6.825
Controlled direct effect (CDE)		
- CDE, setting do(M=0)	-4	-4
- CDE, setting do(M=1)	-4	-9
Pure NDE and Total NIE		
- PNDE	-4	-5.425
- TNIE	-0.9	-1.4
Total NDE and Pure NIE		
- TNDE	-4	-5.925
- PNIE	-0.899999999999999	-0.899999999999999
3-way decomposition		
- PDE	-4	-5.425
- PIE	-0.899999999999999	-0.899999999999999
- MI	0	-0.5

Effects	Without $A * M$ interaction	with $A * M$ interaction
4-way decomposition		
- CDE	-4	-4
- INTref	0.000	-1.425
- INTmed	0.000	-0.5
- PIE	-0.9	-0.9
Marginal randomized		
- marginal rNDE	-3.99999999999999	-5.42499999999999
- marginal rNIE	-0.900000000000006	-1.40000000000001
Conditional randomized		
- conditional rNDE	-3.99999999999999	-5.42499999999999
- conditional rNIE	-0.900000000000006	-1.40000000000001

10.2   True causal quantities with mediator-ouctome confounder affected by the exposure

10.2.1   Average total effects (ATE)

The following function `true.ATE.tv.conf` can be used to run the calculation for the average total effects (ATE).

```r
true.ATE.time.var.conf <- function(interaction = NULL) {
  b <- param.causal.model.2(A.M.interaction = interaction)

  # binary outcome (death)
  S <- cbind(expand.grid(c(0,1),c(0,1),c(0,1), c(0,1)), rep(NA,n=2^4))
  colnames(S) <- list("male","parent_educ","L1","M","sum")
  for (n in 1:16) {
    S[n,"sum"] <- ( ( ( b["b_Y"] +
                    b["b_male_Y"] * S[n,"male"] +
                    b["b_parent_educ_Y"] * S[n,"parent_educ"] +
                    b["b_A_Y"] * 1 +
                    b["b_L1_Y"] * S[n,"L1"] +
                    b["b_M_Y"] * S[n,"M"] +
```

```
                        b["b_AM_Y"] * 1 * S[n,"M"] * b["A.M.inter"] ) *
                    (( b["b_M"] +
                          b["b_male_M"] * S[n,"male"] +
                          b["b_parent_educ_M"] * S[n,"parent_educ"] +
                          b["b_L1_M"] * S[n,"L1"] +
                          b["b_A_M"] * 1 )^( S[n,"M"] )) *
                    (( 1 - (b["b_M"] +
                            b["b_male_M"] * S[n,"male"] +
                            b["b_parent_educ_M"] * S[n,"parent_educ"] +
                            b["b_L1_M"] * S[n,"L1"] +
                            b["b_A_M"] * 1) )^( 1 - S[n,"M"] )) *
                      (( b["b_L1"] +
                            b["b_male_L1"] * S[n,"male"] +
                            b["b_parent_L1"] * S[n,"parent_educ"] +
                            b["b_A_L1"] * 1)^( S[n,"L1"] )) *
                      (( 1 - ( b["b_L1"] +
                               b["b_male_L1"] * S[n,"male"] +
                               b["b_parent_L1"] * S[n,"parent_educ"] +
                               b["b_A_L1"] * 1))^( 1 - S[n,"L1"] )) ) -
                    ( ( b["b_Y"] +
                          b["b_male_Y"] * S[n,"male"] +
                          b["b_parent_educ_Y"] * S[n,"parent_educ"] +
                          b["b_A_Y"] * 0 +
                          b["b_L1_Y"] * S[n,"L1"] +
                          b["b_M_Y"] * S[n,"M"] +
                          b["b_AM_Y"] * 0 * S[n,"M"] * b["A.M.inter"] ) *
                      (( b["b_M"] +
                            b["b_male_M"] * S[n,"male"] +
                            b["b_parent_educ_M"] * S[n,"parent_educ"] +
                            b["b_L1_M"] * S[n,"L1"] +
                            b["b_A_M"] * 0 )^( S[n,"M"] )) *
                      (( 1 - (b["b_M"] +
                              b["b_male_M"] * S[n,"male"] +
                              b["b_parent_educ_M"] * S[n,"parent_educ"] +
                              b["b_L1_M"] * S[n,"L1"] +
                              b["b_A_M"] * 0) )^( 1 - S[n,"M"] )) *
                      (( b["b_L1"] +
                            b["b_male_L1"] * S[n,"male"] +
                            b["b_parent_L1"] * S[n,"parent_educ"] +
                            b["b_A_L1"] * 0)^( S[n,"L1"] )) *
                      (( 1 - ( b["b_L1"] +
                               b["b_male_L1"] * S[n,"male"] +
                               b["b_parent_L1"] * S[n,"parent_educ"] +
                               b["b_A_L1"] * 0))^( 1 - S[n,"L1"] )) ) ) *
        ((b["p_L0_male"])^(S[n,"male"])) *
```

```r
      ((1 - b["p_L0_male"])^(1 - S[n,"male"])) *
      ((b["p_L0_parent_low_educ_lv"])^(S[n,"parent_educ"])) *
      ((1 - b["p_L0_parent_low_educ_lv"])^(1 - S[n,"parent_educ"]))
  }

ATE.death <- sum(S[,"sum"])

# quantitative outcome (QoL)
S <- cbind(expand.grid(c(0,1),c(0,1),c(0,1), c(0,1)), rep(NA,n=2^4))
colnames(S) <- list("male","parent_educ","L1","M","sum")
for (n in 1:16) {
  S[n,"sum"] <- ( ( ( b["mu_Y"] +
                        b["c_male_Y"] * S[n,"male"] +
                        b["c_parent_educ_Y"] * S[n,"parent_educ"] +
                        b["c_A_Y"] * 1 +
                        b["c_L1_Y"] * S[n,"L1"] +
                        b["c_M_Y"] * S[n,"M"] +
                        b["c_AM_Y"] * 1 * S[n,"M"] * b["A.M.inter"] ) *
                      (( b["b_M"] +
                          b["b_male_M"] * S[n,"male"] +
                          b["b_parent_educ_M"] * S[n,"parent_educ"] +
                          b["b_L1_M"] * S[n,"L1"] +
                          b["b_A_M"] * 1 )^( S[n,"M"] )) *
                      (( 1 - (b["b_M"] +
                              b["b_male_M"] * S[n,"male"] +
                              b["b_parent_educ_M"] * S[n,"parent_educ"] +
                              b["b_L1_M"] * S[n,"L1"] +
                              b["b_A_M"] * 1) )^( 1 - S[n,"M"] )) *
                      (( b["b_L1"] +
                          b["b_male_L1"] * S[n,"male"] +
                          b["b_parent_L1"] * S[n,"parent_educ"] +
                          b["b_A_L1"] * 1)^( S[n,"L1"] )) *
                      (( 1 - ( b["b_L1"] +
                              b["b_male_L1"] * S[n,"male"] +
                              b["b_parent_L1"] * S[n,"parent_educ"] +
                              b["b_A_L1"] * 1))^( 1 - S[n,"L1"] )) ) -
                    ( ( b["mu_Y"] +
                        b["c_male_Y"] * S[n,"male"] +
                        b["c_parent_educ_Y"] * S[n,"parent_educ"] +
                        b["c_A_Y"] * 0 +
                        b["c_L1_Y"] * S[n,"L1"] +
                        b["c_M_Y"] * S[n,"M"] +
                        b["c_AM_Y"] * 0 * S[n,"M"] * b["A.M.inter"] ) *
                      (( b["b_M"] +
                          b["b_male_M"] * S[n,"male"] +
```

```
                              b["b_parent_educ_M"] * S[n,"parent_educ"] +
                              b["b_L1_M"] * S[n,"L1"] +
                              b["b_A_M"] * 0 )^( S[n,"M"] )) *
                      (( 1 - (b["b_M"] +
                              b["b_male_M"] * S[n,"male"] +
                              b["b_parent_educ_M"] * S[n,"parent_educ"] +
                              b["b_L1_M"] * S[n,"L1"] +
                              b["b_A_M"] * 0) )^( 1 - S[n,"M"] )) ) *
                      (( b["b_L1"] +
                         b["b_male_L1"] * S[n,"male"] +
                         b["b_parent_L1"] * S[n,"parent_educ"] +
                         b["b_A_L1"] * 0)^( S[n,"L1"] )) *
                      (( 1 - ( b["b_L1"] +
                              b["b_male_L1"] * S[n,"male"] +
                              b["b_parent_L1"] * S[n,"parent_educ"] +
                              b["b_A_L1"] * 0))^( 1 - S[n,"L1"] )) ) *
          ((b["p_L0_male"])^(S[n,"male"])) *
          ((1 - b["p_L0_male"])^(1 - S[n,"male"])) *
          ((b["p_L0_parent_low_educ_lv"])^(S[n,"parent_educ"])) *
          ((1 - b["p_L0_parent_low_educ_lv"])^(1 - S[n,"parent_educ"]))
     }

  ATE.qol <- sum(S[,"sum"])

  return(list(ATE.death = ATE.death, ATE.qol = ATE.qol))
}
```

```
true.ATE2.no.inter <- true.ATE.time.var.conf(interaction = 0)

true.ATE2.with.inter <- true.ATE.time.var.conf(interaction = 1)
```

The average total effects $\text{ATE} = \mathbb{E}(Y_1) - \mathbb{E}(Y_0)$ are:

- 0.0752 for death and -6.26 for quality of life without interaction
- 0.089282 for death and -8.607 for quality of life with interaction

10.2.2 Controlled direct effects (CDE)

The following function `true.CDE.time.var` can be used to run the calculation for controlled direct effects (CDE).

```
true.CDE.time.var <- function(interaction = NULL) {
  b <- param.causal.model.2(A.M.interaction = interaction)
```

```r
# binary outcome (death)
# we estimate both CDE, fixing do(M) = 0 et do(M) = 1 and using the
# corresponding lines in the S matrix
S <- cbind(expand.grid(c(0,1),c(0,1),c(0,1),c(0,1)), rep(NA,n=2^3))
colnames(S) <- list("male","parent_educ","L1","M","sum")
for (n in 1:16) {
  S[n,"sum"] <- ( (( b["b_Y"] +
                      b["b_male_Y"] * S[n,"male"] +
                      b["b_parent_educ_Y"] * S[n,"parent_educ"] +
                      b["b_A_Y"] * 1 +
                      b["b_L1_Y"] * S[n,"L1"] +
                      b["b_M_Y"] * S[n,"M"] +
                      b["b_AM_Y"] * 1 * S[n,"M"] * b["A.M.inter"] ) *
                   (( b["b_L1"] +
                        b["b_male_L1"] * S[n,"male"] +
                        b["b_parent_L1"] * S[n,"parent_educ"] +
                        b["b_A_L1"] * 1)^( S[n,"L1"] )) *
                   (( 1 - ( b["b_L1"] +
                             b["b_male_L1"] * S[n,"male"] +
                             b["b_parent_L1"] * S[n,"parent_educ"] +
                             b["b_A_L1"] * 1))^( 1 - S[n,"L1"] )) ) -
                  (( b["b_Y"] +
                      b["b_male_Y"] * S[n,"male"] +
                      b["b_parent_educ_Y"] * S[n,"parent_educ"] +
                      b["b_A_Y"] * 0 +
                      b["b_L1_Y"] * S[n,"L1"] +
                      b["b_M_Y"] * S[n,"M"] +
                      b["b_AM_Y"] * 0 * S[n,"M"] * b["A.M.inter"] ) *
                   (( b["b_L1"] +
                        b["b_male_L1"] * S[n,"male"] +
                        b["b_parent_L1"] * S[n,"parent_educ"] +
                        b["b_A_L1"] * 0)^( S[n,"L1"] )) *
                   (( 1 - ( b["b_L1"] +
                             b["b_male_L1"] * S[n,"male"] +
                             b["b_parent_L1"] * S[n,"parent_educ"] +
                             b["b_A_L1"] * 0))^( 1 - S[n,"L1"] )) ) ) *
    ((b["p_L0_male"])^(S[n,"male"])) *
    ((1 - b["p_L0_male"])^(1 - S[n,"male"])) *
    ((b["p_L0_parent_low_educ_lv"])^(S[n,"parent_educ"])) *
    ((1 - b["p_L0_parent_low_educ_lv"])^(1 - S[n,"parent_educ"]))
  }

CDE.M0.death <- sum(S[1:8,"sum"])
CDE.M1.death <- sum(S[9:16,"sum"])
```

```r
# quantitative outcome (QoL)
# we estimate both CDE, fixing do(M) = 0 et do(M) = 1 and using the
# corresponding lines in the S matrix
for (n in 1:16) {
  S[n,"sum"] <- ( (( b["mu_Y"] +
                       b["c_male_Y"] * S[n,"male"] +
                       b["c_parent_educ_Y"] * S[n,"parent_educ"] +
                       b["c_A_Y"] * 1 +
                       b["c_L1_Y"] * S[n,"L1"] +
                       b["c_M_Y"] * S[n,"M"] +
                       b["c_AM_Y"] * 1 * S[n,"M"] * b["A.M.inter"] ) *
                     (( b["b_L1"] +
                          b["b_male_L1"] * S[n,"male"] +
                          b["b_parent_L1"] * S[n,"parent_educ"] +
                          b["b_A_L1"] * 1)^( S[n,"L1"] )) *
                     (( 1 - ( b["b_L1"] +
                               b["b_male_L1"] * S[n,"male"] +
                               b["b_parent_L1"] * S[n,"parent_educ"] +
                               b["b_A_L1"] * 1))^( 1 - S[n,"L1"] ))) -
                    (( b["mu_Y"] +
                         b["c_male_Y"] * S[n,"male"] +
                         b["c_parent_educ_Y"] * S[n,"parent_educ"] +
                         b["c_A_Y"] * 0 +
                         b["c_L1_Y"] * S[n,"L1"] +
                         b["c_M_Y"] * S[n,"M"] +
                         b["c_AM_Y"] * 0 * S[n,"M"] * b["A.M.inter"] ) *
                      (( b["b_L1"] +
                           b["b_male_L1"] * S[n,"male"] +
                           b["b_parent_L1"] * S[n,"parent_educ"] +
                           b["b_A_L1"] * 0)^( S[n,"L1"] )) *
                      (( 1 - ( b["b_L1"] +
                                b["b_male_L1"] * S[n,"male"] +
                                b["b_parent_L1"] * S[n,"parent_educ"] +
                                b["b_A_L1"] * 0))^( 1 - S[n,"L1"] )) ) ) *
    ((b["p_L0_male"])^(S[n,"male"])) *
    ((1 - b["p_L0_male"])^(1 - S[n,"male"])) *
    ((b["p_L0_parent_low_educ_lv"])^(S[n,"parent_educ"])) *
    ((1 - b["p_L0_parent_low_educ_lv"])^(1 - S[n,"parent_educ"]))
  }

CDE.M0.qol <- sum(S[1:8,"sum"])
CDE.M1.qol <- sum(S[9:16,"sum"])

return(list(CDE.M0.death = CDE.M0.death, CDE.M1.death = CDE.M1.death,
            CDE.M0.qol = CDE.M0.qol, CDE.M1.qol = CDE.M1.qol))
```

```
}
```

```
true.CDE2.no.inter <- true.CDE.time.var(interaction = 0)
```

```
true.CDE2.with.inter <- true.CDE.time.var(interaction = 1)
```

Setting $do(M = 0)$, the controlled direct effects $\text{CDE}_{M=0} = \mathbb{E}\left(Y_{1,0}\right) - \mathbb{E}\left(Y_{0,0}\right)$ are:

- 0.064 for death and -5 for quality of life without interaction
- 0.064 for death and -5 for quality of life with interaction

Setting $do(M = 1)$, the controlled direct effects $\text{CDE}_{M=1} = \mathbb{E}\left(Y_{1,1}\right) - \mathbb{E}\left(Y_{0,1}\right)$ are:

- 0.064 for death and -5 for quality of life without interaction
- 0.094 for death and -10 for quality of life with interaction

10.2.3 Marginal randomized direct and indirect effects

The following function `true.marg.random.time.var` can be used to run the calculation for the marginal randomized natural direct (marginal MRDE) and indirect effects (marginal MRIE).

```
true.marg.random.time.var <- function(interaction = NULL) {
  b <- param.causal.model.2(A.M.interaction = interaction)

  # marginal distribution of M
  M.S <- cbind(expand.grid(c(0,1),c(0,1),c(0,1),c(0,1),c(0,1)), rep(NA,n=2^5))
  colnames(M.S) <- list("male","parent_educ","L1","M","A","sum")

  for (n in 1:32) {
    M.S[n,"sum"] <- (( b["b_M"] +
                       b["b_male_M"] * M.S[n,"male"] +
                       b["b_parent_educ_M"] * M.S[n,"parent_educ"] +
                       b["b_L1_M"] * M.S[n,"L1"] +
                       b["b_A_M"] * M.S[n,"A"])^( M.S[n,"M"] )) *
      (( 1 - (b["b_M"] +
              b["b_male_M"] * M.S[n,"male"] +
              b["b_parent_educ_M"] * M.S[n,"parent_educ"] +
              b["b_L1_M"] * M.S[n,"L1"] +
              b["b_A_M"] * M.S[n,"A"]) )^( 1 - M.S[n,"M"] )) *
      (( b["b_L1"] +
```

```
        b["b_male_L1"] * M.S[n,"male"] +
        b["b_parent_L1"] * M.S[n,"parent_educ"] +
        b["b_A_L1"] * M.S[n,"A"])^( M.S[n,"L1"] )) *
  (( 1 - ( b["b_L1"] +
            b["b_male_L1"] * M.S[n,"male"] +
            b["b_parent_L1"] * M.S[n,"parent_educ"] +
            b["b_A_L1"] * M.S[n,"A"]))^( 1 - M.S[n,"L1"] ))
}

M0.A0.L00 <- sum(M.S[M.S[,"M"]==0 & M.S[,"A"]==0 & M.S[,"male"]==0 &
                     M.S[,"parent_educ"]==0,"sum"])
M0.A0.L01 <- sum(M.S[M.S[,"M"]==0 & M.S[,"A"]==0 & M.S[,"male"]==0 &
                     M.S[,"parent_educ"]==1,"sum"])
M0.A0.L10 <- sum(M.S[M.S[,"M"]==0 & M.S[,"A"]==0 & M.S[,"male"]==1 &
                     M.S[,"parent_educ"]==0,"sum"])
M0.A0.L11 <- sum(M.S[M.S[,"M"]==0 & M.S[,"A"]==0 & M.S[,"male"]==1 &
                     M.S[,"parent_educ"]==1,"sum"])

M1.A0.L00 <- sum(M.S[M.S[,"M"]==1 & M.S[,"A"]==0 & M.S[,"male"]==0 &
                     M.S[,"parent_educ"]==0,"sum"])
M1.A0.L01 <- sum(M.S[M.S[,"M"]==1 & M.S[,"A"]==0 & M.S[,"male"]==0 &
                     M.S[,"parent_educ"]==1,"sum"])
M1.A0.L10 <- sum(M.S[M.S[,"M"]==1 & M.S[,"A"]==0 & M.S[,"male"]==1 &
                     M.S[,"parent_educ"]==0,"sum"])
M1.A0.L11 <- sum(M.S[M.S[,"M"]==1 & M.S[,"A"]==0 & M.S[,"male"]==1 &
                     M.S[,"parent_educ"]==1,"sum"])

M0.A1.L00 <- sum(M.S[M.S[,"M"]==0 & M.S[,"A"]==1 & M.S[,"male"]==0 &
                     M.S[,"parent_educ"]==0,"sum"])
M0.A1.L01 <- sum(M.S[M.S[,"M"]==0 & M.S[,"A"]==1 & M.S[,"male"]==0 &
                     M.S[,"parent_educ"]==1,"sum"])
M0.A1.L10 <- sum(M.S[M.S[,"M"]==0 & M.S[,"A"]==1 & M.S[,"male"]==1 &
                     M.S[,"parent_educ"]==0,"sum"])
M0.A1.L11 <- sum(M.S[M.S[,"M"]==0 & M.S[,"A"]==1 & M.S[,"male"]==1 &
                     M.S[,"parent_educ"]==1,"sum"])

M1.A1.L00 <- sum(M.S[M.S[,"M"]==1 & M.S[,"A"]==1 & M.S[,"male"]==0 &
                     M.S[,"parent_educ"]==0,"sum"])
M1.A1.L01 <- sum(M.S[M.S[,"M"]==1 & M.S[,"A"]==1 & M.S[,"male"]==0 &
                     M.S[,"parent_educ"]==1,"sum"])
M1.A1.L10 <- sum(M.S[M.S[,"M"]==1 & M.S[,"A"]==1 & M.S[,"male"]==1 &
                     M.S[,"parent_educ"]==0,"sum"])
M1.A1.L11 <- sum(M.S[M.S[,"M"]==1 & M.S[,"A"]==1 & M.S[,"male"]==1 &
                     M.S[,"parent_educ"]==1,"sum"])
```

```r
# binary outcome (death)
S <- cbind(expand.grid(c(0,1),c(0,1),c(0,1),c(0,1)), rep(NA,n=2^4),
           rep(NA,n=2^4), rep(NA,n=2^4))
colnames(S) <- list("male","parent_educ","L1","M","sum.psi11",
                    "sum.psi10", "sum.psi00")
for (n in 1:16) {
  S[n,"sum.psi11"] <-  ( b["b_Y"] +                                    # A=1
                           b["b_male_Y"] * S[n,"male"] +
                           b["b_parent_educ_Y"] * S[n,"parent_educ"] +
                           b["b_A_Y"] * 1 +
                           b["b_L1_Y"] * S[n,"L1"] +
                           b["b_M_Y"] * S[n,"M"] +
                           b["b_AM_Y"] * 1 * S[n,"M"] * b["A.M.inter"] ) *
        ((M1.A1.L00*(S[n,"male"]==0)*(S[n,"parent_educ"]==0) +         # A'=1
            M1.A1.L01*(S[n,"male"]==0)*(S[n,"parent_educ"]==1) +
            M1.A1.L10*(S[n,"male"]==1)*(S[n,"parent_educ"]==0) +
            M1.A1.L11*(S[n,"male"]==1)*(S[n,"parent_educ"]==1) )^( S[n,"M"] )) *
      ((M0.A1.L00*(S[n,"male"]==0)*(S[n,"parent_educ"]==0) +
          M0.A1.L01*(S[n,"male"]==0)*(S[n,"parent_educ"]==1) +
          M0.A1.L10*(S[n,"male"]==1)*(S[n,"parent_educ"]==0) +
          M0.A1.L11*(S[n,"male"]==1)*(S[n,"parent_educ"]==1) )^( 1 - S[n,"M"] )) *
      (( b["b_L1"] +                                                   # A=1
           b["b_male_L1"] * S[n,"male"] +
           b["b_parent_L1"] * S[n,"parent_educ"] +
           b["b_A_L1"] * 1)^( S[n,"L1"] )) *
      (( 1 - ( b["b_L1"] +
                 b["b_male_L1"] * S[n,"male"] +
                 b["b_parent_L1"] * S[n,"parent_educ"] +
                 b["b_A_L1"] * 1))^( 1 - S[n,"L1"] )) *
      ((b["p_L0_male"])^(S[n,"male"])) *
      ((1 - b["p_L0_male"])^(1 - S[n,"male"])) *
      ((b["p_L0_parent_low_educ_lv"])^(S[n,"parent_educ"])) *
      ((1 - b["p_L0_parent_low_educ_lv"])^(1 - S[n,"parent_educ"]))

  S[n,"sum.psi10"] <-  ( b["b_Y"] +                                    # A=1
                           b["b_male_Y"] * S[n,"male"] +
                           b["b_parent_educ_Y"] * S[n,"parent_educ"] +
                           b["b_A_Y"] * 1 +
                           b["b_L1_Y"] * S[n,"L1"] +
                           b["b_M_Y"] * S[n,"M"] +
                           b["b_AM_Y"] * 1 * S[n,"M"] * b["A.M.inter"] ) *
      ((M1.A0.L00*(S[n,"male"]==0)*(S[n,"parent_educ"]==0) +           # A'=0
          M1.A0.L01*(S[n,"male"]==0)*(S[n,"parent_educ"]==1) +
          M1.A0.L10*(S[n,"male"]==1)*(S[n,"parent_educ"]==0) +
          M1.A0.L11*(S[n,"male"]==1)*(S[n,"parent_educ"]==1) )^( S[n,"M"] )) *
```

```
        ((M0.A0.L00*(S[n,"male"]==0)*(S[n,"parent_educ"]==0) +
            M0.A0.L01*(S[n,"male"]==0)*(S[n,"parent_educ"]==1) +
            M0.A0.L10*(S[n,"male"]==1)*(S[n,"parent_educ"]==0) +
            M0.A0.L11*(S[n,"male"]==1)*(S[n,"parent_educ"]==1) )^( 1 - S[n,"M"] )) *
        (( b["b_L1"] +                                                      # A=1
            b["b_male_L1"] * S[n,"male"] +
            b["b_parent_L1"] * S[n,"parent_educ"] +
            b["b_A_L1"] * 1)^( S[n,"L1"] )) *
        (( 1 - ( b["b_L1"] +
                b["b_male_L1"] * S[n,"male"] +
                b["b_parent_L1"] * S[n,"parent_educ"] +
                b["b_A_L1"] * 1))^( 1 - S[n,"L1"] )) *
        ((b["p_L0_male"])^(S[n,"male"])) *
        ((1 - b["p_L0_male"])^(1 - S[n,"male"])) *
        ((b["p_L0_parent_low_educ_lv"])^(S[n,"parent_educ"])) *
        ((1 - b["p_L0_parent_low_educ_lv"])^(1 - S[n,"parent_educ"]))

    S[n,"sum.psi00"] <-  ( b["b_Y"] +                                       # A=0
                            b["b_male_Y"] * S[n,"male"] +
                            b["b_parent_educ_Y"] * S[n,"parent_educ"] +
                            b["b_A_Y"] * 0 +
                            b["b_L1_Y"] * S[n,"L1"] +
                            b["b_M_Y"] * S[n,"M"] +
                            b["b_AM_Y"] * 0 * S[n,"M"] * b["A.M.inter"] ) *
        ((M1.A0.L00*(S[n,"male"]==0)*(S[n,"parent_educ"]==0) +              # A'=0
            M1.A0.L01*(S[n,"male"]==0)*(S[n,"parent_educ"]==1) +
            M1.A0.L10*(S[n,"male"]==1)*(S[n,"parent_educ"]==0) +
            M1.A0.L11*(S[n,"male"]==1)*(S[n,"parent_educ"]==1) )^( S[n,"M"] )) *
        ((M0.A0.L00*(S[n,"male"]==0)*(S[n,"parent_educ"]==0) +
            M0.A0.L01*(S[n,"male"]==0)*(S[n,"parent_educ"]==1) +
            M0.A0.L10*(S[n,"male"]==1)*(S[n,"parent_educ"]==0) +
            M0.A0.L11*(S[n,"male"]==1)*(S[n,"parent_educ"]==1) )^( 1 - S[n,"M"] )) *
        (( b["b_L1"] +                                                      # A=0
            b["b_male_L1"] * S[n,"male"] +
            b["b_parent_L1"] * S[n,"parent_educ"] +
            b["b_A_L1"] * 0)^( S[n,"L1"] )) *
        (( 1 - ( b["b_L1"] +
                b["b_male_L1"] * S[n,"male"] +
                b["b_parent_L1"] * S[n,"parent_educ"] +
                b["b_A_L1"] * 0))^( 1 - S[n,"L1"] )) *
        ((b["p_L0_male"])^(S[n,"male"])) *
        ((1 - b["p_L0_male"])^(1 - S[n,"male"])) *
        ((b["p_L0_parent_low_educ_lv"])^(S[n,"parent_educ"])) *
        ((1 - b["p_L0_parent_low_educ_lv"])^(1 - S[n,"parent_educ"]))
    }
```

```r
mrNDE.death <- sum(S[,"sum.psi10"]) - sum(S[,"sum.psi00"])
mrNIE.death <- sum(S[,"sum.psi11"]) - sum(S[,"sum.psi10"])

# quantitative outcome (QoL)
S <- cbind(expand.grid(c(0,1),c(0,1),c(0,1),c(0,1)), rep(NA,n=2^4),
           rep(NA,n=2^4), rep(NA,n=2^4))
colnames(S) <- list("male","parent_educ","L1","M","sum.psi11", "sum.psi10",
                    "sum.psi00")
for (n in 1:16) {
  S[n,"sum.psi11"] <-  ( b["mu_Y"] +                                    # A=1
                         b["c_male_Y"] * S[n,"male"] +
                         b["c_parent_educ_Y"] * S[n,"parent_educ"] +
                         b["c_A_Y"] * 1 +
                         b["c_L1_Y"] * S[n,"L1"] +
                         b["c_M_Y"] * S[n,"M"] +
                         b["c_AM_Y"] * 1 * S[n,"M"] * b["A.M.inter"] ) *
    ((M1.A1.L00*(S[n,"male"]==0)*(S[n,"parent_educ"]==0) +              # A'=1
        M1.A1.L01*(S[n,"male"]==0)*(S[n,"parent_educ"]==1) +
        M1.A1.L10*(S[n,"male"]==1)*(S[n,"parent_educ"]==0) +
        M1.A1.L11*(S[n,"male"]==1)*(S[n,"parent_educ"]==1) )^( S[n,"M"] )) *
    ((M0.A1.L00*(S[n,"male"]==0)*(S[n,"parent_educ"]==0) +
        M0.A1.L01*(S[n,"male"]==0)*(S[n,"parent_educ"]==1) +
        M0.A1.L10*(S[n,"male"]==1)*(S[n,"parent_educ"]==0) +
        M0.A1.L11*(S[n,"male"]==1)*(S[n,"parent_educ"]==1) )^( 1 - S[n,"M"] )) *
    (( b["b_L1"] +                                                      # A=1
         b["b_male_L1"] * S[n,"male"] +
         b["b_parent_L1"] * S[n,"parent_educ"] +
         b["b_A_L1"] * 1)^( S[n,"L1"] )) *
    (( 1 - ( b["b_L1"] +
               b["b_male_L1"] * S[n,"male"] +
               b["b_parent_L1"] * S[n,"parent_educ"] +
               b["b_A_L1"] * 1))^( 1 - S[n,"L1"] )) *
    ((b["p_L0_male"])^(S[n,"male"])) *
    ((1 - b["p_L0_male"])^(1 - S[n,"male"])) *
    ((b["p_L0_parent_low_educ_lv"])^(S[n,"parent_educ"])) *
    ((1 - b["p_L0_parent_low_educ_lv"])^(1 - S[n,"parent_educ"]))

  S[n,"sum.psi10"] <-  ( b["mu_Y"] +                                    # A=1
                         b["c_male_Y"] * S[n,"male"] +
                         b["c_parent_educ_Y"] * S[n,"parent_educ"] +
                         b["c_A_Y"] * 1 +
                         b["c_L1_Y"] * S[n,"L1"] +
                         b["c_M_Y"] * S[n,"M"] +
                         b["c_AM_Y"] * 1 * S[n,"M"] * b["A.M.inter"] ) *
    ((M1.A0.L00*(S[n,"male"]==0)*(S[n,"parent_educ"]==0) +              # A'=0
```

```
          M1.A0.L01*(S[n,"male"]==0)*(S[n,"parent_educ"]==1) +
          M1.A0.L10*(S[n,"male"]==1)*(S[n,"parent_educ"]==0) +
          M1.A0.L11*(S[n,"male"]==1)*(S[n,"parent_educ"]==1) )^( S[n,"M"] )) *
       ((M0.A0.L00*(S[n,"male"]==0)*(S[n,"parent_educ"]==0) +
          M0.A0.L01*(S[n,"male"]==0)*(S[n,"parent_educ"]==1) +
          M0.A0.L10*(S[n,"male"]==1)*(S[n,"parent_educ"]==0) +
          M0.A0.L11*(S[n,"male"]==1)*(S[n,"parent_educ"]==1) )^( 1 - S[n,"M"] )) *
       (( b["b_L1"] +                                                    # A=1
          b["b_male_L1"] * S[n,"male"] +
          b["b_parent_L1"] * S[n,"parent_educ"] +
          b["b_A_L1"] * 1)^( S[n,"L1"] )) *
       (( 1 - ( b["b_L1"] +
              b["b_male_L1"] * S[n,"male"] +
              b["b_parent_L1"] * S[n,"parent_educ"] +
              b["b_A_L1"] * 1))^( 1 - S[n,"L1"] )) *
       ((b["p_L0_male"])^(S[n,"male"])) *
       ((1 - b["p_L0_male"])^(1 - S[n,"male"])) *
       ((b["p_L0_parent_low_educ_lv"])^(S[n,"parent_educ"])) *
       ((1 - b["p_L0_parent_low_educ_lv"])^(1 - S[n,"parent_educ"]))

   S[n,"sum.psi00"] <-  ( b["mu_Y"] +                                    # A=0
                          b["c_male_Y"] * S[n,"male"] +
                          b["c_parent_educ_Y"] * S[n,"parent_educ"] +
                          b["c_A_Y"] * 0 +
                          b["c_L1_Y"] * S[n,"L1"] +
                          b["c_M_Y"] * S[n,"M"] +
                          b["c_AM_Y"] * 0 * S[n,"M"] * b["A.M.inter"] ) *
       ((M1.A0.L00*(S[n,"male"]==0)*(S[n,"parent_educ"]==0) +            # A'=0
          M1.A0.L01*(S[n,"male"]==0)*(S[n,"parent_educ"]==1) +
          M1.A0.L10*(S[n,"male"]==1)*(S[n,"parent_educ"]==0) +
          M1.A0.L11*(S[n,"male"]==1)*(S[n,"parent_educ"]==1) )^( S[n,"M"] )) *
       ((M0.A0.L00*(S[n,"male"]==0)*(S[n,"parent_educ"]==0) +
          M0.A0.L01*(S[n,"male"]==0)*(S[n,"parent_educ"]==1) +
          M0.A0.L10*(S[n,"male"]==1)*(S[n,"parent_educ"]==0) +
          M0.A0.L11*(S[n,"male"]==1)*(S[n,"parent_educ"]==1) )^( 1 - S[n,"M"] )) *
       (( b["b_L1"] +                                                    # A=0
          b["b_male_L1"] * S[n,"male"] +
          b["b_parent_L1"] * S[n,"parent_educ"] +
          b["b_A_L1"] * 0)^( S[n,"L1"] )) *
       (( 1 - ( b["b_L1"] +
              b["b_male_L1"] * S[n,"male"] +
              b["b_parent_L1"] * S[n,"parent_educ"] +
              b["b_A_L1"] * 0))^( 1 - S[n,"L1"] )) *
       ((b["p_L0_male"])^(S[n,"male"])) *
       ((1 - b["p_L0_male"])^(1 - S[n,"male"])) *
```

```
        ((b["p_L0_parent_low_educ_lv"])^(S[n,"parent_educ"])) *
        ((1 - b["p_L0_parent_low_educ_lv"])^(1 - S[n,"parent_educ"]))
    }

  mrNDE.qol <- sum(S[,"sum.psi10"]) - sum(S[,"sum.psi00"])
  mrNIE.qol <- sum(S[,"sum.psi11"]) - sum(S[,"sum.psi10"])

  return(list(mrNDE.death = mrNDE.death, mrNIE.death = mrNIE.death,
              mrNDE.qol = mrNDE.qol, mrNIE.qol = mrNIE.qol))
}
```

```
true.marg.random2.no.inter <- true.marg.random.time.var(interaction = 0)
```

```
true.marg.random2.with.inter <- true.marg.random.time.var(interaction = 1)
```

The marginal randomized direct effect $\text{MRDE} = \mathbb{E}\left(Y_{1,G_{0|L(0)}}\right) - \mathbb{E}\left(Y_{0,G_{0|L(0)}}\right)$ and the marginal randomized indirect effect $\text{MRIE} = \mathbb{E}\left(Y_{1,G_{1|L(0)}}\right) - \mathbb{E}\left(Y_{1,G_{0|L(0)}}\right)$ are respectively:

- 0.064 and 0.0112 for death without interaction
- 0.073882 and 0.0154000000000001 for death with interaction
- -4.99999999999999 and -1.26 for quality of life without interaction
- -6.64699999999998 and -1.96 for quality of life with interaction

10.2.4 Conditional randomized direct and indirect effects

The following function `true.cond.random.time.var` can be used to run the calculation for the conditional randomized natural direct (CRDE) and the conditional randomized indirect effects (CRIE).

```
true.cond.random.time.var <- function(interaction = NULL) {
  b <- param.causal.model.2(A.M.interaction = interaction)

  # binary outcome (death)
  S <- cbind(expand.grid(c(0,1),c(0,1),c(0,1),c(0,1)), rep(NA,n=2^4),
             rep(NA,n=2^4), rep(NA,n=2^4))
  colnames(S) <- list("male","parent_educ","L1","M","sum.psi11", "sum.psi10",
                      "sum.psi00")
  for (n in 1:16) {
    S[n,"sum.psi11"] <-  ( b["b_Y"] +                                  # A=1
                            b["b_male_Y"] * S[n,"male"] +
                            b["b_parent_educ_Y"] * S[n,"parent_educ"] +
                            b["b_A_Y"] * 1 +
```

```
                                 b["b_L1_Y"] * S[n,"L1"] +
                                 b["b_M_Y"] * S[n,"M"] +
                                 b["b_AM_Y"] * 1 * S[n,"M"] * b["A.M.inter"] ) *
     (( b["b_M"] +                                                                    # A'=1
         b["b_male_M"] * S[n,"male"] +
         b["b_parent_educ_M"] * S[n,"parent_educ"] +
         b["b_L1_M"] * S[n,"L1"] +
         b["b_A_M"] * 1 )^( S[n,"M"] )) *
     (( 1 - (b["b_M"] +
             b["b_male_M"] * S[n,"male"] +
             b["b_parent_educ_M"] * S[n,"parent_educ"] +
             b["b_L1_M"] * S[n,"L1"] +
             b["b_A_M"] * 1 ) )^( 1 - S[n,"M"] )) *
     (( b["b_L1"] +                                                                   # A=1
         b["b_male_L1"] * S[n,"male"] +
         b["b_parent_L1"] * S[n,"parent_educ"] +
         b["b_A_L1"] * 1)^( S[n,"L1"] )) *
     (( 1 - ( b["b_L1"] +
             b["b_male_L1"] * S[n,"male"] +
             b["b_parent_L1"] * S[n,"parent_educ"] +
             b["b_A_L1"] * 1))^( 1 - S[n,"L1"] )) *
     ((b["p_L0_male"])^(S[n,"male"])) *
     ((1 - b["p_L0_male"])^(1 - S[n,"male"])) *
     ((b["p_L0_parent_low_educ_lv"])^(S[n,"parent_educ"])) *
     ((1 - b["p_L0_parent_low_educ_lv"])^(1 - S[n,"parent_educ"]))

  S[n,"sum.psi10"] <-  ( b["b_Y"] +                                                   # A=1
                         b["b_male_Y"] * S[n,"male"] +
                         b["b_parent_educ_Y"] * S[n,"parent_educ"] +
                         b["b_A_Y"] * 1 +
                         b["b_L1_Y"] * S[n,"L1"] +
                         b["b_M_Y"] * S[n,"M"] +
                         b["b_AM_Y"] * 1 * S[n,"M"] * b["A.M.inter"] ) *
     (( b["b_M"] +                                                                    # A'=0
         b["b_male_M"] * S[n,"male"] +
         b["b_parent_educ_M"] * S[n,"parent_educ"] +
         b["b_L1_M"] * S[n,"L1"] +
         b["b_A_M"] * 0 )^( S[n,"M"] )) *
     (( 1 - (b["b_M"] +
             b["b_male_M"] * S[n,"male"] +
             b["b_parent_educ_M"] * S[n,"parent_educ"] +
             b["b_L1_M"] * S[n,"L1"] +
             b["b_A_M"] * 0 ) )^( 1 - S[n,"M"] )) *
     (( b["b_L1"] +                                                                   # A=1
         b["b_male_L1"] * S[n,"male"] +
```

```r
        b["b_parent_L1"] * S[n,"parent_educ"] +
        b["b_A_L1"] * 1)^( S[n,"L1"] )) *
  (( 1 - ( b["b_L1"] +
            b["b_male_L1"] * S[n,"male"] +
            b["b_parent_L1"] * S[n,"parent_educ"] +
            b["b_A_L1"] * 1))^( 1 - S[n,"L1"] )) *
  ((b["p_L0_male"])^(S[n,"male"])) *
  ((1 - b["p_L0_male"])^(1 - S[n,"male"])) *
  ((b["p_L0_parent_low_educ_lv"])^(S[n,"parent_educ"])) *
  ((1 - b["p_L0_parent_low_educ_lv"])^(1 - S[n,"parent_educ"]))

  S[n,"sum.psi00"] <-  ( b["b_Y"] +                                  # A=0
                        b["b_male_Y"] * S[n,"male"] +
                        b["b_parent_educ_Y"] * S[n,"parent_educ"] +
                        b["b_A_Y"] * 0 +
                        b["b_L1_Y"] * S[n,"L1"] +
                        b["b_M_Y"] * S[n,"M"] +
                        b["b_AM_Y"] * 0 * S[n,"M"] * b["A.M.inter"] ) *
  (( b["b_M"] +                                                      # A'=0
        b["b_male_M"] * S[n,"male"] +
        b["b_parent_educ_M"] * S[n,"parent_educ"] +
        b["b_L1_M"] * S[n,"L1"] +
        b["b_A_M"] * 0 )^( S[n,"M"] )) *
  (( 1 - (b["b_M"] +
            b["b_male_M"] * S[n,"male"] +
            b["b_parent_educ_M"] * S[n,"parent_educ"] +
            b["b_L1_M"] * S[n,"L1"] +
            b["b_A_M"] * 0 ) )^( 1 - S[n,"M"] )) *
  (( b["b_L1"] +                                                     # A=0
        b["b_male_L1"] * S[n,"male"] +
        b["b_parent_L1"] * S[n,"parent_educ"] +
        b["b_A_L1"] * 0)^( S[n,"L1"] )) *
  (( 1 - ( b["b_L1"] +
            b["b_male_L1"] * S[n,"male"] +
            b["b_parent_L1"] * S[n,"parent_educ"] +
            b["b_A_L1"] * 0))^( 1 - S[n,"L1"] )) *
  ((b["p_L0_male"])^(S[n,"male"])) *
  ((1 - b["p_L0_male"])^(1 - S[n,"male"])) *
  ((b["p_L0_parent_low_educ_lv"])^(S[n,"parent_educ"])) *
  ((1 - b["p_L0_parent_low_educ_lv"])^(1 - S[n,"parent_educ"]))
  }

 crNDE.death <- sum(S[,"sum.psi10"]) - sum(S[,"sum.psi00"])
 crNIE.death <- sum(S[,"sum.psi11"]) - sum(S[,"sum.psi10"])
```

```r
# quantitative outcome (QoL)
S <- cbind(expand.grid(c(0,1),c(0,1),c(0,1),c(0,1)), rep(NA,n=2^4),
            rep(NA,n=2^4), rep(NA,n=2^4))
colnames(S) <- list("male","parent_educ","L1","M","sum.psi11", "sum.psi10",
                     "sum.psi00")
for (n in 1:16) {
  S[n,"sum.psi11"] <-  ( b["mu_Y"] +                                          # A=1
                          b["c_male_Y"] * S[n,"male"] +
                          b["c_parent_educ_Y"] * S[n,"parent_educ"] +
                          b["c_A_Y"] * 1 +
                          b["c_L1_Y"] * S[n,"L1"] +
                          b["c_M_Y"] * S[n,"M"] +
                          b["c_AM_Y"] * 1 * S[n,"M"] * b["A.M.inter"] ) *
    (( b["b_M"] +                                                             # A'=1
        b["b_male_M"] * S[n,"male"] +
        b["b_parent_educ_M"] * S[n,"parent_educ"] +
        b["b_L1_M"] * S[n,"L1"] +
        b["b_A_M"] * 1 )^( S[n,"M"] )) *
    (( 1 - (b["b_M"] +
            b["b_male_M"] * S[n,"male"] +
            b["b_parent_educ_M"] * S[n,"parent_educ"] +
            b["b_L1_M"] * S[n,"L1"] +
            b["b_A_M"] * 1 ) )^( 1 - S[n,"M"] )) *
    (( b["b_L1"] +                                                            # A=1
        b["b_male_L1"] * S[n,"male"] +
        b["b_parent_L1"] * S[n,"parent_educ"] +
        b["b_A_L1"] * 1)^( S[n,"L1"] )) *
    (( 1 - ( b["b_L1"] +
             b["b_male_L1"] * S[n,"male"] +
             b["b_parent_L1"] * S[n,"parent_educ"] +
             b["b_A_L1"] * 1))^( 1 - S[n,"L1"] )) *
    ((b["p_L0_male"])^(S[n,"male"])) *
    ((1 - b["p_L0_male"])^(1 - S[n,"male"])) *
    ((b["p_L0_parent_low_educ_lv"])^(S[n,"parent_educ"])) *
    ((1 - b["p_L0_parent_low_educ_lv"])^(1 - S[n,"parent_educ"]))

  S[n,"sum.psi10"] <-  ( b["mu_Y"] +                                          # A=1
                          b["c_male_Y"] * S[n,"male"] +
                          b["c_parent_educ_Y"] * S[n,"parent_educ"] +
                          b["c_A_Y"] * 1 +
                          b["c_L1_Y"] * S[n,"L1"] +
                          b["c_M_Y"] * S[n,"M"] +
                          b["c_AM_Y"] * 1 * S[n,"M"] * b["A.M.inter"] ) *
    (( b["b_M"] +                                                             # A'=0
        b["b_male_M"] * S[n,"male"] +
```

```
               b["b_parent_educ_M"] * S[n,"parent_educ"] +
               b["b_L1_M"] * S[n,"L1"] +
               b["b_A_M"] * 0 )^( S[n,"M"] )) *
      (( 1 - (b["b_M"] +
               b["b_male_M"] * S[n,"male"] +
               b["b_parent_educ_M"] * S[n,"parent_educ"] +
               b["b_L1_M"] * S[n,"L1"] +
               b["b_A_M"] * 0 ) )^( 1 - S[n,"M"] )) *
      (( b["b_L1"] +                                                # A=1
          b["b_male_L1"] * S[n,"male"] +
          b["b_parent_L1"] * S[n,"parent_educ"] +
          b["b_A_L1"] * 1)^( S[n,"L1"] )) *
      (( 1 - ( b["b_L1"] +
               b["b_male_L1"] * S[n,"male"] +
               b["b_parent_L1"] * S[n,"parent_educ"] +
               b["b_A_L1"] * 1))^( 1 - S[n,"L1"] )) *
      ((b["p_L0_male"])^(S[n,"male"])) *
      ((1 - b["p_L0_male"])^(1 - S[n,"male"])) *
      ((b["p_L0_parent_low_educ_lv"])^(S[n,"parent_educ"])) *
      ((1 - b["p_L0_parent_low_educ_lv"])^(1 - S[n,"parent_educ"]))

   S[n,"sum.psi00"] <-  ( b["mu_Y"] +                              # A=0
                          b["c_male_Y"] * S[n,"male"] +
                          b["c_parent_educ_Y"] * S[n,"parent_educ"] +
                          b["c_A_Y"] * 0 +
                          b["c_L1_Y"] * S[n,"L1"] +
                          b["c_M_Y"] * S[n,"M"] +
                          b["c_AM_Y"] * 0 * S[n,"M"] * b["A.M.inter"] ) *
      (( b["b_M"] +                                                # A'=0
          b["b_male_M"] * S[n,"male"] +
          b["b_parent_educ_M"] * S[n,"parent_educ"] +
          b["b_L1_M"] * S[n,"L1"] +
          b["b_A_M"] * 0 )^( S[n,"M"] )) *
      (( 1 - (b["b_M"] +
               b["b_male_M"] * S[n,"male"] +
               b["b_parent_educ_M"] * S[n,"parent_educ"] +
               b["b_L1_M"] * S[n,"L1"] +
               b["b_A_M"] * 0 ) )^( 1 - S[n,"M"] )) *
      (( b["b_L1"] +                                                # A=0
          b["b_male_L1"] * S[n,"male"] +
          b["b_parent_L1"] * S[n,"parent_educ"] +
          b["b_A_L1"] * 0)^( S[n,"L1"] )) *
      (( 1 - ( b["b_L1"] +
               b["b_male_L1"] * S[n,"male"] +
               b["b_parent_L1"] * S[n,"parent_educ"] +
```

```
                b["b_A_L1"] * 0))^( 1 - S[n,"L1"] )) *
      ((b["p_L0_male"])^(S[n,"male"])) *
      ((1 - b["p_L0_male"])^(1 - S[n,"male"])) *
      ((b["p_L0_parent_low_educ_lv"])^(S[n,"parent_educ"])) *
      ((1 - b["p_L0_parent_low_educ_lv"])^(1 - S[n,"parent_educ"]))
    }

  crNDE.qol <- sum(S[,"sum.psi10"]) - sum(S[,"sum.psi00"])
  crNIE.qol <- sum(S[,"sum.psi11"]) - sum(S[,"sum.psi10"])

  return(list(crNDE.death = crNDE.death, crNIE.death = crNIE.death,
              crNDE.qol = crNDE.qol, crNIE.qol = crNIE.qol))
}
```

```
true.cond.random2.no.inter <- true.cond.random.time.var(interaction = 0)
```

```
true.cond.random2.with.inter <- true.cond.random.time.var(interaction = 1)
```

The conditional randomized direct effect CRDE $= \mathbb{E}\left(Y_{1,\Gamma_{0|L(0),L(1)}}\right) - \mathbb{E}\left(Y_{0,\Gamma_{0|L(0),L(1)}}\right)$ and conditional randomized indirect effect CRIE $= \mathbb{E}\left(Y_{1,\Gamma_{1|L(0),L(1)}}\right) - \mathbb{E}\left(Y_{1,\Gamma_{0|L(0),L(1)}}\right)$ are respectively:

- 0.0672 and 0.00800000000000001 for death without interaction,
- 0.078282 and 0.011 for death with interaction,
- -5.36 and -0.900000000000006 for quality of life without interaction,
- -7.207 and -1.40000000000001 for quality of life with interaction.

Table 10.2: True values with time varying confounders

Effects	Without $A * M$ interaction	with $A * M$ interaction
Binary outcome		
Average total effect (ATE)	0.0752	0.089282
Controlled direct effect (CDE)		
- CDE, setting do(M=0)	0.064	0.064
- CDE, setting do(M=1)	0.064	0.094

Effects	Without $A * M$ interaction	with $A * M$ interaction
Marginal randomized		
- marginal rNDE	0.064	0.073882
- marginal rNIE	0.0112	0.0154000000000001
Conditional randomized		
- conditional rNDE	0.0672	0.078282
- conditional rNIE	0.0080000000000001	0.011
Quantitative outcome		
Average total effect (ATE)	-6.26	-8.607
Controlled direct effect (CDE)		
- CDE, setting do(M=0)	-5	-5
- CDE, setting do(M=1)	-5	-10
Marginal randomized		
- marginal rNDE	-4.99999999999999	-6.64699999999998
- marginal rNIE	-1.26	-1.96
Conditional randomized		
- conditional rNDE	-5.36	-7.207
- conditional rNIE	-0.900000000000006	-1.40000000000001

Daniel, R M, S N Cousens, B L De Stavola, M G Kenward, and J A C Sterne. 2013. "Methods for Dealing with Time-Dependent Confounding." *Statistics in Medicine* 32 (9): 1584–1618.

Naimi, Ashley I, and Brian W Whitcomb. 2020. "Estimating Risk Ratios and

Risk Differences Using Regression." *American Journal of Epidemiology* 189 (6): 508–10.

Robins, James. 1986. "A New Approach to Causal Inference in Mortality Studies with a Sustained Exposure Period—Application to Control of the Healthy Worker Survivor Effect." *Mathematical Modelling* 7 (9): 1393–1512.

Taubman, Sarah L, James M Robins, Murray A Mittleman, and Miguel A Hernán. 2009. "Intervening on Risk Factors for Coronary Heart Disease: An Application of the Parametric g-Formula." *International Journal of Epidemiology* 38 (6): 1599–1611.

Valeri, Linda, and Tyler J VanderWeele. 2013. "Mediation Analysis Allowing for Exposure-Mediator Interactions and Causal Interpretation: Theoretical Assumptions and Implementation with SAS and SPSS Macros." *Psycol Methods* 18 (2): 137–50.

VanderWeele, Tyler J. 2013. "A Three-Way Decomposition of a Total Effect into Direct, Indirect, and Interactive Effects." *Epidemiology* 13: 224–32.

———. 2014. "A Unification of Mediation and Interaction. A 4-Way Decomposition." *Epidemiology* 25: 749–61.