DGMs of Qian et al. (2020) - Part 2: With Treatment

AUTHOR
Ward B. Eiling

PUBLISHED
October 31, 2024

MODIFIED
November 8, 2024

1 Explanation versus Prediction

In het voorbeeld van Diggle et al. (2002) wat we vorige week hebben besproken, werd vermeld dat een cross-sectionele relatie tussen X_{it} en Y_{it} van interesse kan zijn met longitudinale data als voorspellen het doel is:

> "In many applications the cross-sectional association between Xit and Yit is of substantive interest. For example, in assessing the predictive potential of biomarkers for the detection of cancer, the accuracy of a marker is typically characterized by the cross-sectional sensitivity and specificity. Although alternative predictive models may be developed using longitudinal marker series, these models would not apply to the common clinical setting where only a single measurement is available." (Diggle et al., 2002, p. 256)

Daarentegen legt Qian et al. (2020) juist de nadruk op de voordelen van mixed linear models om tegelijkertijd ook individuele verschillen te kunnen voorspellen, waarbij de focus meer ligt op causaliteit:

> "A particularly appealing feature of random effects models is the ability to predict person-specific random effects, which enables quantitative characterization of between person heterogeneity due to unobserved factors (Schwartz and Stone, 2007, Bolger and Laurenceau, 2013). Understanding such heterogeneity can bring forth new scientific hypotheses for further studies. In addition, the random effects provide a model for the within-person dependence in the time-varying outcome, which improves efficiency in parameter estimation." (Qian et al., 2020, p. 376)

2 Main Simulation of Qian et al. (2020): With Treatment

2.1 Original Section: "4. Simulation"

In the simulation, we considered three generative models (GMs), all of which have an endogenous covariate. In the first two GMs, the endogenous covariate X_{it} equals the previous outcome Y_{it} plus some random noise, so the conditional independence assumption (10) is valid. In GM 3, the endogenous covariate depends directly on b_i, violating assumption (10). The details of the generative models are described below.

In GM1, we considered a simple case with only a random intercept and a random slope for A_{it}, so that $Z_{i(t_0)} = Z_{i(t_2)} = 1$ in model (7). The outcome is generated as:

$$Y_{it+1} = \alpha_0 + \alpha_1 X_{it} + b_{i0} + A_{it}(\beta_0 + \beta_1 X_{it} + b_{i2}) + \epsilon_{it+1}.$$

The random effects $b_{i0} \sim N(0, \sigma_{b0}^2)$ and $b_{i2} \sim N(0, \sigma_{b2}^2)$ are independent of each other. The covariate is generated as $X_{i1} \sim N(0, 1)$, and for $t \geq 2$,

$$X_{it} = Y_{it} + N(0, 1).$$

The randomization probability p_t is constant at $1/2$. The exogenous noise is $\epsilon_{it+1} \sim N(0, \sigma_\epsilon^2)$.

In GM2, we considered the case where $Z_{i(t_0)} = Z_{i(t_2)} = 1$, with time-varying randomization probability. The outcome is generated as:

$$Y_{it+1} = \alpha_0 + \alpha_1 X_{it} + b_{i0} + b_{i1} X_{it} + A_{it}(\beta_0 + \beta_1 X_{it} + b_{i2} + b_{i3} X_{it}) + \epsilon_{it+1}.$$

The random effects $b_{ij} \sim N(0, \sigma_{b_j}^2)$, for $0 \leq j \leq 3$, are independent of each other. The covariate is generated as $X_{i1} \sim N(0, 1)$, and for $t \geq 2$,

$$X_{it} = Y_{it} + N(0, 1).$$

The randomization probability depends on X_{it}:

$$p_t = 0.7 \cdot 1(X_{it} > -1.27) + 0.3 \cdot 1(X_{it} \leq -1.27),$$

where $1(\cdot)$ represents the indicator function, and the cutoff -1.27 was chosen so that p_t equals 0.7 or 0.3 for about half of the time. The exogenous noise is $\epsilon_{it+1} \sim N(0, \sigma_\epsilon^2)$.

GM3 is the same as GM 1, except that the covariate X_{it} depends directly on b_i:

$$X_{i1} \sim N(b_{i0}, 1), \quad X_{it} = Y_{it} + N(b_{i0}, 1) \text{ for } t \geq 2.$$

We chose the following parameter values:

$$\alpha_0 = -2, \quad \alpha_1 = -0.3, \quad \beta_0 = 1, \quad \beta_1 = 0.3,$$

$$\sigma_{b0}^2 = 4, \quad \sigma_{b1}^2 = \frac{1}{4}, \quad \sigma_{b2}^2 = 1, \quad \sigma_{b3}^2 = \frac{1}{4}, \quad \sigma_\epsilon^2 = 1.$$

3 Generative Model 1

3.1 Translation of Notation

In the table below, we will provide the translation of original notation in Qian et al. (2020) to notation more common in psychological research

Parameter	Original	New
Fixed intercept	α_0	γ_{00}
Fixed slope for X_{it}	α_1	γ_{01}
Random intercept	b_{i0}	u_{0i}
Random slope for A_{it}	b_{i2}	u_{2i}
Error term	ϵ_{it+1}	e_{it+1}

Parameter	Original	New
Fixed effect of A_{it}	β_0	γ_{10}
Interaction effect of A_{it} and X_{it}	β_1	γ_{11}
Covariate	X_{it}	Z_{it}
Randomized Treatment	A_{it}	X_{it}

Let’s first state the original model:

$$Y_{it+1} = \alpha_0 + \alpha_1 X_{it} + b_{i0} + A_{it}(\beta_0 + \beta_1 X_{it} + b_{i2}) + \epsilon_{it+1}.$$

Using this new notation, we may thus rewrite GM1 as a within model:

$$Y_{it+1} = \beta_{0i} + \beta_{1i} X_{it} + e_{it+1},$$

where:

$$\begin{aligned} \beta_{0i} &= \gamma_{00} + \gamma_{01} Z_{it} + u_{0i} \quad \text{with} \quad u_{0i} \sim \mathcal{N}(0, \sigma^2_{u0}), \\ \beta_{1i} &= \gamma_{10} + \gamma_{11} Z_{it} + u_{2i} \quad \text{with} \quad u_{2i} \sim \mathcal{N}(0, \sigma^2_{u2}). \end{aligned}$$

Combining these two equations, the model can be expressed as:

$$Y_{it+1} = \gamma_{00} + \gamma_{01} Z_{it} + u_{0i} + X_{it}(\gamma_{10} + \gamma_{11} Z_{it} + u_{2i}) + e_{it+1}.$$

More specifically, the process was generated as follows:

- the random effects $u_{0i} \sim \mathcal{N}(0, 4)$ and $u_{2i} \sim \mathcal{N}(0, 1)$ are independent of each other
- the covariate $Z_{i1} \sim \mathcal{N}(0, 1)$, and for $t \geq 2$, $Z_{it} = Y_{it} + \mathcal{N}(0, 1)$
- the randomization probability p_t is constant at 0.5
- the exogenous noise $e_{it+1} \sim \mathcal{N}(0, 1)$
- the parameter values are $\gamma_{00} = -2$, $\gamma_{01} = -0.3$, $\gamma_{10} = 1$, $\gamma_{11} = 0.3$

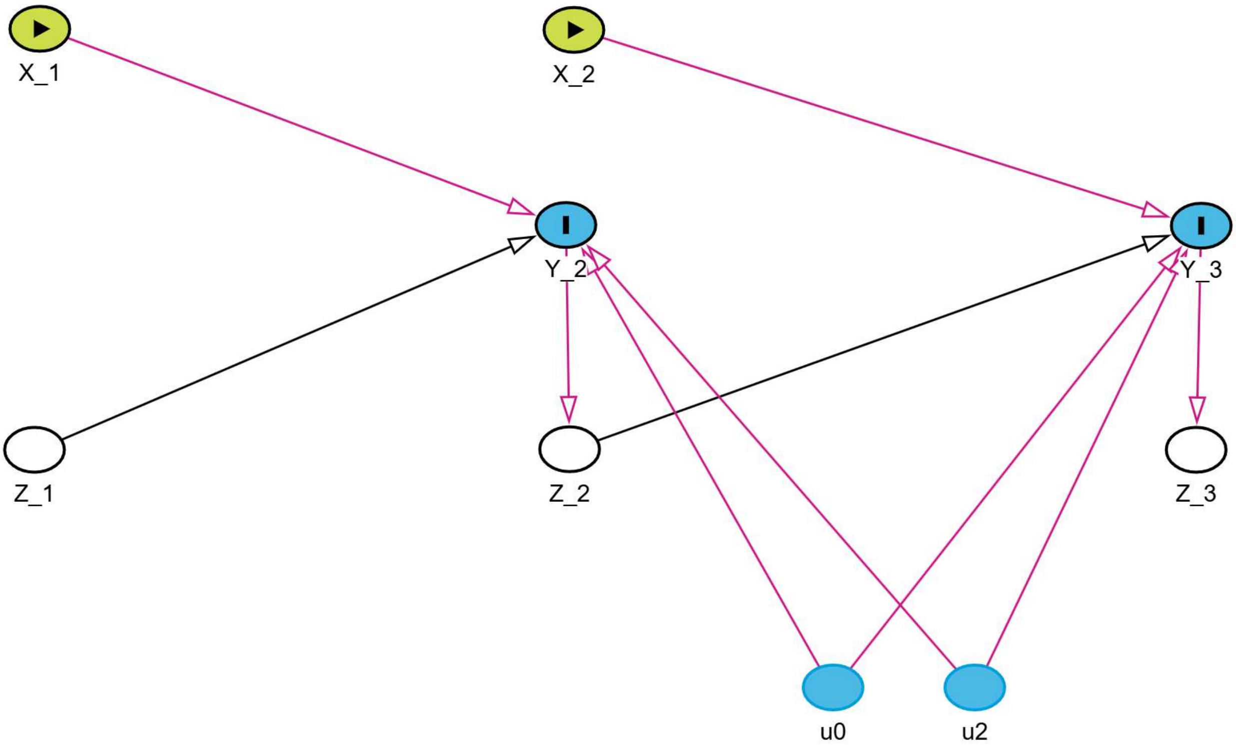
3.2 Visualzing the Model

As mentioned by Ellen in the last meeting (17-10):

> “Conventional DAGs do not only represent main effects but rather the combination of main effects and interactions. Once you have drawn your DAG, you already assume that any variables pointing to the same outcome can modify the effect of the others pointing to the same outcome.” (stackexchange)

So the DAG for the first couple observations, the DAG looks like

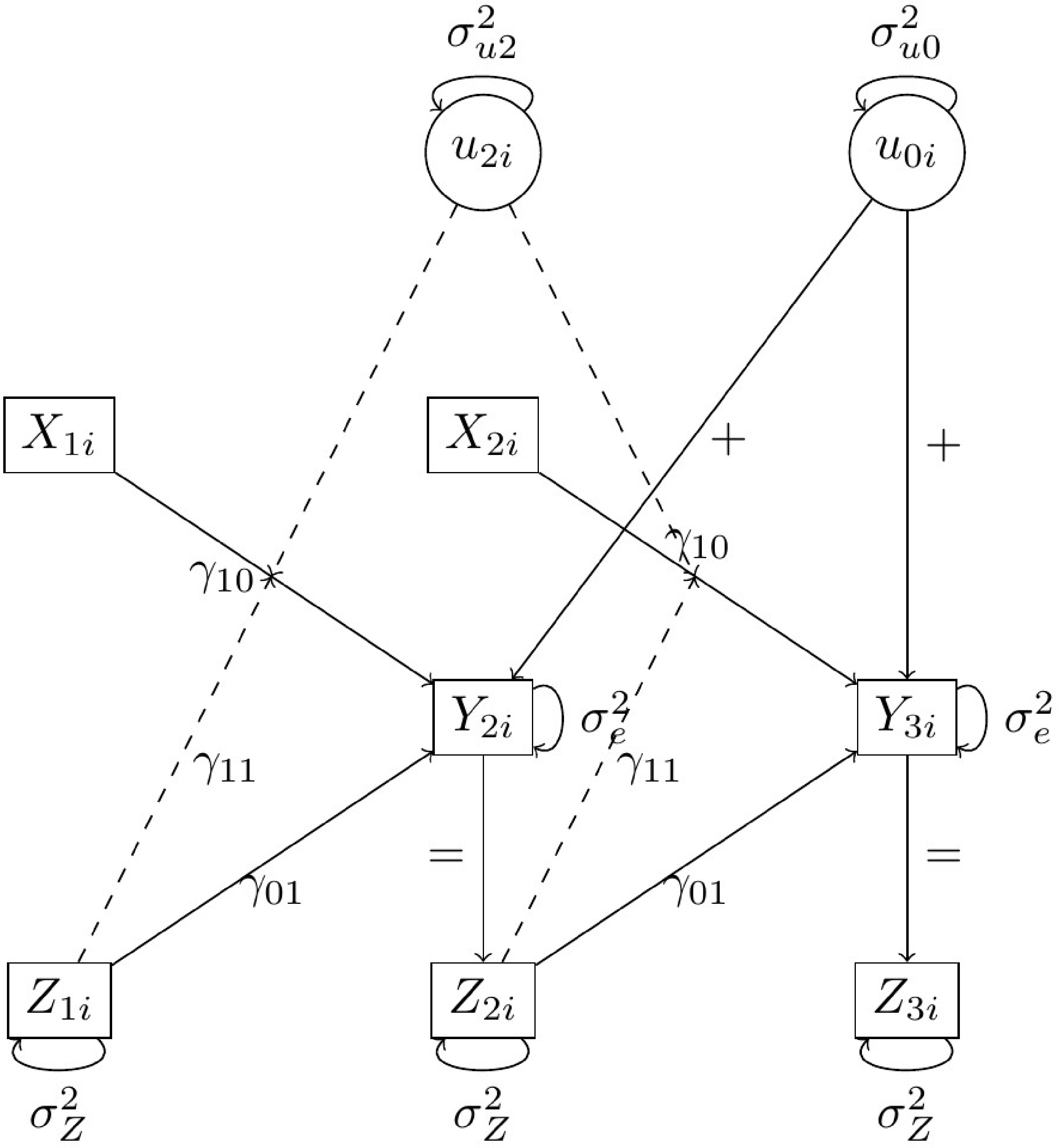
DAG for Generative Model 1



The red arrows here show the biased paths after controlling for the covariate Z_{it}.

Or we can display it as a path diagram, where parameter values are displayed and moderation is shown by the dashed arrow.

Figure 1: Path diagram for Generative Model 1



We can make a couple observations from this path diagram:

- Contrary to the DAG, this path diagram shows the moderation effect (1) of Z_{it} on the relationship between X_{it} and Y_{it+1} and (2) of u_{2i} on the relationship between Z_{it} and Y_{it+1}.
- Similar to the example without treatment in section 2.2, the covariate Z_{it} is determined by the previous value of the outcome Y_{it}—which makes it an endogenous time-varying covariate.

3.3 Data Generating and Estimation

The data generating process for this model is given by

$$Y_{it+1} = \gamma_{00} + \gamma_{01} Z_{it} + u_{0i} + X_{it}(\gamma_{10} + \gamma_{11} Z_{it} + u_{2i}) + e_{it+1}.$$

More specifically, the process was generated as follows:

- the random effects $u_{0i} \sim \mathcal{N}(0,4)$ and $u_{2i} \sim \mathcal{N}(0,1)$ are independent of each other
- the covariate $Z_{i1} \sim \mathcal{N}(0,1)$, and for $t \geq 2$, $Z_{it} = Y_{it} + \mathcal{N}(0,1)$
- the randomization probability p_t is constant at 0.5
- the exogenous noise $e_{it+1} \sim \mathcal{N}(0,1)$
- the parameter values are $\gamma_{00} = -2$, $\gamma_{01} = -0.3$, $\gamma_{10} = 1$, $\gamma_{11} = 0.3$

```r
dgm1 <- function(sample_size, total_T) {

    # Parameters
    gamma_00 <- -2   # Fixed intercept
    gamma_01 <- -0.3 # Fixed slope for Z
    gamma_10 <- 1    # Fixed effect of treatment (X)
    gamma_11 <- 0.3  # Interaction effect between treatment (X) and covariate (Z)
    sigma_u0 <- 2    # SD of random intercept (u_0i)
    sigma_u2 <- 1    # SD of random slope for treatment (u_2i)
    sigma_e <- 1     # SD of error term (e_{it+1})
    prob_x <- 0.5    # Randomization probability for treatment (X)

    # Data frame setup
    df_names <- c("userid", "day", "Z", "prob_X", "X", "Y", "u0", "u2", "e", "delta")
    dta <- data.frame(matrix(NA, nrow = sample_size * total_T, ncol = length(df_names)))
    names(dta) <- df_names

    # Assign userid and day
    dta$userid <- rep(1:sample_size, each = total_T)
    dta$day <- rep(1:total_T, times = sample_size)

    # Generate uncorrelated random effects
    u_0i <- rnorm(sample_size, mean = 0, sd = sigma_u0)
    u_2i <- rnorm(sample_size, mean = 0, sd = sigma_u2)

    # Data generation for each time point
    for (t in 1:total_T) {
        # Row index for day t for every subject
        row_index <- seq(from = t, by = total_T, length = sample_size)

        # Generate Z based on the process described
        if (t == 1) {
            dta$Z[row_index] <- rnorm(sample_size)
        } else {
            dta$Z[row_index] <- dta$Y[row_index_lag1] + rnorm(sample_size)
        }

        # Set fixed probability for treatment assignment
        dta$prob_X[row_index] <- rep(prob_x, sample_size)

        # Treatment assignment
        dta$X[row_index] <- rbinom(sample_size, 1, dta$prob_X[row_index])

        # Error term
        dta$e[row_index] <- rnorm(sample_size, mean = 0, sd = sigma_e)

        # Treatment effect (delta)
        dta$delta[row_index] <- gamma_10 + gamma_11 * dta$Z[row_index] + u_2i

        # Outcome Y
        dta$Y[row_index] <- gamma_00 + gamma_01 * dta$Z[row_index] + u_0i +
                            dta$X[row_index] * dta$delta[row_index] + dta$e[row_index]

        # Store random effects
        dta$u0[row_index] <- u_0i
        dta$u2[row_index] <- u_2i

        # Update row index for lagged Y
        row_index_lag1 <- row_index
    }

    return(dta)
}

# Run the data generation function
set.seed(123987)
data <- dgm1(sample_size = 100000, total_T = 10)
saveRDS(data, "data/Qian_GM1_data.rds")
```

```r
# load data
data <- readRDS("data/Qian_GM1_data.rds")

# Fit models
gee_indep <- geeglm(Y ~ Z + X + Z*X, id = userid, data = data, corstr = "independence")
gee_exch <- geeglm(Y ~ Z + X + Z*X, id = userid, data = data, corstr = "exchangeable")
gee_ar1 <- geeglm(Y ~ Z + X + Z*X, id = userid, data = data, corstr = "ar1")
mlm_mle <- lmer(Y ~ Z + X + Z*X + (1 + X | userid), data = data, REML = FALSE)

gamma_10 <- 1    # Fixed effect of treatment (X)
treatment_effect <- c(coef(gee_indep)[3], coef(gee_exch)[3], coef(gee_ar1)[3], fixef(mlm_mle)[

df_treatment <- data.frame(row.names = c("GEE (Indep)", "GEE (Exch)", "GEE (AR1)", "MLM (MLE)"
readRDS("data/Qian_GM1_results.rds")
```

Generating Model 1 - Treatment Effect Estimates

	treatment_effect	true_effect
GEE (Indep)	1.093	1
GEE (Exch)	1.089	1
GEE (AR1)	0.993	1
MLM (MLE)	1.006	1