Stat 571 HW4

Dongyang Wang

2023-02-17

```
rm(list=ls())
set.seed(42)
```

Question 1

Question 1.1

```
library(lme4)
```

```
## Loading required package: Matrix
```

```
sixcity <- read.table("sixcity.dat", header = F)
colnames(sixcity) = c("wheezing", "id", "age", "smoking")
glmm <- glmer(wheezing ~ smoking*age + (1|id), family=binomial, data=sixcity)
summary(glmm)$coefficients[,c(1,2,4)]
```

```
##                Estimate Std. Error     Pr(>|z|)
## (Intercept) -3.4017099 0.27883736 3.122768e-34
## smoking      0.4782426 0.29925558 1.100191e-01
## age         -0.2170390 0.08678107 1.238450e-02
## smoking:age  0.1046484 0.13911975 4.519198e-01
```

Age is significant. Smoking is not nor the interaction. log odds of wheezing is expected to decrease by 0.2170390 for 1 year increase in age.

Question 1.2

```
library(gee)
gee_exch <- gee(wheezing ~ age * smoking,
        id = id, corstr = "exchangeable",
        family= binomial, data = sixcity)
```

```
## Beginning Cgee S-function, @(#) geeformula.q 4.13 98/01/27
```

```
## running glm to get initial regression estimate
```

```
## (Intercept)         age     smoking age:smoking
##  -1.9008426  -0.1412531   0.3139540   0.0708441
```

```
summary(gee_exch)$coefficients[,c(1,4)]
```

```
##                Estimate Robust S.E.
## (Intercept) -1.90049539  0.11908696
## age         -0.14123592  0.05820089
## smoking      0.31382583  0.18784180
## age:smoking  0.07083185  0.08827886
```

```
res = cbind(summary(glmm)$coefficients[,c(1,2)], summary(gee_exch)$coefficients[,c(1,4)])
colnames(res) <- c("GLMM Estimate", "GLMM SE", "GEE Estimate", "GEE SE (Robust)")
res
```

```
##             GLMM Estimate    GLMM SE GEE Estimate GEE SE (Robust)
## (Intercept)    -3.4017099 0.27883736  -1.90049539      0.11908696
## smoking         0.4782426 0.29925558  -0.14123592      0.05820089
## age            -0.2170390 0.08678107   0.31382583      0.18784180
## smoking:age     0.1046484 0.13911975   0.07083185      0.08827886
```

It appears that the estimates are wildly different, as well as their standard errors. The interpretations are different in the sense that the GLMM needs to take into account the random effect on each observation. Also, note that in the setup that the GLMM is based on conditional expectation while the GEE is based on marginal model.

Question 2

The data generating process is similar to HW1. I recycled some of the code. To reiterate the logic: Under the random intercept model, since $var(Y) = 1 = \theta + \sigma^2$ and $corr(Y_{ij}, Y_{ik}) = \rho = \frac{\theta}{\theta+\sigma^2}$, we solve the equations and get $\theta = \rho$ and $\sigma^2 = 1 - \rho$. In this way, we can generate the x, e, b seperately and use a linear relationship we choose to generate the y values, without the need to sample y directly but achieving the same results. For the dichotomous outcome, I further follow Maria's approach in the office hour as detailed on Canvas for data generation.

```
library(geepack)
# Set beta to 0.5 and 1
beta1 = 0.5
beta0 = 1
p = 0.5

# For testing
m = 10
n = 5

gen.one <- function(m,n){

  total = m*n

  # Generate the variables
  x = rnorm(total, 0, 1)
  b = rep(rnorm(m, mean = 0, sd = sqrt(p)),n)
  e = rnorm(total, mean = 0, sd = sqrt(1-p))
  expo = beta0 + beta1*x + b + e
  prob_y = exp(expo)/(1+exp(expo))
  temp_y = runif(total)
  y = ifelse(temp_y >= prob_y, 0, 1)

  # GLMM
  lmm <- glmer(y ~ x + (1|b), family=binomial)

  # GEE
  gee = geeglm(y~ x, id = b, corstr = "exchangeable")

  # Estimate variance for efficiency
  lmm_var0 = vcov(lmm)[1,1]
```

```
  lmm_var1 = vcov(lmm)[2,2]
  gee_var0 = vcov(gee)[1,1]
  gee_var1 = vcov(gee)[2,2]

  # Estimate coefficients
  lmm_coef0 = fixef(lmm)[1]
  lmm_coef1 = fixef(lmm)[2]
  gee_coef0 = coef(gee)[1]
  gee_coef1 = coef(gee)[2]

  # Estimate bias
  lmm_bias0 = lmm_coef0 - beta0
  lmm_bias1 = lmm_coef1 - beta1
  gee_bias0 = gee_coef0 - beta0
  gee_bias1 = gee_coef1 - beta1

  return(data.frame(m = m, n = n,
                    lmm_var0 = lmm_var0, lmm_var1 = lmm_var1,
                    gee_var0 = gee_var0, gee_var1 = gee_var1,
                    lmm_bias0 = lmm_bias0, lmm_bias1 = lmm_bias1,
                    gee_bias0 = gee_bias0, gee_bias1 = gee_bias1
                    ) )
}
```

```
nrep = 1000

res <- do.call(rbind, lapply(c(1:nrep), function(nrep){
gen.one(m,n)
}))
mean_res <- colMeans(res)

simulation_res = as.data.frame(mean_res)
simulation_res
```

```
##               mean_res
## m         10.000000000
## n          5.000000000
## lmm_var0   0.194575752
## lmm_var1   0.151066777
## gee_var0   0.004021989
## gee_var1   0.003854795
## lmm_bias0  0.012458728
## lmm_bias1  0.027962856
## gee_bias0 -0.309638634
## gee_bias1 -0.414775411
```

As we can easily observe that the variance of the GEE model is significantly lower for both parameters. But the biases seem to be very low for the glmm method.

The merits of the logistic mixed model is that it accounts for the same individuals. On the other hand, GEE does not require likelihood, and we are flexible in choosing the right corrleation structure.

If given more time, we can possibly explore some survival analysis context as well as modifying the number n or the correlation structure.