PBHL-B574_simulation_code

Yao Chen

2023-04-24

Simulation example 1 code

correlated data when $\rho = 0.5$.

```r
library(PGEE) # requiring `mvtnorm` and `MASS` packages
set.seed(12345) # for reproducibility
# correlated normal responses
#True parameter vector

# covariate matrices
ar1_cor <- function(n, rho) {
  exponent <- abs(matrix(1:n - 1, nrow = n, ncol = n, byrow = TRUE) - 
                    (1:n - 1))
  rho^exponent
}
exch_cor <- function(n, rho) {
  mat <- matrix(rho, nrow = n, ncol = n)
  diag(mat) <- 1
  return(mat)
}

# strength of within cluster correlation
rho_values <- c(0.5, 0.8)

# Create the function to simulate data set 
# consisting of N=200 study participants
# with 4 observations per participant
# p = 200 covariates
simulate<-function(N, rho){
  # The number of repeated measurements is 4
  timepoints=4
  # The number of covariates is 200
  p <- 200
  # b_0
  beta0 <- c(2, 3, 1.5, 2, rep(0, p - 4))
  # Simulate errors from MVN
  e_i <- c(sapply(1, function(x) t(mvrnorm(n = N, mu = rep(0,timepoints), 
Sigma = exch_cor(4,rho)))) )
  # Generate ID variable
  id <- c(sapply(1:N, function(x) rep(x, timepoints)))
  # Simulate x_ij
  # x_ij,1 from Bernoulli(0.5)
  x_1 <- c(sapply(1: N, function(x) rep(t(rbinom(1, 1, 0.5)),
                                        timepoints)))
  # x_ij,2-200 from MVN
  x_2 <- matrix(sapply(1:(p-1), function(x) t(mvrnorm(n = N, mu =
rep(0,timepoints),
                                                      Sigma =
ar1_cor(4,0.5)))), ncol=c(p-1))
  colnames(x_2) <- paste0("x", 2:200)
  X <- cbind(x_1,x_2)
  # Simulate response
  y<- X %*% beta0 +e_i
  # Return data frame
  data.frame(id,y,X)
}
# generate simulation data with correlated normal responses
data_sim1 <- simulate(N=200, rho=rho_values[1])
data_sim2 <- simulate(N=200, rho=rho_values[2])

colnames(data_sim1)[1:10]
head(data_sim1,5)[1:10]

formula <- "y ~.-id"
family <- gaussian(link = "identity")
lambda.vec <- seq(0.1,0.9,0.1)
# find the optimum lambda
cv_sim1 <- CVfit(formula = formula, id = id, data = data_sim1, family =
family, scale.fix = TRUE,
            scale.value = 1, fold = 4, lambda.vec = lambda.vec, pindex =
c(1,2), eps = 10^-6,
            maxiter = 30, tol = 10^-3)
print(cv_sim1) # 0.4 is the best lambda
names(cv_sim1)
cv_sim1$lam.opt
cv_sim2 <- CVfit(formula = formula, id = id, data = data_sim2, family =
family, scale.fix = TRUE,
            scale.value = 1, fold = 4, lambda.vec = lambda.vec, pindex =
c(1,2), eps = 10^-6,
            maxiter = 30, tol = 10^-3)
print(cv_sim2) # 0.5 is the best lambda
names(cv_sim2)
cv_sim2$lam.opt

# number of generated data set for each set up = 100
nsim=100
b0 <- c(2, 3, 1.5, 2, rep(0, 200 - 4))

# Perform the simulation - PGEE and rho=0.5
# Perform the simulation - rho=0.5
for(i in 1:nsim){
  print(i)
  N=200
  rho=rho_values[1] # rho=0.5, 0.8
  data<-simulate(N, rho)
  myfit1_indp <- PGEE(formula = formula, id = id, data = data, na.action =
NULL,
                      family = family, corstr = "independence", Mv = NULL,
                      beta_int = c(rep(0,dim(data)[2]-1)), R = NULL,
scale.fix = TRUE,
                      scale.value = 1, lambda = cv_sim1$lam.opt, pindex =
c(1,2), eps = 10^-6,
                      maxiter = 30, tol = 10^-6, silent = TRUE)
  myfit1_exch <- PGEE(formula = formula, id = id, data = data, na.action =
NULL,
                      family = family, corstr = "exchangeable", Mv = NULL,
                      beta_int = c(rep(0,dim(data)[2]-1)), R = NULL,
scale.fix = TRUE,
                      scale.value = 1, lambda = cv_sim1$lam.opt, pindex =
c(1,2), eps = 10^-6,
                      maxiter = 30, tol = 10^-6, silent = TRUE)
  myfit1_ar1 <- PGEE(formula = formula, id = id, data = data, na.action =
NULL,
                     family = family, corstr = "AR-1", Mv = NULL,
                     beta_int = c(rep(0,dim(data)[2]-1)), R = NULL, scale.fix
= TRUE,
                     scale.value = 1, lambda = cv_sim1$lam.opt, pindex =
c(1,2), eps = 10^-6,
                     maxiter = 30, tol = 10^-6, silent = TRUE)
  pgee_b1 <- coef(myfit1_indp)
  pgee_b2 <- coef(myfit1_exch)
  pgee_b3 <- coef(myfit1_ar1)
  myfit2_indp <- MGEE(formula = formula, id = id, data = data, na.action =
NULL,
                      family = family, corstr = "independence", Mv = NULL,
                      beta_int = c(rep(0,dim(data)[2]-1)), R = NULL,
scale.fix = TRUE,
                      scale.value = 1,
                      maxiter = 30, tol = 10^-6, silent = TRUE)
  myfit2_exch <- MGEE(formula = formula, id = id, data = data, na.action =
NULL,
                      family = family, corstr = "exchangeable", Mv = NULL,
                      beta_int = c(rep(0,dim(data)[2]-1)), R = NULL,
scale.fix = TRUE,
                      scale.value = 1,
                      maxiter = 30, tol = 10^-6, silent = TRUE)
  myfit2_ar1 <- MGEE(formula = formula, id = id, data = data, na.action =
NULL,
                     family = family, corstr = "AR-1", Mv = NULL,
                     beta_int = c(rep(0,dim(data)[2]-1)), R = NULL, scale.fix
= TRUE,
                     scale.value = 1,
                     maxiter = 30, tol = 10^-6, silent = TRUE)
  gee_b1 <- coef(myfit2_indp)
  gee_b2 <- coef(myfit2_exch)
  gee_b3 <- coef(myfit2_ar1)
  myfit3_indp <- MGEE(formula = formula, id = id, data = data, na.action =
NULL,
                      family = family, corstr = "independence", Mv = NULL,
                      beta_int = c(0, b0), R = NULL, scale.fix = TRUE,
                      scale.value = 1,
                      maxiter = 30, tol = 10^-6, silent = TRUE)
  myfit3_exch <- MGEE(formula = formula, id = id, data = data, na.action =
NULL,
                      family = family, corstr = "exchangeable", Mv = NULL,
                      beta_int = c(0, b0), R = NULL, scale.fix = TRUE,
                      scale.value = 1,
                      maxiter = 30, tol = 10^-6, silent = TRUE)
  myfit3_ar1 <- MGEE(formula = formula, id = id, data = data, na.action =
NULL,
                     family = family, corstr = "AR-1", Mv = NULL,
                     beta_int = c(0, b0), R = NULL, scale.fix = TRUE,
                     scale.value = 1,
                     maxiter = 30, tol = 10^-6, silent = TRUE)
  oracle_b1 <- coef(myfit3_indp)
  oracle_b2 <- coef(myfit3_exch)
  oracle_b3 <- coef(myfit3_ar1)
  if(i==1){
    pgee1_beta1<-pgee_b1
    pgee1_beta2<-pgee_b2
    pgee1_beta3<-pgee_b3
    gee1_beta1<-gee_b1
    gee1_beta2<-gee_b2
    gee1_beta3<-gee_b3
    oracle1_beta1<-oracle_b1
    oracle1_beta2<-oracle_b2
    oracle1_beta3<-oracle_b3
  } else {
    pgee1_beta1<-rbind(pgee1_beta1,pgee_b1)
    pgee1_beta2<-rbind(pgee1_beta2,pgee_b2)
    pgee1_beta3<-rbind(pgee1_beta3,pgee_b3)
    gee1_beta1<-rbind(gee1_beta1,gee_b1)
    gee1_beta2<-rbind(gee1_beta2,gee_b2)
    gee1_beta3<-rbind(gee1_beta3,gee_b3)
    oracle1_beta1<-rbind(oracle1_beta1,oracle_b1)
    oracle1_beta2<-rbind(oracle1_beta2,oracle_b2)
    oracle1_beta3<-rbind(oracle1_beta3,oracle_b3)
  }}

# MSE when rho=0.5
pgee1_mse <- c(mean(colSums((pgee1_beta1[,-1] - b0)^2))/nrow(pgee1_beta1),
               mean(colSums((pgee1_beta2[,-1] - b0)^2))/nrow(pgee1_beta2),
               mean(colSums((pgee1_beta3[,-1] - b0)^2))/nrow(pgee1_beta3))
names(pgee1_mse) <- c("indp", "exch", "ar1")
gee1_mse <- c(mean(colSums((gee1_beta1[,-1] - b0)^2))/nrow(gee1_beta1),
              mean(colSums((gee1_beta2[,-1] - b0)^2))/nrow(gee1_beta2),
              mean(colSums((gee1_beta3[,-1] - b0)^2))/nrow(gee1_beta3))
names(gee1_mse) <- c("indp", "exch", "ar1")
oracle1_mse <- c(mean(colSums((oracle1_beta1[,-1] -
b0)^2))/nrow(oracle1_beta1),
                 mean(colSums((oracle1_beta2[,-1] -
b0)^2))/nrow(oracle1_beta2),
                 mean(colSums((oracle1_beta3[,-1] -
b0)^2))/nrow(oracle1_beta3))
names(oracle1_mse) <- c("indp", "exch", "ar1")

# U/O/EXACT when rho=0.5
num_indp_pgee1 <- apply(pgee1_beta1[,-1], 1, function(x) sum(abs(x) > 10^-3))
num_exch_pgee1 <- apply(pgee1_beta2[,-1], 1, function(x) sum(abs(x) > 10^-3))
num_ar1_pgee1 <- apply(pgee1_beta3[,-1], 1, function(x) sum(abs(x) > 10^-3))
prop_pgee1_indp <- c(mean(num_indp_pgee1 < 4), mean(num_indp_pgee1 > 4),
mean(num_indp_pgee1 == 4))
prop_pgee1_exch <- c(mean(num_exch_pgee1 < 4), mean(num_exch_pgee1 > 4),
mean(num_exch_pgee1 == 4))
prop_pgee1_ar1 <- c(mean(num_ar1_pgee1 < 4), mean(num_ar1_pgee1 > 4),
mean(num_ar1_pgee1 == 4))
prop_pgee1 <- rbind(prop_pgee1_indp, prop_pgee1_exch, prop_pgee1_ar1)
colnames(prop_pgee1) <- c("U", "O", "EXACT")
rownames(prop_pgee1) <- c("pgee1_indp", "pgee1_exch", "pgee1_ar1")
prop_pgee1
num_indp_gee1 <- apply(gee1_beta1[,-1], 1, function(x) sum(abs(x) > 10^-3))
num_exch_gee1 <- apply(gee1_beta2[,-1], 1, function(x) sum(abs(x) > 10^-3))
num_ar1_gee1 <- apply(gee1_beta3[,-1], 1, function(x) sum(abs(x) > 10^-3))
prop_gee1_indp <- c(mean(num_indp_gee1 < 4), mean(num_indp_gee1 > 4),
mean(num_indp_gee1 == 4))
prop_gee1_exch <- c(mean(num_exch_gee1 < 4), mean(num_exch_gee1 > 4),
mean(num_exch_gee1 == 4))
prop_gee1_ar1 <- c(mean(num_ar1_gee1 < 4), mean(num_ar1_gee1 > 4),
mean(num_ar1_gee1 == 4))
prop_gee1 <- rbind(prop_gee1_indp, prop_gee1_exch, prop_gee1_ar1)
colnames(prop_gee1) <- c("U", "O", "EXACT")
rownames(prop_gee1) <- c("gee1_indp", "gee1_exch", "gee1_ar1")
prop_gee1
# TP  when rho=0.5
mean(apply(pgee1_beta1[,2:5], 1, function(x) sum(abs(x) > 10^-3)))
mean(apply(pgee1_beta2[,2:5], 1, function(x) sum(abs(x) > 10^-3)))
mean(apply(pgee1_beta3[,2:5], 1, function(x) sum(abs(x) > 10^-3)))
mean(apply(gee1_beta1[,2:5], 1, function(x) sum(abs(x) > 10^-3)))
mean(apply(gee1_beta2[,2:5], 1, function(x) sum(abs(x) > 10^-3)))
mean(apply(gee1_beta3[,2:5], 1, function(x) sum(abs(x) > 10^-3)))
# FP  when rho=0.5
```

```
mean(apply(pgee1_beta1[,-c(1:5)], 1, function(x) sum(abs(x) > 10^-3)))
mean(apply(pgee1_beta2[,-c(1:5)], 1, function(x) sum(abs(x) > 10^-3)))
mean(apply(pgee1_beta3[,-c(1:5)], 1, function(x) sum(abs(x) > 10^-3)))
mean(apply(gee1_beta1[,-c(1:5)], 1, function(x) sum(abs(x) > 10^-3)))
mean(apply(gee1_beta2[,-c(1:5)], 1, function(x) sum(abs(x) > 10^-3)))
mean(apply(gee1_beta3[,-c(1:5)], 1, function(x) sum(abs(x) > 10^-3)))
```

correlated data when $\rho = 0.8$.

```
# Perform the simulation - rho=0.8
for(i in 1:nsim){
  print(i)
  N=200
  rho=rho_values[2] # rho=0.5, 0.8
  data<-simulate(N, rho)
  myfit1_indp <- PGEE(formula = formula, id = id, data = data, na.action =
NULL,
                      family = family, corstr = "independence", Mv = NULL,
                      beta_int = c(rep(0,dim(data)[2]-1)), R = NULL,
scale.fix = TRUE,
                      scale.value = 1, lambda = cv_sim1$lam.opt, pindex =
c(1,2), eps = 10^-6,
                      maxiter = 30, tol = 10^-6, silent = TRUE)
  myfit1_exch <- PGEE(formula = formula, id = id, data = data, na.action =
NULL,
                      family = family, corstr = "exchangeable", Mv = NULL,
                      beta_int = c(rep(0,dim(data)[2]-1)), R = NULL,
scale.fix = TRUE,
                      scale.value = 1, lambda = cv_sim1$lam.opt, pindex =
c(1,2), eps = 10^-6,
                      maxiter = 30, tol = 10^-6, silent = TRUE)
  myfit1_ar1 <- PGEE(formula = formula, id = id, data = data, na.action =
NULL,
                     family = family, corstr = "AR-1", Mv = NULL,
                     beta_int = c(rep(0,dim(data)[2]-1)), R = NULL, scale.fix
= TRUE,
                     scale.value = 1, lambda = cv_sim1$lam.opt, pindex =
c(1,2), eps = 10^-6,
                     maxiter = 30, tol = 10^-6, silent = TRUE)
  pgee_b1 <- coef(myfit1_indp)
  pgee_b2 <- coef(myfit1_exch)
  pgee_b3 <- coef(myfit1_ar1)
  myfit2_indp <- MGEE(formula = formula, id = id, data = data, na.action =
NULL,
                      family = family, corstr = "independence", Mv = NULL,
                      beta_int = c(rep(0,dim(data)[2]-1)), R = NULL,
scale.fix = TRUE,
                      scale.value = 1,
                      maxiter = 30, tol = 10^-6, silent = TRUE)
  myfit2_exch <- MGEE(formula = formula, id = id, data = data, na.action =
NULL,
                      family = family, corstr = "exchangeable", Mv = NULL,
                      beta_int = c(rep(0,dim(data)[2]-1)), R = NULL,
scale.fix = TRUE,
                      scale.value = 1,
                      maxiter = 30, tol = 10^-6, silent = TRUE)
  myfit2_ar1 <- MGEE(formula = formula, id = id, data = data, na.action =
NULL,
                     family = family, corstr = "AR-1", Mv = NULL,
                     beta_int = c(rep(0,dim(data)[2]-1)), R = NULL, scale.fix
= TRUE,
                     scale.value = 1,
                     maxiter = 30, tol = 10^-6, silent = TRUE)
  gee_b1 <- coef(myfit2_indp)
  gee_b2 <- coef(myfit2_exch)
  gee_b3 <- coef(myfit2_ar1)
  myfit3_indp <- MGEE(formula = formula, id = id, data = data, na.action =
NULL,
                      family = family, corstr = "independence", Mv = NULL,
                      beta_int = c(0, b0), R = NULL, scale.fix = TRUE,
                      scale.value = 1,
                      maxiter = 30, tol = 10^-6, silent = TRUE)
  myfit3_exch <- MGEE(formula = formula, id = id, data = data, na.action =
NULL,
                      family = family, corstr = "exchangeable", Mv = NULL,
                      beta_int = c(0, b0), R = NULL, scale.fix = TRUE,
                      scale.value = 1,
                      maxiter = 30, tol = 10^-6, silent = TRUE)
  myfit3_ar1 <- MGEE(formula = formula, id = id, data = data, na.action =
NULL,
                     family = family, corstr = "AR-1", Mv = NULL,
                     beta_int = c(0, b0), R = NULL, scale.fix = TRUE,
                     scale.value = 1,
                     maxiter = 30, tol = 10^-6, silent = TRUE)
  oracle_b1 <- coef(myfit3_indp)
  oracle_b2 <- coef(myfit3_exch)
  oracle_b3 <- coef(myfit3_ar1)
  if(i==1){
    pgee2_beta1<-pgee_b1
    pgee2_beta2<-pgee_b2
    pgee2_beta3<-pgee_b3
    gee2_beta1<-gee_b1
    gee2_beta2<-gee_b2
    gee2_beta3<-gee_b3
    oracle2_beta1<-oracle_b1
    oracle2_beta2<-oracle_b2
    oracle2_beta3<-oracle_b3
  } else {
    pgee2_beta1<-rbind(pgee2_beta1,pgee_b1)
    pgee2_beta2<-rbind(pgee2_beta2,pgee_b2)
    pgee2_beta3<-rbind(pgee2_beta3,pgee_b3)
    gee2_beta1<-rbind(gee2_beta1,gee_b1)
    gee2_beta2<-rbind(gee2_beta2,gee_b2)
    gee2_beta3<-rbind(gee2_beta3,gee_b3)
    oracle2_beta1<-rbind(oracle2_beta1,oracle_b1)
    oracle2_beta2<-rbind(oracle2_beta2,oracle_b2)
    oracle2_beta3<-rbind(oracle2_beta3,oracle_b3)
  }}

#######################  test: initial beta values = 0  ####################
############################################################################
############################################################################
############################################################################
############################################################################
for(i in 1:2){
  print(i)
  N=200
  rho=rho_values[1] # rho=0.5, 0.8
  data<-simulate(N, rho)
  myfit1_indp <- PGEE(formula = formula, id = id, data = data, na.action =
NULL,
                      family = family, corstr = "independence", Mv = NULL,
                      beta_int = NULL, R = NULL, scale.fix = TRUE,
                      scale.value = 1, lambda = cv_sim1$lam.opt, pindex =
c(1,2), eps = 10^-6,
                      maxiter = 30, tol = 10^-6, silent = TRUE)
  myfit1_exch <- PGEE(formula = formula, id = id, data = data, na.action =
NULL,
                      family = family, corstr = "exchangeable", Mv = NULL,
                      beta_int = NULL, R = NULL, scale.fix = TRUE,
                      scale.value = 1, lambda = cv_sim1$lam.opt, pindex =
c(1,2), eps = 10^-6,
                      maxiter = 30, tol = 10^-6, silent = TRUE)
  myfit1_ar1 <- PGEE(formula = formula, id = id, data = data, na.action =
NULL,
                     family = family, corstr = "AR-1", Mv = NULL,
                     beta_int = NULL, R = NULL, scale.fix = TRUE,
                     scale.value = 1, lambda = cv_sim1$lam.opt, pindex =
c(1,2), eps = 10^-6,
                     maxiter = 30, tol = 10^-6, silent = TRUE)
  pgee_b1 <- coef(myfit1_indp)
  pgee_b2 <- coef(myfit1_exch)
  pgee_b3 <- coef(myfit1_ar1)
  myfit2_indp <- MGEE(formula = formula, id = id, data = data, na.action =
NULL,
                      family = family, corstr = "independence", Mv = NULL,
                      beta_int = NULL, R = NULL, scale.fix = TRUE,
                      scale.value = 1,
                      maxiter = 30, tol = 10^-6, silent = TRUE)
  myfit2_exch <- MGEE(formula = formula, id = id, data = data, na.action =
NULL,
                      family = family, corstr = "exchangeable", Mv = NULL,
                      beta_int = NULL, R = NULL, scale.fix = TRUE,
                      scale.value = 1,
                      maxiter = 30, tol = 10^-6, silent = TRUE)
  myfit2_ar1 <- MGEE(formula = formula, id = id, data = data, na.action =
NULL,
                     family = family, corstr = "AR-1", Mv = NULL,
                     beta_int = NULL,
                     R = NULL, scale.fix = TRUE,
                     scale.value = 1,
                     maxiter = 30, tol = 10^-6, silent = TRUE)
  gee_b1 <- coef(myfit2_indp)
  gee_b2 <- coef(myfit2_exch)
  gee_b3 <- coef(myfit2_ar1)
  myfit3_indp <- MGEE(formula = formula, id = id, data = data, na.action =
NULL,
                      family = family, corstr = "independence", Mv = NULL,
                      beta_int = c(b0), R = NULL, scale.fix = TRUE,
                      scale.value = 1,
                      maxiter = 30, tol = 10^-6, silent = TRUE)
  myfit3_exch <- MGEE(formula = formula, id = id, data = data, na.action =
NULL,
                      family = family, corstr = "exchangeable", Mv = NULL,
                      beta_int = c(b0), R = NULL, scale.fix = TRUE,
                      scale.value = 1,
                      maxiter = 30, tol = 10^-6, silent = TRUE)
  myfit3_ar1 <- MGEE(formula = formula, id = id, data = data, na.action =
NULL,
                     family = family, corstr = "AR-1", Mv = NULL,
                     beta_int = c(b0), R = NULL, scale.fix = TRUE,
                     scale.value = 1,
                     maxiter = 30, tol = 10^-6, silent = TRUE)
  oracle_b1 <- coef(myfit3_indp)
  oracle_b2 <- coef(myfit3_exch)
  oracle_b3 <- coef(myfit3_ar1)
  if(i==1){
    pgee1_beta1<-pgee_b1
    pgee1_beta2<-pgee_b2
    pgee1_beta3<-pgee_b3
    gee1_beta1<-gee_b1
    gee1_beta2<-gee_b2
    gee1_beta3<-gee_b3
    oracle1_beta1<-oracle_b1
    oracle1_beta2<-oracle_b2
    oracle1_beta3<-oracle_b3
  } else {
    pgee1_beta1<-rbind(pgee1_beta1,pgee_b1)
    pgee1_beta2<-rbind(pgee1_beta2,pgee_b2)
    pgee1_beta3<-rbind(pgee1_beta3,pgee_b3)
    gee1_beta1<-rbind(gee1_beta1,gee_b1)
    gee1_beta2<-rbind(gee1_beta2,gee_b2)
    gee1_beta3<-rbind(gee1_beta3,gee_b3)
    oracle1_beta1<-rbind(oracle1_beta1,oracle_b1)
    oracle1_beta2<-rbind(oracle1_beta2,oracle_b2)
    oracle1_beta3<-rbind(oracle1_beta3,oracle_b3)
  }}
```

real-world data

```
library(PGEE) # requiring `mvtnorm` and `MASS` packages
# load data
data(yeastG1)
data = yeastG1
# get the column names
colnames(data)[1:9]
# see some portion of yeast G1 data
head(data,5)[1:9]
# define the input arguments
formula <- "y ~.-id"
family <- gaussian(link = "identity")
lambda.vec <- seq(0.01,0.2,0.01)
# find the optimum lambda
cv <- CVfit(formula = formula, id = id, data = data, family = family, 
scale.fix = TRUE,
            scale.value = 1, fold = 4, lambda.vec = lambda.vec, pindex = 
c(1,2), eps = 10^-6,
            maxiter = 30, tol = 10^-6)
# print the results
print(cv)
# see the returned values by CVfit
names(cv)
# get the optimum lambda
cv$lam.opt
# fit the PGEE model
myfit1 <- PGEE(formula = formula, id = id, data = data, na.action = NULL,
               family = family, corstr = "independence", Mv = NULL,
               beta_int = c(rep(0,dim(data)[2]-1)), R = NULL, scale.fix = 
TRUE,
               scale.value = 1, lambda = cv$lam.opt, pindex = c(1,2), eps = 
10^-6,
               maxiter = 30, tol = 10^-6, silent = TRUE)
# get the values returned by myfit object
names(myfit1)

# see a portion of the results returned by coef(summary(myfit1))
head(coef(summary(myfit1)),7)

# see the variables which have non-zero coefficients
index1 <- which(abs(coef(summary(myfit1))[,"Estimate"]) > 10^-3)
names(abs(coef(summary(myfit1))[index1,"Estimate"]))
# see the PGEE summary statistics of these non-zero variables
```

```r
coef(summary(myfit1))[index1,]
# fit the GEE model
myfit2 <- MGEE(formula = formula, id = id, data = data, na.action = NULL,
               family = family, corstr = "independence", Mv = NULL,
               beta_int = c(rep(0,dim(data)[2]-1)), R = NULL, scale.fix =
TRUE,
               scale.value = 1, maxiter = 30, tol = 10^-6, silent = TRUE)
# see the significantly associated TFs in PGEE analysis
names(which(abs(coef(summary(myfit1))[index1,"Robust z"]) > 1.96))

# see the significantly associated TFs in GEE analysis
names(which(abs(coef(summary(myfit2))[,"Robust z"]) > 1.96))
```