Simulation

Sunsik Kim

Simulation size setting

```
set.seed(10); R = 10 # R : Truncation level
N = 50; T = 4; D = 7
Z = make_Z(rep(T, N))
```

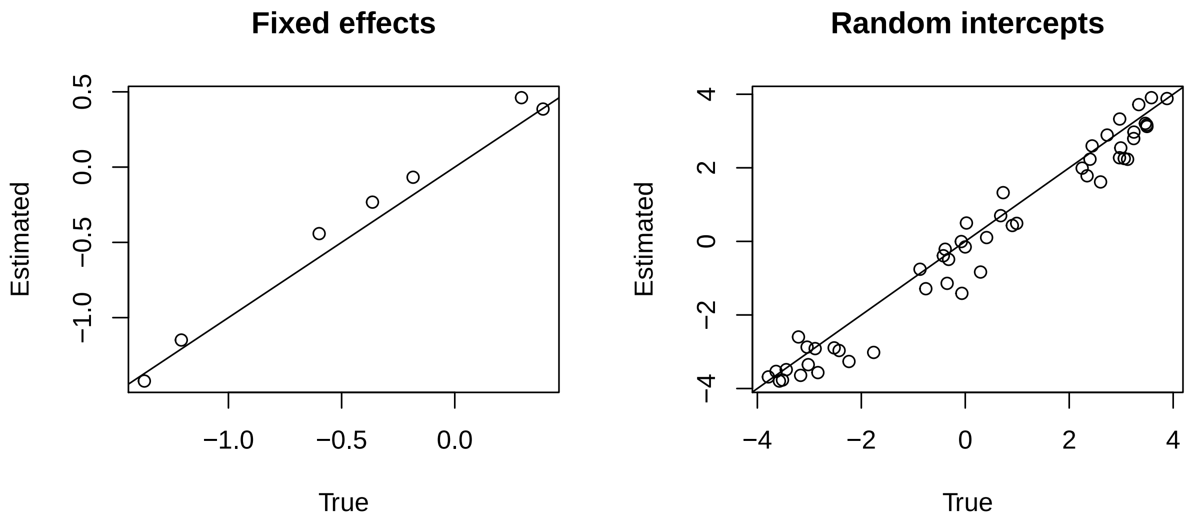
Normal DPM in LMM

Data generation

```
beta = rnorm(D+1)
w = matrix(rnorm(N*T*D), N*T, D)
# Represent random intercept as mixture of 3 normal distributions
assigner = runif(N)
mu1 = 3; mu2 = 0; mu3 = -3
u = rnorm(N, mu1, 0.49)
u[assigner > 0.33] = rnorm(sum(assigner > 0.33), mu2, 0.49)
u[assigner > 0.66] = rnorm(sum(assigner > 0.66), mu3, 0.49)
y = cbind(1, w)%*%beta + Z%*%u + rnorm(nrow(Z))
```

Result presentation

```
## [1] "0.1257 seconds elapsed."
```



Normal DPM in BSA-LMM

Data generation

```r
beta = rnorm(D+1)
w = matrix(rnorm(N*T*D), N*T, D)
f = function(x) -3*(x-1.5)^4
x = 3*runif(N*T); ord = order(x)
```

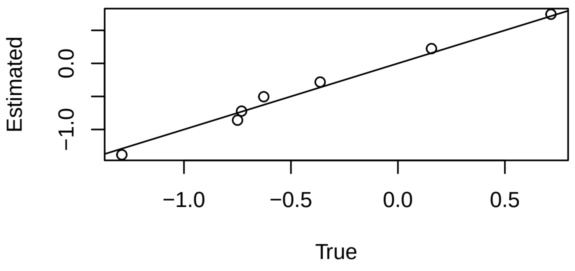
Generating random intercepts

```r
# Ordinary LMM
u_lmm = rnorm(N)
y_lmm = cbind(1, w)%*%beta + Z%*%u_lmm + f(x) + rnorm(nrow(Z))
# DPM : Represent random intercept as mixture of 3 normal distributions
assigner = runif(N)
mu1 = 3; mu2 = 0; mu3 = -3
u_dpm = rnorm(N, mu1)
u_dpm[assigner > 0.33] = rnorm(sum(assigner > 0.33), mu2)
u_dpm[assigner > 0.66] = rnorm(sum(assigner > 0.66), mu3)
y_dpm = cbind(1, w)%*%beta + Z%*%u_dpm + f(x) + rnorm(nrow(Z))
```

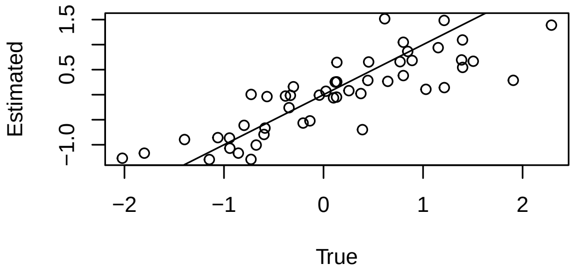
Result presentation

```
## [1] "0.3838 seconds elapsed."
```

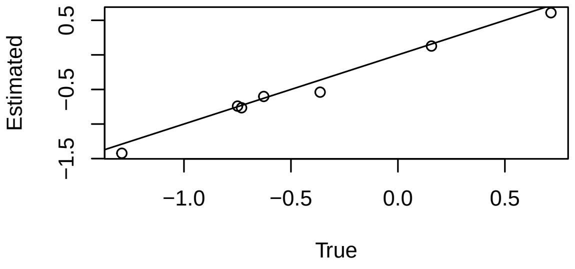
Fixed effects(BSA−LMM)



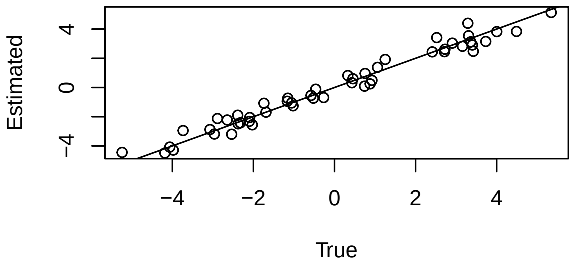
Random intercepts(BSA−LMM)

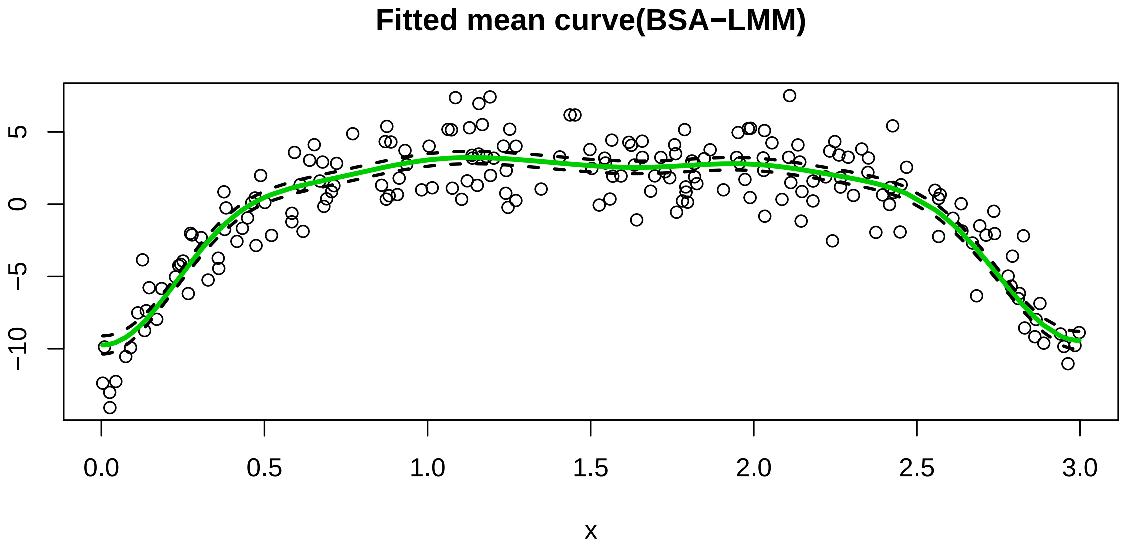


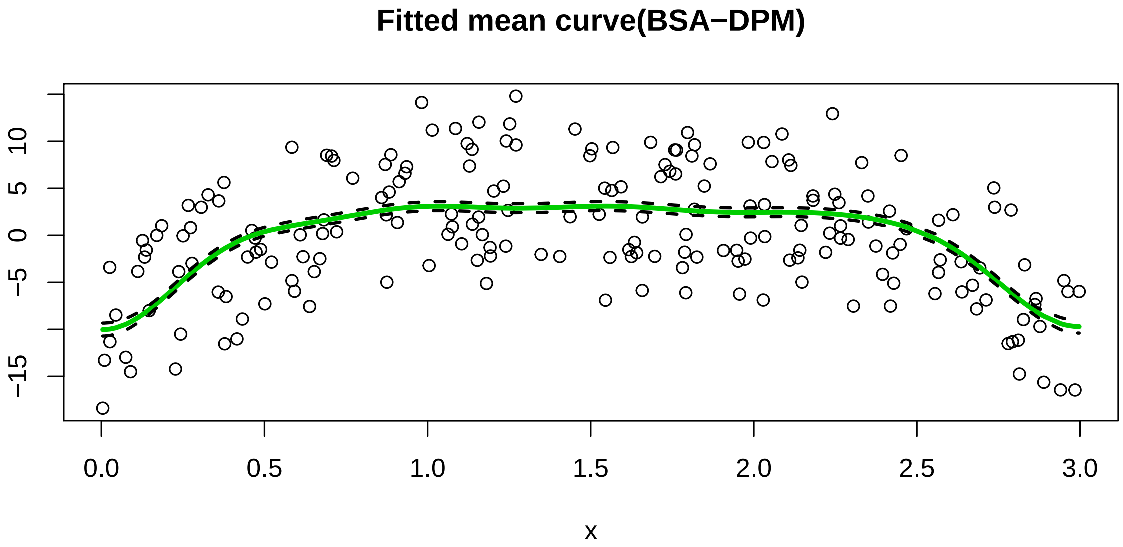
Fixed effects(BSA−DPM)



Random intercepts(BSA−DPM)




Fitted mean curve(BSA−LMM)


Fitted mean curve(BSA−DPM)