Simulations

Model

For each study $s = 1, \cdots, S$, we have observations $i = 1, \cdots, n_s$. Each observation has $p = 1, \cdots, P$ characteristics (for example, gene expressions).

MOM-SS

$$Y = A\alpha^\top + X\beta^\top + F\Phi^\top + E$$

where

$$A = \begin{bmatrix} \mathbb{1}_{n_1} & 0 & \cdots & 0 \\ 0 & \mathbb{1}_{n_2} & \cdots & 0 \\ \vdots & 0 & \cdots & 0 \\ 0 & 0 & \cdots & \mathbb{1}_{n_S} \end{bmatrix}, \quad \alpha^\top = \begin{bmatrix} \alpha_{11} & \cdots & \alpha_{P1} \\ \vdots & \ddots & \vdots \\ \alpha_{1S} & \cdots & \alpha_{PS} \end{bmatrix}$$

BMSFA model

$$Y = F\Phi^\top + L\Lambda^\top + E$$

where

$$L = \text{diag}(L_1, \cdots, L_{M_s}), \quad \Lambda = \text{diag}(\Lambda_1, \cdots, \Lambda_S)$$

$$L_s = \begin{bmatrix} l_{1s} & \cdots & l_{1Ms} \\ \vdots & \ddots & \vdots \\ l_{n_s 1} & \cdots & l_{n_s M_s} \end{bmatrix}, \quad \Lambda_s = \begin{bmatrix} \lambda_{11} & \cdots & \lambda_{1M_s} \\ \vdots & \ddots & \vdots \\ \lambda_{P1} & \cdots & \lambda_{PM_s} \end{bmatrix}$$

Installing different packages

```r
# install.packages("remotes")
# remotes::install_github("rdevito/MSFA")
library(MSFA)

#install.packages("devtools")
#library(devtools)
#devtools::install_github("AleAviP/BFR.BE")
library(BFR.BE)

# Other packages
library(tidyverse)
library(matlab)
```

Senerio 1: 4 studies/batches (S=4), 500 total samples (N=500), 50 genes expressions/nutrition (P=50), 2 known covariates(Q=2), 5 common latent factors(K=5).

Aim

The data is designed for the MOM-SS. But can the BMSFA also work?

- Can the common latent factors in BMSFA captures the effect of the known covariates without supplying the X?
- Can the study specific latent factors in BMSFA captures the effect similar to the random intercepts in the MOM-SS?

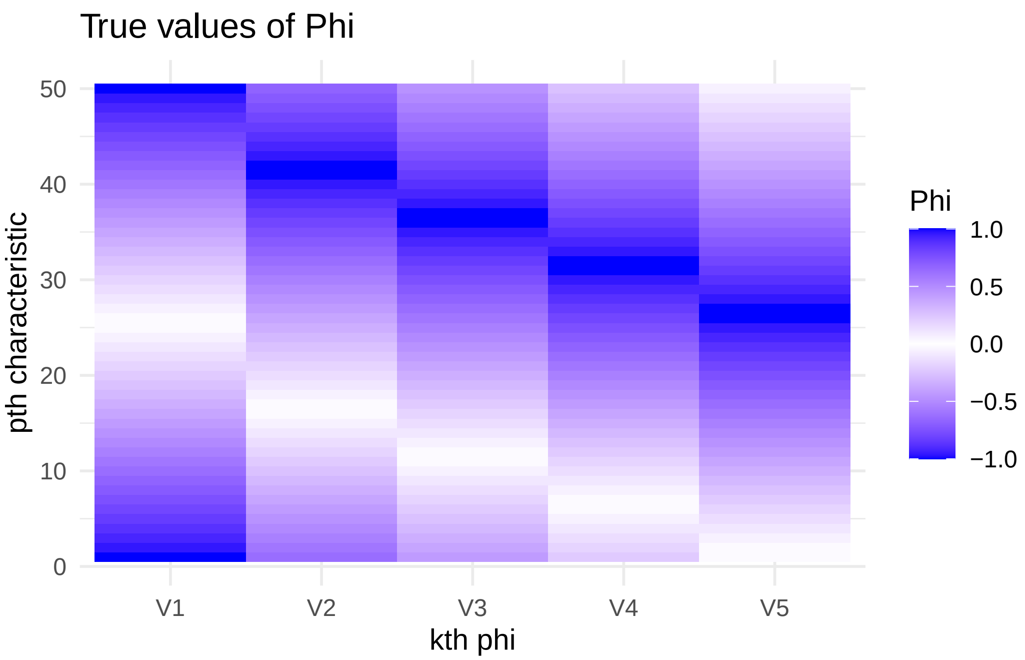
```r
set.seed(5)
S <- 4
N <- 500
P <- 50
Q <- 2
K <- 5

# Number of observations in each study
n_s <- rmultinom(1, N, prob = rep(1/S, S))

# Specify the matrix dimensions
A <- matrix(0, nrow = N, ncol = S)
alpha <- matrix(0, nrow = P, ncol = S)
```

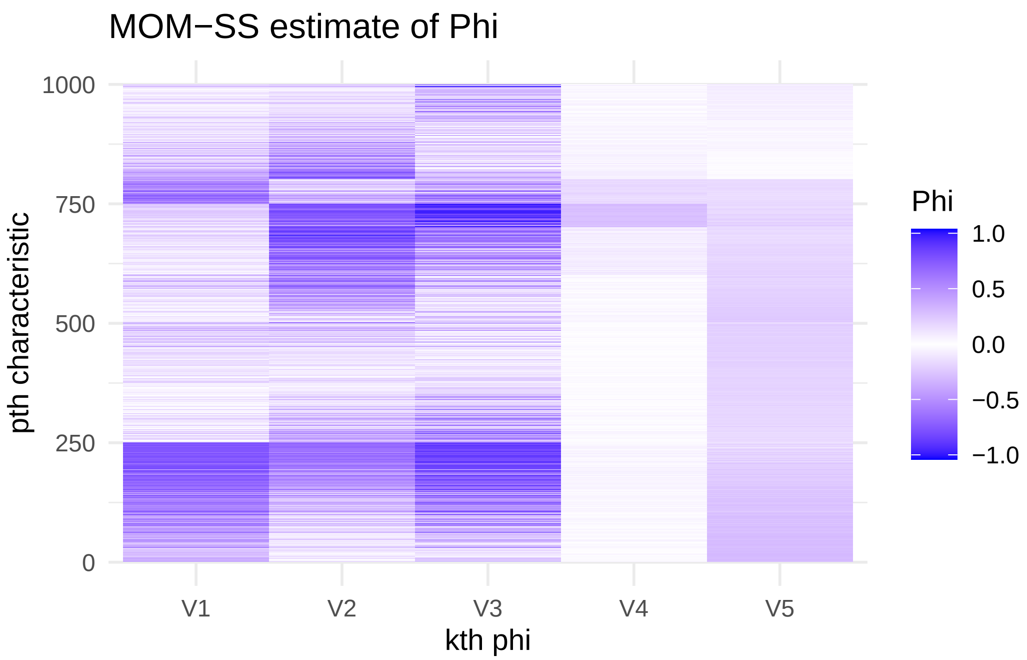
```
X <- matrix(0, nrow = N, ncol = Q)
Beta <- matrix(0, nrow = P, ncol = Q)
F_matrix <- matrix(0, nrow = N, ncol = K)
Phi <- matrix(0, nrow = P, ncol = K)
E <- matrix(0, nrow = N, ncol = P)
Sigma <- matrix(0, nrow = P, ncol = S)


# Generate the true values of the parameters and the variables
# study-specific variables and parameters
for(s in 1:S){
  last_index_of_s <- sum(n_s[1:s])
  if(s==1){
    first_index_of_s <- 1
  }else{
    first_index_of_s <- (sum(n_s[1:(s-1)])+1)
  }

  # study-specific parameters
  alpha[, s] <- rep(s+1, P)
  Sigma[, s] <- s * 0.2

  # Variables
  for(i in first_index_of_s:last_index_of_s){
      A[i, s] <- 1
      X[i,] <- runif(Q, 0, 3)
      E[i,] = mvrnorm(1, rep(0, P),
                 diag(Sigma[, s], nrow = P),
                 tol = 1e-6, empirical = FALSE, EISPACK = FALSE)
  }
}

# common variables and parameters
Beta = matrix(c(rep(-2, round(P/2)), rep(2, P - round(P/2))),
              rnorm(P, 0, 1),
              nrow = P, ncol = Q)
grid <- seq(-1, 1,length.out = P)
Phi <- matrix(grid, nrow = P, ncol = K)
rate<-trunc(P/(K*2))
for(k in 2:K){
  Phi[,k]<-grid[c((k*rate):P, 1:(k*rate-1))]
}
```

```r
F_matrix <- rmvnorm(N, numeric(K), diag(K))

# Generate matrix Y
Y <- A%*%t(alpha)+ X%*%t(Beta) + F_matrix%*%t(Phi)+ E
```

```r
# Do not run it again.
#Fit the MOM-SS model
result_MOMSS_sen1 <- BFR.BE.EM.CV(x = Y, v = X, b = A, q = 5)
```

```r
#Fit the BMSFA model
# Reorgainze the data for BMSFA
Y_BMSFA <- list()
for(s in 1:S){
  Y_BMSFA[[s]] <- Y[which(A[,s]==1),]
}

# Centering the variables
for(s in 1:S){
  Y_BMSFA[[s]] <- scale(Y_BMSFA[[s]], center = TRUE, scale = FALSE)
}
set.seed(10)
result_BMSFA_sen1 <- sp_msfa(Y_BMSFA, k = 5, j_s = c(2,1,1,1), trace = FALSE)
```



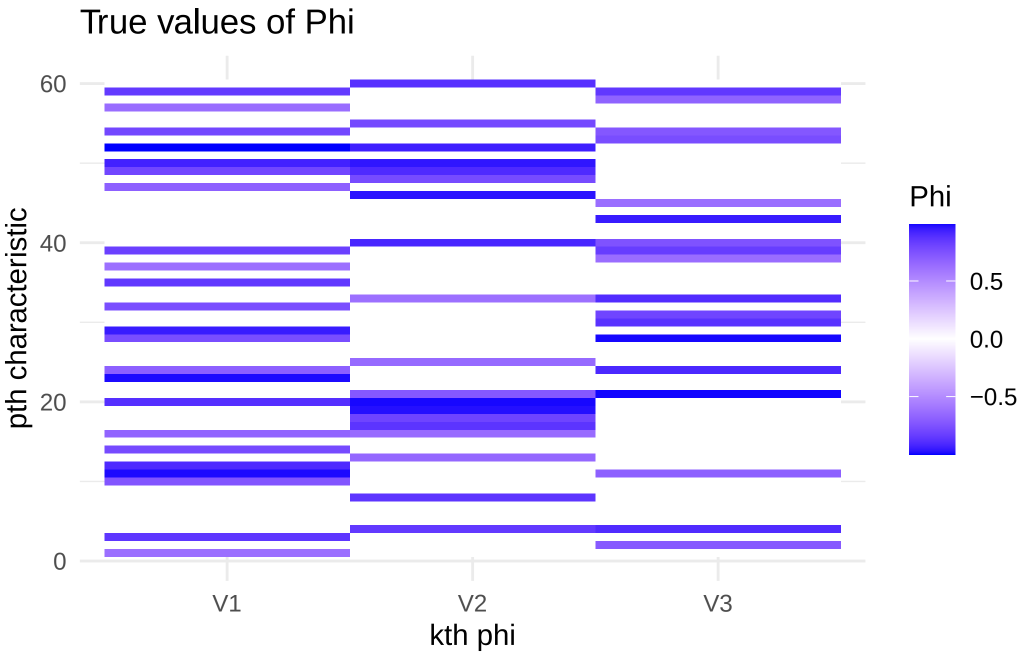


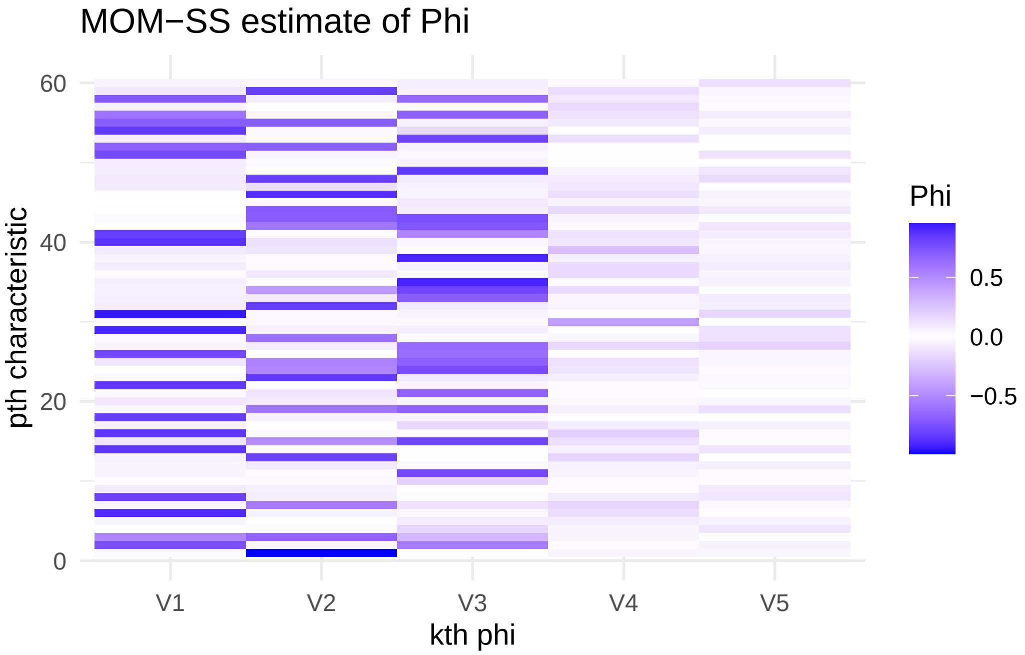
Senerio 2: 2 studies/batches (S=4), 50 samples(N=50), 1000 genes expressions (P=60), no known covariates, 3 common latent factors(K=3), 1 and 4 study-specific factors for the two studies, respectivtly.

```
S <- 4
p <- 60
k <- 3
j_s <- rep(1, 4)
n_s <- c(10, 15, 12, 14)
theta <- rep(0, length = p)
PH <- as.vector(zeros(p, k))
noZEROc <- (p / 3) * k
studyc <- runif(noZEROc, 0.6, 1)
sign <- sample(x = length(studyc), size = (length(studyc) / 2))
studyc[sign] <- studyc[sign] * (-1)
positionc <- sample(x = k * p, size = length(studyc))
PH[positionc] <- studyc
Phi <- matrix(PH, p, k)
L <- noZERO <- study <- position <- Lambda_s <- Psi_s <- Sigma_s <- X_s <- list()

for(s in 1:S){
    L[[s]] <- as.vector(zeros(p, j_s[s]))
    noZERO[[s]] <- (p / 3) * j_s[s]
    study[[s]] <- runif(noZERO[[s]], -1, 1)
    position[[s]] <- sample(x = p * j_s[s], size = length(study[[s]]))
    L[[s]][position[[s]]] <- study[[s]]
    Lambda_s[[s]] <- matrix(L[[s]], p, j_s[s])
    Psi_s[[s]] <- diag(runif(p, 0, 1), p)
    Sigma_s[[s]] <- tcrossprod(Phi)  + tcrossprod(Lambda_s[[s]])  + Psi_s[[s]]
    X_s[[s]] <- mvrnorm(n_s[s], theta, Sigma_s[[s]])}
```

```
#Fit the BMSFA model
set.seed(10)
result_BMSFA_sen2 <- sp_msfa(X_s, k = 10, j_s = c(1, 4), trace = FALSE)
```

```
# Fit MOM-SS model
# transform the data
Y <- do.call(rbind, X_s)
A <- matrix(0, nrow = sum(n_s), ncol = S)
for(s in 1:S){
  if(s==1){
    first_index_of_s <- 1
  }else{
    first_index_of_s <- (sum(n_s[1:(s-1)])+1)
  }
  last_index_of_s <- sum(n_s[1:s])
  A[first_index_of_s:last_index_of_s, s] <- 1
}
result_MOMSS_sen2 <- BFR.BE.EM.CV(x = Y, v = NULL, b = A, q = 5)
```


True values of Phi
pth characteristic
0
20
40
60
V1
V2
V3
kth phi
Phi
0.5
0.0
−0.5


MOM−SS estimate of Phi
pth characteristic
0
20
40
60
V1
V2
V3
V4
V5
kth phi
Phi
0.5
0.0
−0.5