Simulation Study

Kevin Korsurat

2023-06-16

```r
### Function: Simulating the data based on the scenario
f_data_sim <- function(sim_seed, scenario_index){

  ### place for storing result.
  actual_clus <- NULL
  dat <- NULL

  set.seed(sim_seed)

  if(! scenario_index %in% 1:4){
    warning("invalid scenario. we have only 4 scenarios")
  } else {
    if(scenario_index == 1){
      actual_clus <- sample(1:2, 500, replace = TRUE)
      dat <- rnorm(500, c(-5, 5)[actual_clus])
    } else if(scenario_index == 2){
      actual_clus <- sample(1:5, 500, replace = TRUE)
      dat <- rnorm(500, (c(0, 7.5, 15, 25, 35))[actual_clus])
    } else if(scenario_index == 3){
      actual_clus <- sample(1:2, 500, replace = TRUE)
      dat <- rnorm(500, c(-5, 5)[actual_clus], 3)
    } else {
      actual_clus <- sample(1:5, 500, replace = TRUE)
      dat <- rnorm(500, (c(0, 7.5, 15, 25, 35)[actual_clus])/2, 1)
    }
  }

  ### return the simulated data
  result <- data.frame(actual_clus, dat)
  return(result)
}

### Function: Compute average silhouette for k clusters
### https://uc-r.github.io/kmeans_clustering
avg_sil <- function(k, data_clus) {
  km.res <- kmeans(data_clus, centers = k)
  ss <- silhouette(km.res$cluster, dist(data_clus))
  mean(ss[, 3])
}

### Function: Calculate the BIC for EM algorithm
k_EM_BIC <- function(data_clus, k, em_opt){
```

```
  ### Initialize the model
  init_EM <- init.EM(data_clus, nclass = k, EMC = em_opt,
                     stable.solution = TRUE, min.n = 1, min.n.iter = 10,
                     method = c("Rnd.EM"))
  ### Calculate BIC
  em.bic(scale(data_list$dat), init_EM)
}
```

Hyperparameter choosing (SFDMM)

I have chosen the set of hyperparameters based on the sensitivity analysis. Based on the sensitivity analysis, the model works well if we choose something that looks like a noninformative prior.

- $K_{max} = 10$
- $\sigma_0^2 = 100$
- $a_\sigma = b_\sigma = 0.01$
- $\xi = 1$
- $a_\theta = b_\theta = 1$
- the number of launch step is 10

Then, I will test this set of the hyperparameter on all cases for both raw and scaled dataset.

```
### Raw
for(i in 1:4){
  dat_sim <- f_data_sim(31807, i)
  dat_y <- as.numeric(scale(dat_sim$dat, center = FALSE, scale = FALSE))
  model <- SFDMM_model(iter = 10000, K_max = 10, init_assign = rep(0, 500),
                       y = dat_y, a0 = 0.01, b0 = 0.01, mu0 = 0, s20 = 100,
                       xi0 = 1, a_theta = 1, b_theta = 1, launch_iter = 10,
                       print_iter = 10001)
  print(paste0("Scenario ", i, " (Raw Data)"))
  table(salso(model$iter_assign[-(1:5000), ]), dat_sim$actual_clus) %>% print()
  print(" ")
}
```

```
## [1] "Scenario 1 (Raw Data)"
##
##       1   2
##   1   0 254
##   2 246   0
## [1] " "
## [1] "Scenario 2 (Raw Data)"
##
##       1   2   3   4   5
##   1 102   0   0   0   0
##   2   0   0  98   0   0
##   3   0   0   0   0  99
##   4   0   0   0 100   0
##   5   0 101   0   0   0
## [1] " "
## [1] "Scenario 3 (Raw Data)"
##
```

```
##       1   2
##   1   2 232
##   2 244  22
## [1] " "
## [1] "Scenario 4 (Raw Data)"
##
##       1   2   3   4   5
##   1 102  10   0   0   0
##   2   0   5  94   0   0
##   3   0   0   0   0  97
##   4   0   0   0 100   2
##   5   0  86   4   0   0
## [1] " "
```

```
### Scaled
for(i in 1:4){
  dat_sim <- f_data_sim(31807, i)
  dat_y <- as.numeric(scale(dat_sim$dat, center = TRUE, scale = TRUE))
  model <- SFDMM_model(iter = 10000, K_max = 10, init_assign = rep(0, 500),
                       y = dat_y, a0 = 0.01, b0 = 0.01, mu0 = 0, s20 = 100,
                       xi0 = 1, a_theta = 1, b_theta = 1, launch_iter = 10,
                       print_iter = 10001)
  print(paste0("Scenario ", i, " (Scaled Data)"))
  table(salso(model$iter_assign[-(1:5000), ]), dat_sim$actual_clus) %>% print()
  print(" ")
}
```

```
## [1] "Scenario 1 (Scaled Data)"
##
##       1   2
##   1   0 254
##   2 246   0
## [1] " "
## [1] "Scenario 2 (Scaled Data)"
##
##       1   2   3   4   5
##   1 102   0   0   0   0
##   2   0   0  98   0   0
##   3   0   0   0   0  99
##   4   0   0   0 100   0
##   5   0 101   0   0   0
## [1] " "
## [1] "Scenario 3 (Scaled Data)"
##
##       1   2
##   1   3 234
##   2 243  20
## [1] " "
## [1] "Scenario 4 (Scaled Data)"
##
##       1   2   3   4   5
##   1 102  10   0   0   0
##   2   0   4  94   0   0
##   3   0   0   0   0  97
```

```
##   4   0   0   0 100   2
##   5   0  87   4   0   0
## [1] " "
```

Other models

- We will compare our model with the other methods.
- K-means and PAM
  - Choose K from the range of 2 to 10 (K_{max})
  - Use the silhouette to determine the value of k. (the highest average silhouette).
- EM
  - Choose K with the lowest BIC.
- AntMAN
- Dirichlet Process

```
for(i in 1:4){

  print(paste0("============== Scenario ", i, " (Raw Data) =============="))

  dat_sim <- f_data_sim(31807, i)
  dat_y <- as.numeric(scale(dat_sim$dat, center = FALSE, scale = FALSE))

  ### K-mean
  k_means_sil <- rep(NA, 9)
  for(i in 2:10){
    k_means_sil[(i-1)] <- avg_sil(i, dat_y)
  }
  km_method <- kmeans(dat_y, which.max(k_means_sil) + 1)
  print("K-means: ")
  table(km_method$cluster, dat_sim$actual_clus) %>% print()
  print(" ")

  ### PAM
  pam_sil <- rep(NA, 9)
  for(i in 2:10){
    pam_sil[(i-1)] <- mean(silhouette(pam(dat_y, i))[, 3])
  }
  pam_method <- kmeans(dat_y, which.max(pam_sil) + 1)
  print("PAM: ")
  table(pam_method$cluster, dat_sim$actual_clus) %>% print()
  print(" ")

  ### EM
  em_option <- .EMControl(short.iter = 1)
  em_BIC <- rep(NA, 9)
  for(i in 2:10){
    em_BIC[(i-1)] <- k_EM_BIC(data.frame(dat_y), i, em_option)
  }
  EM_opt <- which.min(em_BIC) + 1
  em_method <- emcluster(data.frame(dat_y), emobj = init.EM(data.frame(dat_y), nclass = EM_opt,
                                                            EMC = em_option, stable.solution = TRUE,
```

```
                                                        min.n = 1, min.n.iter = 10,
                                                        method = c("Rnd.EM")),
                          EMC = em_option, assign.class = TRUE)$class
  print("EM: ")
  table(em_method, dat_sim$actual_clus) %>% print()
  print(" ")

  ### AntMAN
  AntMAN_MCMC <- AM_mcmc_parameters(niter = 10000, burnin = 5000, thin = 1,
                                    verbose = 1, output = c("CI", "K"),
                                    parallel = FALSE, output_dir = NULL)
  data_hyper <- AM_mix_hyperparams_uninorm(m0 = 0, k0 = 1, nu0 = 0.01, sig02 = 0.01)
  cluster_hyper <- AM_mix_weights_prior_gamma(a = 1, b = 1)
  AntMAN_mod<- AntMAN::AM_mcmc_fit(y = dat_y, initial_clustering = rep(1, 500),
                                   mix_kernel_hyperparams = data_hyper,
                                   mix_weight_prior = cluster_hyper,
                                   mcmc_parameters = AntMAN_MCMC)
  AntMAN_method <- as.numeric(salso(AM_clustering(AntMAN_mod), maxNClusters = 10))
  print("AntMAN: ")
  table(AntMAN_method, dat_sim$actual_clus) %>% print()
  print(" ")

  ### DP
  dp_mod <- DirichletProcessGaussian(as.matrix(dat_y),
                                     g0Priors = c(0, 1, 0.01, 0.01), alphaPriors = c(1, 1))
  dp_fit <- Fit(dp_mod, 10000, updatePrior = FALSE, progressBar = TRUE)
  dp_clus <- matrix(NA, nrow = 5000, ncol = 500)
  for(i in 1:5000){
    dp_clus[i, ] <- dp_fit$labelsChain[[(5000 + i)]]
  }
  print("DP: ")
  table(salso(dp_clus, maxNClusters = 10), dat_sim$actual_clus) %>% print()
  print(" ")
}
```

```
## [1] "============== Scenario 1 (Raw Data) =============="
## [1] "K-means: "
##
##       1   2
##   1 246   0
##   2   0 254
## [1] " "
## [1] "PAM: "
##
##       1   2
##   1   0 254
##   2 246   0
## [1] " "
## [1] "EM: "
##
## em_method   1   2
##         1 246   0
##         2   0 254
```

```
## [1] " "
## [1] "AntMAN: "
##
## AntMAN_method   1   2
##             1 246 254
## [1] " "
##   |                                                                      |
## [1] "DP: "
##
##       1   2
##   1   0 254
##   2 246   0
## [1] " "
## [1] "============== Scenario 2 (Raw Data) =============="
## [1] "K-means: "
##
##       1   2   3   4   5
##   1   0   0  98   0   0
##   2 102   0   0   0   0
##   3   0   0   0   0  99
##   4   0   0   0 100   0
##   5   0 101   0   0   0
## [1] " "
## [1] "PAM: "
##
##       1   2   3   4   5
##   1   0   0  98   0   0
##   2   0 101   0   0   0
##   3   0   0   0 100   0
##   4   0   0   0   0  99
##   5 102   0   0   0   0
## [1] " "
## [1] "EM: "
##
## em_method   1   2   3   4   5
##         1   0   0   0   0   3
##         2   0   0   0   0   3
##         3   0   0   0   0  51
##         4 102 101  98   0   0
##         5   0   0   0   0  40
##         6   0   0   0 100   0
##         7   0   0   0   0   2
## [1] " "
## [1] "AntMAN: "
##
## AntMAN_method   1   2   3   4   5
##             1 102 101  98 100  99
## [1] " "
##   |                                                                      |
## [1] "DP: "
##
##       1   2   3   4   5
##   1 102   0   0   0   0
##   2   0   0  98 100  99
```

```
##   3   0 101   0   0   0
## [1] " "
## [1] "============== Scenario 3 (Raw Data) =============="
## [1] "K-means: "
##
##       1   2
##   1 240  13
##   2   6 241
## [1] " "
## [1] "PAM: "
##
##       1   2
##   1   6 241
##   2 240  13
## [1] " "
## [1] "EM: "
##
## em_method   1   2
##         1   2 230
##         2 244  24
## [1] " "
## [1] "AntMAN: "
##
## AntMAN_method   1   2
##             1 246 254
## [1] " "
##   |                                                                 |
## [1] "DP: "
##
##       1   2
##   1   3 233
##   2 243  21
## [1] " "
## [1] "============== Scenario 4 (Raw Data) =============="
## [1] "K-means: "
##
##       1   2   3   4   5
##   1   0   4  94   0   0
##   2  99   8   0   0   0
##   3   0   0   0   0  97
##   4   0   0   0 100   2
##   5   3  89   4   0   0
## [1] " "
## [1] "PAM: "
##
##       1   2   3   4   5
##   1   0   4  94   0   0
##   2   3  89   4   0   0
##   3   0   0   0 100   2
##   4   0   0   0   0  97
##   5  99   8   0   0   0
## [1] " "
## [1] "EM: "
##
```

```
## em_method   1   2   3   4   5
##         1   0   0  19   0   0
##         2   0   3  21   0   0
##         3   0   0  49   0   0
##         4 102  98   6   0   0
##         5   0   0   3 100  99
## [1] " "
## [1] "AntMAN: "
##
## AntMAN_method   1   2   3   4   5
##             1 102 101  98 100  99
## [1] " "
##   |                                                                      |
## [1] "DP: "
##
##       1   2   3   4   5
##   1  95   4   0   0   0
##   2   7  97  98   0   0
##   3   0   0   0 100  99
## [1] " "
```

```r
for(i in 1:4){

  print(paste0("============== Scenario ", i, " (Scaled Data) =============="))

  dat_sim <- f_data_sim(31807, i)
  dat_y <- as.numeric(scale(dat_sim$dat, center = TRUE, scale = TRUE))

  ### K-mean
  k_means_sil <- rep(NA, 9)
  for(i in 2:10){
    k_means_sil[(i-1)] <- avg_sil(i, dat_y)
  }
  km_method <- kmeans(dat_y, which.max(k_means_sil) + 1)
  print("K-means: ")
  table(km_method$cluster, dat_sim$actual_clus) %>% print()
  print(" ")

  ### PAM
  pam_sil <- rep(NA, 9)
  for(i in 2:10){
    pam_sil[(i-1)] <- mean(silhouette(pam(dat_y, i))[, 3])
  }
  pam_method <- kmeans(dat_y, which.max(pam_sil) + 1)
  print("PAM: ")
  table(pam_method$cluster, dat_sim$actual_clus) %>% print()
  print(" ")

  ### EM
  em_option <- .EMControl(short.iter = 1)
  em_BIC <- rep(NA, 9)
  for(i in 2:10){
    em_BIC[(i-1)] <- k_EM_BIC(data.frame(dat_y), i, em_option)
  }
```

```
  EM_opt <- which.min(em_BIC) + 1
  em_method <- emcluster(data.frame(dat_y), emobj = init.EM(data.frame(dat_y), nclass = EM_opt,
                                                            EMC = em_option, stable.solution = TRUE,
                                                            min.n = 1, min.n.iter = 10,
                                                            method = c("Rnd.EM")),
                         EMC = em_option, assign.class = TRUE)$class
  print("EM: ")
  table(em_method, dat_sim$actual_clus) %>% print()
  print(" ")

  ### AntMAN
  AntMAN_MCMC <- AM_mcmc_parameters(niter = 10000, burnin = 5000, thin = 1,
                                    verbose = 1, output = c("CI", "K"),
                                    parallel = FALSE, output_dir = NULL)
  data_hyper <- AM_mix_hyperparams_uninorm(m0 = 0, k0 = 1, nu0 = 0.01, sig02 = 0.01)
  cluster_hyper <- AM_mix_weights_prior_gamma(a = 1, b = 1)
  AntMAN_mod<- AntMAN::AM_mcmc_fit(y = dat_y, initial_clustering = rep(1, 500),
                                   mix_kernel_hyperparams = data_hyper,
                                   mix_weight_prior = cluster_hyper,
                                   mcmc_parameters = AntMAN_MCMC)
  AntMAN_method <- as.numeric(salso(AM_clustering(AntMAN_mod), maxNClusters = 10))
  print("AntMAN: ")
  table(AntMAN_method, dat_sim$actual_clus) %>% print()
  print(" ")

  ### DP
  dp_mod <- DirichletProcessGaussian(as.matrix(dat_y),
                                     g0Priors = c(0, 1, 0.01, 0.01), alphaPriors = c(1, 1))
  dp_fit <- Fit(dp_mod, 10000, updatePrior = FALSE, progressBar = TRUE)
  dp_clus <- matrix(NA, nrow = 5000, ncol = 500)
  for(i in 1:5000){
    dp_clus[i, ] <- dp_fit$labelsChain[[(5000 + i)]]
  }
  print("DP: ")
  table(salso(dp_clus, maxNClusters = 10), dat_sim$actual_clus) %>% print()
  print(" ")
}
```

```
## [1] "============== Scenario 1 (Scaled Data) =============="
## [1] "K-means: "
##
##       1   2
##   1 246   0
##   2   0 254
## [1] " "
## [1] "PAM: "
##
##       1   2
##   1   0 254
##   2 246   0
## [1] " "
## [1] "EM: "
##
```

```
## em_method   1   2
##         1   0 254
##         2 246   0
## [1] " "
## [1] "AntMAN: "
##
## AntMAN_method   1   2
##             1 246 254
## [1] " "
##   |                                                                      |
## [1] "DP: "
##
##       1   2
##   1   0 254
##   2 246   0
## [1] " "
## [1] "============== Scenario 2 (Scaled Data) =============="
## [1] "K-means: "
##
##       1   2   3   4   5
##   1   0   0  98   0   0
##   2 102   0   0   0   0
##   3   0   0   0   0  99
##   4   0   0   0 100   0
##   5   0 101   0   0   0
## [1] " "
## [1] "PAM: "
##
##       1   2   3   4   5
##   1   0   0  98   0   0
##   2   0 101   0   0   0
##   3   0   0   0 100   0
##   4   0   0   0   0  99
##   5 102   0   0   0   0
## [1] " "
## [1] "EM: "
##
## em_method   1   2   3   4   5
##         1   0   0   0   0  99
##         2   0   0  33   0   0
##         3   0   0  39   0   0
##         4 102   0   0   0   0
##         5   0 101   0   0   0
##         6   0   0   0 100   0
##         7   0   0  26   0   0
## [1] " "
## [1] "AntMAN: "
##
## AntMAN_method   1   2   3   4   5
##             1 102 101  98 100  99
## [1] " "
##   |                                                                      |
## [1] "DP: "
##
```

```
##       1   2   3   4   5
##   1 102 101   0   0   0
##   2   0   0  98   0   0
##   3   0   0   0   0  99
##   4   0   0   0 100   0
## [1] " "
## [1] "============== Scenario 3 (Scaled Data) =============="
## [1] "K-means: "
##
##       1   2
##   1 240  13
##   2   6 241
## [1] " "
## [1] "PAM: "
##
##       1   2
##   1   6 241
##   2 240  13
## [1] " "
## [1] "EM: "
##
## em_method   1   2
##         1   2 230
##         2 244  24
## [1] " "
## [1] "AntMAN: "
##
## AntMAN_method   1   2
##             1 246 254
## [1] " "
##   |                                                                      |
## [1] "DP: "
##
##       1   2
##   1   3 232
##   2 243  22
## [1] " "
## [1] "============== Scenario 4 (Scaled Data) =============="
## [1] "K-means: "
##
##       1   2   3   4   5
##   1   0   4  94   0   0
##   2  99   8   0   0   0
##   3   0   0   0   0  97
##   4   0   0   0 100   2
##   5   3  89   4   0   0
## [1] " "
## [1] "PAM: "
##
##       1   2   3   4   5
##   1   0   4  94   0   0
##   2   3  89   4   0   0
##   3   0   0   0 100   2
##   4   0   0   0   0  97
```

```
##   5  99   8   0   0   0
## [1] " "
## [1] "EM: "
##
## em_method   1   2   3   4   5
##         1   0  86   4   0   0
##         2   0   5  94   0   0
##         3 102  10   0   0   0
##         4   0   0   0 100   2
##         5   0   0   0   0  97
## [1] " "
## [1] "AntMAN: "
##
## AntMAN_method   1   2   3   4   5
##             1 102 101  98 100  99
## [1] " "
##   |                                                                      |
## [1] "DP: "
##
##       1   2   3   4   5
##   1 102  97   5   0   0
##   2   0   4  93   0   0
##   3   0   0   0   0  98
##   4   0   0   0 100   1
## [1] " "
```