multivariate_norm_simulation_study

Tarek El-Hajjaoui

2023-05-14

```
library(MASS)
set.seed(0)  # for reproducibility
# Independent and equal Sigma
Sigma_ind_eq = matrix(c(1.5, 0, 0,
                        0, 1.5, 0,
                        0, 0, 1.5),nrow=3)

# Positively dependent Sigma
Sigma_pos_dep = matrix(c(1.0, 0.8, 0.6,
                         0.8, 1.0, 0.4,
                         0.6, 0.4, 1.0), nrow=3)


# Negatively dependent Sigma
Sigma_neg_dep = matrix(c(1.0, -0.2, -0.2,
                         -0.2, 1.0, -0.2,
                         -0.2, -0.2, 1.0), nrow=3)

# Independent and unequal Sigma
Sigma_ind_uneq = matrix(c(2, 0, 0,
                        0, 6, 0,
                        0, 0, 10), nrow=3)
```

Appendix

- Note: Only the helper functions and the use of the helper functions for 1 of simulations and its figure is demonstrated for conciseness.

Appendix: Sigmas

1. Sigma 1 (M1): Independent and equal variance

$$\Sigma = \begin{pmatrix} 1.5 & 0 & 0 \\ 0 & 1.5 & 0 \\ 0 & 0 & 1.5 \end{pmatrix}$$

2. Sigma 2 (M2): Positively dependent

$$\Sigma = \begin{pmatrix} 1 & 0.8 & 0.6 \\ 0.8 & 1 & 0.4 \\ 0.6 & 0.4 & 1 \end{pmatrix}$$

3. Sigma 3 (M3): Negatively dependent

$$\Sigma = \begin{pmatrix} 1 & -0.2 & -0.2 \\ -0.2 & 1 & -0.2 \\ -0.2 & -0.2 & 1 \end{pmatrix}$$

4. Sigma 4 (M4): Independent and unequal variance:

$$\Sigma = \begin{pmatrix} 2 & 0 & 0 \\ 0 & 6 & 0 \\ 0 & 0 & 10 \end{pmatrix}$$

Appendix: Code

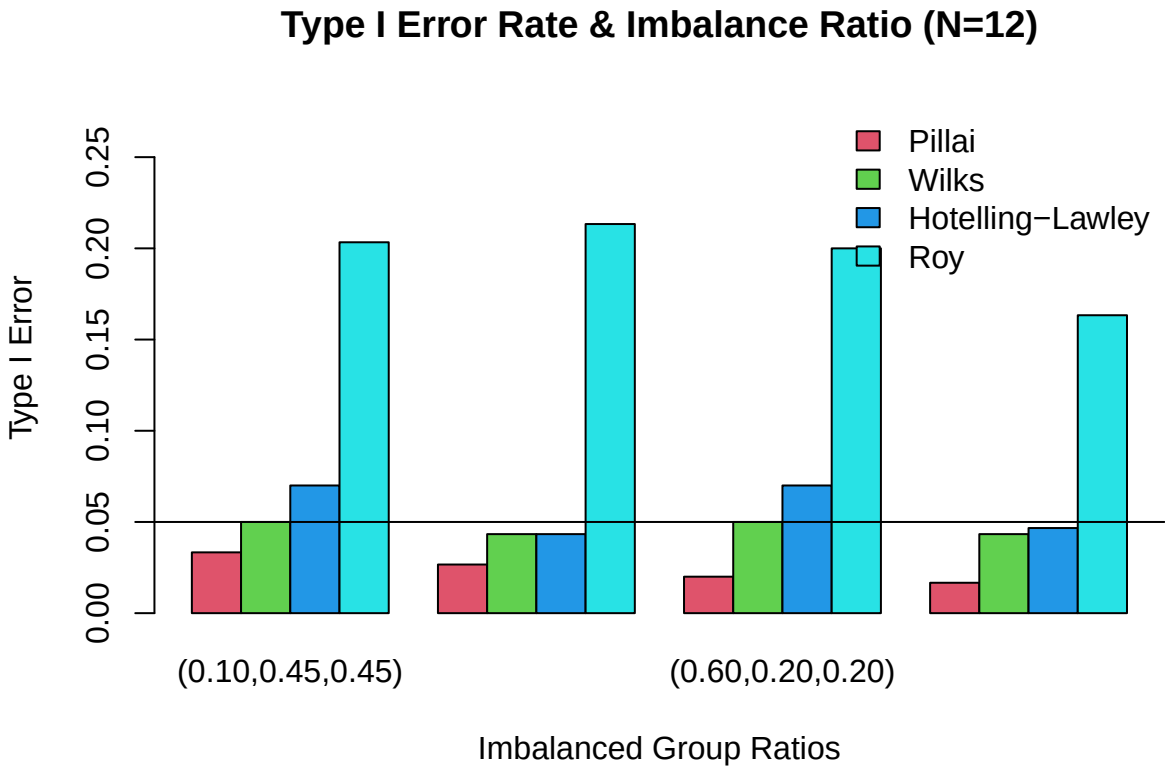
Helper Functions

```
gen_imbalanced_data <- function(n1, n2, n3, G, mu1, mu2, mu3, Sigma=diag(1.5, 3)){
  Y <- c()
  Y <- rbind(Y, mvrnorm(n1, mu1, Sigma))
  Y <- rbind(Y, mvrnorm(n2, mu2, Sigma))
  Y <- rbind(Y, mvrnorm(n3, mu3, Sigma))
  Y
}
test_imbalanced <- function(n1, n2, n3, G, Y){
  groups <- rep(c(paste("Group", 1)), each=n1)
  groups <- c(groups, rep(c(paste("Group", 2)), each = n2))
  groups <- c(groups, rep(c(paste("Group", 3)), each = n3))
  obj <- manova(Y ~ groups)
  tests <- c("Pillai", "Wilks", "Hotelling-Lawley", "Roy")
  reject <- rep(0, 4)
  for (t in 1:length(tests)){
      # position [1,6] where p-value is
      reject[t] <- summary(obj, test = tests[t])$stats[1,6]<0.05
    }
  reject # if reject, then 1 else 0
}
```

```r
simulate_imbalanced <- function(B, n, w1, w2, w3, G, mu1, mu2, mu3, Sigma=diag(1.5, 3)){
  # calculate the proportions for each group
  p1 <- w1 / (w1 + w2 + w3)
  p2 <- w2 / (w1 + w2 + w3)
  p3 <- w3 / (w1 + w2 + w3)
  proportions <- c(p1, p2, p3) # adjust proportions to ensure they sum to 1
  proportions <- proportions / sum(proportions)
  # multiply adjusted proportions by the total population size
  n1 <- round(proportions[1] * n)
  n2 <- round(proportions[2] * n)
  n3 <- round(proportions[3] * n)
  results <- rep(0, 4)
  for (b in 1:B){
      Y <- gen_imbalanced_data(n1,n2,n3,G,mu1,mu2,mu3,Sigma)
      results <- results + test_imbalanced(n1,n2,n3,G,Y)
  }
  results/B
}
```

Figure 2: Type I Error Rate of Imbalanced Groups.

```r
set.seed(0);tests <- c("Pillai", "Wilks", "Hotelling-Lawley", "Roy")
degrees <- c(1,2,3,4);n<-12
n12_alpha_imbalanced_degree <- matrix(0, 4, 4)
n12_alpha_imbalanced_degree[1, ] <- simulate_imbalanced(B=300,n=n,
                                                        w1=0.10,w2=0.45,w3=0.45,
                                                        G=3,mu1=c(0,0,0),mu2=c(0,0,0),
                                                            mu3=c(0,0,0),Sigma=Sigma_ind_eq)
n12_alpha_imbalanced_degree[2, ] <- simulate_imbalanced(B=300,n=n,
                                                        w1=0.20,w2=0.40,w3=0.40,
                                                        G=3,mu1=c(0,0,0),mu2=c(0,0,0),
                                                            mu3=c(0,0,0),Sigma=Sigma_ind_eq)
n12_alpha_imbalanced_degree[3, ] <- simulate_imbalanced(B=300,n=n,
                                                        w1=0.60,w2=0.20,w3=0.20,
                                                        G=3,mu1=c(0,0,0),mu2=c(0,0,0),
                                                            mu3=c(0,0,0),Sigma=Sigma_ind_eq)
n12_alpha_imbalanced_degree[4, ] <- simulate_imbalanced(B=300,n=n,
                                                        w1=0.90,w2=0.05,w3=0.05,
                                                        G=3,mu1=c(0,0,0),mu2=c(0,0,0),
                                                            mu3=c(0,0,0),Sigma=Sigma_ind_eq)
barplot(matrix(n12_alpha_imbalanced_degree, nrow = length(degrees), byrow = TRUE),
        beside = TRUE,
        ylim = c(0, 0.28),
        xlab = "Imbalanced Group Ratios",
        ylab = "Type I Error",
        main = "Type I Error Rate & Imbalance Ratio (N=12)",
        col = c(2:5),
        names.arg = c("(0.10,0.45,0.45)", "(0.20,0.40,0.40)", "(0.60,0.20,0.20)", "(0.90,0.20,0.20)"))
legend("topright",legend = tests,fill = c(2:5),bty = "n");abline(h = 0.05)
```


Type I Error Rate & Imbalance Ratio (N=12)
Type I Error
0.00
0.05
0.10
0.15
0.20
0.25
Pillai
Wilks
Hotelling−Lawley
Roy
(0.10,0.45,0.45)
(0.60,0.20,0.20)
Imbalanced Group Ratios