Version 2: 2 clusters; Fix cluster label switching u,v; unif; fix w [0.001,1]; only force ordering of means in 1 sample

Simulate data

```
I <- 50
K <- 2
S <- 10

# choose diffuse priors for gamma
a_gamma <- 2
b_gamma <- 10

avrg <- a_gamma * b_gamma
std.dv <- sqrt(a_gamma*b_gamma^2)
g_range = seq(0, avrg + 5*std.dv, 0.01)
g_y = dgamma(g_range, a_gamma, rate = 1/b_gamma)
#plot(g_range, g_y, type = "l", ylim=c(0, max(g_y) + 0.01))

set.seed(123)

a <- matrix(NA, nrow=K, ncol=S)
b <- matrix(NA, nrow=K, ncol=S)
for (s in 1:S) {
  a[, s] <- rgamma(K, a_gamma, rate = 1/b_gamma)
  b[, s] <- rgamma(K, a_gamma, rate = 1/b_gamma)
}

# reorder a,b matrices to match ordering of means (U) in S1
U <- a/(a+b)
V <- a+b
U.ordered <- U[order(U[,1]), ]
a.ordered <- a[order(U[,1]),]
b.ordered <- b[order(U[,1]),]
V.ordered <- V[order(U[,1]),]

pi <- as.vector(rdirichlet(1, rep(1, K)))
z <- sample(1:K, size = I, replace = T, prob = pi)

w <- matrix(NA, nrow=I, ncol=S)
for (s in 1:S) {
  w[, s] <- rbeta(I, a.ordered[,s][z], b.ordered[,s][z])
}

tcn <- matrix(2, nrow=I, ncol=S)
m <- matrix(rep(sample(1:2, size = I, replace = T), S), nrow=I, ncol=S)

calcTheta <- function(m, tcn, w) {
```

```
  (m * w) / (tcn * w + 2*(1-w))
}
theta <- calcTheta(m, tcn, w)

n <- replicate(S, rpois(I, 100))
y <- matrix(NA, nrow=I, ncol=S)
for (i in 1:I) {
  for (s in 1:S) {
    y[i, s] <- rbinom(1, n[i, s], theta[i,s])
  }
}
```

JAGS

```
jags.file <- file.path(models.dir, "v2_uv_unif_fix2.jags")

test.data <- list("I" = I, "S" = S, "K" = K,
                  "y" = y, "n" = n,
                  "m" = m, "tcn" = tcn)
jags.m <- jags.model(jags.file, test.data,
                     n.chains = 1,
                     inits = list(".RNG.name" = "base::Wichmann-Hill",
                                  ".RNG.seed" = 123))
```

```
## Compiling model graph
##    Resolving undeclared variables
##    Allocating nodes
## Graph information:
##    Observed stochastic nodes: 500
##    Unobserved stochastic nodes: 591
##    Total graph size: 8440
##
## Initializing model
```

```
params <- c("z", "w", "U", "V")
samps <- coda.samples(jags.m, params, n.iter=10000, thin=10)
s <- summary(samps)
effectiveSize(samps)
```

```
##      U[1,1]      U[2,1]      U[1,2]      U[2,2]      U[1,3]      U[2,3]
##  146.944614    5.997426   60.184976    5.043509    3.159442  505.893389
##      U[1,4]      U[2,4]      U[1,5]      U[2,5]      U[1,6]      U[2,6]
##    1.504475   33.404314    3.777986  881.226729 1000.000000   29.117591
##      U[1,7]      U[2,7]      U[1,8]      U[2,8]      U[1,9]      U[2,9]
##   12.497438   31.247274    3.083553    5.426385    3.199115   10.527595
##     U[1,10]     U[2,10]      V[1,1]      V[2,1]      V[1,2]      V[2,2]
##    2.964935   16.209653  379.541376  861.786122  654.339249  619.900535
##      V[1,3]      V[2,3]      V[1,4]      V[2,4]      V[1,5]      V[2,5]
##   26.727707   40.539815    4.171156   16.398334   25.091179    9.253076
##      V[1,6]      V[2,6]      V[1,7]      V[2,7]      V[1,8]      V[2,8]
##   13.200426 1000.000000  571.796637 1000.000000   10.972346   13.634779
##      V[1,9]      V[2,9]     V[1,10]     V[2,10]      w[1,1]      w[2,1]
##    5.078545 1000.000000    5.473132  355.106039  752.609040 1000.000000
```

```
##      w[3,1]      w[4,1]      w[5,1]      w[6,1]      w[7,1]      w[8,1]
## 1000.000000 1000.000000 1000.000000  902.949060 1088.182615 1000.000000
##      w[9,1]     w[10,1]     w[11,1]     w[12,1]     w[13,1]     w[14,1]
## 1000.000000  667.225111 1000.000000 1000.000000 1000.000000 1000.000000
##     w[15,1]     w[16,1]     w[17,1]     w[18,1]     w[19,1]     w[20,1]
## 1000.000000 1000.000000 1135.009447 1000.000000 1000.000000 1000.000000
##     w[21,1]     w[22,1]     w[23,1]     w[24,1]     w[25,1]     w[26,1]
## 1000.000000 1000.000000  904.021431  900.772679 1000.000000 1027.757904
##     w[27,1]     w[28,1]     w[29,1]     w[30,1]     w[31,1]     w[32,1]
## 1102.011636 1000.000000 1000.000000 1292.136672 1000.000000  791.140537
##     w[33,1]     w[34,1]     w[35,1]     w[36,1]     w[37,1]     w[38,1]
##  894.287036 1000.000000 1000.000000 1000.000000  728.555697  952.933539
##     w[39,1]     w[40,1]     w[41,1]     w[42,1]     w[43,1]     w[44,1]
## 1000.000000 1000.000000 1000.000000 1000.000000 1000.000000  870.073256
##     w[45,1]     w[46,1]     w[47,1]     w[48,1]     w[49,1]     w[50,1]
## 1000.000000 1000.000000 1000.000000 1000.000000 1000.000000 1000.000000
##      w[1,2]      w[2,2]      w[3,2]      w[4,2]      w[5,2]      w[6,2]
## 1000.000000  315.125220  784.095152 1000.000000  256.588696  679.215235
##      w[7,2]      w[8,2]      w[9,2]     w[10,2]     w[11,2]     w[12,2]
## 1000.000000 1104.213431   78.086905 1042.902426 1000.000000  501.302612
##     w[13,2]     w[14,2]     w[15,2]     w[16,2]     w[17,2]     w[18,2]
## 1000.000000  910.354816  902.285768 1000.000000  622.685287 1000.000000
##     w[19,2]     w[20,2]     w[21,2]     w[22,2]     w[23,2]     w[24,2]
## 1000.000000 1032.457177  768.191557  668.610702  636.154905  265.141651
##     w[25,2]     w[26,2]     w[27,2]     w[28,2]     w[29,2]     w[30,2]
## 1000.000000 1000.000000  542.267841  843.968028  803.092629  646.030206
##     w[31,2]     w[32,2]     w[33,2]     w[34,2]     w[35,2]     w[36,2]
##  904.436452 1000.000000 1144.368979 1000.000000 1000.000000  470.377688
##     w[37,2]     w[38,2]     w[39,2]     w[40,2]     w[41,2]     w[42,2]
## 1189.152002 1000.000000  884.928894 1000.000000  879.299212  628.249947
##     w[43,2]     w[44,2]     w[45,2]     w[46,2]     w[47,2]     w[48,2]
## 1387.618547  708.854769  846.360872 1000.000000  864.127910  773.507616
##     w[49,2]     w[50,2]      w[1,3]      w[2,3]      w[3,3]      w[4,3]
##  807.390663  834.356215 1000.000000   97.303943  737.067608 1000.000000
##      w[5,3]      w[6,3]      w[7,3]      w[8,3]      w[9,3]     w[10,3]
##  109.579205  191.663033  710.355589 1000.000000   74.387022 1109.137534
##     w[11,3]     w[12,3]     w[13,3]     w[14,3]     w[15,3]     w[16,3]
## 1000.000000  299.200793  882.566954 1000.000000  881.745025 1000.000000
##     w[17,3]     w[18,3]     w[19,3]     w[20,3]     w[21,3]     w[22,3]
##   74.696779 1122.031861  265.783424  457.351226  451.413635   68.352759
##     w[23,3]     w[24,3]     w[25,3]     w[26,3]     w[27,3]     w[28,3]
##  114.895655  116.525181  892.551295 1000.000000   87.860219  855.159897
##     w[29,3]     w[30,3]     w[31,3]     w[32,3]     w[33,3]     w[34,3]
##   68.431529   88.399188  877.150625  474.068225  816.299022  437.364125
##     w[35,3]     w[36,3]     w[37,3]     w[38,3]     w[39,3]     w[40,3]
## 1000.000000  182.412143 1000.000000  337.864586   88.084805  762.279090
##     w[41,3]     w[42,3]     w[43,3]     w[44,3]     w[45,3]     w[46,3]
## 1000.000000   51.046332  722.404415  849.321273 1000.000000 1000.000000
##     w[47,3]     w[48,3]     w[49,3]     w[50,3]      w[1,4]      w[2,4]
## 1000.000000  721.705776  759.122385 1000.000000  889.491066  879.429911
##      w[3,4]      w[4,4]      w[5,4]      w[6,4]      w[7,4]      w[8,4]
##  308.873438  377.659525  895.488865   43.492740  548.404276  346.339746
##      w[9,4]     w[10,4]     w[11,4]     w[12,4]     w[13,4]     w[14,4]
## 1000.000000   63.961968  158.293434 1226.334480 1000.000000  768.967884
```

```
##     w[15,4]     w[16,4]     w[17,4]     w[18,4]     w[19,4]     w[20,4]
##  811.619342 1000.000000  502.451940 1000.000000  771.909835 1360.560722
##     w[21,4]     w[22,4]     w[23,4]     w[24,4]     w[25,4]     w[26,4]
##  273.391047    8.534488 1133.060129   69.576490  433.444746  224.776157
##     w[27,4]     w[28,4]     w[29,4]     w[30,4]     w[31,4]     w[32,4]
##  889.298389 1376.286874  938.625266  460.125294 1000.000000   93.370781
##     w[33,4]     w[34,4]     w[35,4]     w[36,4]     w[37,4]     w[38,4]
## 1000.000000  176.070700  876.119805  839.465584 1000.000000  374.783553
##     w[39,4]     w[40,4]     w[41,4]     w[42,4]     w[43,4]     w[44,4]
## 1000.000000  446.665507  121.841488   14.737969 1226.533256 1000.000000
##     w[45,4]     w[46,4]     w[47,4]     w[48,4]     w[49,4]     w[50,4]
##   92.105671  631.127798   91.665502  183.883825  858.090699 1055.287685
##      w[1,5]      w[2,5]      w[3,5]      w[4,5]      w[5,5]      w[6,5]
##  836.744946  158.405431 1000.000000  578.971927 1190.558165   55.540417
##      w[7,5]      w[8,5]      w[9,5]     w[10,5]     w[11,5]     w[12,5]
## 1000.000000 1000.000000  168.425789 1000.000000  831.403766   57.167869
##     w[13,5]     w[14,5]     w[15,5]     w[16,5]     w[17,5]     w[18,5]
##  201.462212 1121.762021  894.689074 1000.000000   82.135168  676.091343
##     w[19,5]     w[20,5]     w[21,5]     w[22,5]     w[23,5]     w[24,5]
## 1097.777137  939.137969  209.245813  182.595351 1000.000000 1000.000000
##     w[25,5]     w[26,5]     w[27,5]     w[28,5]     w[29,5]     w[30,5]
##  259.173468 1000.000000  604.083325 1041.004812 1000.000000  847.643352
##     w[31,5]     w[32,5]     w[33,5]     w[34,5]     w[35,5]     w[36,5]
##  118.811404 1000.000000 1000.000000  608.112652 1000.000000   20.574996
##     w[37,5]     w[38,5]     w[39,5]     w[40,5]     w[41,5]     w[42,5]
## 1000.000000  717.005709 1000.000000 1000.000000   54.647713  299.393635
##     w[43,5]     w[44,5]     w[45,5]     w[46,5]     w[47,5]     w[48,5]
## 1172.508735 1000.000000  995.282579  888.263634 1000.000000  790.833515
##     w[49,5]     w[50,5]      w[1,6]      w[2,6]      w[3,6]      w[4,6]
## 1000.000000 1000.000000 1033.717172  269.824904 1000.000000 1000.000000
##      w[5,6]      w[6,6]      w[7,6]      w[8,6]      w[9,6]     w[10,6]
##  235.877266  833.874912  810.857210 1100.139687 1000.000000 1124.049272
##     w[11,6]     w[12,6]     w[13,6]     w[14,6]     w[15,6]     w[16,6]
## 1092.047058  119.521955 1000.000000 1000.000000 1000.000000 1000.000000
##     w[17,6]     w[18,6]     w[19,6]     w[20,6]     w[21,6]     w[22,6]
## 1000.000000 1000.000000  834.547661  834.314118 1000.000000  876.478740
##     w[23,6]     w[24,6]     w[25,6]     w[26,6]     w[27,6]     w[28,6]
## 1000.000000  110.966523 1000.000000 1133.159866 1033.439735 1144.976604
##     w[29,6]     w[30,6]     w[31,6]     w[32,6]     w[33,6]     w[34,6]
## 1000.000000  313.460446  708.806339 1000.000000 1000.000000  823.233372
##     w[35,6]     w[36,6]     w[37,6]     w[38,6]     w[39,6]     w[40,6]
## 1130.534415  859.365365 1256.209880 1000.000000 1109.843772 1000.000000
##     w[41,6]     w[42,6]     w[43,6]     w[44,6]     w[45,6]     w[46,6]
## 1000.000000  483.401478 1000.000000 1000.000000 1000.000000 1000.000000
##     w[47,6]     w[48,6]     w[49,6]     w[50,6]      w[1,7]      w[2,7]
## 1084.703509 1109.581535 1000.000000 1000.000000 1000.000000 1000.000000
##      w[3,7]      w[4,7]      w[5,7]      w[6,7]      w[7,7]      w[8,7]
## 1000.000000 1096.161318 1000.000000 1000.000000  720.236534 1000.000000
##      w[9,7]     w[10,7]     w[11,7]     w[12,7]     w[13,7]     w[14,7]
## 1000.000000 1000.000000 1000.000000 1000.000000 1000.000000 1000.000000
##     w[15,7]     w[16,7]     w[17,7]     w[18,7]     w[19,7]     w[20,7]
##  901.379320 1139.983001 1000.000000 1117.982849 1000.000000 1000.000000
##     w[21,7]     w[22,7]     w[23,7]     w[24,7]     w[25,7]     w[26,7]
## 1000.000000  904.229237  813.411059 1000.000000 1000.000000 1000.000000
```

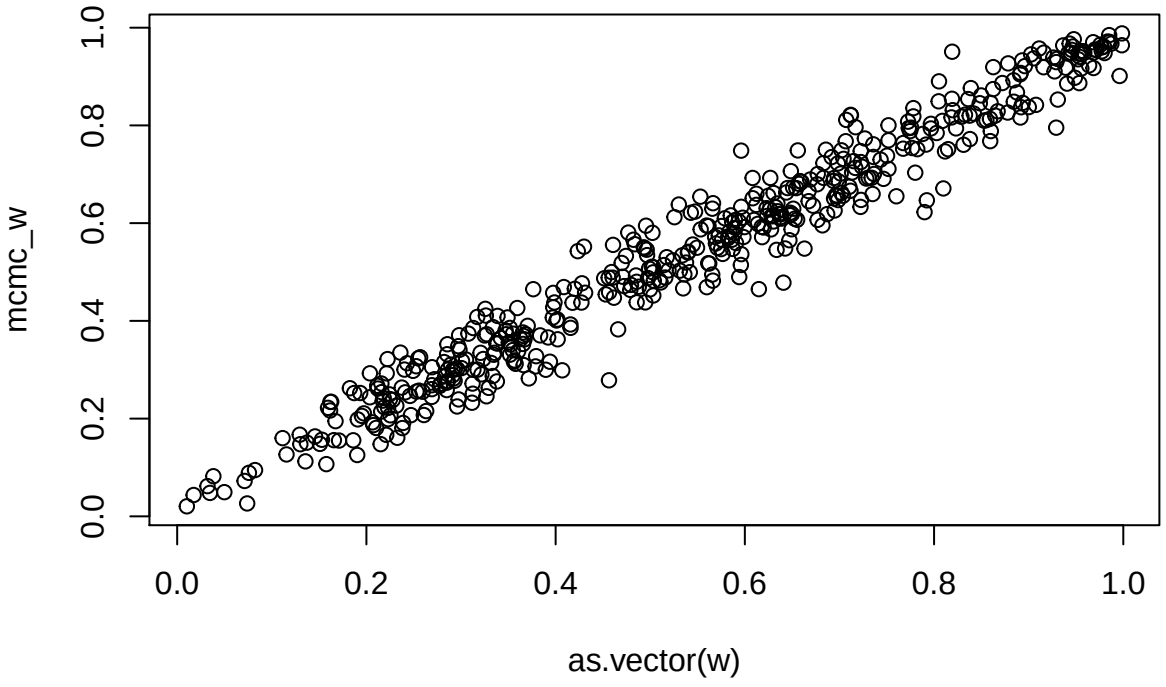
```
##     w[27,7]     w[28,7]     w[29,7]     w[30,7]     w[31,7]     w[32,7]
## 1000.000000  831.264871  749.285620  905.447032 1050.457275 1000.000000
##     w[33,7]     w[34,7]     w[35,7]     w[36,7]     w[37,7]     w[38,7]
## 1000.000000 1000.000000 1111.234490 1000.000000 1000.000000 1000.000000
##     w[39,7]     w[40,7]     w[41,7]     w[42,7]     w[43,7]     w[44,7]
## 1000.000000 1000.000000 1000.000000 1000.000000 1000.000000 1182.041073
##     w[45,7]     w[46,7]     w[47,7]     w[48,7]     w[49,7]     w[50,7]
## 1000.000000 1000.000000 1000.000000 1000.000000 1067.592687  817.575466
##      w[1,8]      w[2,8]      w[3,8]      w[4,8]      w[5,8]      w[6,8]
##  961.160000   13.021064  130.446861   45.525233   36.557496   59.373844
##      w[7,8]      w[8,8]      w[9,8]     w[10,8]     w[11,8]     w[12,8]
##  216.507236 1000.000000   51.461317   30.967450   18.152921   78.438044
##     w[13,8]     w[14,8]     w[15,8]     w[16,8]     w[17,8]     w[18,8]
##   66.916440  153.203823  156.933232 1000.000000   16.744108 1000.000000
##     w[19,8]     w[20,8]     w[21,8]     w[22,8]     w[23,8]     w[24,8]
##   23.215412  704.570847  313.053781  855.471791   26.496144   17.957391
##     w[25,8]     w[26,8]     w[27,8]     w[28,8]     w[29,8]     w[30,8]
##  571.188243 1000.000000   10.615425  100.035373 1000.000000   24.066320
##     w[31,8]     w[32,8]     w[33,8]     w[34,8]     w[35,8]     w[36,8]
## 1000.000000   91.521786  897.198722  962.874117  868.054812   78.340907
##     w[37,8]     w[38,8]     w[39,8]     w[40,8]     w[41,8]     w[42,8]
##   97.589447   50.058901   89.816581  862.958065  456.744538   69.192033
##     w[43,8]     w[44,8]     w[45,8]     w[46,8]     w[47,8]     w[48,8]
## 1000.000000 1000.000000 1238.894084   23.676844   22.920752  130.477981
##     w[49,8]     w[50,8]      w[1,9]      w[2,9]      w[3,9]      w[4,9]
##  323.547967   57.620704 1000.000000 1000.000000 1142.113349 1000.000000
##      w[5,9]      w[6,9]      w[7,9]      w[8,9]      w[9,9]     w[10,9]
## 1116.752325   44.161403   23.298568 1127.824034   45.310970   69.484741
##     w[11,9]     w[12,9]     w[13,9]     w[14,9]     w[15,9]     w[16,9]
## 1000.000000 1071.088643   13.760082  730.225128 1000.000000  206.432230
##     w[17,9]     w[18,9]     w[19,9]     w[20,9]     w[21,9]     w[22,9]
##  821.425890 1000.000000   27.594369 1000.000000  154.617423   48.962865
##     w[23,9]     w[24,9]     w[25,9]     w[26,9]     w[27,9]     w[28,9]
## 1095.507418 1000.000000  106.694610 1000.000000 1000.000000 1000.000000
##     w[29,9]     w[30,9]     w[31,9]     w[32,9]     w[33,9]     w[34,9]
## 1000.000000   21.761956  490.412561   64.581696  880.227729   51.874390
##     w[35,9]     w[36,9]     w[37,9]     w[38,9]     w[39,9]     w[40,9]
##  901.458368 1000.000000   24.495890  278.694914   18.794363 1000.000000
##     w[41,9]     w[42,9]     w[43,9]     w[44,9]     w[45,9]     w[46,9]
## 1000.000000  344.940069 1000.000000  290.055402  365.931131   16.339498
##     w[47,9]     w[48,9]     w[49,9]     w[50,9]     w[1,10]     w[2,10]
## 1290.125772  811.628816  119.157686 1000.000000 1000.000000   65.106975
##     w[3,10]     w[4,10]     w[5,10]     w[6,10]     w[7,10]     w[8,10]
##   90.817809   21.669456  313.225455 1000.000000 1000.000000  603.915416
##     w[9,10]    w[10,10]    w[11,10]    w[12,10]    w[13,10]    w[14,10]
##   32.506365  117.338133 1000.000000 1000.000000 1000.000000 1000.000000
##    w[15,10]    w[16,10]    w[17,10]    w[18,10]    w[19,10]    w[20,10]
##  619.929392 1000.000000 1000.000000 1000.000000 1000.000000  974.966029
##    w[21,10]    w[22,10]    w[23,10]    w[24,10]    w[25,10]    w[26,10]
##   40.295117  289.492736 1000.000000   38.874846  623.809380  319.786501
##    w[27,10]    w[28,10]    w[29,10]    w[30,10]    w[31,10]    w[32,10]
## 1000.000000   64.157845  487.232179  749.339455  115.874005  521.759530
##    w[33,10]    w[34,10]    w[35,10]    w[36,10]    w[37,10]    w[38,10]
##  458.963366 1000.000000 1000.000000  238.379568  379.241101 1000.000000
```

```
##    w[39,10]    w[40,10]    w[41,10]    w[42,10]    w[43,10]    w[44,10]
##   21.262205  195.597646  443.929379  113.193626 1116.335920 1000.000000
##    w[45,10]    w[46,10]    w[47,10]    w[48,10]    w[49,10]    w[50,10]
##  214.659874  132.962705  187.397298 1000.000000  140.384787 1000.000000
##        z[1]        z[2]        z[3]        z[4]        z[5]        z[6]
##    0.000000    1.552061    0.000000    0.000000    1.552061    1.552061
##        z[7]        z[8]        z[9]       z[10]       z[11]       z[12]
##    0.000000    0.000000    1.552061    0.000000    0.000000    1.552061
##       z[13]       z[14]       z[15]       z[16]       z[17]       z[18]
##    0.000000    0.000000    0.000000    0.000000    1.552061    0.000000
##       z[19]       z[20]       z[21]       z[22]       z[23]       z[24]
##    0.000000    0.000000    1.552061    1.552061    1.552061    1.552061
##       z[25]       z[26]       z[27]       z[28]       z[29]       z[30]
##    1.552061    0.000000    1.553403    1.552061    1.552061    1.552061
##       z[31]       z[32]       z[33]       z[34]       z[35]       z[36]
##    0.000000    0.000000    0.000000    0.000000    0.000000    1.552061
##       z[37]       z[38]       z[39]       z[40]       z[41]       z[42]
##    0.000000    0.000000    0.000000    0.000000    1.552061    1.552061
##       z[43]       z[44]       z[45]       z[46]       z[47]       z[48]
##    0.000000    0.000000    0.000000    0.000000    0.000000    0.000000
##       z[49]       z[50]
##    0.000000    0.000000
```

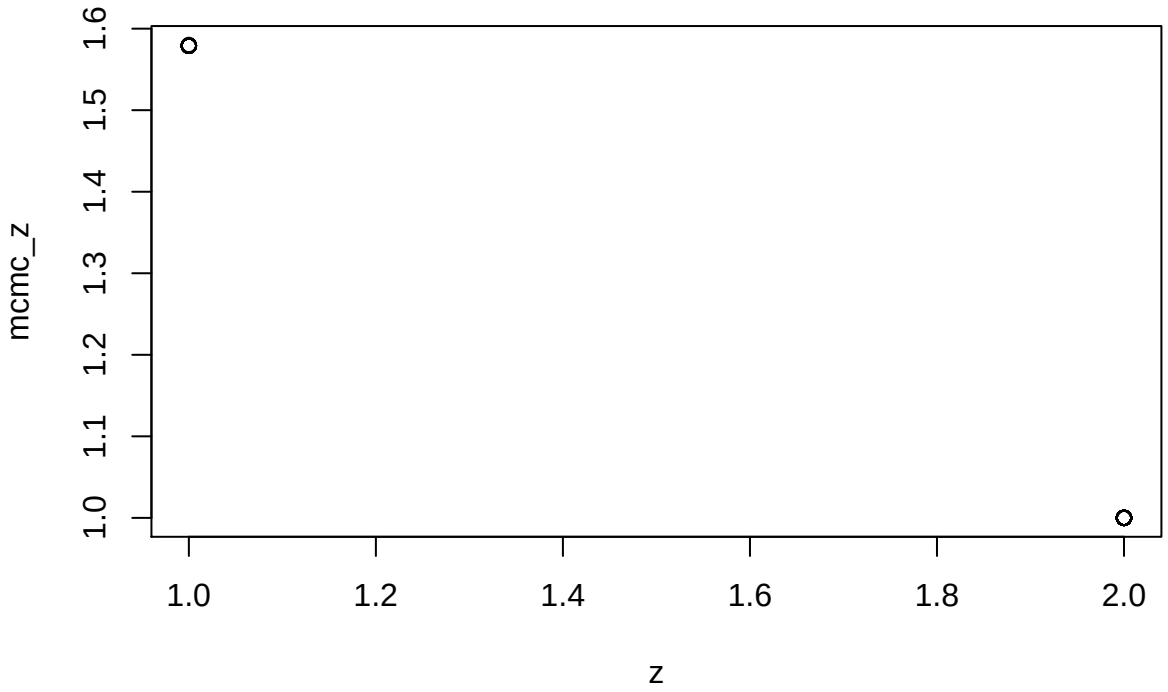
```
pdf(file.path(trace.dir, paste0(runName, "_trace.pdf")))
plot(samps)
dev.off()
```

```
## pdf
##   2
```

```
mcmc_vals <- s$statistics
mcmc_w <- mcmc_vals[substr(rownames(mcmc_vals), 1, 1) == "w", "Mean"]
plot(as.vector(w), mcmc_w, type = "p")
```



```r
mcmc_z <- as.vector(mcmc_vals[substr(rownames(mcmc_vals), 1, 1) == "z", "Mean"])
#mcmc_z <- round(mcmc_z, 0)
plot(z, mcmc_z, type = "p")
```

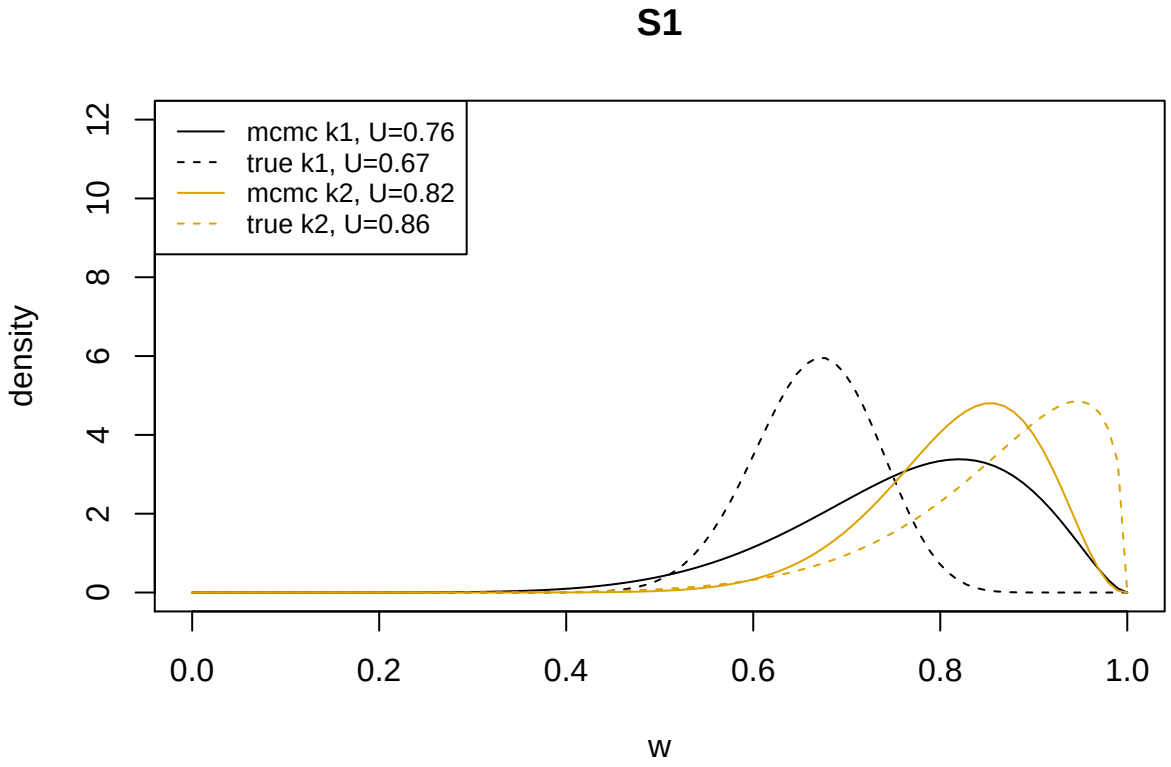


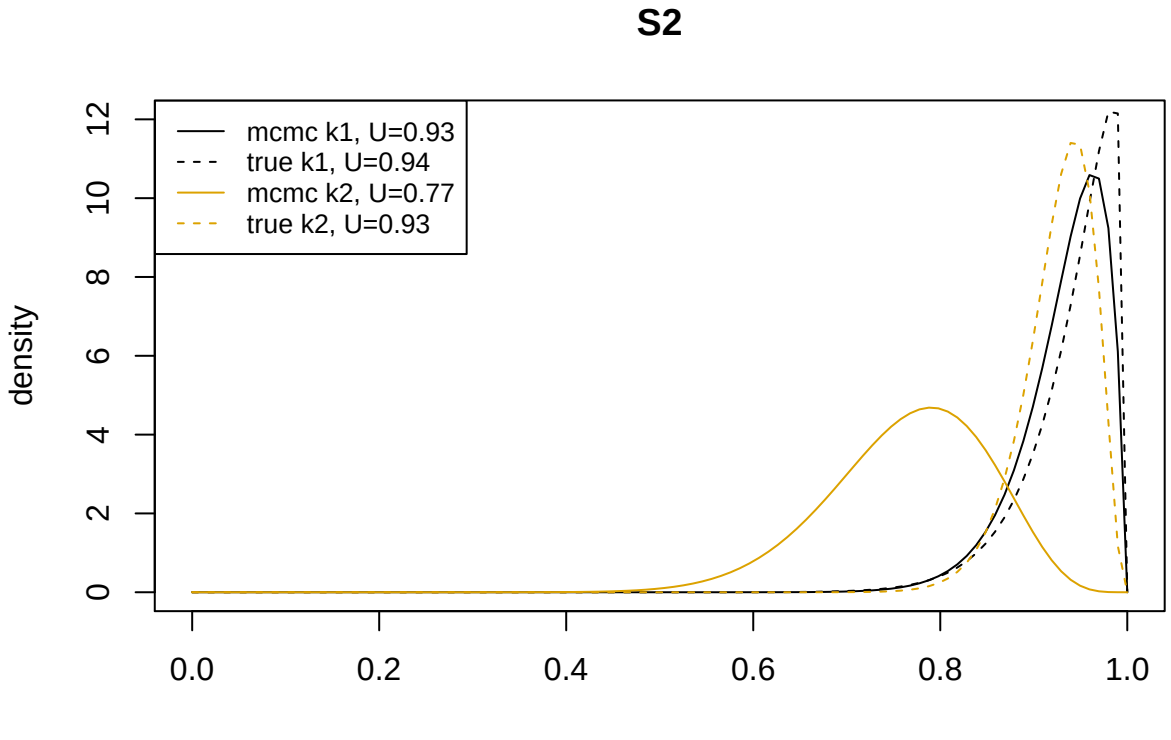

```r
mcmc_U <- mcmc_vals[substr(rownames(mcmc_vals), 1, 1) == "U", "Mean"]
mcmc_U <- matrix(mcmc_U, nrow=K)
mcmc_V <- mcmc_vals[substr(rownames(mcmc_vals), 1, 1) == "V", "Mean"]
mcmc_V <- matrix(mcmc_V, nrow=K)

mcmc_U
```

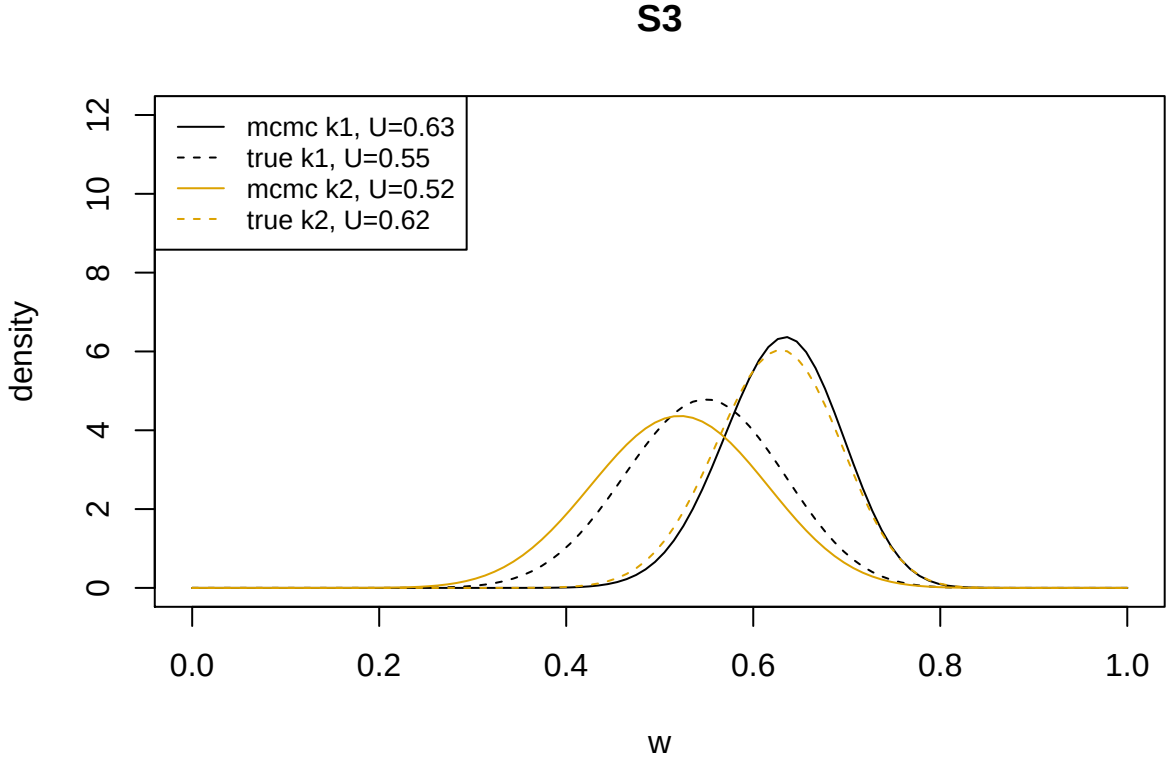
```
##           [,1]      [,2]      [,3]      [,4]      [,5]      [,6]      [,7]
## [1,] 0.7649695 0.9321478 0.6299403 0.3332673 0.6405793 0.2316686 0.3013665
## [2,] 0.8170933 0.7657157 0.5199298 0.6072313 0.4714489 0.3523815 0.4141651
##           [,8]      [,9]     [,10]
## [1,] 0.5952107 0.3620452 0.5070573
## [2,] 0.6890116 0.6465535 0.4062999
```

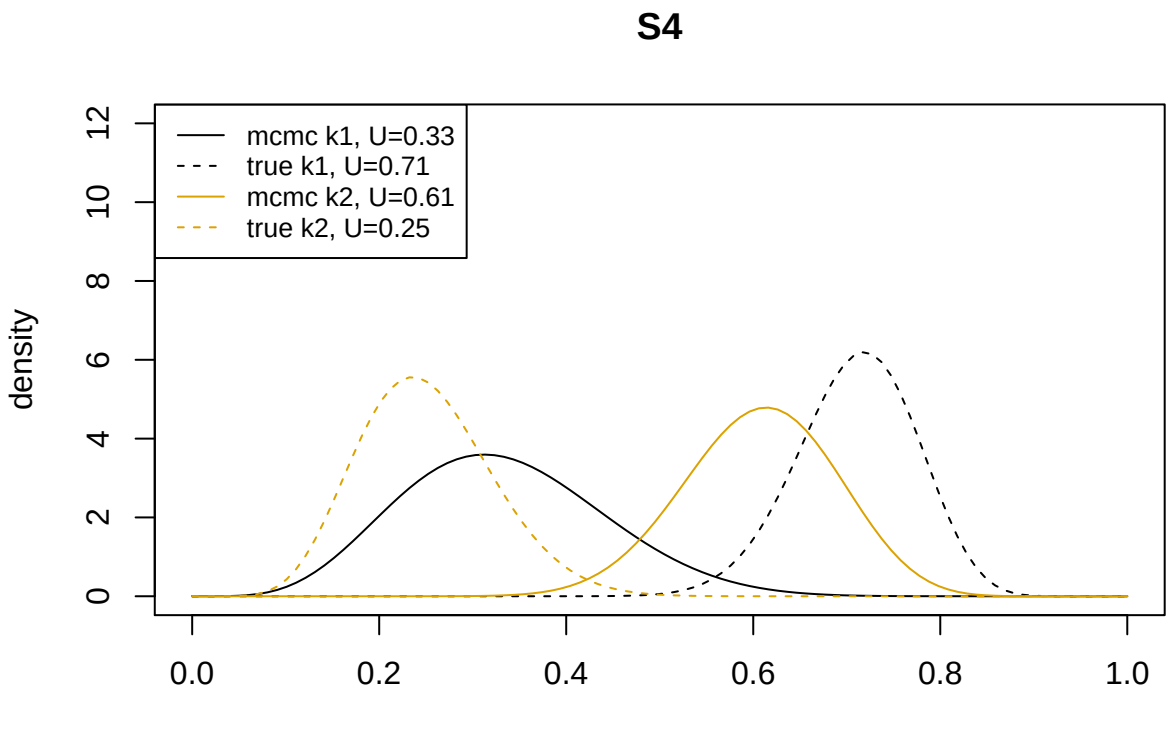
```r
p <- seq(0, 1, length = 100)
colors <- c("#000000", "#DCA200", "#8FA7ED", "#9D847A", "#A47901")
for (s in 1:S) {
  for (k in 1:K) {
    if (k == 1) {
      # plot mcmc mean U,V
      plot(p, dbeta(p, mcmc_U[k,s] * mcmc_V[k,s], (1-mcmc_U[k,s])*mcmc_V[k,s]),
           main = paste0("S", s),
           ylab = "density", xlab = "w", type = "l", col = colors[k],
           ylim = c(0, 12))
      # plot truth
      lines(p, dbeta(p, a.ordered[k,s], b.ordered[k,s]), type = "l", col = colors[k], lty=2)
      # add legend
      allU <- round(as.vector(rbind(mcmc_U[,s], U.ordered[,s])), digits = 2)
```

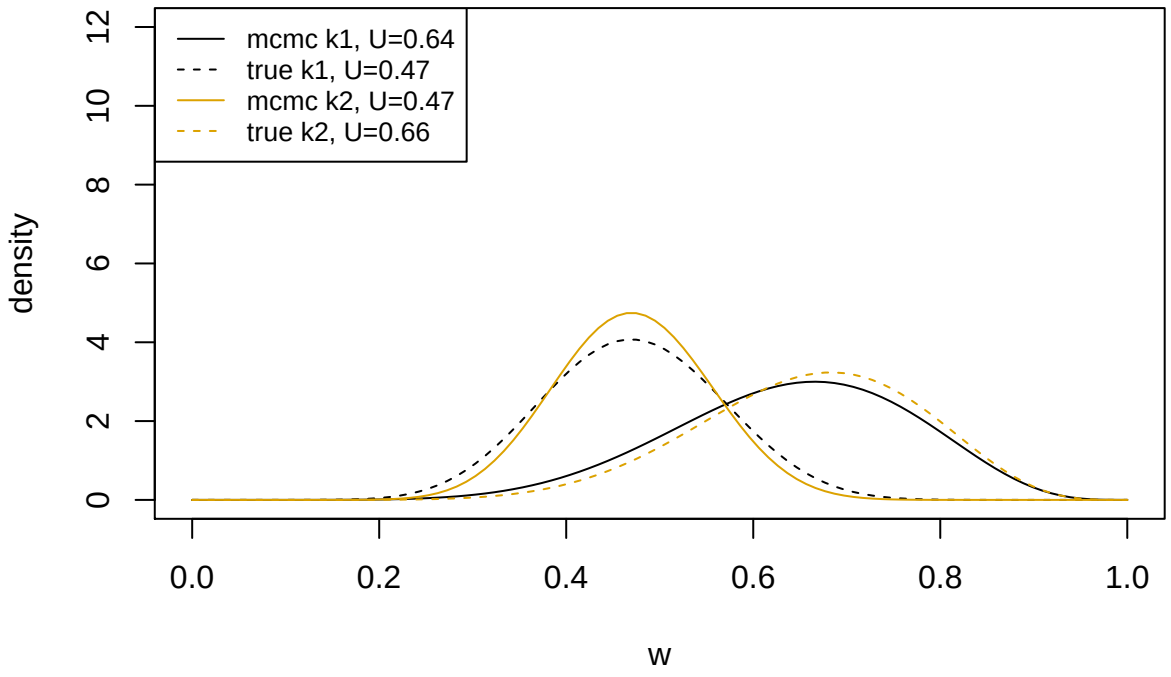
```
      legend(x = "topleft",
             legend = paste0(c("mcmc k", "true k"), rep(1:K, each=2), ", U=", allU),
             col = colors[rep(1:K, each=2)],
             lty = rep(1:2, K),
             cex=0.8)
    } else {
      # plot mcmc mean U,V
      lines(p, dbeta(p, mcmc_U[k,s] * mcmc_V[k,s], (1-mcmc_U[k,s])*mcmc_V[k,s]),
            type = "l", col = colors[k])
      # plot truth
      lines(p, dbeta(p, a.ordered[k,s], b.ordered[k,s]), type = "l", col = colors[k], lty=2)
    }
  }
}
```

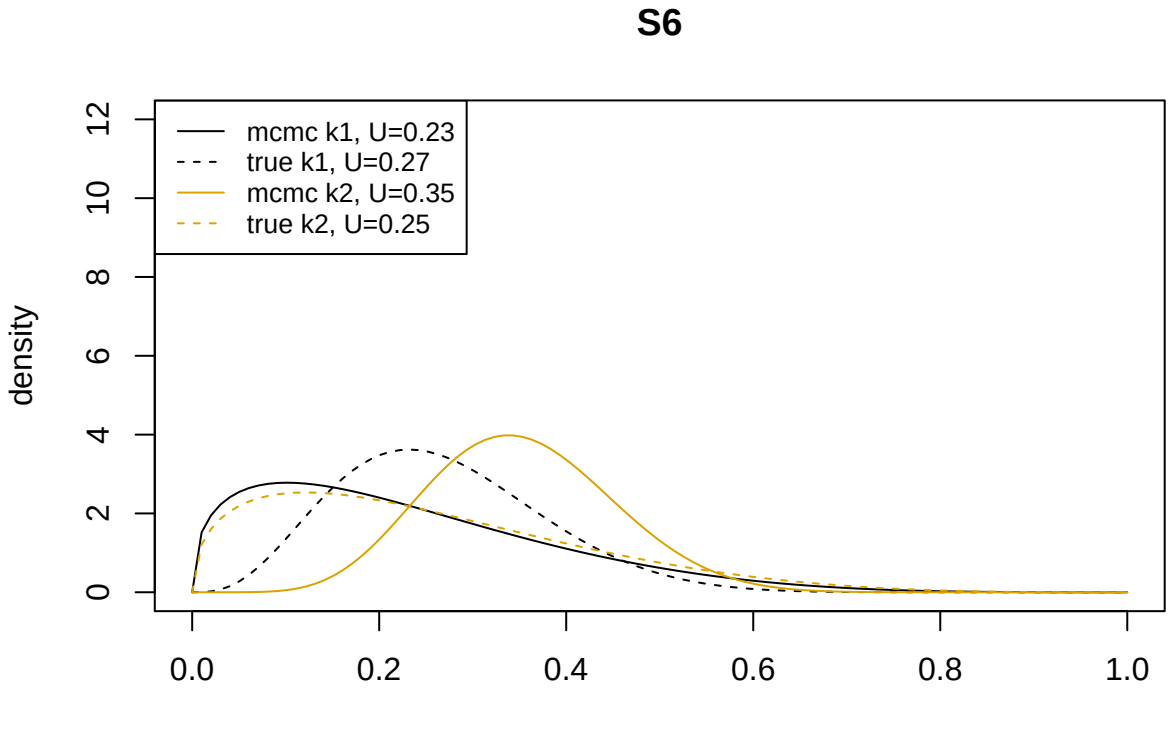


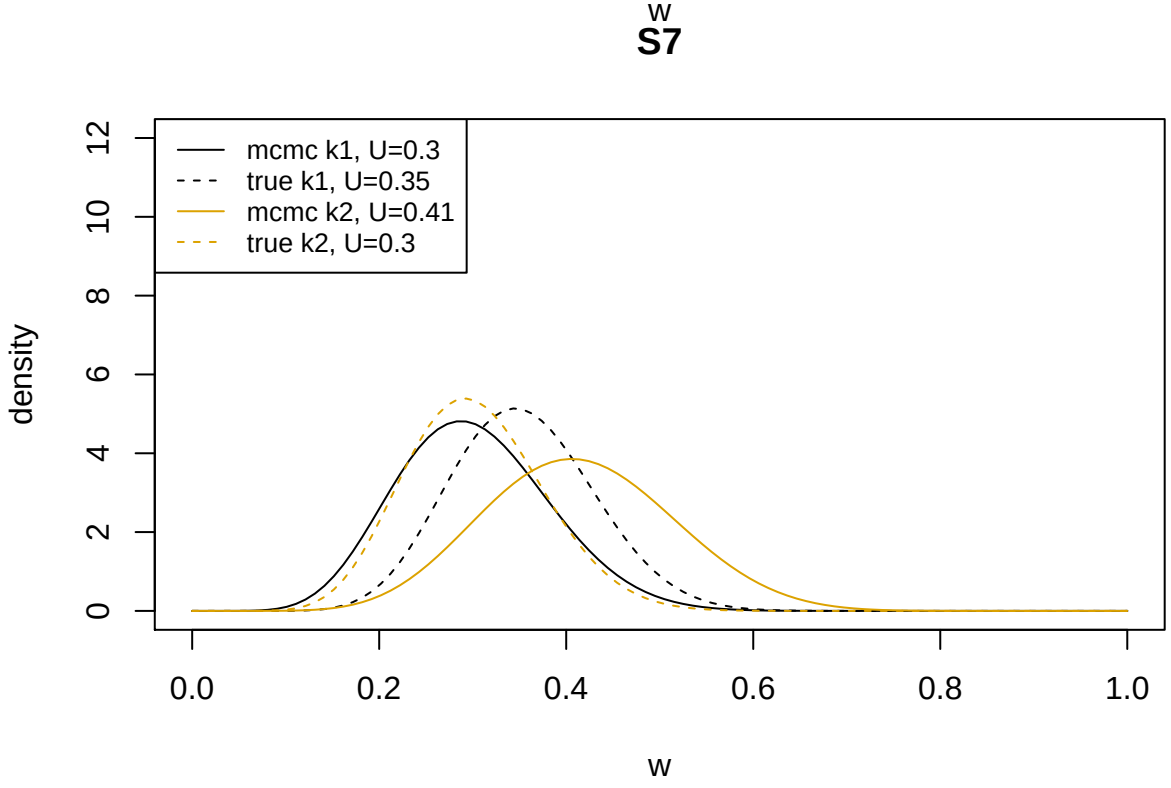

S2
mcmc k1, U=0.93
true k1, U=0.94
mcmc k2, U=0.77
true k2, U=0.93
density
0
2
4
6
8
10
12
0.0
0.2
0.4
0.6
0.8
1.0
w

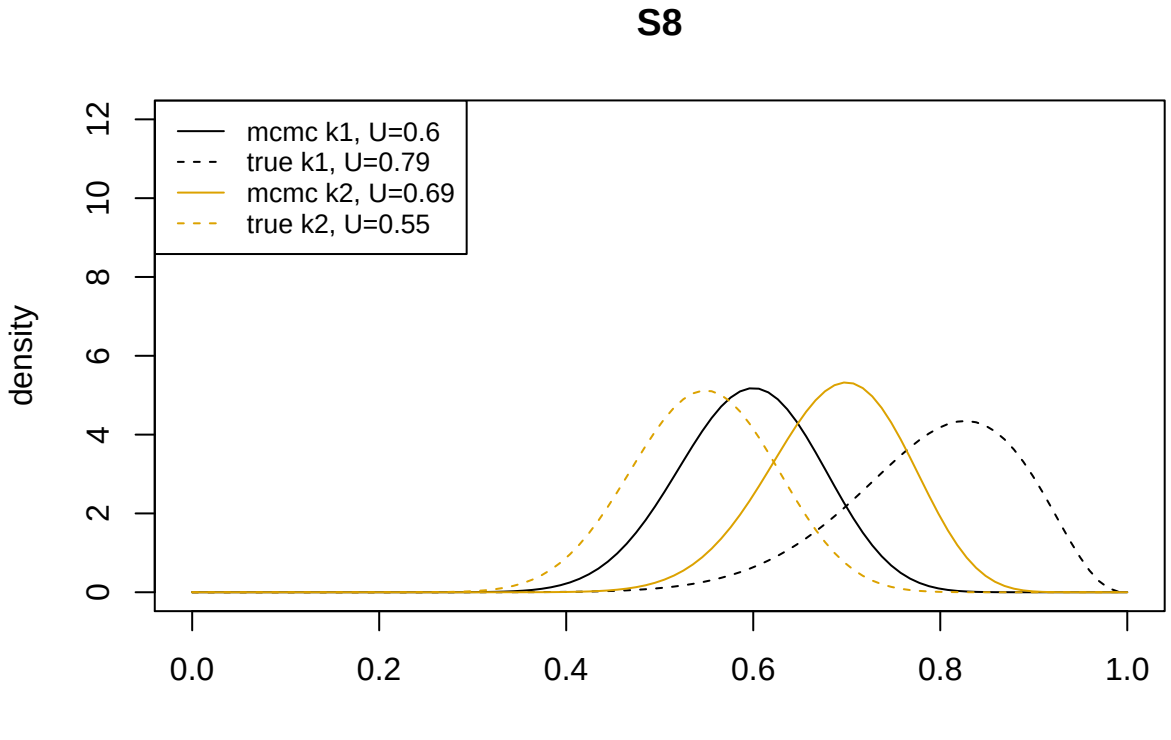

S3
mcmc k1, U=0.63
true k1, U=0.55
mcmc k2, U=0.52
true k2, U=0.62
density
0
2
4
6
8
10
12
0.0
0.2
0.4
0.6
0.8
1.0
w

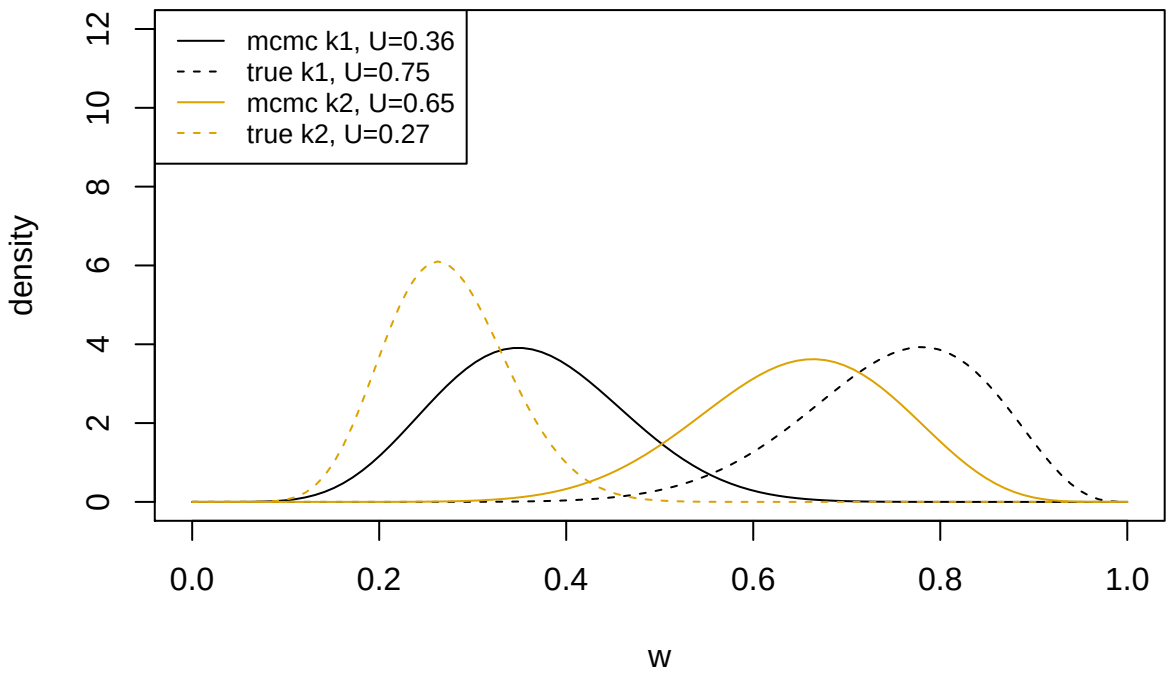

S4
mcmc k1, U=0.33
true k1, U=0.71
mcmc k2, U=0.61
true k2, U=0.25
density
0
2
4
6
8
10
12
0.0
0.2
0.4
0.6
0.8
1.0
w

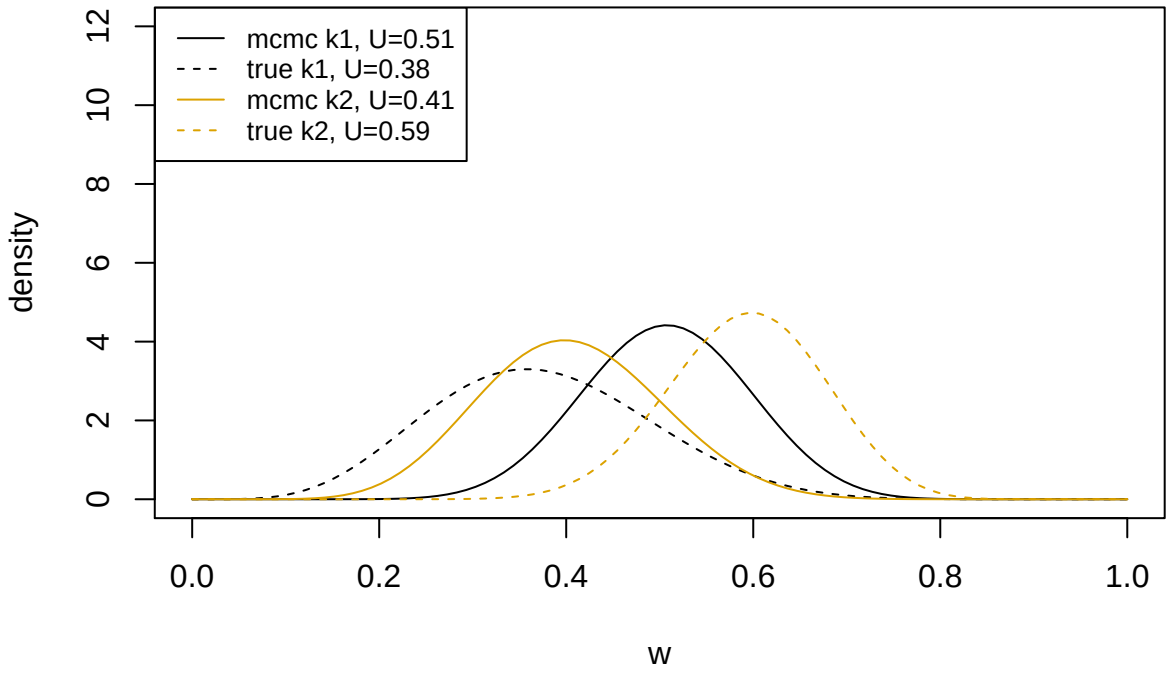

S5
mcmc k1, U=0.64
true k1, U=0.47
mcmc k2, U=0.47
true k2, U=0.66
density
0
2
4
6
8
10
12
0.0
0.2
0.4
0.6
0.8
1.0
w


S6
mcmc k1, U=0.23
true k1, U=0.27
mcmc k2, U=0.35
true k2, U=0.25
density
0
2
4
6
8
10
12
0.0
0.2
0.4
0.6
0.8
1.0
w


S7
mcmc k1, U=0.3
true k1, U=0.35
mcmc k2, U=0.41
true k2, U=0.3
density
0
2
4
6
8
10
12
0.0
0.2
0.4
0.6
0.8
1.0
w


S8
mcmc k1, U=0.6
true k1, U=0.79
mcmc k2, U=0.69
true k2, U=0.55
density
0
2
4
6
8
10
12
0.0
0.2
0.4
0.6
0.8
1.0
w


S9
mcmc k1, U=0.36
true k1, U=0.75
mcmc k2, U=0.65
true k2, U=0.27
density
0
2
4
6
8
10
12
0.0
0.2
0.4
0.6
0.8
1.0
w


S10
mcmc k1, U=0.51
true k1, U=0.38
mcmc k2, U=0.41
true k2, U=0.59
density
0
2
4
6
8
10
12
0.0
0.2
0.4
0.6
0.8
1.0
w