Version 3: 2 clusters

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, "v3_uv_unif.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: 8441
##
## Initializing model
```

```
params <- c("z", "w", "a", "b", "U", "V")
samps <- coda.samples(jags.m, params, n.iter=5000, thin=5)
s <- summary(samps)
effectiveSize(samps)
```

```
##    U[1,1]    U[2,1]    U[1,2]    U[2,2]    U[1,3]    U[2,3]    U[1,4]
## 1000.0000 1000.0000  256.9799  702.7069  842.9127  778.0261  738.5546
##    U[2,4]    U[1,5]    U[2,5]    U[1,6]    U[2,6]    U[1,7]    U[2,7]
##  875.4037  641.6113 1000.0000 1000.0000 1000.0000  863.7917 1000.0000
##    U[1,8]    U[2,8]    U[1,9]    U[2,9]   U[1,10]   U[2,10]    V[1,1]
##  719.0297  706.0704  747.8796 1000.0000  965.4304  834.1777  674.9626
##    V[2,1]    V[1,2]    V[2,2]    V[1,3]    V[2,3]    V[1,4]    V[2,4]
##  693.2280  215.5293  531.4926  681.9607  818.0726  760.2652 1048.9193
##    V[1,5]    V[2,5]    V[1,6]    V[2,6]    V[1,7]    V[2,7]    V[1,8]
##  849.5364  818.4299  731.6126  863.2861  807.2281  729.1037  703.1355
##    V[2,8]    V[1,9]    V[2,9]   V[1,10]   V[2,10]    a[1,1]    a[2,1]
##  780.7133  696.1936  758.4579  727.9651  707.3924  674.0870  702.0654
##    a[1,2]    a[2,2]    a[1,3]    a[2,3]    a[1,4]    a[2,4]    a[1,5]
##  219.3795  529.5931  684.4693  815.7422  768.6839 1000.0000  727.5818
##    a[2,5]    a[1,6]    a[2,6]    a[1,7]    a[2,7]    a[1,8]    a[2,8]
##  820.6246  723.2479 1000.0000  739.3986  777.0393  703.3992  791.1617
##    a[1,9]    a[2,9]   a[1,10]   a[2,10]    b[1,1]    b[2,1]    b[1,2]
##  711.2780  759.7431  754.3253  706.8336  687.2736  679.0334  178.5581
```

```
##    b[2,2]    b[1,3]    b[2,3]    b[1,4]    b[2,4]    b[1,5]    b[2,5]
##  597.4439  685.6306  820.4003  740.5690 1041.0045  855.5054  824.2036
##    b[1,6]    b[2,6]    b[1,7]    b[2,7]    b[1,8]    b[2,8]    b[1,9]
##  739.1395  862.4106  807.5965  730.0136  699.6708  771.2174  670.8544
##    b[2,9]   b[1,10]   b[2,10]    w[1,1]    w[2,1]    w[3,1]    w[4,1]
##  761.3971  719.9837  711.1355 1117.8111  852.0500 1000.0000 1000.0000
##    w[5,1]    w[6,1]    w[7,1]    w[8,1]    w[9,1]   w[10,1]   w[11,1]
## 1000.0000  767.0638 1000.0000 1000.0000  874.9197  885.8370 1000.0000
##   w[12,1]   w[13,1]   w[14,1]   w[15,1]   w[16,1]   w[17,1]   w[18,1]
## 1000.0000 1000.0000 1000.0000  892.8943 1000.0000 1255.3910 1000.0000
##   w[19,1]   w[20,1]   w[21,1]   w[22,1]   w[23,1]   w[24,1]   w[25,1]
## 1000.0000 1095.3388 1000.0000  894.4930 1000.0000 1000.0000 1102.8960
##   w[26,1]   w[27,1]   w[28,1]   w[29,1]   w[30,1]   w[31,1]   w[32,1]
##  903.6346 1200.1514  876.2983  722.3097 1065.7744 1088.9724 1000.0000
##   w[33,1]   w[34,1]   w[35,1]   w[36,1]   w[37,1]   w[38,1]   w[39,1]
##  853.1317  880.4167 1000.0000  900.1388 1000.0000 1000.0000 1000.0000
##   w[40,1]   w[41,1]   w[42,1]   w[43,1]   w[44,1]   w[45,1]   w[46,1]
## 1000.0000  793.5808 1000.0000 1000.0000 1016.3167 1170.4587 1000.0000
##   w[47,1]   w[48,1]   w[49,1]   w[50,1]    w[1,2]    w[2,2]    w[3,2]
##  910.3685  905.9003 1000.0000 1000.0000 1000.0000  416.2398 1000.0000
##    w[4,2]    w[5,2]    w[6,2]    w[7,2]    w[8,2]    w[9,2]   w[10,2]
##  927.2276  584.5166  556.4299 1000.0000 1000.0000  482.4029 1000.0000
##   w[11,2]   w[12,2]   w[13,2]   w[14,2]   w[15,2]   w[16,2]   w[17,2]
## 1000.0000  325.3585 1073.3861  848.4840  869.0474  696.1138  414.8242
##   w[18,2]   w[19,2]   w[20,2]   w[21,2]   w[22,2]   w[23,2]   w[24,2]
## 1272.0751  872.6353 1000.0000  666.5313  430.9048  605.2182  435.1609
##   w[25,2]   w[26,2]   w[27,2]   w[28,2]   w[29,2]   w[30,2]   w[31,2]
## 1000.0000 1000.0000  652.1767  869.4190  823.6072  419.9912 1000.0000
##   w[32,2]   w[33,2]   w[34,2]   w[35,2]   w[36,2]   w[37,2]   w[38,2]
## 1000.0000 1000.0000 1000.0000  836.7131  378.9625  908.2138 1000.0000
##   w[39,2]   w[40,2]   w[41,2]   w[42,2]   w[43,2]   w[44,2]   w[45,2]
##  811.6932  869.1298  853.8004  535.0187 1000.0000 1000.0000  848.3316
##   w[46,2]   w[47,2]   w[48,2]   w[49,2]   w[50,2]    w[1,3]    w[2,3]
## 1000.0000 1000.0000  887.4292 1000.0000  859.5032 1000.0000  895.9346
##    w[3,3]    w[4,3]    w[5,3]    w[6,3]    w[7,3]    w[8,3]    w[9,3]
##  891.0538 1000.0000 1000.0000  887.7219 1000.0000 1291.4518  910.1926
##   w[10,3]   w[11,3]   w[12,3]   w[13,3]   w[14,3]   w[15,3]   w[16,3]
## 1000.0000 1000.0000  904.4895 1000.0000 1039.2549 1158.1962  876.5526
##   w[17,3]   w[18,3]   w[19,3]   w[20,3]   w[21,3]   w[22,3]   w[23,3]
## 1275.9902 1000.0000 1000.0000 1000.0000  899.7196 1070.6746 1095.0845
##   w[24,3]   w[25,3]   w[26,3]   w[27,3]   w[28,3]   w[29,3]   w[30,3]
##  899.4313 1000.0000 1000.0000  850.6619 1000.0000 1000.0000 1000.0000
##   w[31,3]   w[32,3]   w[33,3]   w[34,3]   w[35,3]   w[36,3]   w[37,3]
## 1269.2284  905.3265 1000.0000  956.0225 1000.0000  991.6190  810.5799
##   w[38,3]   w[39,3]   w[40,3]   w[41,3]   w[42,3]   w[43,3]   w[44,3]
##  933.9721  877.3477 1000.0000 1000.0000  880.1645 1000.0000 1000.0000
##   w[45,3]   w[46,3]   w[47,3]   w[48,3]   w[49,3]   w[50,3]    w[1,4]
##  965.6218  638.2399  876.2324  688.3011 1000.0000  909.3068 1000.0000
##    w[2,4]    w[3,4]    w[4,4]    w[5,4]    w[6,4]    w[7,4]    w[8,4]
## 1289.8194 1000.0000  785.6230  820.1322 1000.0000 1000.0000 1000.0000
##    w[9,4]   w[10,4]   w[11,4]   w[12,4]   w[13,4]   w[14,4]   w[15,4]
## 1004.4024 1476.4806 1000.0000 1000.0000 1000.0000 1000.0000 1000.0000
##   w[16,4]   w[17,4]   w[18,4]   w[19,4]   w[20,4]   w[21,4]   w[22,4]
## 1000.0000 1000.0000 1000.0000  881.1063 1078.6278  887.3904  899.3639
```

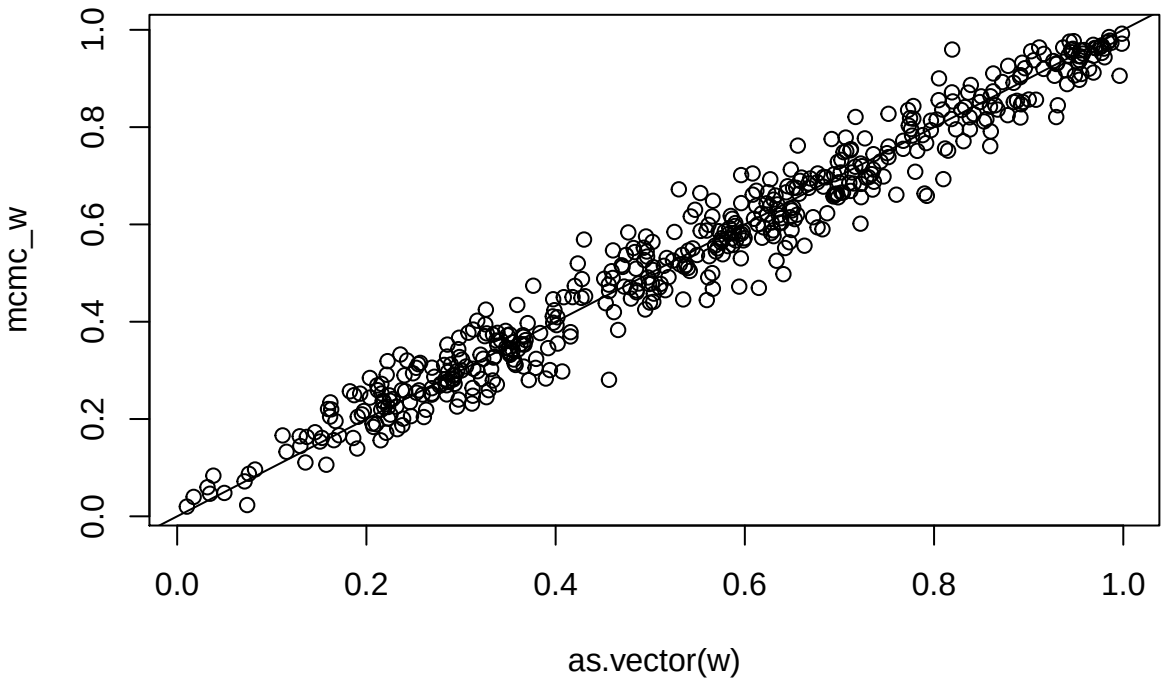
```
##   w[23,4]   w[24,4]   w[25,4]   w[26,4]   w[27,4]   w[28,4]   w[29,4]
## 1000.0000 1000.0000 1000.0000  912.1256  905.5903 1124.4850 1140.6956
##   w[30,4]   w[31,4]   w[32,4]   w[33,4]   w[34,4]   w[35,4]   w[36,4]
## 1000.0000 1000.0000 1000.0000 1155.9722 1000.0000 1000.0000 1000.0000
##   w[37,4]   w[38,4]   w[39,4]   w[40,4]   w[41,4]   w[42,4]   w[43,4]
## 1000.0000 1000.0000  852.2908 1000.0000  916.9343  913.2479 1000.0000
##   w[44,4]   w[45,4]   w[46,4]   w[47,4]   w[48,4]   w[49,4]   w[50,4]
## 1000.0000 1000.0000 1000.0000 1000.0000 1000.0000 1000.0000 1000.0000
##    w[1,5]    w[2,5]    w[3,5]    w[4,5]    w[5,5]    w[6,5]    w[7,5]
## 1112.2081 1000.0000 1000.0000 1000.0000 1000.0000 1000.0000 1000.0000
##    w[8,5]    w[9,5]   w[10,5]   w[11,5]   w[12,5]   w[13,5]   w[14,5]
## 1089.1390 1000.0000 1000.0000 1000.0000  907.7326 1000.0000 1000.0000
##   w[15,5]   w[16,5]   w[17,5]   w[18,5]   w[19,5]   w[20,5]   w[21,5]
##  989.5152 1000.0000  764.9651  857.9399 1000.0000 1000.0000 1000.0000
##   w[22,5]   w[23,5]   w[24,5]   w[25,5]   w[26,5]   w[27,5]   w[28,5]
## 1000.0000 1000.0000 1000.0000 1000.0000 1000.0000 1000.0000  835.2400
##   w[29,5]   w[30,5]   w[31,5]   w[32,5]   w[33,5]   w[34,5]   w[35,5]
## 1000.0000 1000.0000  867.7227 1000.0000 1000.0000 1000.0000  859.2941
##   w[36,5]   w[37,5]   w[38,5]   w[39,5]   w[40,5]   w[41,5]   w[42,5]
## 1436.7994 1000.0000 1000.0000 1000.0000 1000.0000 1000.0000 1000.0000
##   w[43,5]   w[44,5]   w[45,5]   w[46,5]   w[47,5]   w[48,5]   w[49,5]
##  857.1885  847.7568 1000.0000 1000.0000  747.5111 1000.0000  894.4079
##   w[50,5]    w[1,6]    w[2,6]    w[3,6]    w[4,6]    w[5,6]    w[6,6]
## 1000.0000 1000.0000 1011.1857 1000.0000 1000.0000  899.1890 1000.0000
##    w[7,6]    w[8,6]    w[9,6]   w[10,6]   w[11,6]   w[12,6]   w[13,6]
##  830.0012 1180.5176 1000.0000 1000.0000 1000.0000 1000.0000 1000.0000
##   w[14,6]   w[15,6]   w[16,6]   w[17,6]   w[18,6]   w[19,6]   w[20,6]
## 1000.0000 1000.0000 1000.0000 1000.0000 1172.1708 1000.0000 1000.0000
##   w[21,6]   w[22,6]   w[23,6]   w[24,6]   w[25,6]   w[26,6]   w[27,6]
## 1000.0000 1000.0000 1000.0000 1000.0000 1511.5329 1000.0000 1000.0000
##   w[28,6]   w[29,6]   w[30,6]   w[31,6]   w[32,6]   w[33,6]   w[34,6]
## 1000.0000 1254.5453 1000.0000  789.9170 1030.2300 1000.0000  909.7065
##   w[35,6]   w[36,6]   w[37,6]   w[38,6]   w[39,6]   w[40,6]   w[41,6]
##  824.3293 1000.0000 1000.0000  886.6264 1000.0000 1000.0000 1000.0000
##   w[42,6]   w[43,6]   w[44,6]   w[45,6]   w[46,6]   w[47,6]   w[48,6]
## 1000.0000 1000.0000 1000.0000 1000.0000 1000.0000 1000.0000 1000.0000
##   w[49,6]   w[50,6]    w[1,7]    w[2,7]    w[3,7]    w[4,7]    w[5,7]
## 1000.0000 1000.0000  869.4549 1000.0000 1200.1193  784.6369 1000.0000
##    w[6,7]    w[7,7]    w[8,7]    w[9,7]   w[10,7]   w[11,7]   w[12,7]
##  846.1752  896.9306 1000.0000 1134.8039 1000.0000 1000.0000 1000.0000
##   w[13,7]   w[14,7]   w[15,7]   w[16,7]   w[17,7]   w[18,7]   w[19,7]
## 1000.0000 1007.7490 1000.0000 1000.0000 1000.0000  828.0019 1120.6972
##   w[20,7]   w[21,7]   w[22,7]   w[23,7]   w[24,7]   w[25,7]   w[26,7]
## 1000.0000 1000.0000 1103.7198 1000.0000 1000.0000  858.6898 1000.0000
##   w[27,7]   w[28,7]   w[29,7]   w[30,7]   w[31,7]   w[32,7]   w[33,7]
## 1000.0000 1000.0000 1000.0000 1000.0000 1000.0000 1000.0000 1000.0000
##   w[34,7]   w[35,7]   w[36,7]   w[37,7]   w[38,7]   w[39,7]   w[40,7]
##  870.1161 1025.6971 1000.0000  981.3826 1000.0000  485.2174 1000.0000
##   w[41,7]   w[42,7]   w[43,7]   w[44,7]   w[45,7]   w[46,7]   w[47,7]
##  608.6606 1386.3704  569.1887 1067.0675 1000.0000  985.1446 1000.0000
##   w[48,7]   w[49,7]   w[50,7]    w[1,8]    w[2,8]    w[3,8]    w[4,8]
##  910.5211  863.7091 1000.0000 1000.0000  879.9173  777.6573 1000.0000
##    w[5,8]    w[6,8]    w[7,8]    w[8,8]    w[9,8]   w[10,8]   w[11,8]
## 1000.0000 1000.0000 1000.0000 1000.0000  896.2324 1000.0000 1400.0887
```

```
##   w[12,8]   w[13,8]   w[14,8]   w[15,8]   w[16,8]   w[17,8]   w[18,8]
## 1000.0000 1000.0000 1000.0000  961.7502 1000.0000  911.3392 1000.0000
##   w[19,8]   w[20,8]   w[21,8]   w[22,8]   w[23,8]   w[24,8]   w[25,8]
## 1000.0000  874.7106 1000.0000  883.4973 1000.0000 1000.0000 1000.0000
##   w[26,8]   w[27,8]   w[28,8]   w[29,8]   w[30,8]   w[31,8]   w[32,8]
## 1000.0000 1000.0000  812.6587  982.1948 1000.0000 1000.0000 1000.0000
##   w[33,8]   w[34,8]   w[35,8]   w[36,8]   w[37,8]   w[38,8]   w[39,8]
## 1000.0000  919.5174 1000.0000  861.9298 1000.0000  868.9756 1000.0000
##   w[40,8]   w[41,8]   w[42,8]   w[43,8]   w[44,8]   w[45,8]   w[46,8]
## 1000.0000 1000.0000 1000.0000 1110.6697 1000.0000 1000.0000 1000.0000
##   w[47,8]   w[48,8]   w[49,8]   w[50,8]    w[1,9]    w[2,9]    w[3,9]
## 1000.0000 1325.5667  693.3665 1000.0000 1275.4240  868.1897 1000.0000
##    w[4,9]    w[5,9]    w[6,9]    w[7,9]    w[8,9]    w[9,9]   w[10,9]
## 1000.0000  865.2610 1000.0000  910.3242 1000.0000 1000.0000 1000.0000
##   w[11,9]   w[12,9]   w[13,9]   w[14,9]   w[15,9]   w[16,9]   w[17,9]
## 1000.0000 1000.0000 1110.3206  860.6891 1000.0000 1289.6938 1000.0000
##   w[18,9]   w[19,9]   w[20,9]   w[21,9]   w[22,9]   w[23,9]   w[24,9]
## 1000.0000 1000.0000  797.4038 1000.0000 1000.0000  905.3118 1000.0000
##   w[25,9]   w[26,9]   w[27,9]   w[28,9]   w[29,9]   w[30,9]   w[31,9]
## 1000.0000 1000.0000  912.4811  841.8031 1000.0000 1000.0000  898.5178
##   w[32,9]   w[33,9]   w[34,9]   w[35,9]   w[36,9]   w[37,9]   w[38,9]
## 1000.0000 1000.0000 1000.0000 1000.0000  994.3702 1000.0000 1000.0000
##   w[39,9]   w[40,9]   w[41,9]   w[42,9]   w[43,9]   w[44,9]   w[45,9]
##  775.1422 1000.0000  909.5212 1000.0000 1000.0000 1000.0000 1113.5995
##   w[46,9]   w[47,9]   w[48,9]   w[49,9]   w[50,9]   w[1,10]   w[2,10]
##  908.9459 1000.0000 1171.3111 1000.0000  846.7027 1000.0000 1000.0000
##   w[3,10]   w[4,10]   w[5,10]   w[6,10]   w[7,10]   w[8,10]   w[9,10]
## 1000.0000  909.0988 1000.0000 1000.0000 1000.0000 1000.0000  826.3748
##  w[10,10]  w[11,10]  w[12,10]  w[13,10]  w[14,10]  w[15,10]  w[16,10]
##  951.2758 1000.0000 1000.0000  909.1276 1000.0000 1000.0000 1112.8219
##  w[17,10]  w[18,10]  w[19,10]  w[20,10]  w[21,10]  w[22,10]  w[23,10]
## 1182.0908 1000.0000 1000.0000 1000.0000  831.9050 1000.0000 1000.0000
##  w[24,10]  w[25,10]  w[26,10]  w[27,10]  w[28,10]  w[29,10]  w[30,10]
## 1468.6851 1000.0000 1000.0000 1395.3689 1000.0000  805.8156  884.6763
##  w[31,10]  w[32,10]  w[33,10]  w[34,10]  w[35,10]  w[36,10]  w[37,10]
##  954.8265  900.9876 1000.0000 1000.0000 1000.0000 1000.0000 1003.4923
##  w[38,10]  w[39,10]  w[40,10]  w[41,10]  w[42,10]  w[43,10]  w[44,10]
## 1000.0000 1000.0000 1188.0188 1000.0000  890.8982 1027.7424 2075.7946
##  w[45,10]  w[46,10]  w[47,10]  w[48,10]  w[49,10]  w[50,10]      z[1]
## 1000.0000 1124.9216 1000.0000 1000.0000 1000.0000 1123.6960    0.0000
##      z[2]      z[3]      z[4]      z[5]      z[6]      z[7]      z[8]
##    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
##      z[9]     z[10]     z[11]     z[12]     z[13]     z[14]     z[15]
##    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
##     z[16]     z[17]     z[18]     z[19]     z[20]     z[21]     z[22]
##    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
##     z[23]     z[24]     z[25]     z[26]     z[27]     z[28]     z[29]
##    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
##     z[30]     z[31]     z[32]     z[33]     z[34]     z[35]     z[36]
##    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
##     z[37]     z[38]     z[39]     z[40]     z[41]     z[42]     z[43]
##    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
##     z[44]     z[45]     z[46]     z[47]     z[48]     z[49]     z[50]
##    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
```

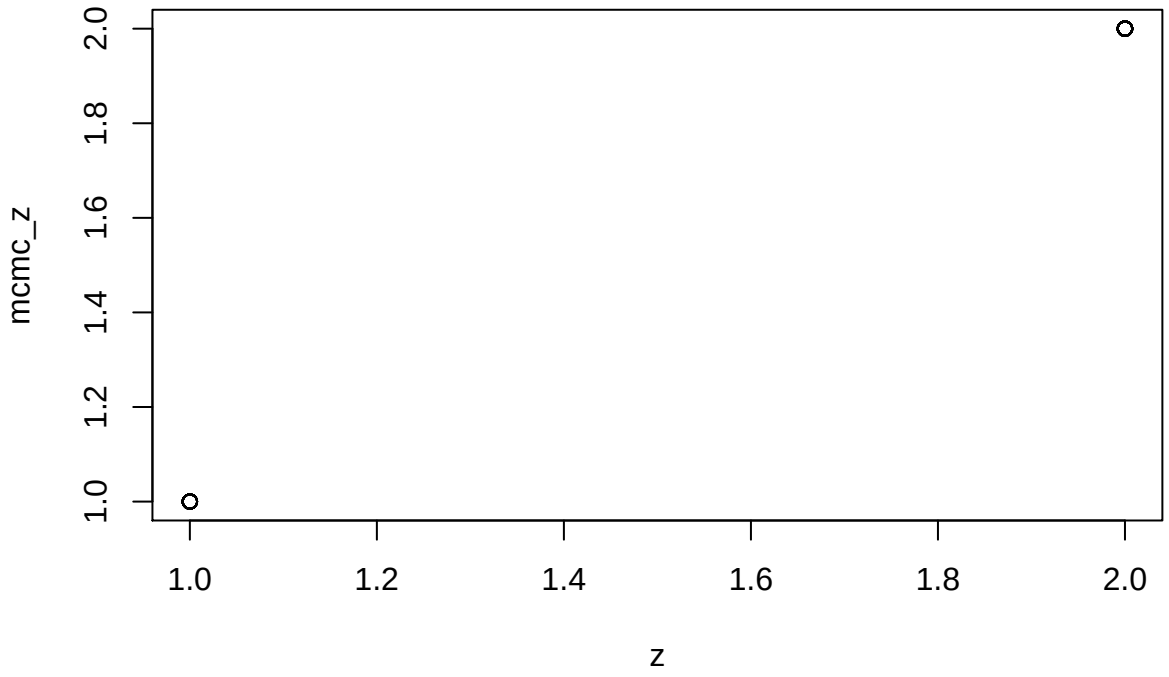
```
pdf(file.path(trace.dir, paste0(runName, "_trace.pdf")))
plot(samps)
dev.off()
```

```
## 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")
abline(a=0, b=1)
```




```
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")
```

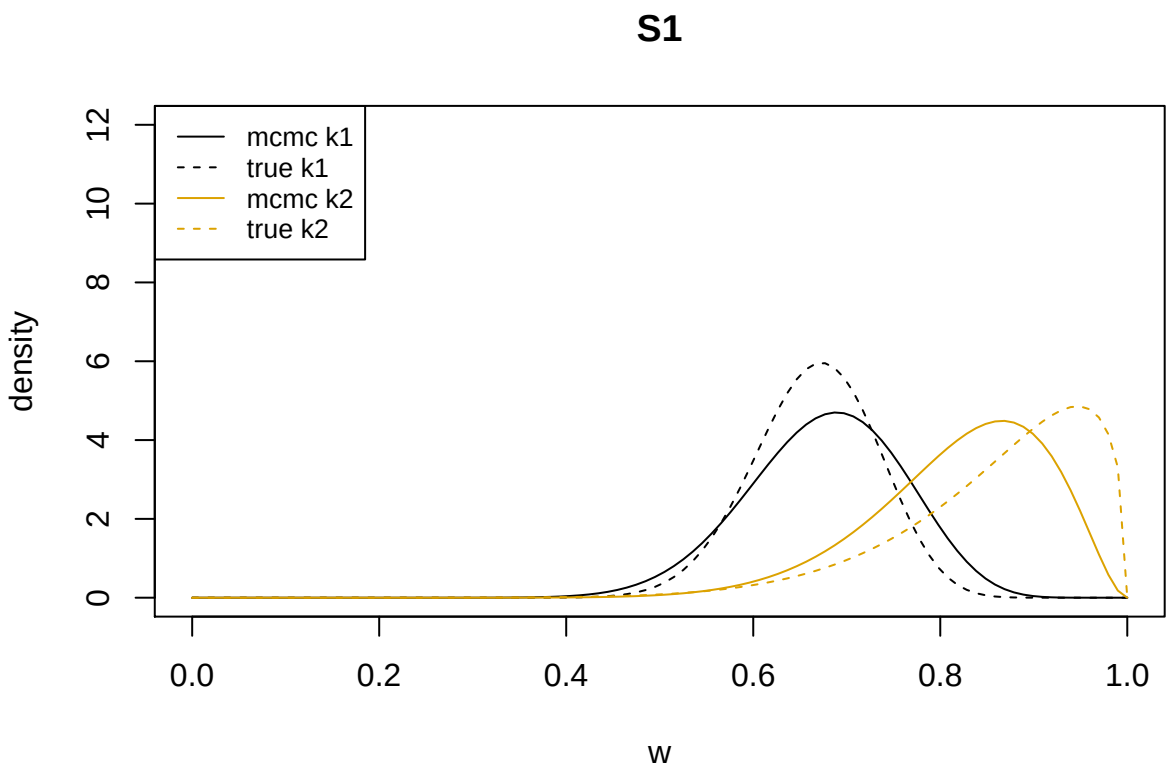


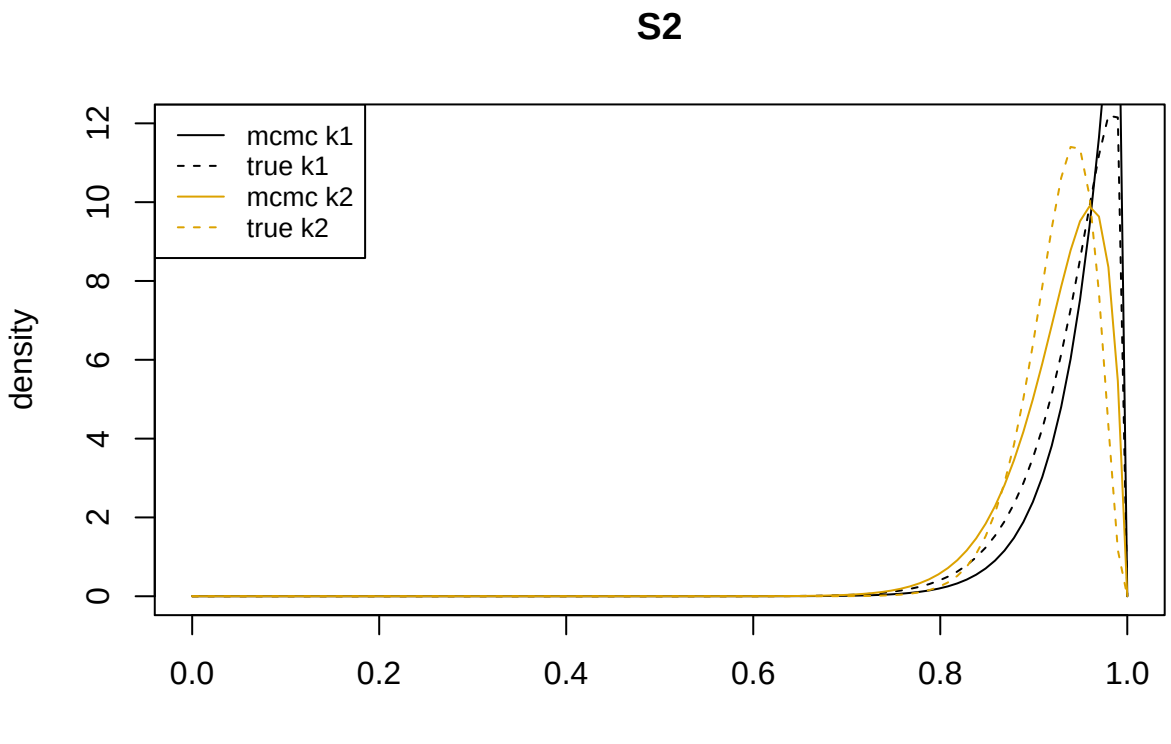

```
mcmc_U <- mcmc_vals[substr(rownames(mcmc_vals), 1, 1) == "U", "Mean"]
mcmc_U <- matrix(mcmc_U, nrow=K)
mcmc_V <- mcmc_vals[substr(rownames(mcmc_vals), 1, 1) == "V", "Mean"]
mcmc_V <- matrix(mcmc_V, nrow=K)

mcmc_U
```

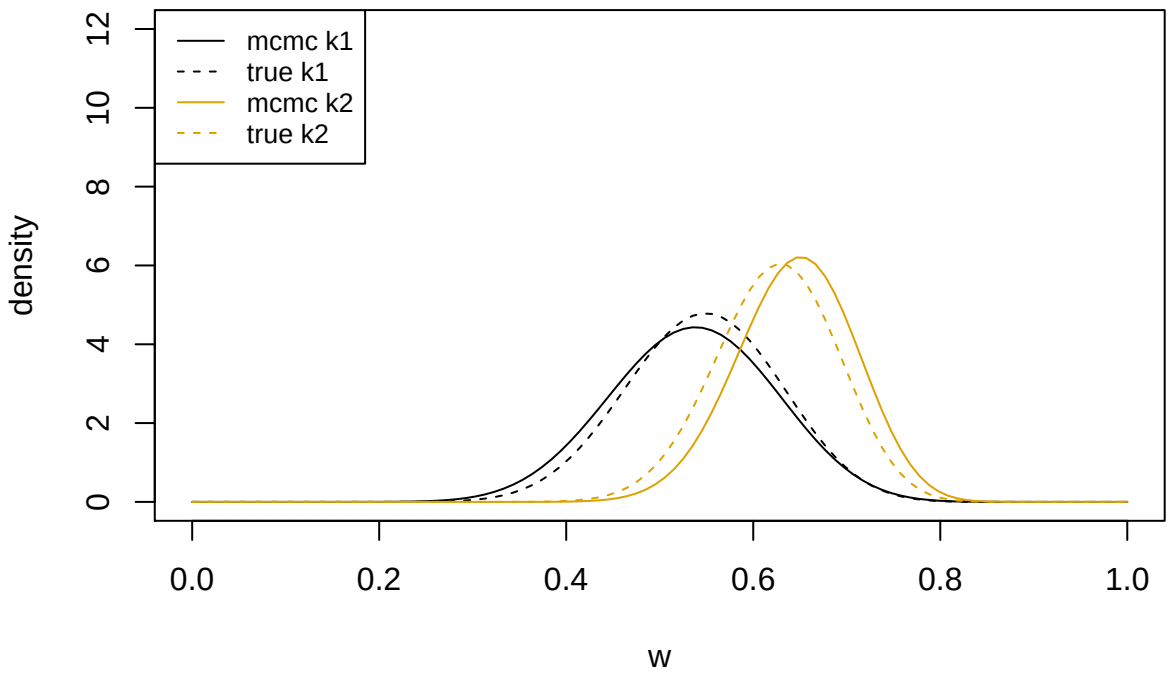
```
##           [,1]      [,2]      [,3]      [,4]      [,5]      [,6]      [,7]
## [1,] 0.6766632 0.9560501 0.5356466 0.6918874 0.4497956 0.2468128 0.3479960
## [2,] 0.8203972 0.9272124 0.6454016 0.2676675 0.6738327 0.2274741 0.2941534
##           [,8]      [,9]     [,10]
## [1,] 0.8277652 0.7478772 0.3196716
## [2,] 0.5536041 0.2885470 0.5406944
```

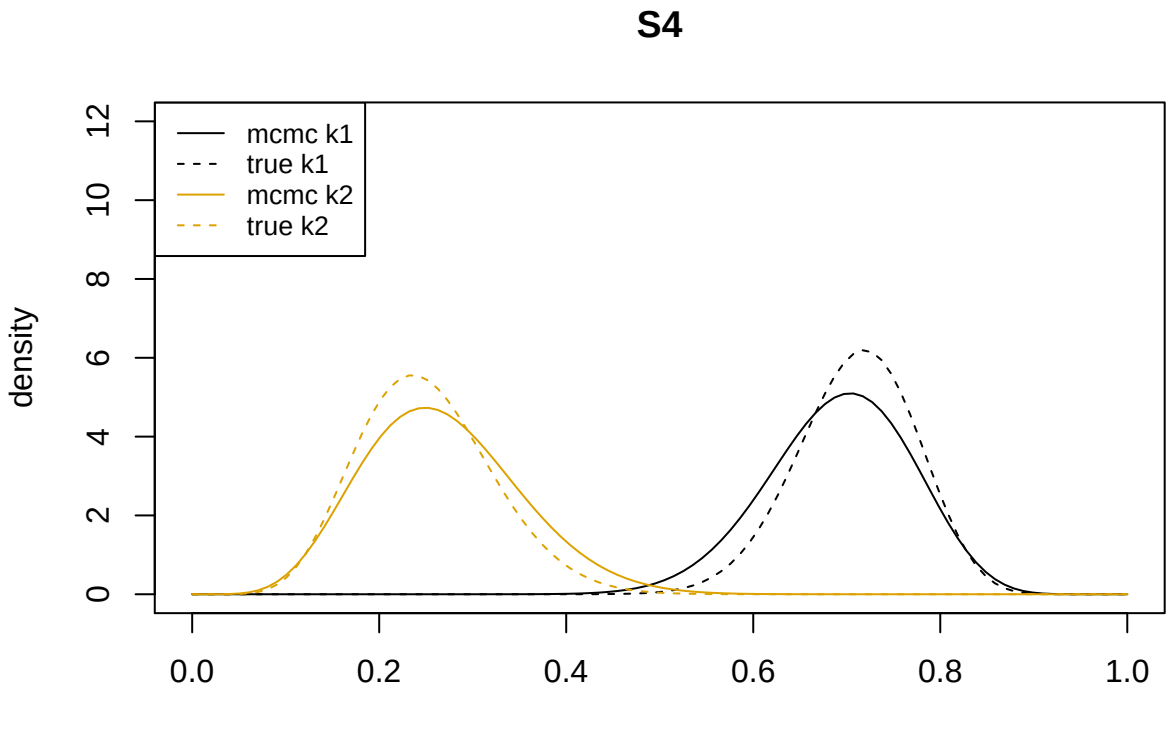
```
p <- seq(0, 1, length = 100)
colors <- c("#000000", "#DCA200", "#8FA7ED", "#9D847A", "#A47901")
for (s in 1:S) {
  for (k in 1:K) {
    if (k == 1) {
      # plot mcmc mean U,V
      plot(p, dbeta(p, mcmc_U[k,s] * mcmc_V[k,s], (1-mcmc_U[k,s])*mcmc_V[k,s]),
           main = paste0("S", s),
           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
      legend(x = "topleft",
             legend = paste0(c("mcmc k", "true k"), rep(1:K, each=2)),
             col = colors[rep(1:K, each=2)],
             lty = rep(1:2, K),
             cex=0.8)
    } else {
```

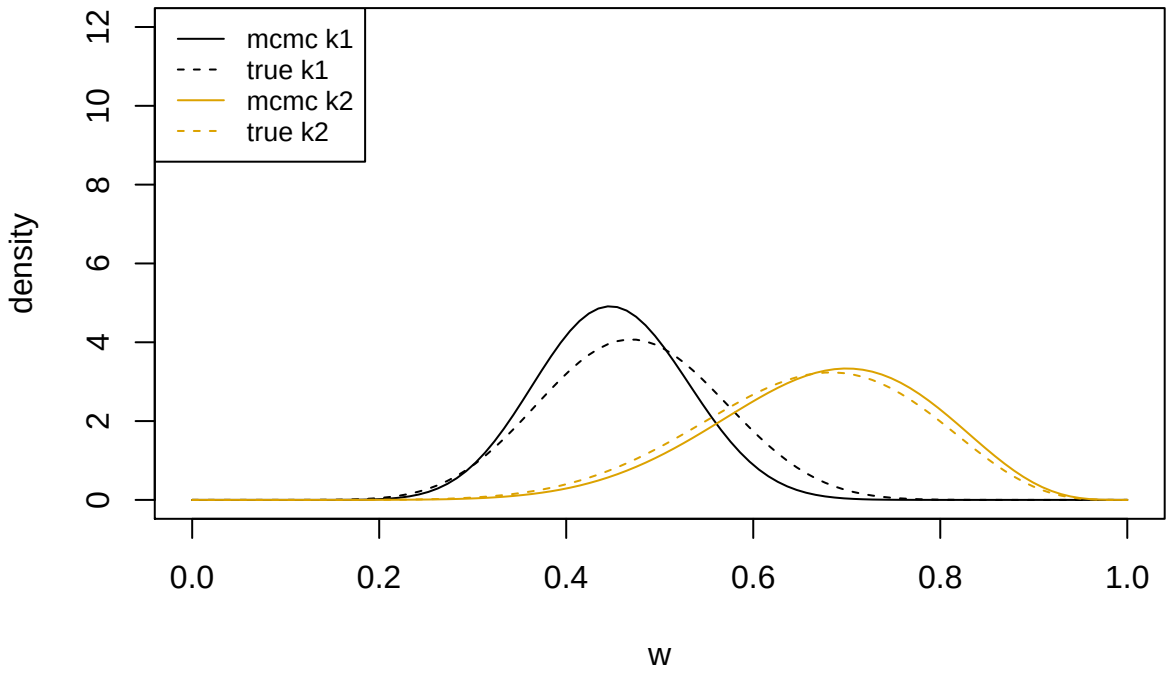
```
      # plot mcmc mean U,V
      lines(p, dbeta(p, mcmc_U[k,s] * mcmc_V[k,s], (1-mcmc_U[k,s])*mcmc_V[k,s]),
            type = "l", col = colors[k])
      # plot truth
      lines(p, dbeta(p, a.ordered[k,s], b.ordered[k,s]), type = "l", col = colors[k], lty=2)
    }
  }
}
```

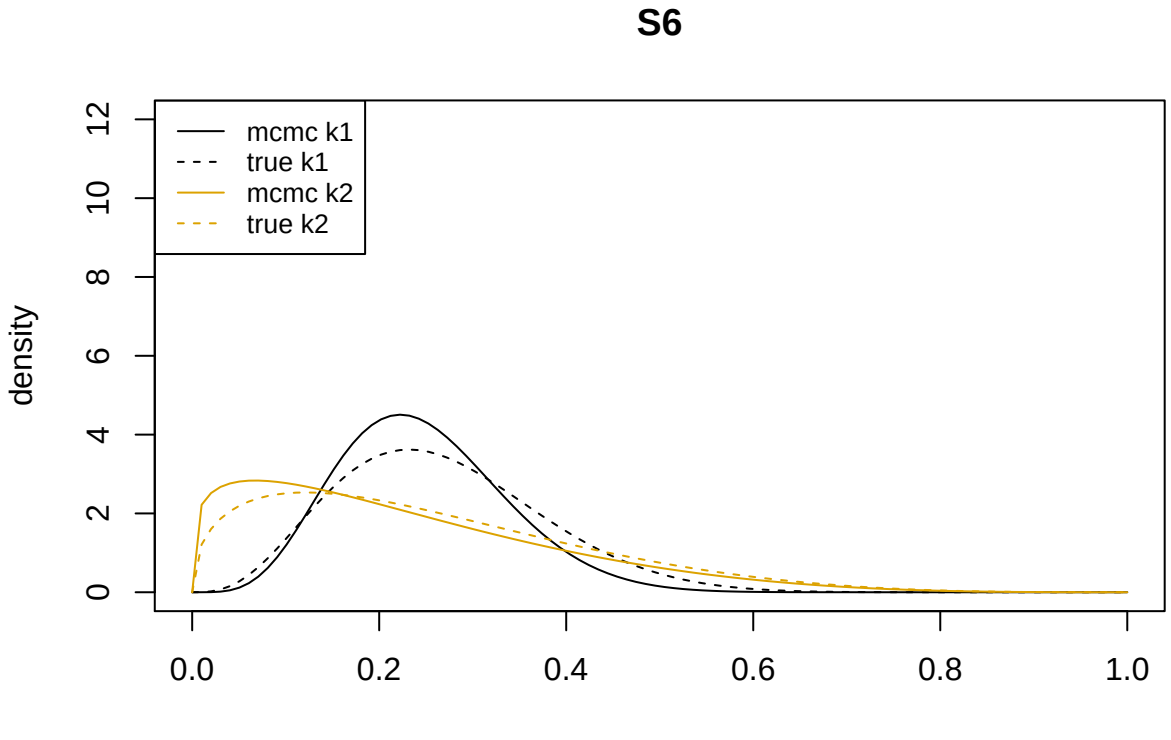


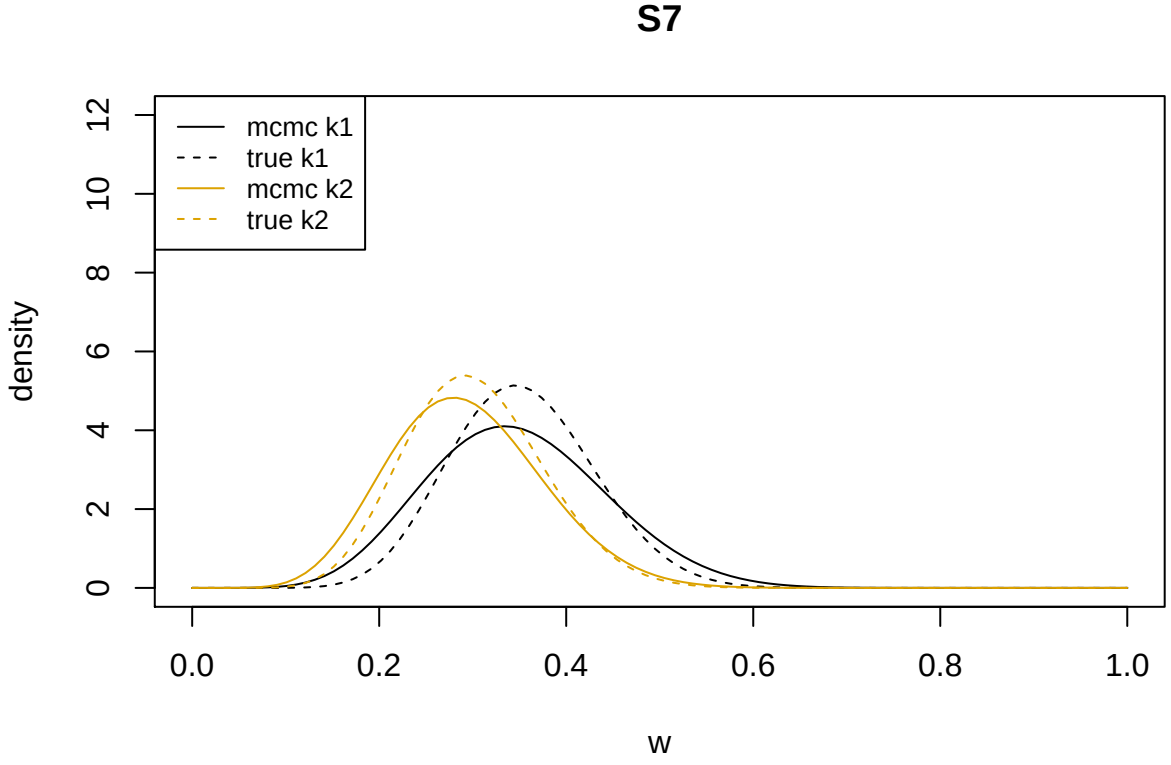

S2
mcmc k1
true k1
mcmc k2
true k2
density
0
2
4
6
8
10
12
0.0
0.2
0.4
0.6
0.8
1.0
w

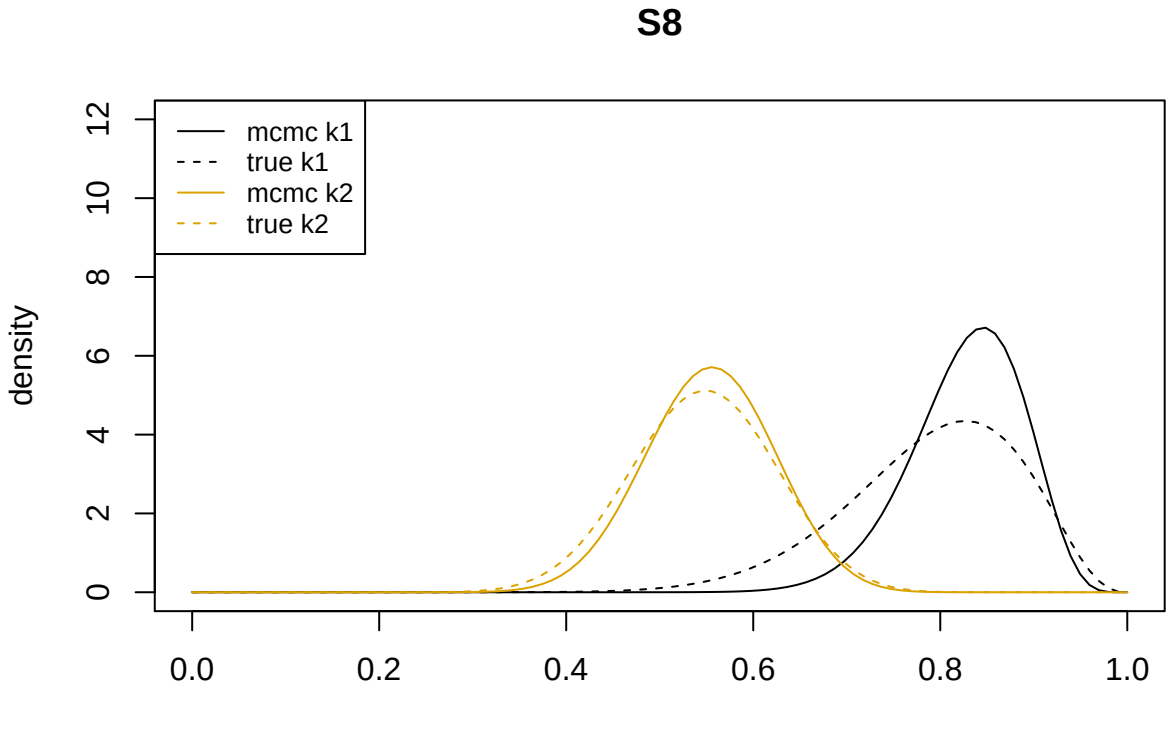

S3
mcmc k1
true k1
mcmc k2
true k2
density
0
2
4
6
8
10
12
0.0
0.2
0.4
0.6
0.8
1.0
w

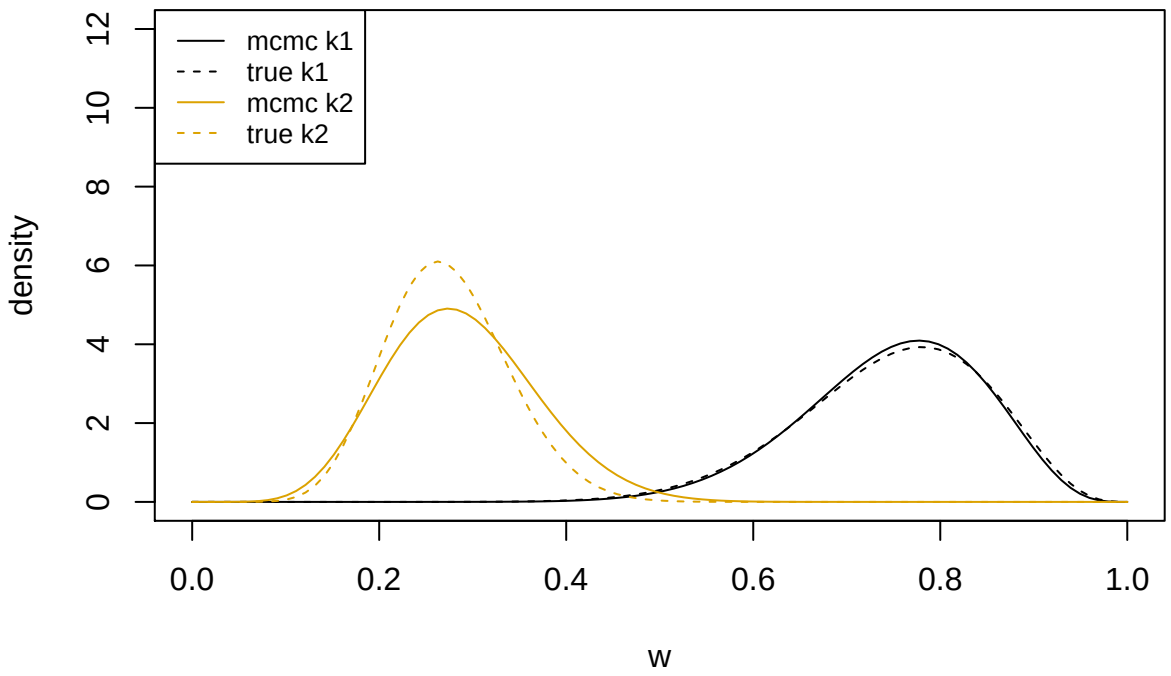

S4
mcmc k1
true k1
mcmc k2
true k2
density
0
2
4
6
8
10
12
0.0
0.2
0.4
0.6
0.8
1.0
w

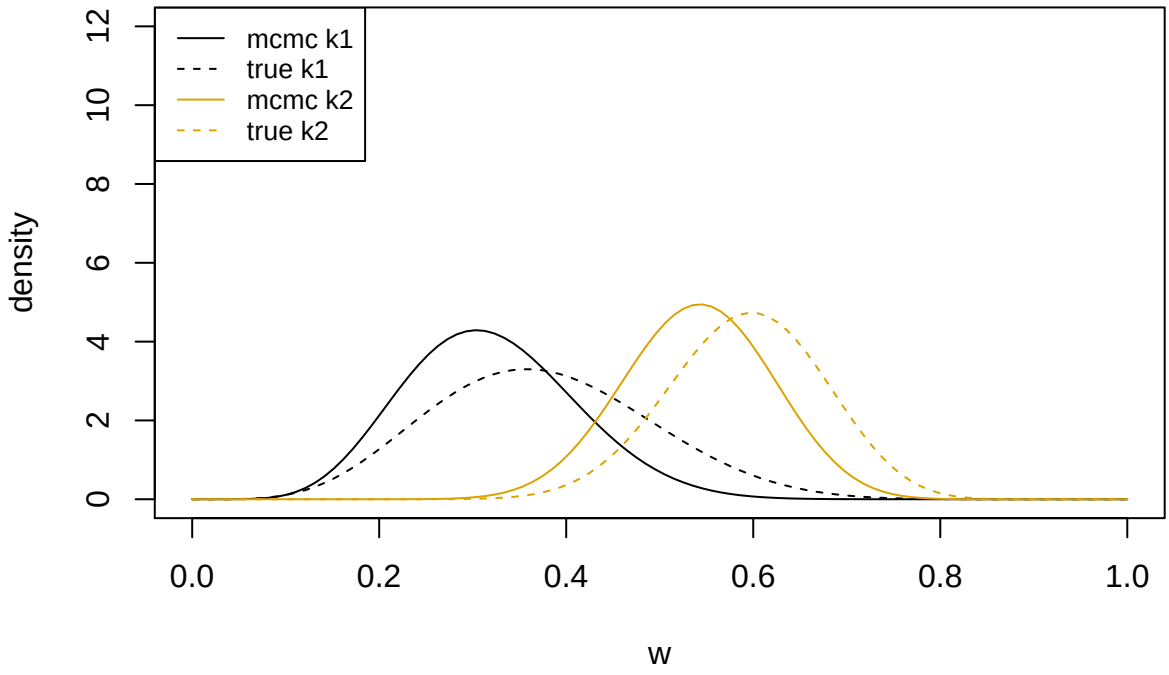

S5
mcmc k1
true k1
mcmc k2
true k2
density
0
2
4
6
8
10
12
0.0
0.2
0.4
0.6
0.8
1.0
w


S6
mcmc k1
true k1
mcmc k2
true k2
density
0
2
4
6
8
10
12
0.0
0.2
0.4
0.6
0.8
1.0
w


S7
mcmc k1
true k1
mcmc k2
true k2
density
0
2
4
6
8
10
12
0.0
0.2
0.4
0.6
0.8
1.0
w


S8
mcmc k1
true k1
mcmc k2
true k2
density
0
2
4
6
8
10
12
0.0
0.2
0.4
0.6
0.8
1.0
w


S9
mcmc k1
true k1
mcmc k2
true k2
density
0
2
4
6
8
10
12
0.0
0.2
0.4
0.6
0.8
1.0
w


S10
mcmc k1
true k1
mcmc k2
true k2
density
0
2
4
6
8
10
12
0.0
0.2
0.4
0.6
0.8
1.0
w