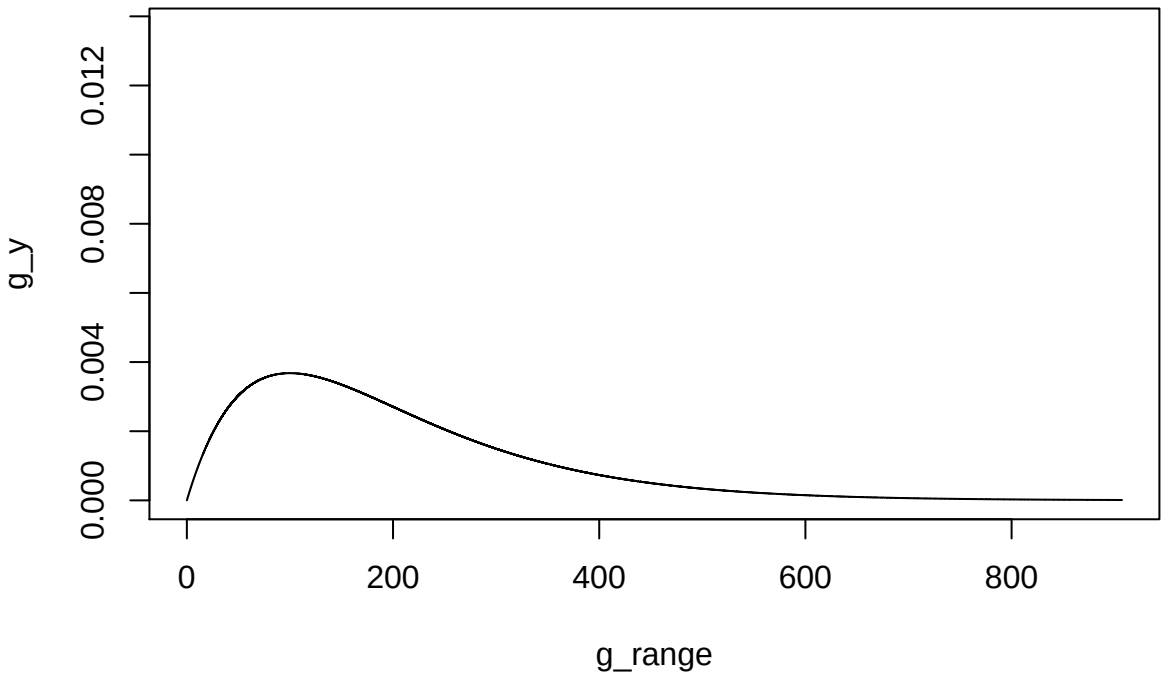
Version 3: 4 clusters; Start chain at true values

Simulate data

```
I <- 50
K <- 4
S <- 10

# choose diffuse priors for gamma
a_gamma <- 2
b_gamma <- 100

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)
trueVals <- list(z = z, w = w, U = U.ordered, V = V.ordered)
jags.m <- jags.model(jags.file, test.data,
                     n.chains = 1,
                     inits = trueVals)
```

```
## Compiling model graph
##    Resolving undeclared variables
##    Allocating nodes
## Graph information:
##    Observed stochastic nodes: 500
##    Unobserved stochastic nodes: 631
##    Total graph size: 8511
##
## Initializing model
```

```
params <- c("z", "w", "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[3,1]    U[4,1]    U[1,2]    U[2,2]    U[3,2]
##  824.8590 1000.0000 1122.3630  570.0951  773.8009  743.4059 1000.0000
##    U[4,2]    U[1,3]    U[2,3]    U[3,3]    U[4,3]    U[1,4]    U[2,4]
##  855.4362  677.6171  952.0267  847.3254  892.4762 1000.0000  785.3142
##    U[3,4]    U[4,4]    U[1,5]    U[2,5]    U[3,5]    U[4,5]    U[1,6]
##  903.3408  851.6784  884.3738  791.6811 1000.0000  869.8115  868.5460
##    U[2,6]    U[3,6]    U[4,6]    U[1,7]    U[2,7]    U[3,7]    U[4,7]
##  811.4524 1000.0000  807.5239  905.9553  750.5871  883.1492  896.0149
##    U[1,8]    U[2,8]    U[3,8]    U[4,8]    U[1,9]    U[2,9]    U[3,9]
##  820.4713  738.9755  904.2757  822.6604  680.8574  885.5605 1000.0000
##    U[4,9]   U[1,10]   U[2,10]   U[3,10]   U[4,10]    V[1,1]    V[2,1]
##  832.9122  899.6473  713.8772 1000.0000  723.7422  762.0330  734.5863
##    V[3,1]    V[4,1]    V[1,2]    V[2,2]    V[3,2]    V[4,2]    V[1,3]
## 1000.0000  797.0539  764.0588  881.6053  708.6376  767.8568  716.4760
##    V[2,3]    V[3,3]    V[4,3]    V[1,4]    V[2,4]    V[3,4]    V[4,4]
##  914.4930  861.1268 1000.0000  829.1130  728.7577  909.2654  912.1397
##    V[1,5]    V[2,5]    V[3,5]    V[4,5]    V[1,6]    V[2,6]    V[3,6]
## 1000.0000  765.1699  831.1822  722.4576  797.2712  891.2556  865.3527
##    V[4,6]    V[1,7]    V[2,7]    V[3,7]    V[4,7]    V[1,8]    V[2,8]
##  999.3725  906.3361  721.1502  874.3283 1026.7412  843.8823 1141.3940
##    V[3,8]    V[4,8]    V[1,9]    V[2,9]    V[3,9]    V[4,9]   V[1,10]
## 1000.0000  902.4251 1000.0000  700.2458  896.9207  798.5557  863.7057
##   V[2,10]   V[3,10]   V[4,10]    w[1,1]    w[2,1]    w[3,1]    w[4,1]
##  709.5176  879.5487  903.8160 1000.0000 1165.2390 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  653.8366 1000.0000 1000.0000  887.2430 1144.5157 1121.4372
##   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 1000.0000  894.3952  813.0101 1122.1703
##   w[19,1]   w[20,1]   w[21,1]   w[22,1]   w[23,1]   w[24,1]   w[25,1]
## 1000.0000 1480.5459 1000.0000 1000.0000 1144.7035 1000.0000  825.1922
##   w[26,1]   w[27,1]   w[28,1]   w[29,1]   w[30,1]   w[31,1]   w[32,1]
## 1000.0000 1000.0000 1000.0000 1233.9758 1000.0000  904.0179 1000.0000
##   w[33,1]   w[34,1]   w[35,1]   w[36,1]   w[37,1]   w[38,1]   w[39,1]
## 1000.0000  888.1010 1292.4506 1000.0000 1000.0000 1000.0000 1099.1772
##   w[40,1]   w[41,1]   w[42,1]   w[43,1]   w[44,1]   w[45,1]   w[46,1]
##  885.3900 1000.0000 1130.5196  822.5855 1000.0000  947.6693 1031.0499
##   w[47,1]   w[48,1]   w[49,1]   w[50,1]    w[1,2]    w[2,2]    w[3,2]
##  940.6353 1000.0000 1000.0000  787.1942 1000.0000 1000.0000 1000.0000
##    w[4,2]    w[5,2]    w[6,2]    w[7,2]    w[8,2]    w[9,2]   w[10,2]
##  843.7971  913.2800 1000.0000  794.2380  802.8197 1000.0000 1267.4954
##   w[11,2]   w[12,2]   w[13,2]   w[14,2]   w[15,2]   w[16,2]   w[17,2]
## 1019.2734 1000.0000  806.3044  871.5754 1000.0000  908.5325 1000.0000
##   w[18,2]   w[19,2]   w[20,2]   w[21,2]   w[22,2]   w[23,2]   w[24,2]
## 1189.3886 1000.0000 1000.0000 1000.0000 1000.0000 1000.0000 1000.0000
##   w[25,2]   w[26,2]   w[27,2]   w[28,2]   w[29,2]   w[30,2]   w[31,2]
##  997.8520 1000.0000 1000.0000 1000.0000  854.9457 1000.0000 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  773.3692 1148.3563 1015.8708 1000.0000 1000.0000
##   w[39,2]   w[40,2]   w[41,2]   w[42,2]   w[43,2]   w[44,2]   w[45,2]
## 1000.0000 1122.4471 1000.0000 1000.0000 1000.0000  887.4853 1000.0000
##   w[46,2]   w[47,2]   w[48,2]   w[49,2]   w[50,2]    w[1,3]    w[2,3]
```

```
## 1000.0000  894.1919 1000.0000  903.4465  790.6403 1000.0000 1000.0000
##    w[3,3]    w[4,3]    w[5,3]    w[6,3]    w[7,3]    w[8,3]    w[9,3]
## 1000.0000 1000.0000  905.8965 1000.0000 1000.0000 1139.2165  847.5091
##   w[10,3]   w[11,3]   w[12,3]   w[13,3]   w[14,3]   w[15,3]   w[16,3]
## 1000.0000 1053.4069 1000.0000 1000.0000 1000.0000 1323.9734 1000.0000
##   w[17,3]   w[18,3]   w[19,3]   w[20,3]   w[21,3]   w[22,3]   w[23,3]
##  911.9403  888.2510 1000.0000 1000.0000 1000.0000 1000.0000  844.5719
##   w[24,3]   w[25,3]   w[26,3]   w[27,3]   w[28,3]   w[29,3]   w[30,3]
##  830.7378 1000.0000 1000.0000 1000.0000 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]
##  887.8167 1000.0000 1000.0000 1125.5027 1000.0000  781.9314 1000.0000
##   w[38,3]   w[39,3]   w[40,3]   w[41,3]   w[42,3]   w[43,3]   w[44,3]
## 1000.0000  890.4852 1000.0000 1000.0000 1000.0000 1000.0000 1045.7993
##   w[45,3]   w[46,3]   w[47,3]   w[48,3]   w[49,3]   w[50,3]    w[1,4]
## 1000.0000 1000.0000 1000.0000 1000.0000 1000.0000 1235.2223 1000.0000
##    w[2,4]    w[3,4]    w[4,4]    w[5,4]    w[6,4]    w[7,4]    w[8,4]
## 1000.0000 1000.0000 1000.0000 1000.0000 1000.0000  859.7030 1315.7938
##    w[9,4]   w[10,4]   w[11,4]   w[12,4]   w[13,4]   w[14,4]   w[15,4]
## 1000.0000 1000.0000  897.0244 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 1000.0000 1099.9566 1000.0000 1000.0000
##   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 1041.3798 1071.1415 1322.3405 1066.9368 1000.0000
##   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 1115.4589 1000.0000 1000.0000 1034.1306 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  906.3789 1000.0000 1000.0000  845.4595  805.5520 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 1076.6870 1000.0000  860.5011 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]
## 1000.0000 1000.0000 1000.0000 1000.0000 1076.3803  792.8326 1000.0000
##    w[8,5]    w[9,5]   w[10,5]   w[11,5]   w[12,5]   w[13,5]   w[14,5]
## 1000.0000  817.8165 1000.0000  795.4027  892.9866 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]
## 1000.0000 1000.0000 1000.0000 1000.0000 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]
##  805.1937  858.5528 1000.0000 1234.8514  888.9383 1000.0000  802.6385
##   w[29,5]   w[30,5]   w[31,5]   w[32,5]   w[33,5]   w[34,5]   w[35,5]
##  905.7567 1000.0000 1000.0000 1013.1435  883.0151 1000.0000 1000.0000
##   w[36,5]   w[37,5]   w[38,5]   w[39,5]   w[40,5]   w[41,5]   w[42,5]
## 1000.0000 1000.0000  907.7714 1000.0000 1000.0000  861.8868 1000.0000
##   w[43,5]   w[44,5]   w[45,5]   w[46,5]   w[47,5]   w[48,5]   w[49,5]
## 1000.0000  854.6283 1000.0000 1000.0000 1000.0000 1100.1420 1109.2453
##   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 1289.2833 1000.0000 1000.0000 1000.0000 1000.0000
##    w[7,6]    w[8,6]    w[9,6]   w[10,6]   w[11,6]   w[12,6]   w[13,6]
## 1000.0000 1000.0000  857.0519  894.2778  977.8315 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  678.1459 1165.1737 1000.0000 1000.0000 1000.0000  879.9876
##   w[21,6]   w[22,6]   w[23,6]   w[24,6]   w[25,6]   w[26,6]   w[27,6]
## 1213.8018  895.9659 1485.4551 1125.5871  857.6942  816.0906 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 1000.0000 1000.0000 1000.0000 1000.0000 1000.0000 1000.0000
##   w[35,6]   w[36,6]   w[37,6]   w[38,6]   w[39,6]   w[40,6]   w[41,6]
```

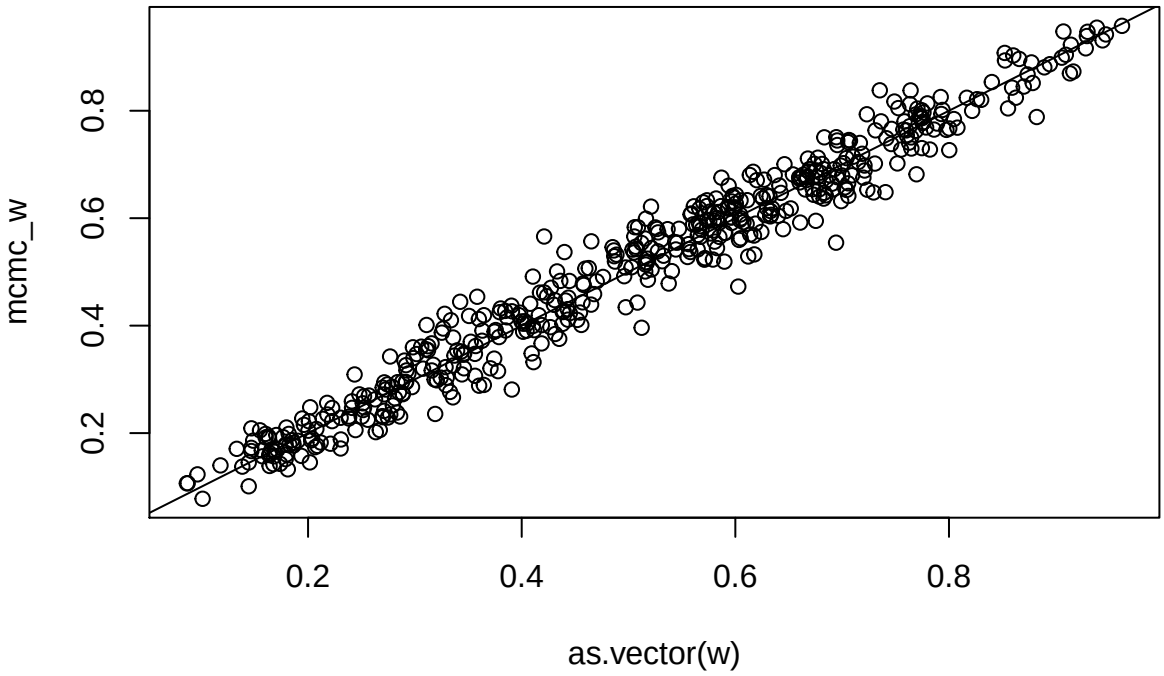
```
## 1000.0000  900.9138  727.6785 1000.0000 1189.0154 1000.0000 1000.0000
##   w[42,6]   w[43,6]   w[44,6]   w[45,6]   w[46,6]   w[47,6]   w[48,6]
## 1000.0000 1000.0000 1174.1511 1538.4433 1000.0000  791.3820 1000.0000
##   w[49,6]   w[50,6]    w[1,7]    w[2,7]    w[3,7]    w[4,7]    w[5,7]
##  826.8098 1000.0000 1000.0000 1000.0000 1998.3508 1000.0000 1000.0000
##    w[6,7]    w[7,7]    w[8,7]    w[9,7]   w[10,7]   w[11,7]   w[12,7]
## 1000.0000 1000.0000 1000.0000  836.3524 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 1000.0000 1000.0000  848.9852 1000.0000 1000.0000 1000.0000
##   w[20,7]   w[21,7]   w[22,7]   w[23,7]   w[24,7]   w[25,7]   w[26,7]
##  828.0505 1000.0000 1000.0000 1000.0000 1000.0000 1000.0000 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 1151.2327 1000.0000  906.0542 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]
##  904.7782 1000.0000 1000.0000 1000.0000  867.4358  872.7508 1227.3514
##   w[41,7]   w[42,7]   w[43,7]   w[44,7]   w[45,7]   w[46,7]   w[47,7]
## 1000.0000 1000.0000 1173.2900 1000.0000 1000.0000 1000.0000  909.6508
##   w[48,7]   w[49,7]   w[50,7]    w[1,8]    w[2,8]    w[3,8]    w[4,8]
## 1000.0000 1000.0000 1000.0000 1320.7171 1000.0000 1000.0000  944.8989
##    w[5,8]    w[6,8]    w[7,8]    w[8,8]    w[9,8]   w[10,8]   w[11,8]
##  893.5089  825.7379  879.2616  884.3888  871.1645 1139.1424 1000.0000
##   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  907.5177 1000.0000 1000.0000 1000.0000  802.0154
##   w[19,8]   w[20,8]   w[21,8]   w[22,8]   w[23,8]   w[24,8]   w[25,8]
## 1110.0539 1000.0000 1041.4970 1000.0000  903.9516 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]
##  933.4378 1045.9418 1056.8983 1000.0000 1000.0000 1000.0000 1102.6235
##   w[33,8]   w[34,8]   w[35,8]   w[36,8]   w[37,8]   w[38,8]   w[39,8]
## 1000.0000  882.7981 1000.0000  879.0079 1000.0000 1000.0000  962.7366
##   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  845.6628  905.8915 1149.7919  906.9104 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 1000.0000 1000.0000 1000.0000 1000.0000 1000.0000 1000.0000
##    w[4,9]    w[5,9]    w[6,9]    w[7,9]    w[8,9]    w[9,9]   w[10,9]
## 1024.6700  905.5542 1000.0000 1000.0000 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 1000.0000 1000.0000 1000.0000  847.5864 1099.7639
##   w[18,9]   w[19,9]   w[20,9]   w[21,9]   w[22,9]   w[23,9]   w[24,9]
## 1077.1693 1000.0000 1061.1832 1000.0000 1000.0000 1000.0000 1244.9454
##   w[25,9]   w[26,9]   w[27,9]   w[28,9]   w[29,9]   w[30,9]   w[31,9]
## 1000.0000  912.2399 1000.0000 1000.0000 1000.0000 1000.0000 1000.0000
##   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 1515.4593 1000.0000 1000.0000 1000.0000 1010.2661
##   w[39,9]   w[40,9]   w[41,9]   w[42,9]   w[43,9]   w[44,9]   w[45,9]
##  703.6186 1000.0000 1000.0000 1094.0513  888.5477 1000.0000 1000.0000
##   w[46,9]   w[47,9]   w[48,9]   w[49,9]   w[50,9]   w[1,10]   w[2,10]
## 1035.3232 1094.8581  737.0093 1022.0052 1000.0000 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  852.8233 1000.0000 1000.0000 1226.1767 1000.0000 1000.0000
##  w[10,10]  w[11,10]  w[12,10]  w[13,10]  w[14,10]  w[15,10]  w[16,10]
## 1000.0000  904.7942  891.9409 1000.0000 1000.0000 1000.0000 1038.7589
##  w[17,10]  w[18,10]  w[19,10]  w[20,10]  w[21,10]  w[22,10]  w[23,10]
## 1399.1671 1000.0000 1000.0000  898.8041 1000.0000 1216.7520 1102.6830
##  w[24,10]  w[25,10]  w[26,10]  w[27,10]  w[28,10]  w[29,10]  w[30,10]
```

```
## 1000.0000  908.2210  911.6568 1000.0000 1000.0000 1000.0000 1000.0000
##  w[31,10]  w[32,10]  w[33,10]  w[34,10]  w[35,10]  w[36,10]  w[37,10]
## 1000.0000 1000.0000  856.0824  834.3187  891.1343 1000.0000 1000.0000
##  w[38,10]  w[39,10]  w[40,10]  w[41,10]  w[42,10]  w[43,10]  w[44,10]
## 1317.2519 1013.8986 1000.0000 1000.0000 1098.6374  818.1584 1000.0000
##  w[45,10]  w[46,10]  w[47,10]  w[48,10]  w[49,10]  w[50,10]      z[1]
## 1019.4989 1212.4996 1000.0000 1221.7952 1000.0000 1000.0000    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 1000.0000    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
```

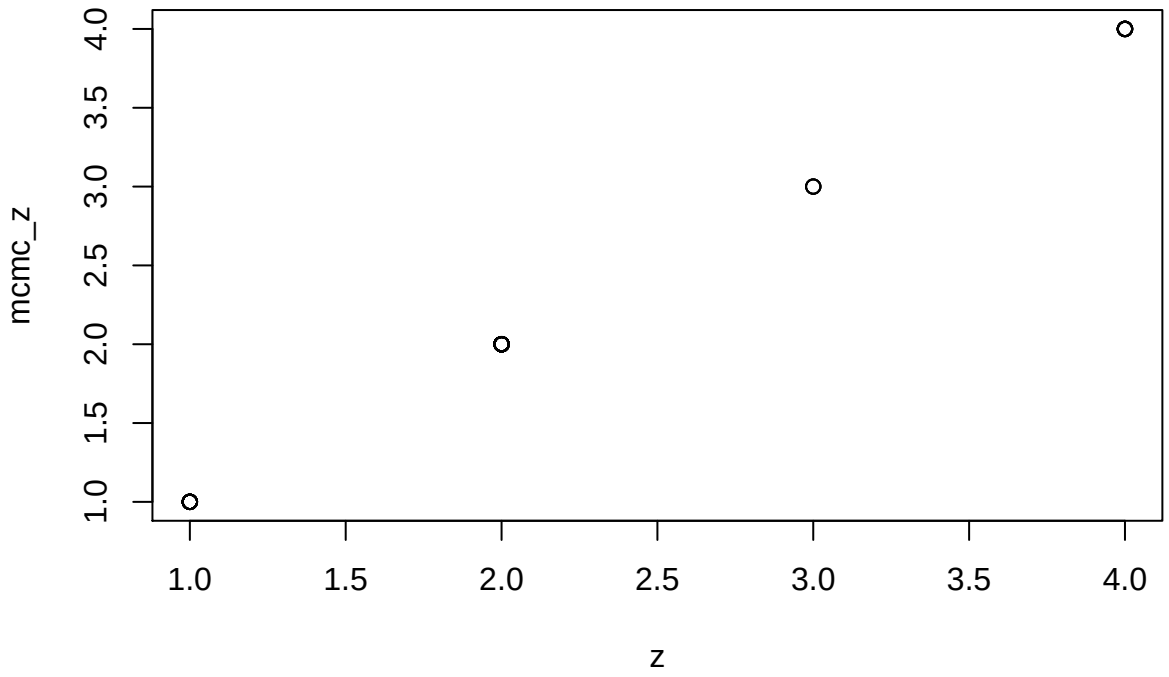
```
pdf(file.path(trace.dir, paste0(runName, "_trace.pdf")))
plot(samps)
dev.off()
```

```
## 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)
```

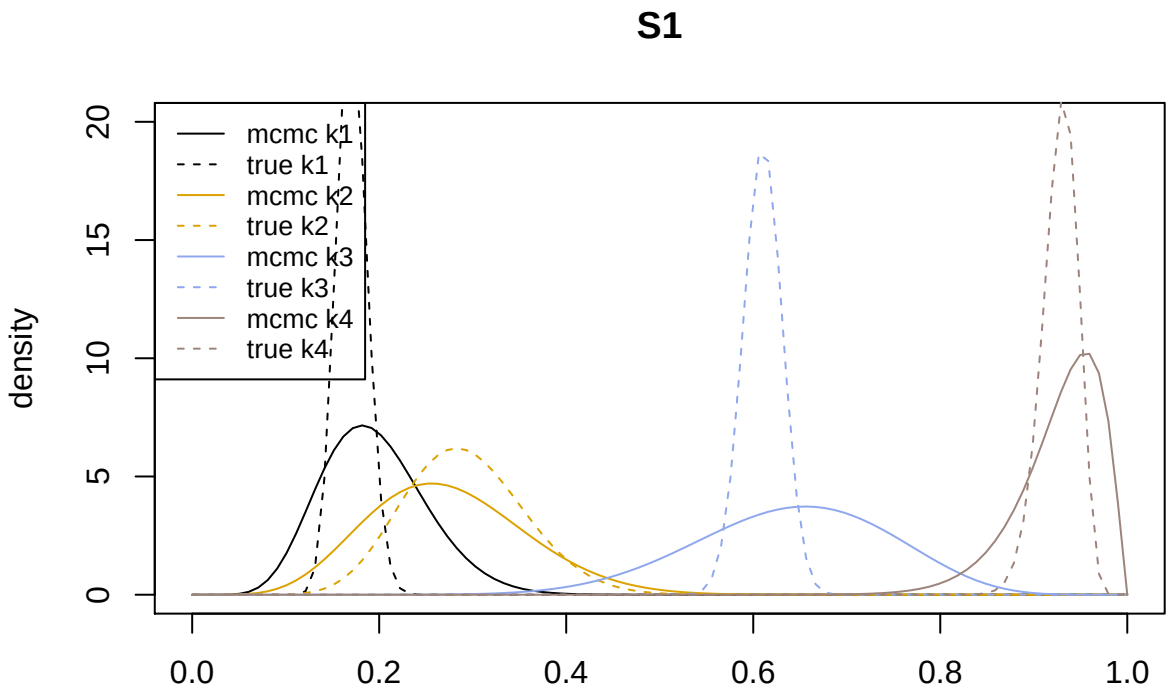


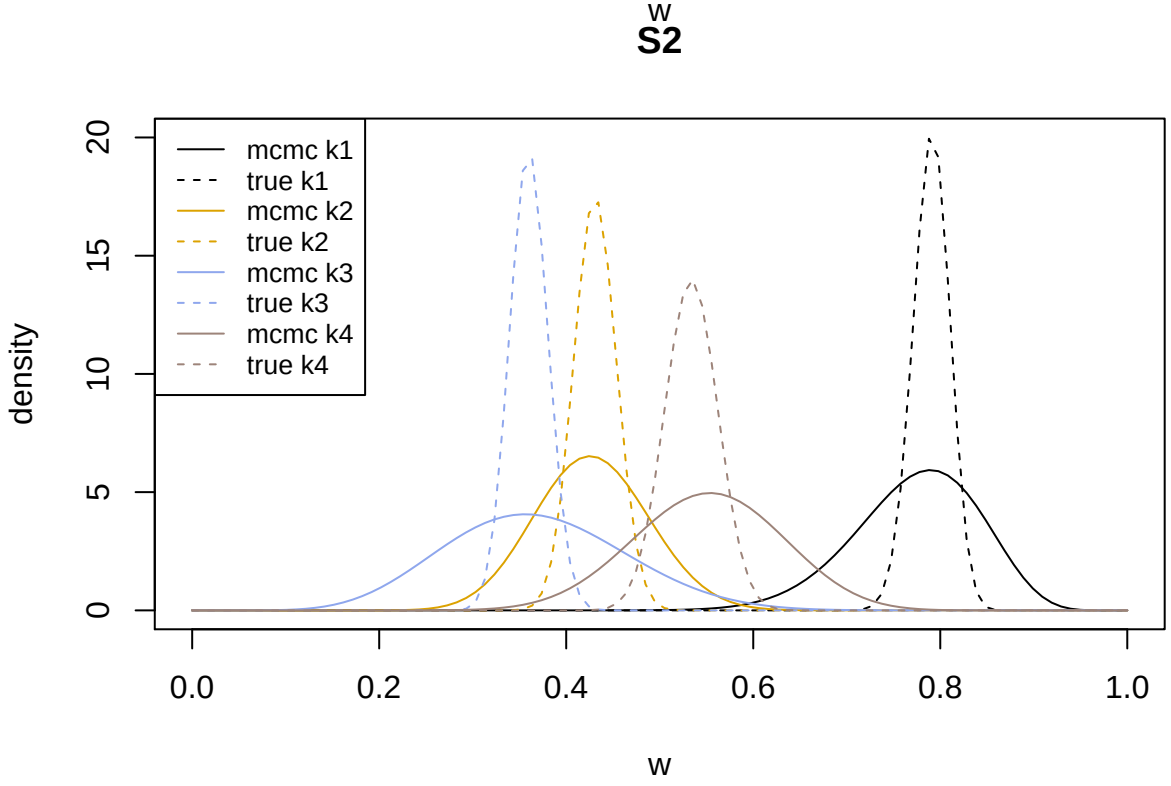
```r
mcmc_z <- as.vector(mcmc_vals[substr(rownames(mcmc_vals), 1, 1) == "z", "Mean"])
#mcmc_z <- round(mcmc_z, 0)
plot(z, mcmc_z, type = "p")
```

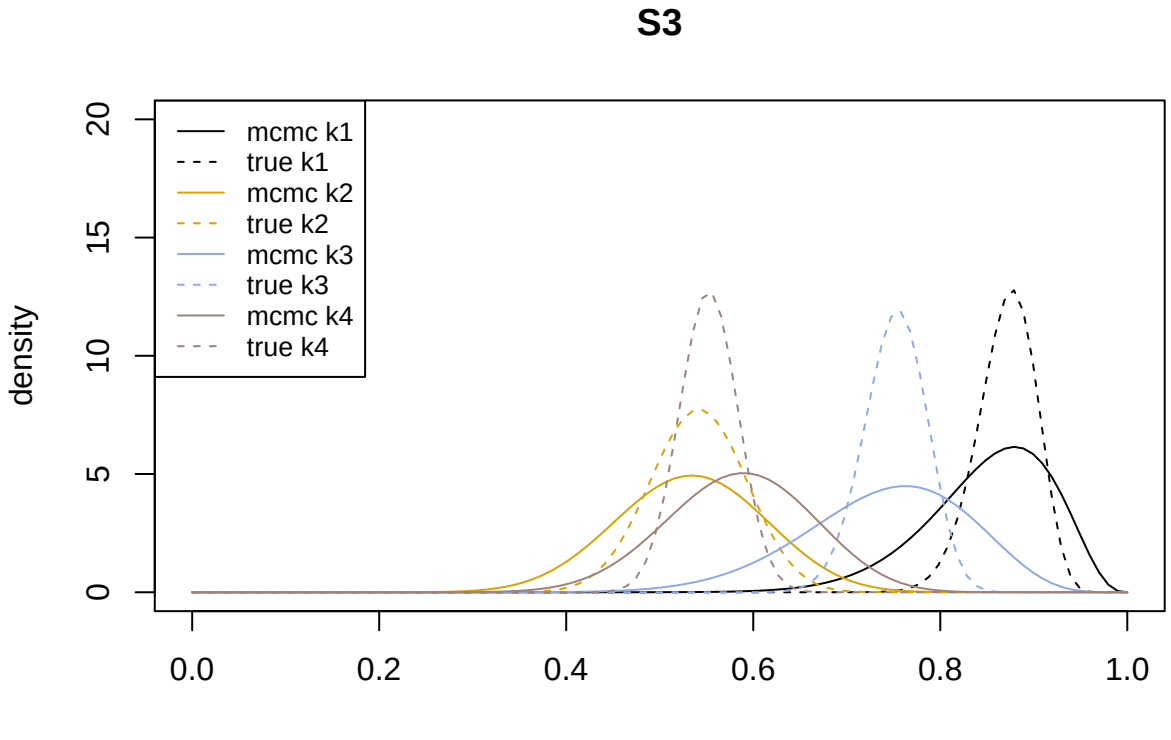


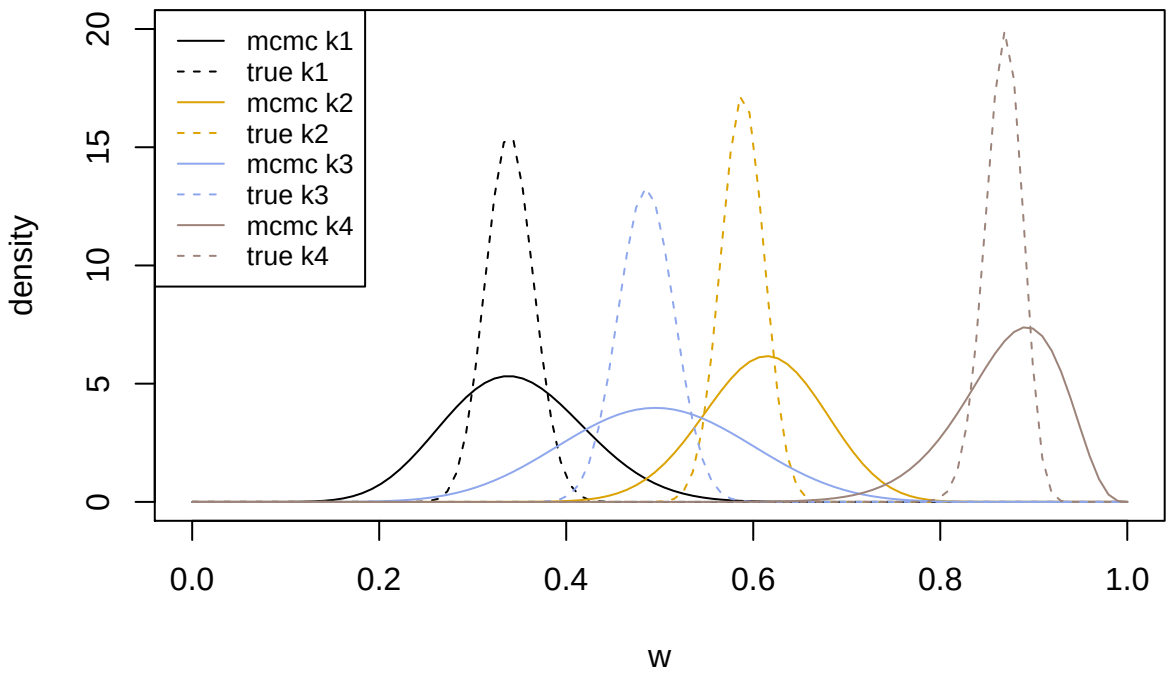

```r
mcmc_U <- mcmc_vals[substr(rownames(mcmc_vals), 1, 1) == "U", "Mean"]
mcmc_U <- matrix(mcmc_U, nrow=K)
mcmc_V <- mcmc_vals[substr(rownames(mcmc_vals), 1, 1) == "V", "Mean"]
mcmc_V <- matrix(mcmc_V, nrow=K)

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, 20))
      # 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
```

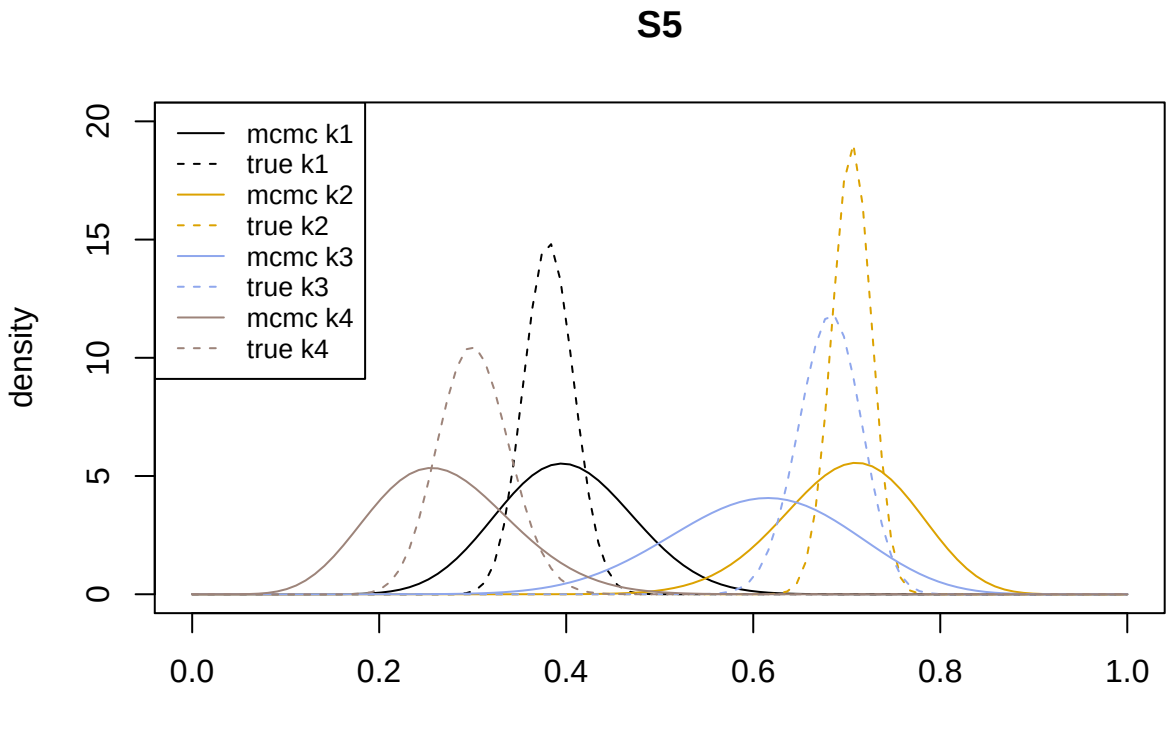
```
      lines(p, dbeta(p, a.ordered[k,s], b.ordered[k,s]), type = "l", col = colors[k], lty=2)
    }
  }
}
```

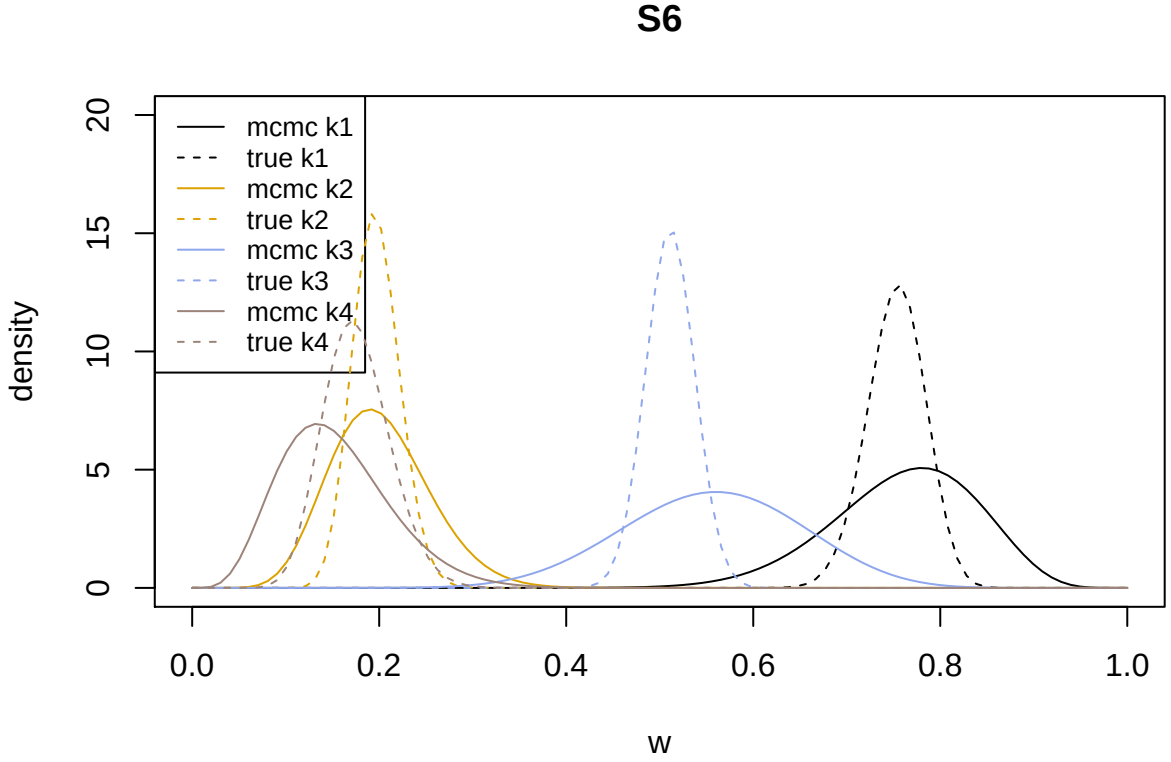


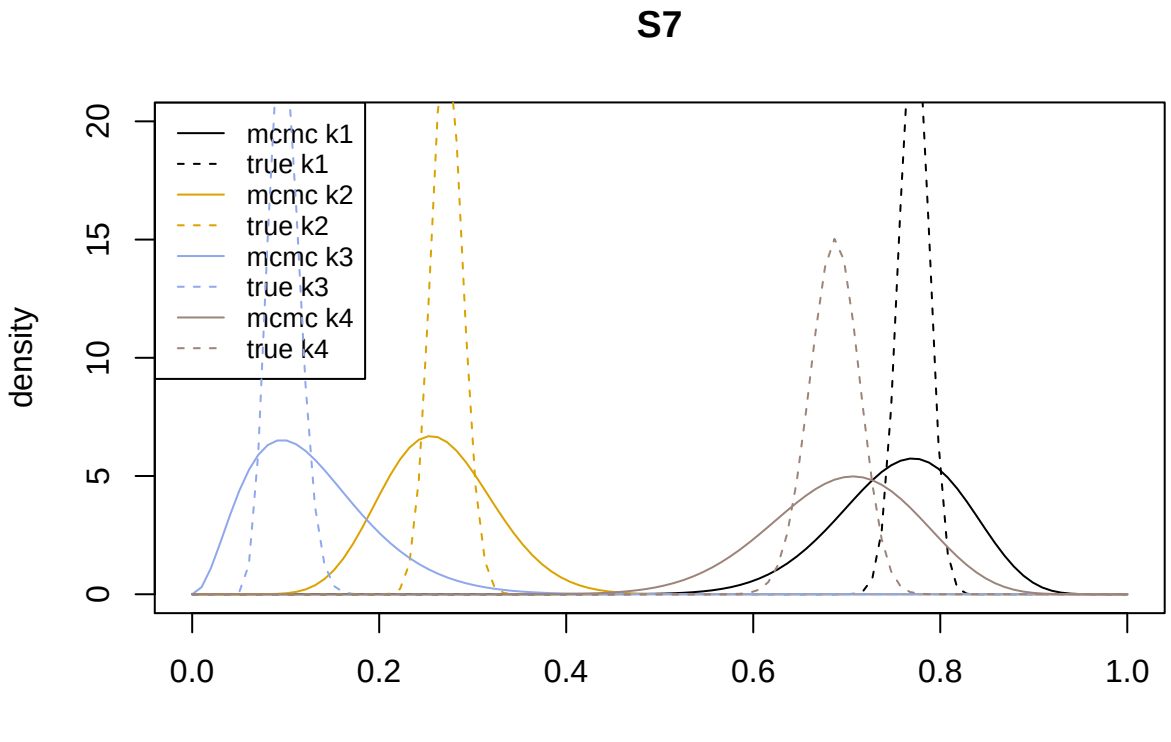


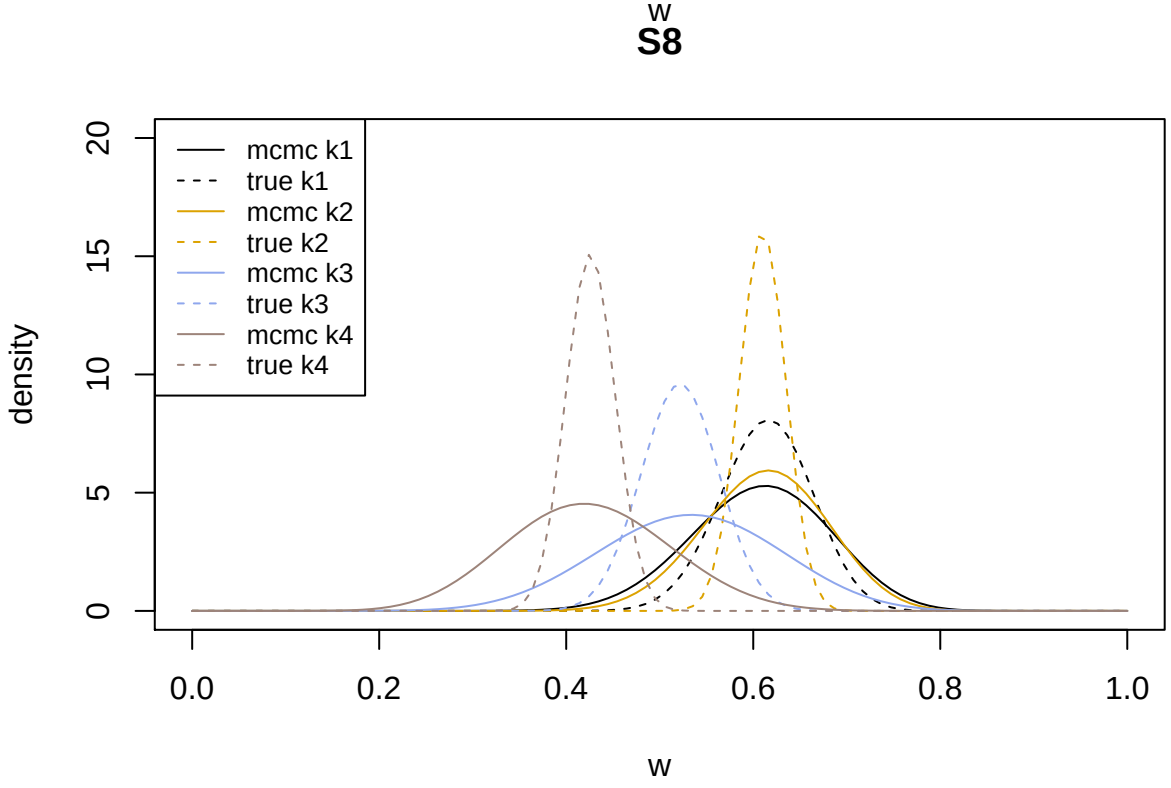

S3
mcmc k1
true k1
mcmc k2
true k2
mcmc k3
true k3
mcmc k4
true k4
density
0
5
10
15
20
0.0
0.2
0.4
0.6
0.8
1.0
w

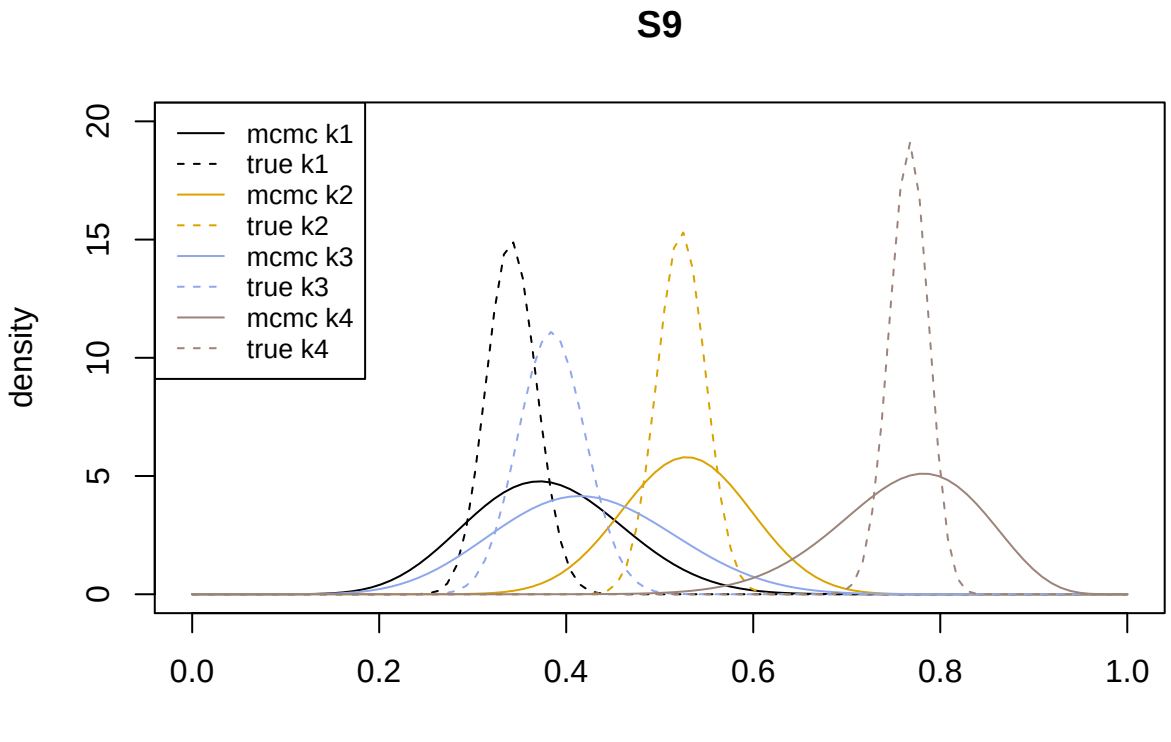

S4
mcmc k1
true k1
mcmc k2
true k2
mcmc k3
true k3
mcmc k4
true k4
density
0
5
10
15
20
0.0
0.2
0.4
0.6
0.8
1.0
w

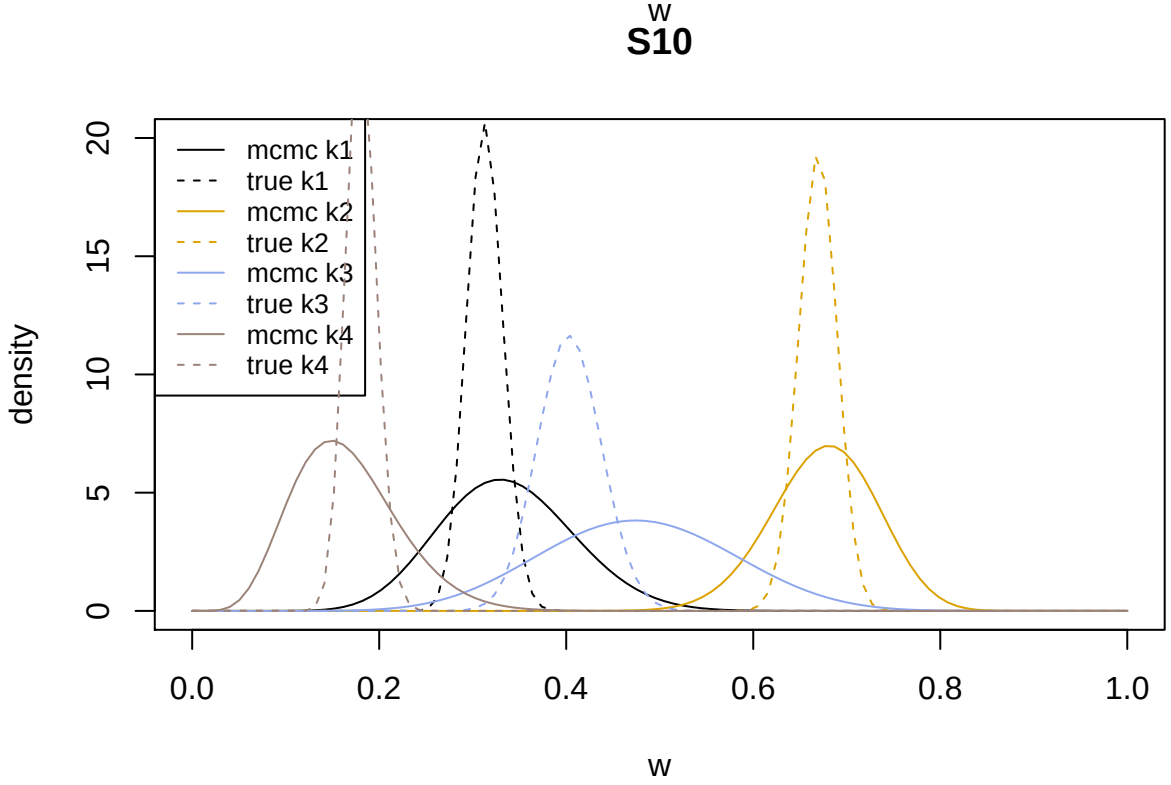

S5
mcmc k1
true k1
mcmc k2
true k2
mcmc k3
true k3
mcmc k4
true k4
density
0
5
10
15
20
0.0
0.2
0.4
0.6
0.8
1.0
w


S6
mcmc k1
true k1
mcmc k2
true k2
mcmc k3
true k3
mcmc k4
true k4
density
0
5
10
15
20
0.0
0.2
0.4
0.6
0.8
1.0
w


S7
mcmc k1
true k1
mcmc k2
true k2
mcmc k3
true k3
mcmc k4
true k4
density
0
5
10
15
20
0.0
0.2
0.4
0.6
0.8
1.0
w


S8
mcmc k1
true k1
mcmc k2
true k2
mcmc k3
true k3
mcmc k4
true k4
density
0
5
10
15
20
0.0
0.2
0.4
0.6
0.8
1.0
w


S9
mcmc k1
true k1
mcmc k2
true k2
mcmc k3
true k3
mcmc k4
true k4
density
0
5
10
15
20
0.0
0.2
0.4
0.6
0.8
1.0
w


S10
mcmc k1
true k1
mcmc k2
true k2
mcmc k3
true k3
mcmc k4
true k4
density
0
5
10
15
20
0.0
0.2
0.4
0.6
0.8
1.0
w