HW05.Rmd

Shashi

February 12, 2018

I have executed these exercises on my own and written the answers in my own words. Signed: Shashi Shankar

1.

```
kappa = seq( 0 , 150 , length=1001 )
source("DBDA2E-utilities.R")
```

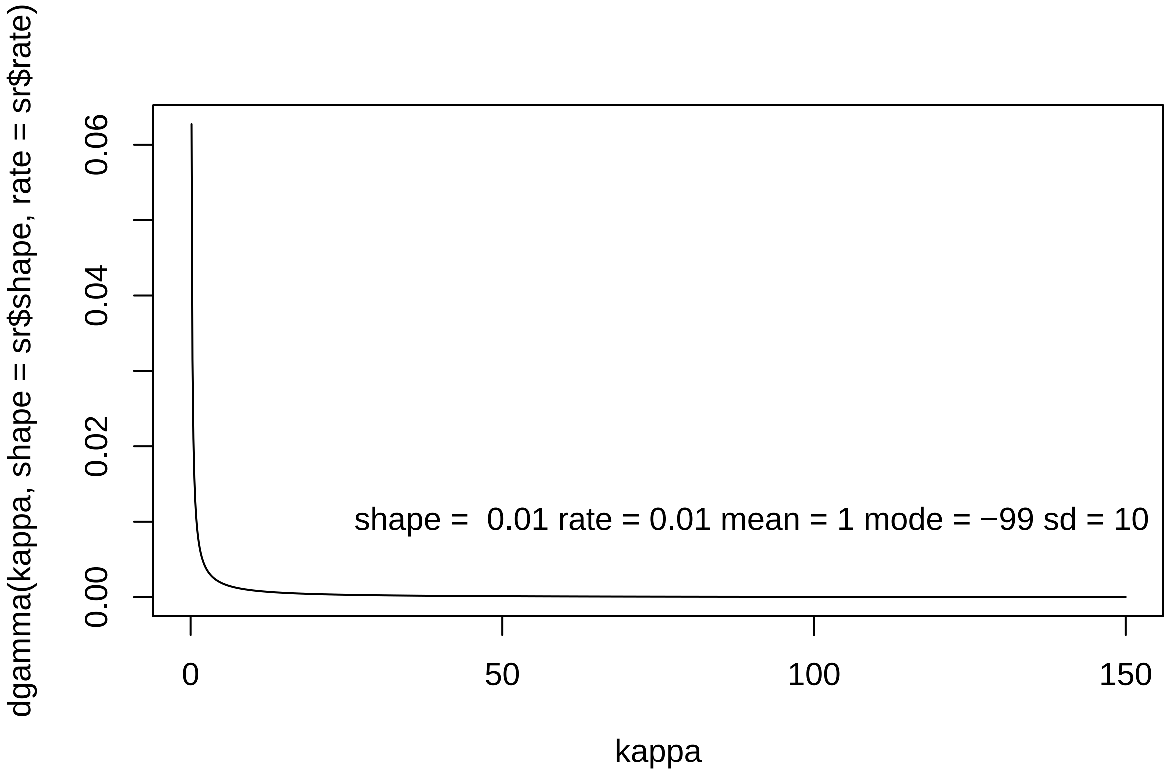
```
##
## *********************************************************************
## Kruschke, J. K. (2015). Doing Bayesian Data Analysis, Second Edition:
## A Tutorial with R, JAGS, and Stan. Academic Press / Elsevier.
## *********************************************************************
```

```
## Loading required package: coda
```

```
## Linked to JAGS 4.2.0
```

```
## Loaded modules: basemod,bugs
```

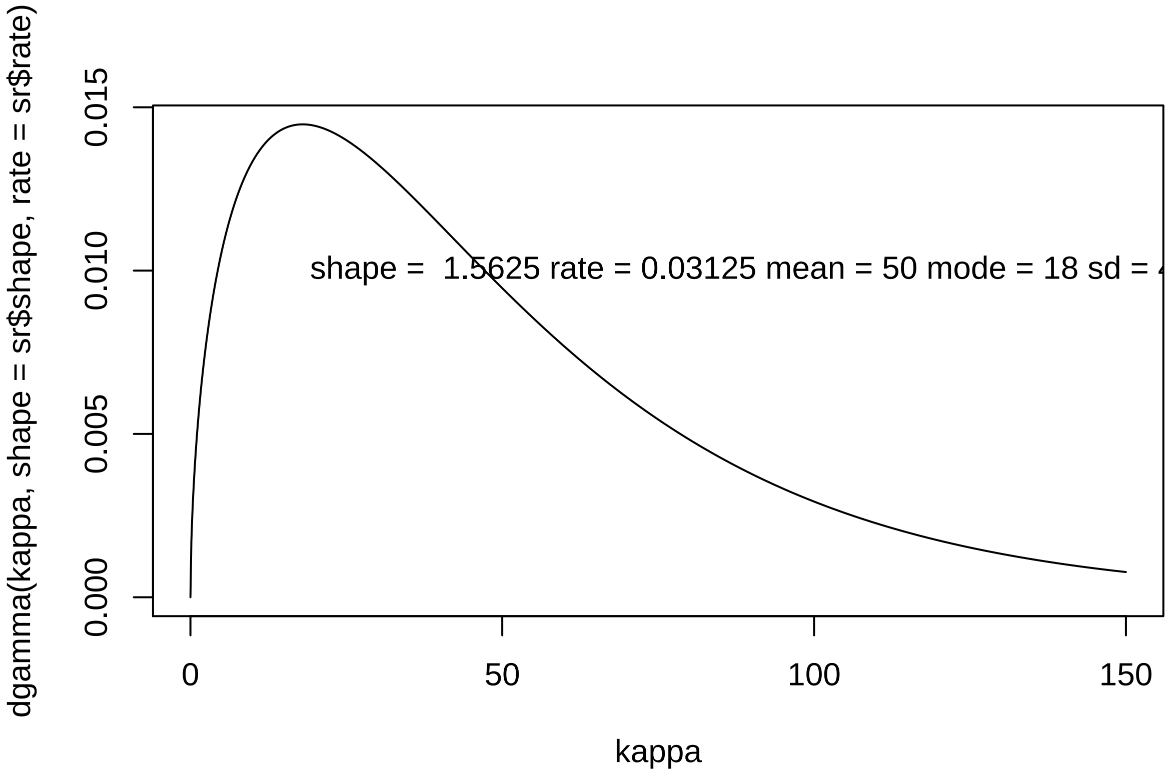
```
mean <- 1
sd <- 10
sr = gammaShRaFromMeanSD( mean=1 , sd=10 )
mode = (sr$shape - 1)/sr$rate
plot( kappa , dgamma( kappa , shape=sr$shape , rate=sr$rate ) , type="l")
text(90,0.01, paste("shape = ", toString(sr$shape) , "rate =", toString(sr$rate), "mean =", toString(mea
```




```
kappa = seq( 0 , 150 , length=1001 )
source("DBDA2E-utilities.R")
```

```
##
## *********************************************************************
## Kruschke, J. K. (2015). Doing Bayesian Data Analysis, Second Edition:
## A Tutorial with R, JAGS, and Stan. Academic Press / Elsevier.
## *********************************************************************
```

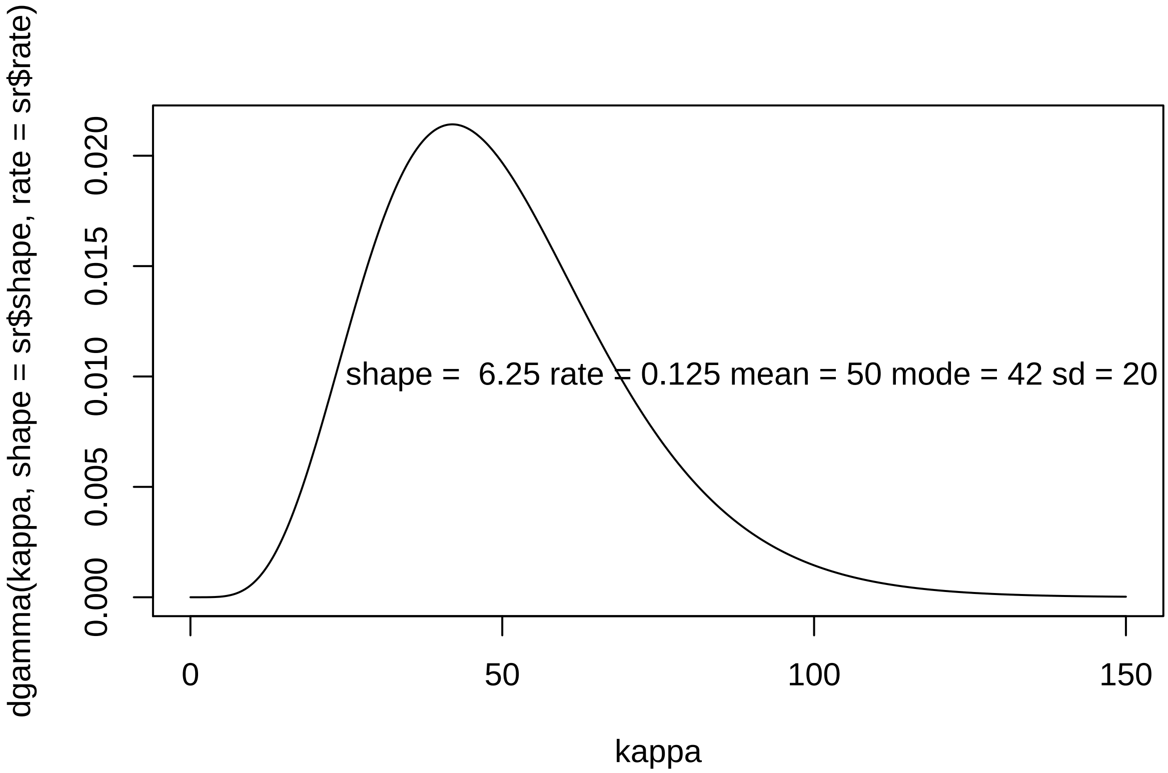
```
mode <- 18
sd <- 40
sr = gammaShRaFromModeSD( mode=18 , sd=40 )
mean = sr$shape/sr$rate
plot( kappa , dgamma( kappa , shape=sr$shape , rate=sr$rate ) , type="l")
text(90,0.01, paste("shape = ", toString(sr$shape) , "rate =", toString(sr$rate), "mean =", toString(mea
```




```
kappa = seq( 0 , 150 , length=1001 )
source("DBDA2E-utilities.R")
```

```
##
## *********************************************************************
## Kruschke, J. K. (2015). Doing Bayesian Data Analysis, Second Edition:
## A Tutorial with R, JAGS, and Stan. Academic Press / Elsevier.
## *********************************************************************
```

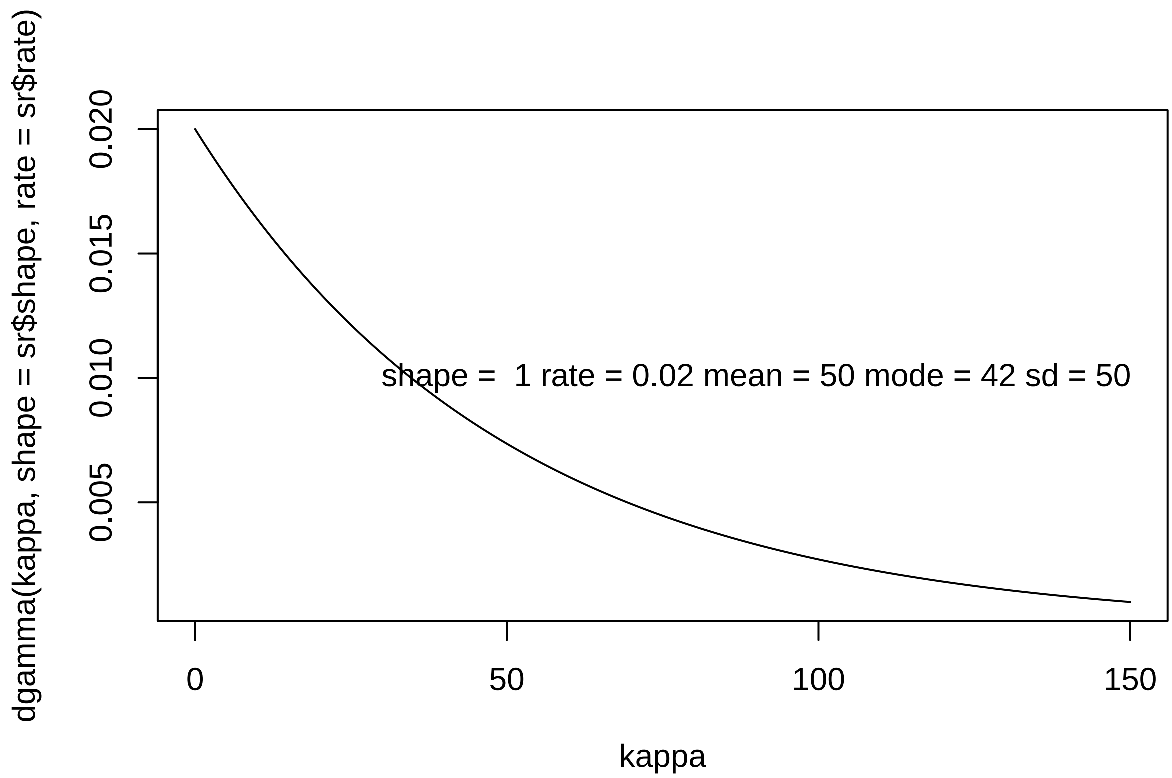
```
mode <- 42
sd <- 20
sr = gammaShRaFromModeSD( mode=42 , sd=20 )
mean = sr$shape/sr$rate
plot( kappa , dgamma( kappa , shape=sr$shape , rate=sr$rate ) , type="l")
text(90,0.01, paste("shape = ", toString(sr$shape) , "rate =", toString(sr$rate), "mean =", toString(mea
```




```r
kappa = seq( 0 , 150 , length=1001 )
source("DBDA2E-utilities.R")
```

```
##
## *********************************************************************
## Kruschke, J. K. (2015). Doing Bayesian Data Analysis, Second Edition:
## A Tutorial with R, JAGS, and Stan. Academic Press / Elsevier.
## *********************************************************************
```

```r
mean <- 50
sd <- 50
mode = (sr$shape - 1)/sr$rate
sr = gammaShRaFromMeanSD( mean=50 , sd=50 )
mean = sr$shape/sr$rate
plot( kappa , dgamma( kappa , shape=sr$shape , rate=sr$rate ) , type="l")
text(90,0.01, paste("shape = ", toString(sr$shape) , "rate =", toString(sr$rate), "mean =", toString(mea
```



2A. Proportion of head for coins 1,2,3 and 4 are 0.25, 0.50, 0.50, and 0.75 respectively.

2B.

```
s = c( 1,1,1,1, 2,2,2,2, 3,3,3,3, 4,4,4,4 ) # subject indicator for each datum
y = c( 1,0,0,0, 1,1,0,0, 1,1,0,0, 1,1,1,0 ) # value of each datum
theta = c( 0.25 , 0.50 , 0.50 , 0.75 )
omega = 0.5
kappa = 2.0
# lik (below) is likelihood.
lik = 1.0 # intialize
for ( sIdx in unique(s) ) {
# To understand the next line, unpack it from the inside out. Consider the first
# time through the for loop, when sIdx is 1. What is s==sIdx? What is y[s==sIdx]?
# What is theta[sIdx]? What is theta[sIdx]^y[s==sIdx]? etc.
# What is this line computing in Eqn 9.10?
lik = lik * prod( theta[sIdx]^y[s==sIdx] * (1-theta[sIdx])^(1-y[s==sIdx]) )
}
a = omega*(kappa-2)+1
b = (1-omega)*(kappa-2)+1
# What is the next line computing in Eqn 9.10?
lik = lik * prod( theta^(a-1) * (1-theta)^(b-1) / beta(a,b) )
```

```r
show(lik)
```

```
## [1] 4.345179e-05
```

Shape of the Beta distribution is flat. These parameter values do not constitute any reasonable shrinkage relative to the data proportions.

2C.

```r
s = c( 1,1,1,1, 2,2,2,2, 3,3,3,3, 4,4,4,4 ) # subject indicator for each datum
y = c( 1,0,0,0, 1,1,0,0, 1,1,0,0, 1,1,1,0 ) # value of each datum
theta = c( 0.35 , 0.50 , 0.50 , 0.65 )
omega = 0.5
kappa = 20.0
# lik (below) is likelihood.
lik = 1.0 # intialize
for ( sIdx in unique(s) ) {
# To understand the next line, unpack it from the inside out. Consider the first
# time through the for loop, when sIdx is 1. What is s==sIdx? What is y[s==sIdx]?
# What is theta[sIdx]? What is theta[sIdx]^y[s==sIdx]? etc.
# What is this line computing in Eqn 9.10?
lik = lik * prod( theta[sIdx]^y[s==sIdx] * (1-theta[sIdx])^(1-y[s==sIdx]) )
}
a = omega*(kappa-2)+1
b = (1-omega)*(kappa-2)+1
# What is the next line computing in Eqn 9.10?
lik = lik * prod( theta^(a-1) * (1-theta)^(b-1) / beta(a,b) )

show(lik)
```

```
## [1] 0.001019145
```

Shape of the Beta distribution is peaked.Parameter values in part c yield a higher likelihood value for the data. When kappa is larger, the individual theta values must be shrunken more.

3.

```r
source("Jags-Ydich-XnomSsubj-MbinomBetaOmegaKappa-Example.R")
```

```
##
## *********************************************************************
## Kruschke, J. K. (2015). Doing Bayesian Data Analysis, Second Edition:
## A Tutorial with R, JAGS, and Stan. Academic Press / Elsevier.
## *********************************************************************
##
## Calling 3 simulations using the parallel method...
## Following the progress of chain 1 (the program will wait for all
## chains to finish before continuing):
## Welcome to JAGS 4.2.0 on Tue Feb 13 01:36:22 2018
## JAGS is free software and comes with ABSOLUTELY NO WARRANTY
## Loading module: basemod: ok
## Loading module: bugs: ok
## . . Reading data file data.txt
```

```
## . Compiling model graph
##    Resolving undeclared variables
##    Allocating nodes
## Graph information:
##    Observed stochastic nodes: 28
##    Unobserved stochastic nodes: 30
##    Total graph size: 239
## . Reading parameter file inits1.txt
## . Initializing model
## . Adapting 500
## -------------------------------------------------| 500
## ++++++++++++++++++++++++++++++++++++++++++++++++++ 100%
## Adaptation successful
## . Updating 500
## -------------------------------------------------| 500
## ************************************************** 100%
## . . . . Updating 66670
## -------------------------------------------------| 66650
## ************************************************** 100%
## * 100%
## . . . . Updating 0
## . Deleting model
## .
## All chains have finished
## Simulation complete.  Reading coda files...
## Coda files loaded successfully
## Finished running the simulation
##    user  system elapsed
##    2.22    0.37   16.39
```

```
##                         Mean    Median        Mode     ESS HDImass
## omega             0.43660840  0.436380  0.43459396 11587.7    0.95
## kappa            59.07611672 59.444100 80.41875967 13404.0    0.95
## theta[1]          0.38148434  0.385305  0.39470797 15800.1    0.95
## theta[2]          0.39896354  0.401982  0.41215184 17136.6    0.95
## theta[3]          0.41539481  0.416082  0.42406655 17312.7    0.95
## theta[4]          0.41536504  0.415837  0.41703893 16624.8    0.95
## theta[5]          0.41578363  0.416448  0.41181900 17437.3    0.95
## theta[6]          0.41535949  0.416176  0.42448668 17253.2    0.95
## theta[7]          0.41530822  0.416820  0.42244144 17881.6    0.95
## theta[8]          0.41535402  0.416834  0.42391393 17155.7    0.95
## theta[9]          0.41488253  0.415588  0.40758333 18005.4    0.95
## theta[10]         0.41573415  0.416430  0.41818729 17290.9    0.95
## theta[11]         0.43250689  0.432355  0.43549793 17927.2    0.95
## theta[12]         0.43253952  0.432688  0.44393362 18332.3    0.95
## theta[13]         0.43315474  0.432744  0.42064925 17828.2    0.95
## theta[14]         0.43318550  0.432510  0.42437922 17953.9    0.95
## theta[15]         0.43246072  0.432140  0.43435353 17741.8    0.95
## theta[16]         0.44985881  0.447718  0.43844311 17369.4    0.95
## theta[17]         0.44970940  0.447768  0.44289418 17583.6    0.95
## theta[18]         0.45061039  0.449043  0.45189460 17487.8    0.95
## theta[19]         0.45042418  0.448944  0.45144929 19746.3    0.95
## theta[20]         0.45081402  0.449629  0.44867291 17573.2    0.95
## theta[21]         0.44982469  0.449013  0.45091455 18031.8    0.95
```

```
## theta[22]            0.45042511  0.449347  0.44890458 18519.1    0.95
## theta[23]            0.46690268  0.464260  0.46603992 18087.2    0.95
## theta[24]            0.46766484  0.465409  0.45650759 18254.3    0.95
## theta[25]            0.48533028  0.481407  0.48133251 17979.2    0.95
## theta[26]            0.48408701  0.480356  0.47647378 16523.1    0.95
## theta[27]            0.48457453  0.480538  0.47572360 16608.2    0.95
## theta[28]            0.50208917  0.496958  0.48583313 17281.3    0.95
## theta[1]-theta[14]  -0.05170116 -0.047566 -0.03442738 18952.1    0.95
## theta[1]-theta[28]  -0.12060483 -0.109148 -0.09917428 16000.5    0.95
## theta[14]-theta[28] -0.06890367 -0.063555 -0.06072951 18512.8    0.95
##                       HDIlow    HDIhigh CompVal PcntGtCompVal ROPElow
## omega                 0.369849   0.503041     0.5      3.109845      NA
## kappa                18.482500 101.983000      NA            NA      NA
## theta[1]              0.235743   0.525622     0.5      4.384781      NA
## theta[2]              0.251855   0.536004     0.5      7.149643      NA
## theta[3]              0.272302   0.553474     0.5     10.999450      NA
## theta[4]              0.278479   0.552419     0.5     10.954452      NA
## theta[5]              0.275141   0.553344     0.5     11.024449      NA
## theta[6]              0.275404   0.550042     0.5     10.639468      NA
## theta[7]              0.278628   0.552679     0.5     10.549473      NA
## theta[8]              0.273096   0.551658     0.5     10.899455      NA
## theta[9]              0.278072   0.553095     0.5     10.744463      NA
## theta[10]             0.278645   0.556353     0.5     10.819459      NA
## theta[11]             0.299250   0.571053     0.5     16.004200      NA
## theta[12]             0.296856   0.571987     0.5     16.124194      NA
## theta[13]             0.298030   0.571797     0.5     16.374181      NA
## theta[14]             0.290858   0.567833     0.5     16.219189      NA
## theta[15]             0.291946   0.563511     0.5     15.984201      NA
## theta[16]             0.308512   0.586395     0.5     22.893855      NA
## theta[17]             0.313390   0.593095     0.5     22.653867      NA
## theta[18]             0.314472   0.591046     0.5     22.843858      NA
## theta[19]             0.314153   0.589374     0.5     22.718864      NA
## theta[20]             0.315483   0.593166     0.5     22.978851      NA
## theta[21]             0.305700   0.585342     0.5     22.603870      NA
## theta[22]             0.318945   0.594191     0.5     23.113844      NA
## theta[23]             0.332766   0.614458     0.5     30.288486      NA
## theta[24]             0.327795   0.609069     0.5     31.213439      NA
## theta[25]             0.346004   0.636755     0.5     39.513024      NA
## theta[26]             0.342017   0.631945     0.5     39.093045      NA
## theta[27]             0.341872   0.629361     0.5     39.483026      NA
## theta[28]             0.357385   0.655331     0.5     48.422579      NA
## theta[1]-theta[14]   -0.240068   0.128246     0.0     28.768562      NA
## theta[1]-theta[28]   -0.340845   0.079730     0.0     10.919454      NA
## theta[14]-theta[28] -0.268782   0.118653     0.0     23.838808      NA
##                     ROPEhigh PcntLtROPE PcntInROPE PcntGtROPE
## omega                     NA         NA         NA         NA
## kappa                     NA         NA         NA         NA
## theta[1]                  NA         NA         NA         NA
## theta[2]                  NA         NA         NA         NA
## theta[3]                  NA         NA         NA         NA
## theta[4]                  NA         NA         NA         NA
## theta[5]                  NA         NA         NA         NA
## theta[6]                  NA         NA         NA         NA
## theta[7]                  NA         NA         NA         NA
```

```
## theta[8]                NA         NA         NA         NA
## theta[9]                NA         NA         NA         NA
## theta[10]               NA         NA         NA         NA
## theta[11]               NA         NA         NA         NA
## theta[12]               NA         NA         NA         NA
## theta[13]               NA         NA         NA         NA
## theta[14]               NA         NA         NA         NA
## theta[15]               NA         NA         NA         NA
## theta[16]               NA         NA         NA         NA
## theta[17]               NA         NA         NA         NA
## theta[18]               NA         NA         NA         NA
## theta[19]               NA         NA         NA         NA
## theta[20]               NA         NA         NA         NA
## theta[21]               NA         NA         NA         NA
## theta[22]               NA         NA         NA         NA
## theta[23]               NA         NA         NA         NA
## theta[24]               NA         NA         NA         NA
## theta[25]               NA         NA         NA         NA
## theta[26]               NA         NA         NA         NA
## theta[27]               NA         NA         NA         NA
## theta[28]               NA         NA         NA         NA
## theta[1]-theta[14]      NA         NA         NA         NA
## theta[1]-theta[28]      NA         NA         NA         NA
## theta[14]-theta[28]     NA         NA         NA         NA
```

Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.

```
##
## *********************************************************************
## Kruschke, J. K. (2015). Doing Bayesian Data Analysis, Second Edition:
## A Tutorial with R, JAGS, and Stan. Academic Press / Elsevier.
## *********************************************************************
##
## Calling 3 simulations using the parallel method...
## Following the progress of chain 1 (the program will wait for all
## chains to finish before continuing):
## Welcome to JAGS 4.2.0 on Tue Feb 13 01:36:44 2018
## JAGS is free software and comes with ABSOLUTELY NO WARRANTY
## Loading module: basemod: ok
## Loading module: bugs: ok
## . . Reading data file data.txt
## . Compiling model graph
##    Resolving undeclared variables
##    Allocating nodes
## Graph information:
##    Observed stochastic nodes: 28
##    Unobserved stochastic nodes: 30
##    Total graph size: 239
## . Reading parameter file inits1.txt
## . Initializing model
## . Adapting 500
## -------------------------------------------------| 500
## ++++++++++++++++++++++++++++++++++++++++++++++++++ 100%
## Adaptation successful
```

```
## . Updating 500
## -------------------------------------------------| 500
## ************************************************** 100%
## . . . . Updating 66670
## -------------------------------------------------| 66650
## ************************************************** 100%
## * 100%
## . . . . Updating 0
## . Deleting model
## .
## All chains have finished
## Simulation complete.  Reading coda files...
## Coda files loaded successfully
## Finished running the simulation
##    user  system elapsed
##    2.42    0.40   16.99
```

```
##                            Mean    Median        Mode     ESS HDImass
## omega                0.43647149  0.436175  0.42838190 11705.7    0.95
## kappa               59.30915080 59.823400 70.31079120 13016.0    0.95
## theta[1]             0.38172868  0.385122  0.40009817 15486.0    0.95
## theta[2]             0.39862481  0.400254  0.39917720 17280.0    0.95
## theta[3]             0.41505286  0.416094  0.41587651 16880.7    0.95
## theta[4]             0.41510323  0.416215  0.41964837 17207.6    0.95
## theta[5]             0.41659906  0.417517  0.41799460 17027.4    0.95
## theta[6]             0.41549619  0.416412  0.41566942 17273.8    0.95
## theta[7]             0.41490076  0.415424  0.40869516 17518.0    0.95
## theta[8]             0.41523064  0.415570  0.41347395 17739.7    0.95
## theta[9]             0.41520088  0.416113  0.41817136 17549.8    0.95
## theta[10]            0.41527636  0.416564  0.42124799 17060.0    0.95
## theta[11]            0.43254703  0.432215  0.43275464 18723.4    0.95
## theta[12]            0.43258245  0.432030  0.42716137 17631.0    0.95
## theta[13]            0.43280004  0.432931  0.43682610 18028.3    0.95
## theta[14]            0.43219157  0.431819  0.43099541 18160.7    0.95
## theta[15]            0.43247645  0.432291  0.42118123 17902.0    0.95
## theta[16]            0.44949060  0.447893  0.44208295 18401.3    0.95
## theta[17]            0.44941191  0.447792  0.45249395 18107.8    0.95
## theta[18]            0.44899151  0.447771  0.44731743 17891.9    0.95
## theta[19]            0.44994844  0.448411  0.44976714 17759.1    0.95
## theta[20]            0.45005678  0.448921  0.44866029 18318.2    0.95
## theta[21]            0.44917645  0.447849  0.44632361 17891.7    0.95
## theta[22]            0.45016305  0.448545  0.45216555 17165.3    0.95
## theta[23]            0.46710639  0.464921  0.46674077 16408.6    0.95
## theta[24]            0.46724183  0.464841  0.45863540 17562.0    0.95
## theta[25]            0.48353287  0.480394  0.47531948 17982.6    0.95
## theta[26]            0.48383578  0.479999  0.47754342 16498.6    0.95
## theta[27]            0.48475702  0.480645  0.47552334 14680.9    0.95
## theta[28]            0.50088674  0.496279  0.49251653 15807.1    0.95
## theta[1]-theta[14]  -0.05046289 -0.046289 -0.04205044 18282.8    0.95
## theta[1]-theta[28]  -0.11915806 -0.107616 -0.09273484 15932.4    0.95
## theta[14]-theta[28] -0.06869517 -0.063413 -0.06609809 18703.1    0.95
##                        HDIlow    HDIhigh CompVal PcntGtCompVal ROPElow
## omega                0.368410   0.501242     0.5      3.034848      NA
## kappa               18.793400 101.987000      NA            NA      NA
```

```
## theta[1]              0.231524   0.521920     0.5      4.399780      NA
## theta[2]              0.264161   0.543366     0.5      6.984651      NA
## theta[3]              0.278754   0.554263     0.5     10.624469      NA
## theta[4]              0.279996   0.552325     0.5     10.709465      NA
## theta[5]              0.278945   0.555243     0.5     11.014449      NA
## theta[6]              0.278057   0.552066     0.5     10.489476      NA
## theta[7]              0.270254   0.547852     0.5     10.724464      NA
## theta[8]              0.277790   0.553066     0.5     10.964452      NA
## theta[9]              0.275194   0.551057     0.5     11.229439      NA
## theta[10]             0.276659   0.553613     0.5     10.464477      NA
## theta[11]             0.298920   0.569920     0.5     16.004200      NA
## theta[12]             0.301434   0.573782     0.5     15.824209      NA
## theta[13]             0.290636   0.564068     0.5     16.074196      NA
## theta[14]             0.294858   0.571150     0.5     16.004200      NA
## theta[15]             0.293537   0.565784     0.5     16.004200      NA
## theta[16]             0.310793   0.586480     0.5     22.648868      NA
## theta[17]             0.314163   0.590247     0.5     22.278886      NA
## theta[18]             0.313117   0.585760     0.5     22.323884      NA
## theta[19]             0.313779   0.586682     0.5     22.553872      NA
## theta[20]             0.313943   0.587254     0.5     23.133843      NA
## theta[21]             0.314855   0.593303     0.5     22.658867      NA
## theta[22]             0.315924   0.587975     0.5     22.983851      NA
## theta[23]             0.332847   0.614177     0.5     30.408480      NA
## theta[24]             0.331132   0.613377     0.5     30.183491      NA
## theta[25]             0.342634   0.628432     0.5     39.228039      NA
## theta[26]             0.340756   0.629268     0.5     38.748063      NA
## theta[27]             0.344705   0.634520     0.5     39.123044      NA
## theta[28]             0.357809   0.653018     0.5     47.842608      NA
## theta[1]-theta[14]  -0.237396   0.133909     0.0     29.598520      NA
## theta[1]-theta[28]  -0.342661   0.075301     0.0     11.059447      NA
## theta[14]-theta[28] -0.263021   0.121706     0.0     23.628819      NA
##                     ROPEhigh PcntLtROPE PcntInROPE PcntGtROPE
## omega                     NA         NA         NA         NA
## kappa                     NA         NA         NA         NA
## theta[1]                  NA         NA         NA         NA
## theta[2]                  NA         NA         NA         NA
## theta[3]                  NA         NA         NA         NA
## theta[4]                  NA         NA         NA         NA
## theta[5]                  NA         NA         NA         NA
## theta[6]                  NA         NA         NA         NA
## theta[7]                  NA         NA         NA         NA
## theta[8]                  NA         NA         NA         NA
## theta[9]                  NA         NA         NA         NA
## theta[10]                 NA         NA         NA         NA
## theta[11]                 NA         NA         NA         NA
## theta[12]                 NA         NA         NA         NA
## theta[13]                 NA         NA         NA         NA
## theta[14]                 NA         NA         NA         NA
## theta[15]                 NA         NA         NA         NA
## theta[16]                 NA         NA         NA         NA
## theta[17]                 NA         NA         NA         NA
## theta[18]                 NA         NA         NA         NA
## theta[19]                 NA         NA         NA         NA
## theta[20]                 NA         NA         NA         NA
```

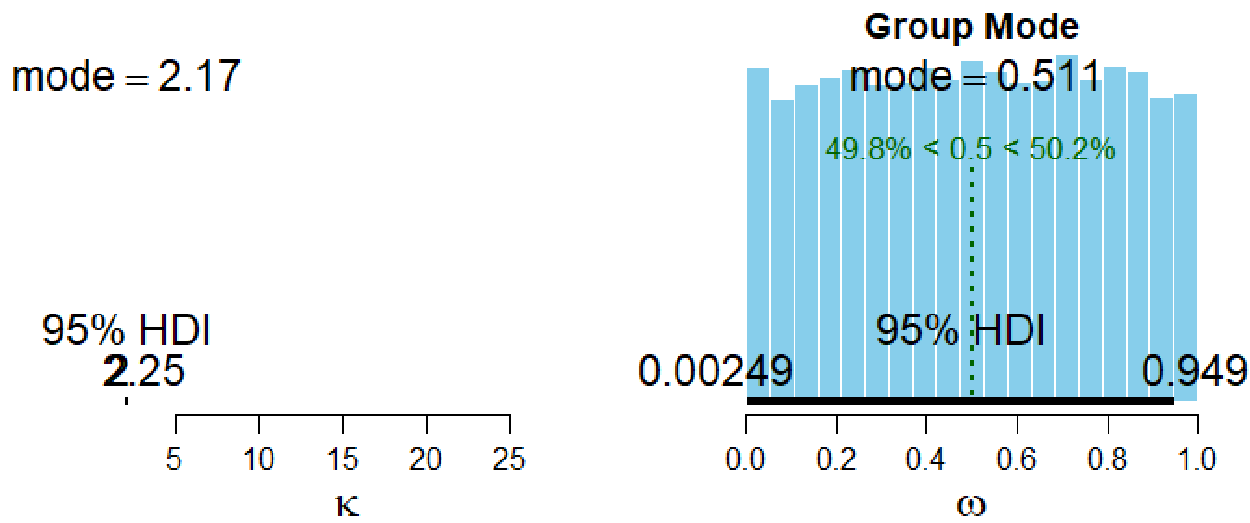
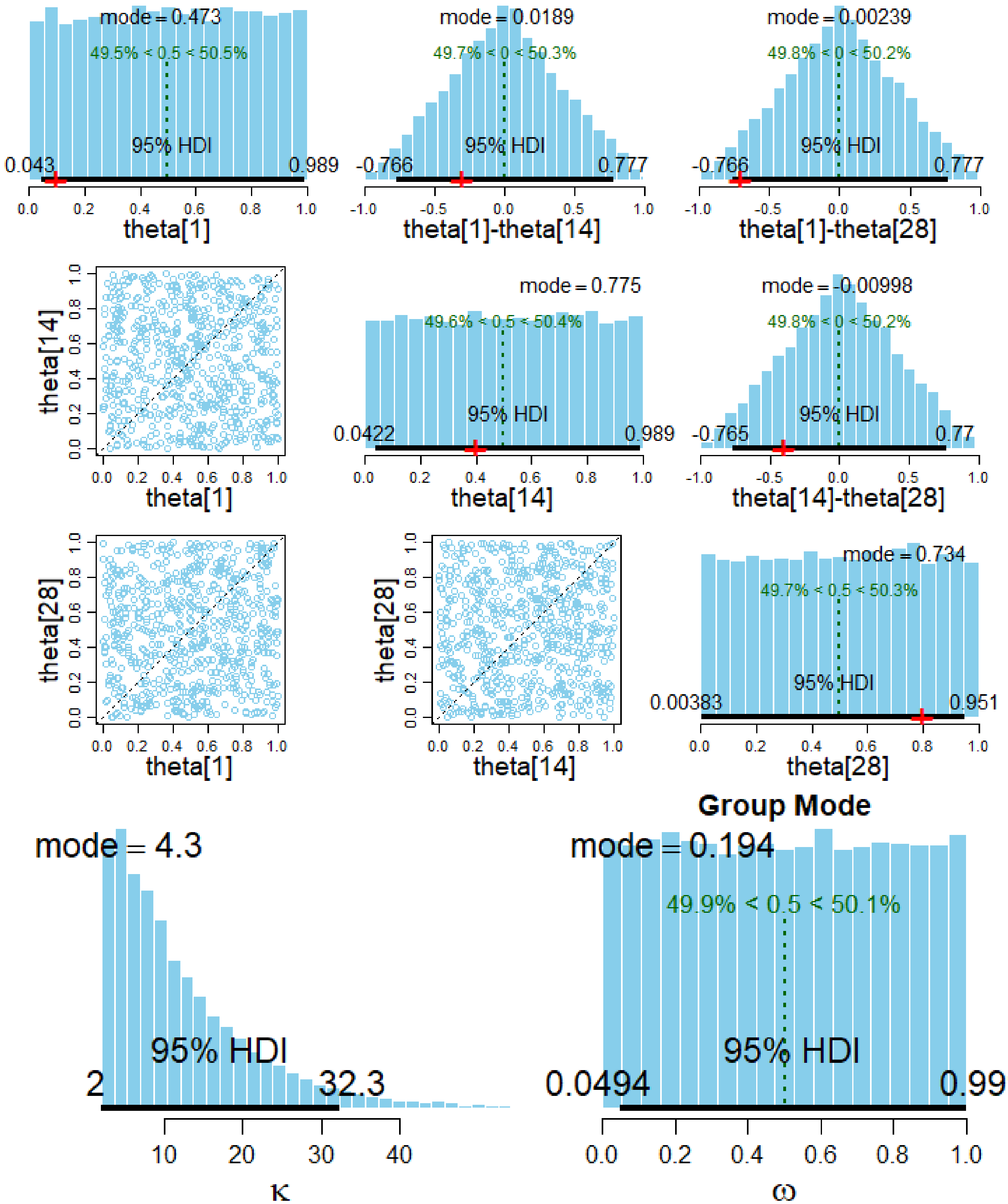
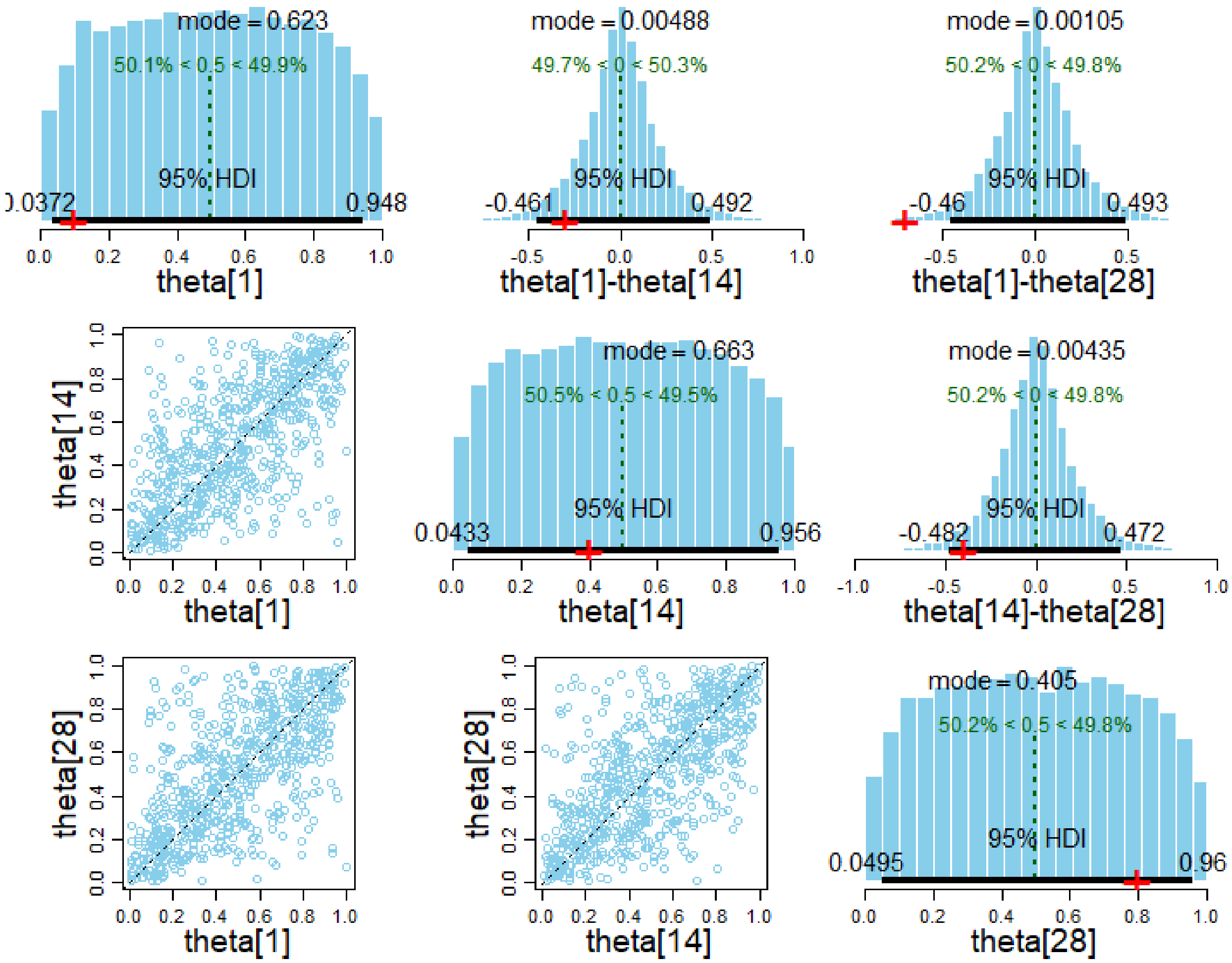


Figure 1:

```
## theta[21]                 NA          NA          NA          NA
## theta[22]                 NA          NA          NA          NA
## theta[23]                 NA          NA          NA          NA
## theta[24]                 NA          NA          NA          NA
## theta[25]                 NA          NA          NA          NA
## theta[26]                 NA          NA          NA          NA
## theta[27]                 NA          NA          NA          NA
## theta[28]                 NA          NA          NA          NA
## theta[1]-theta[14]        NA          NA          NA          NA
## theta[1]-theta[28]        NA          NA          NA          NA
## theta[14]-theta[28]       NA          NA          NA          NA
```


mode = 0.473
49.5% < 0.5 < 50.5%
95% HDI
0.043
0.989
theta[1]
mode = 0.0189
49.7% < 0 < 50.3%
95% HDI
-0.766
0.777
theta[1]-theta[14]
mode = 0.00239
49.8% < 0 < 50.2%
95% HDI
-0.766
0.777
theta[1]-theta[28]
theta[14]
theta[1]
mode = 0.775
49.6% < 0.5 < 50.4%
95% HDI
0.0422
0.989
theta[14]
mode = -0.00998
49.8% < 0 < 50.2%
95% HDI
-0.765
0.77
theta[14]-theta[28]
theta[28]
theta[1]
theta[28]
theta[14]
mode = 0.734
49.7% < 0.5 < 50.3%
95% HDI
0.00383
0.951
theta[28]
mode = 4.3
95% HDI
2
32.3
κ
Group Mode
mode = 0.194
49.9% < 0.5 < 50.1%
95% HDI
0.0494
0.99
ω



kappa does not get too small When the prior has mode=1. But when the prior has mean=1, kappa has a very high probability of being very small. The two different thetas can have opposite extremes when kappa is very small in the case where prior has mean = 1.

3B. I think the prior with mean=1 is more appropriate as it sets uniform prior for the individual thetas and sets a broad prior on the differences of thetas.

4.

```r
# Generate the data frame:
# N.B.: The functions below expect the data to be a data frame,
# with one component being a vector of integer 0,1 values,
# and one component being a factor of subject identifiers.
headsTails = c( rep(1,30),rep(0,100-30),
rep(1,40),rep(0,100-40),
rep(1,50),rep(0,100-50),
rep(1,60),rep(0,100-60),
rep(1,70),rep(0,100-70) )
subjID = factor( c( rep("A",100),
rep("B",100),
```

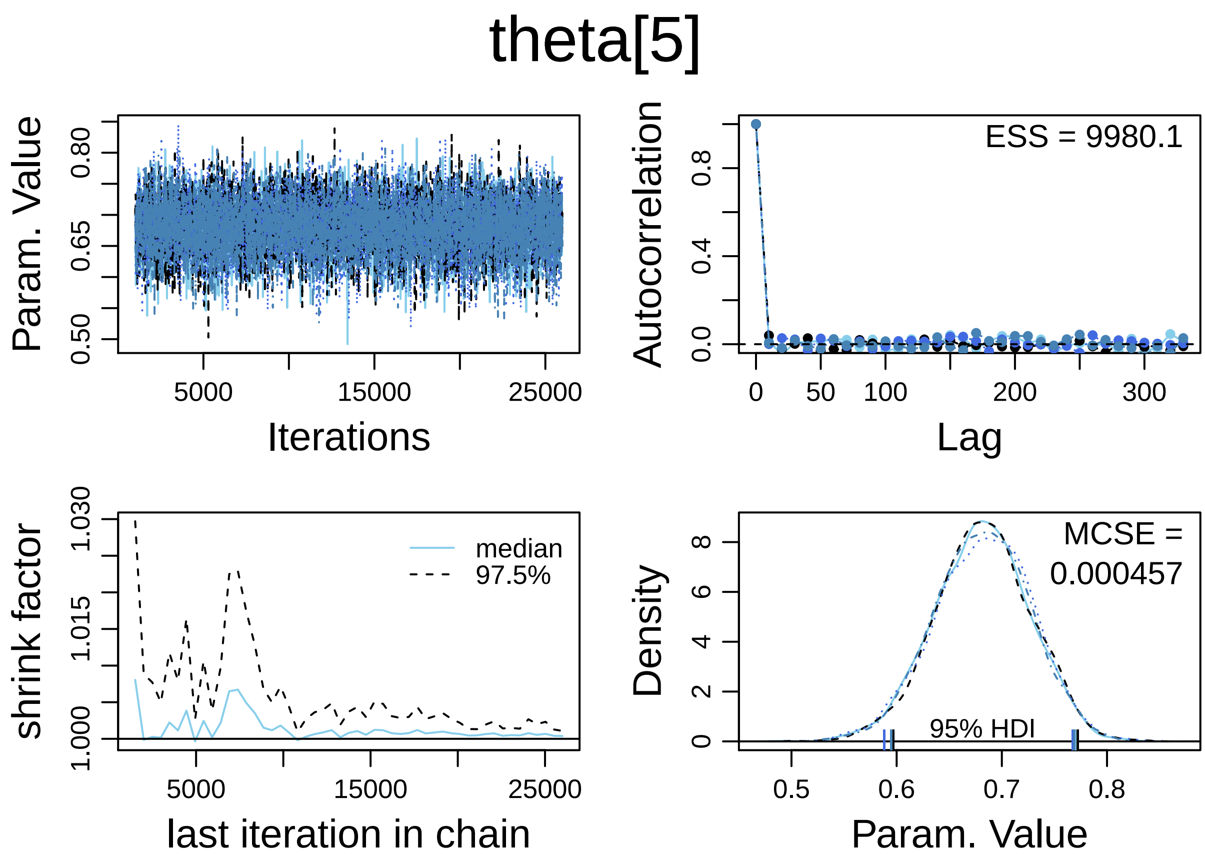
```r
rep("C",100),
rep("D",100),
rep("E",100) ) )
myData = data.frame( y=headsTails , s=subjID )
#-------------------------------------------------------------------------------
# Load the relevant model into R's working memory:
source("Jags-Ydich-XnomSsubj-MbernBetaOmegaKappa.R")
```

```
##
## *********************************************************************
## Kruschke, J. K. (2015). Doing Bayesian Data Analysis, Second Edition:
## A Tutorial with R, JAGS, and Stan. Academic Press / Elsevier.
## *********************************************************************
```

```r
fileNameRoot = "Exercise4-"
graphFileType = "png"
# Generate the MCMC chain:
mcmcCoda = genMCMC( data=myData , sName="s" , yName="y" ,
numSavedSteps=10000 , saveName=fileNameRoot , thinSteps=10 )
```

```
## Compiling model graph
##    Resolving undeclared variables
##    Allocating nodes
## Graph information:
##    Observed stochastic nodes: 500
##    Unobserved stochastic nodes: 7
##    Total graph size: 1046
##
## Initializing model
##
## Burning in the MCMC chain...
## Sampling final MCMC chain...
```

```r
# Display diagnostics of chain, for specified parameters:
parameterNames = varnames(mcmcCoda) # get all parameter names for reference
for ( parName in parameterNames[c(1:3,length(parameterNames))] ) {
diagMCMC( codaObject=mcmcCoda , parName=parName ,
saveName=fileNameRoot , saveType=graphFileType )
}
# Get summary statistics of chain:
summaryInfo = smryMCMC( mcmcCoda , compVal=0.5 ,
diffIdVec=c(1,2,3,4,5), compValDiff=0.0,
saveName=fileNameRoot )
```




```
##                          Mean      Median        Mode     ESS HDImass
## omega              0.49863090  0.49934503  0.49860267 10000.0    0.95
## kappa             10.57502329  9.09678361  6.41932765 10000.0    0.95
## theta[1]           0.31843846  0.31736134  0.30970425 10000.0    0.95
## theta[2]           0.40897635  0.40849883  0.40978728 10000.0    0.95
## theta[3]           0.49966294  0.49987618  0.50076156  9968.6    0.95
## theta[4]           0.59084310  0.59136335  0.58945713  9293.1    0.95
## theta[5]           0.68160619  0.68245928  0.68244190  9709.7    0.95
## theta[1]-theta[2] -0.09053789 -0.09033178 -0.08814586 10000.0    0.95
## theta[1]-theta[3] -0.18122448 -0.18105125 -0.17534931 10000.0    0.95
## theta[1]-theta[4] -0.27240464 -0.27287014 -0.27264018 10000.0    0.95
## theta[1]-theta[5] -0.36316772 -0.36399943 -0.36339517 10000.0    0.95
## theta[2]-theta[3] -0.09068659 -0.09107404 -0.08547912  9989.4    0.95
## theta[2]-theta[4] -0.18186675 -0.18232850 -0.18468106 10000.0    0.95
## theta[2]-theta[5] -0.27262984 -0.27269753 -0.26691658 10000.0    0.95
## theta[3]-theta[4] -0.09118016 -0.09093902 -0.09426929 10000.0    0.95
## theta[3]-theta[5] -0.18194325 -0.18236992 -0.18350112 10000.0    0.95
## theta[4]-theta[5] -0.09076309 -0.09060728 -0.08768527 10000.0    0.95
##                       HDIlow     HDIhigh CompVal PcntGtCompVal ROPElow
## omega              0.2322431  0.76606607     0.5         49.70      NA
## kappa              2.0127149 23.13535749      NA            NA      NA
## theta[1]           0.2294930  0.40568617     0.5          0.00      NA
## theta[2]           0.3179471  0.50266225     0.5          3.06      NA
## theta[3]           0.4087967  0.59844113     0.5         49.89      NA
## theta[4]           0.5010180  0.68916026     0.5         96.83      NA
## theta[5]           0.5945671  0.77146228     0.5         99.99      NA
```

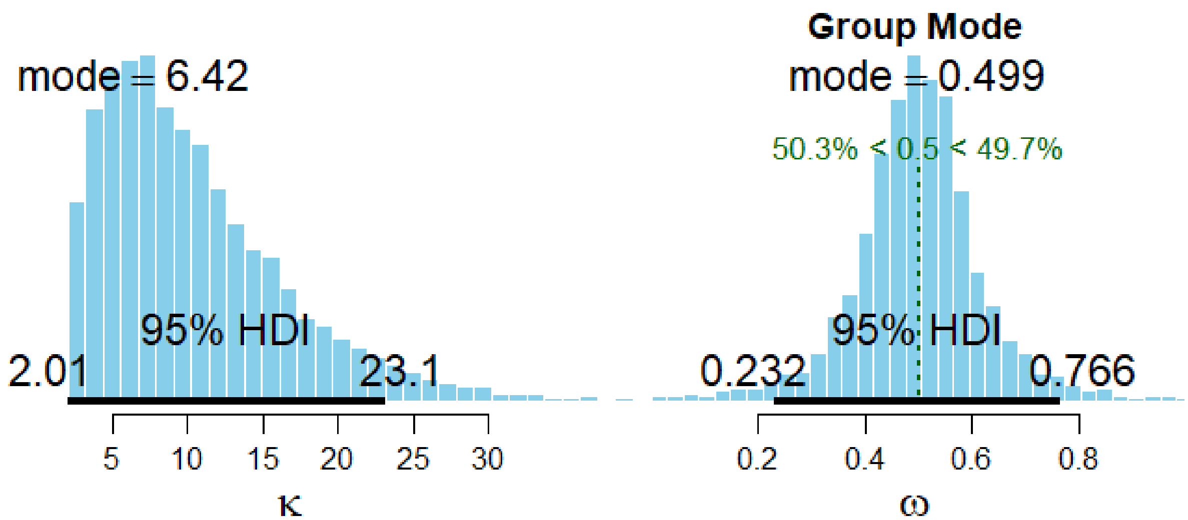
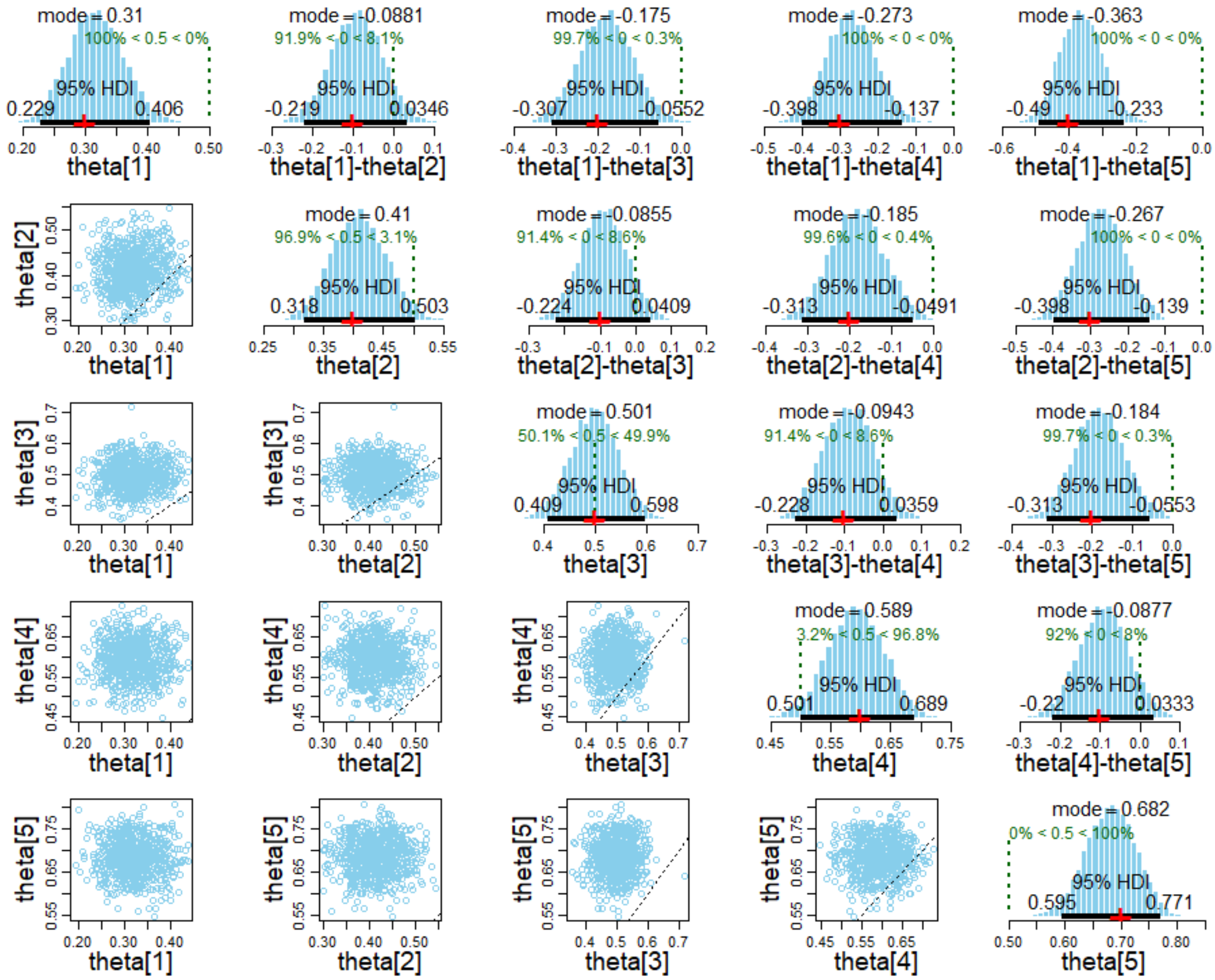


Figure 2:

```
## theta[1]-theta[2] -0.2193714  0.03457214     0.0           8.10      NA
## theta[1]-theta[3] -0.3072866 -0.05521081     0.0           0.34      NA
## theta[1]-theta[4] -0.3978799 -0.13690871     0.0           0.00      NA
## theta[1]-theta[5] -0.4895484 -0.23283192     0.0           0.00      NA
## theta[2]-theta[3] -0.2237148  0.04085347     0.0           8.60      NA
## theta[2]-theta[4] -0.3128959 -0.04911971     0.0           0.36      NA
## theta[2]-theta[5] -0.3975533 -0.13940951     0.0           0.00      NA
## theta[3]-theta[4] -0.2276643  0.03588625     0.0           8.60      NA
## theta[3]-theta[5] -0.3133622 -0.05531593     0.0           0.32      NA
## theta[4]-theta[5] -0.2200607  0.03328258     0.0           8.04      NA
##                   ROPEhigh PcntLtROPE PcntInROPE PcntGtROPE
## omega                   NA         NA         NA         NA
## kappa                   NA         NA         NA         NA
## theta[1]                NA         NA         NA         NA
## theta[2]                NA         NA         NA         NA
## theta[3]                NA         NA         NA         NA
## theta[4]                NA         NA         NA         NA
## theta[5]                NA         NA         NA         NA
## theta[1]-theta[2]       NA         NA         NA         NA
## theta[1]-theta[3]       NA         NA         NA         NA
## theta[1]-theta[4]       NA         NA         NA         NA
## theta[1]-theta[5]       NA         NA         NA         NA
## theta[2]-theta[3]       NA         NA         NA         NA
## theta[2]-theta[4]       NA         NA         NA         NA
## theta[2]-theta[5]       NA         NA         NA         NA
## theta[3]-theta[4]       NA         NA         NA         NA
## theta[3]-theta[5]       NA         NA         NA         NA
## theta[4]-theta[5]       NA         NA         NA         NA
```

```r
# Display posterior information:
plotMCMC( mcmcCoda , data=myData , sName="s" , yName="y" ,
compVal=0.5 , #rope=c(0.45,0.55) ,
diffIdVec=c(1,2,3,4,5), compValDiff=0.0, #ropeDiff = c(-0.05,0.05) ,
saveName=fileNameRoot , saveType=graphFileType )
```



mode = 0.31
100% < 0.5 < 0%
95% HDI
0.229
0.406
theta[1]
mode = -0.0881
91.9% < 0 < 8.1%
95% HDI
-0.219
0.0346
theta[1]-theta[2]
mode = -0.175
99.7% < 0 < 0.3%
95% HDI
-0.307
-0.0552
theta[1]-theta[3]
mode = -0.273
100% < 0 < 0%
95% HDI
-0.398
-0.137
theta[1]-theta[4]
mode = -0.363
100% < 0 < 0%
95% HDI
-0.49
-0.233
theta[1]-theta[5]
theta[2]
theta[1]
mode = 0.41
96.9% < 0.5 < 3.1%
95% HDI
0.318
0.503
theta[2]
mode = -0.0855
91.4% < 0 < 8.6%
95% HDI
-0.224
0.0409
theta[2]-theta[3]
mode = -0.185
99.6% < 0 < 0.4%
95% HDI
-0.313
-0.0491
theta[2]-theta[4]
mode = -0.267
100% < 0 < 0%
95% HDI
-0.398
-0.139
theta[2]-theta[5]
theta[3]
theta[1]
theta[3]
theta[2]
mode = 0.501
50.1% < 0.5 < 49.9%
95% HDI
0.409
0.598
theta[3]
mode = -0.0943
91.4% < 0 < 8.6%
95% HDI
-0.228
0.0359
theta[3]-theta[4]
mode = -0.184
99.7% < 0 < 0.3%
95% HDI
-0.313
-0.0553
theta[3]-theta[5]
theta[4]
theta[1]
theta[4]
theta[2]
theta[4]
theta[3]
mode = 0.589
3.2% < 0.5 < 96.8%
95% HDI
0.501
0.689
theta[4]
mode = -0.0877
92% < 0 < 8%
95% HDI
-0.22
0.0333
theta[4]-theta[5]
theta[5]
theta[1]
theta[5]
theta[2]
theta[5]
theta[3]
theta[5]
theta[4]
mode = 0.682
0% < 0.5 < 100%
95% HDI
0.595
0.771
theta[5]

Figure 3:

The posterior showed very small shrinkage when prior on kappa was set with mean=1 as the prior emphasizes small kappa values. But, There is significant shrinkage, when prior uniform was used. Bayesian posterior distribution provides more comprehensive and explicit description of the uncertainty of the estimate for all the parameters compared to MLE.

5A.

```
z=3
N=9
omega1 = 0.25
omega2 = 1 - omega1
kappa = 12
p1 = 0.5
p2 = 1 - p1
a1 = omega1*(kappa - 2) + 1
b1 = (1 - omega1) * (kappa - 2) + 1
pD = function(z,N,a,b) { exp( lbeta(z+a,N-z+b) - lbeta(a,b) ) }
a2 = omega2*(kappa - 2) + 1
b2 = (1 - omega2) * (kappa - 2) + 1
Bayes_factor = (pD(z,N, a1, b1) * p1)/(pD(z,N, a2, b2) * p2)
post_tails = Bayes_factor/ (1 + Bayes_factor)
post_heads = 1 - post_tails

show(Bayes_factor)
```

```
## [1] 4.683258
```

```
show(post_tails)
```

```
## [1] 0.8240446
```

```
show(post_heads)
```

```
## [1] 0.1759554
```

The posterior is exactly the opposite to the example given in the book. The posterior odds are 0.213 against the head-biased factory, which is to say 4.68 (i.e., 1/0.213) in favor of the tail-biased factory

5B.

```
z=6
N=9
omega1 = 0.25
omega2 = 1 - omega1
kappa = 120
p1 = 0.5
p2 = 1 - p1
a1 = omega1*(kappa - 2) + 1
b1 = (1 - omega1) * (kappa - 2) + 1
pD = function(z,N,a,b) { exp( lbeta(z+a,N-z+b) - lbeta(a,b) ) }
a2 = omega2*(kappa - 2) + 1
b2 = (1 - omega2) * (kappa - 2) + 1
Bayes_factor = (pD(z,N, a1, b1) * p1)/(pD(z,N, a2, b2) * p2)
```

```
post_tails = Bayes_factor/ (1 + Bayes_factor)
post_heads = 1 - post_tails

show(Bayes_factor)
```

```
## [1] 0.05020039
```

```
show(post_tails)
```

```
## [1] 0.04780077
```

```
show(post_heads)
```

```
## [1] 0.9521992
```

The posterior is 0.05 in favor of the tail-biased factory.

5C.

```
z=6
N=9
omega1 = 0.025
omega2 = 1 - omega1
kappa = 120
p1 = 0.5
p2 = 1 - p1
a1 = omega1*(kappa - 2) + 1
b1 = (1 - omega1) * (kappa - 2) + 1
pD = function(z,N,a,b) { exp( lbeta(z+a,N-z+b) - lbeta(a,b) ) }
a2 = omega2*(kappa - 2) + 1
b2 = (1 - omega2) * (kappa - 2) + 1
Bayes_factor = (pD(z,N, a1, b1) * p1)/(pD(z,N, a2, b2) * p2)
post_tails = Bayes_factor/ (1 + Bayes_factor)
post_heads = 1 - post_tails

show(Bayes_factor)
```

```
## [1] 0.0002858371
```

```
show(post_tails)
```

```
## [1] 0.0002857554
```

```
show(post_heads)
```

```
## [1] 0.9997142
```

The posterior is 0.0002 in favor of the tail-biased factory ie Factory is head biased.

5D.

```
z=6
N=9
omega1 = 0.025
omega2 = 1 - omega1
```

```r
kappa = 120
p1 = 0.05
p2 = 1 - p1
a1 = omega1*(kappa - 2) + 1
b1 = (1 - omega1) * (kappa - 2) + 1
pD = function(z,N,a,b) { exp( lbeta(z+a,N-z+b) - lbeta(a,b) ) }
a2 = omega2*(kappa - 2) + 1
b2 = (1 - omega2) * (kappa - 2) + 1
Bayes_factor = (pD(z,N, a1, b1) * p1)/(pD(z,N, a2, b2) * p2)
post_tails = Bayes_factor/ (1 + Bayes_factor)
post_heads = 1 - post_tails

show(Bayes_factor)
```

```
## [1] 1.504406e-05
```

```r
show(post_tails)
```

```
## [1] 1.504383e-05
```

```r
show(post_heads)
```

```
## [1] 0.999985
```

The posterior is negligibly in favor of the tail-biased factory ie Factory is higly head biased.