Stats205_HW3

Yushang Lai

2/6/2020

Problem 1

```
options(digits=5)
Prior.1 <- epi.betabuster(mode = 0.75, conf = 0.05, greaterthan = F, x = 0.6)
Prior.2 <- epi.betabuster(mode = 0.01, conf = 0.99, greaterthan = F, x = 0.02)
Prior.3 <- epi.betabuster(mode = 1, conf = 0.01, greaterthan = F, x = 0.8)


capture.output(as.data.frame(Prior.1, row.names = 'Prior One'))
```

```
## [1] "          shape1 shape2 mode    mean  median   lower  upper  variance"
## [2] "Prior One 23.567 8.5223 0.75 0.73442 0.73934 0.57161 0.8696 0.0058946"
```

```
capture.output(as.data.frame(Prior.2, row.names = 'Prior Two'))
```

```
## [1] "          shape1 shape2 mode     mean   median     lower    upper   variance"
## [2] "Prior Two 11.035 994.47 0.01 0.010975 0.010652 0.0055035 0.018276 1.0784e-05"
```

```
capture.output(as.data.frame(Prior.3, row.names = 'Prior Three'))
```

```
## [1] "            shape1 shape2 mode    mean  median   lower   upper  variance"
## [2] "Prior Three 20.638      1    1 0.95379 0.96697 0.83632 0.99877 0.0019471"
```

```
# knitr::kable(rbind(Prior.1, Prior.2, Prior.3) , digits = rep.int(4, 8))
df <- rbind(Prior.1, Prior.2, Prior.3)
pander(df)
```

Table 1: Table continues below

	shape1	shape2	mode	mean	median	lower
Prior.1	23.567	8.5223	0.75	0.73442	0.73934	0.57161
Prior.2	11.035	994.47	0.01	0.010975	0.010652	0.0055035
Prior.3	20.638	1	1	0.95379	0.96697	0.83632

	upper	variance
Prior.1	0.8696	0.0058946
Prior.2	0.018276	1.0784e-05
Prior.3	0.99877	0.0019471

```r
panderOptions('digits', 4)
```

Piror One is

$$Beta(23.567, 8.5223)$$

Piror two is

$$Beta(11.035, 994.47)$$

Piror three is

$$Beta(20.638, 1)$$

Problem 2

##(1)

$$y_i \sim Bin(n_i, \theta_i) \qquad ind$$
$$\theta_i \sim Beta(\alpha, \beta) \qquad i.i.d$$
$$\alpha, \beta \sim \mu, \eta$$
$$\mu, \eta \sim LN(m, \frac{1}{c}), Beta(a, b)$$

##(2)

```r
ESP.data=read.csv("./GanzStudiesUsed-56.csv", header=T)
head(ESP.data)
```

```
##   ï..n hits
## 1   32   14
## 2    7    6
## 3   30   13
## 4   30    7
## 5   20    2
## 6   10    9
```

```r
jags_model ="model{
for( i in 1 : N ) {
Y[i] ~ dbin(theta[i], n[i])
theta[i] ~ dbeta(alpha, beta)
}
alpha = eta * mu
beta = eta * (1-mu)
eta ~ dlnorm(m, 1/C)
mu ~ dbeta(a, b)
}"
jags.data = list(Y = ESP.data$hits, n = ESP.data$ï..n, N = dim(ESP.data)[1], a=20,b=40,m=0,C=3)
jags.param <- c("theta", "alpha", "beta","eta","mu")

jagsfit <- jags(data=jags.data, n.chains=5, inits=NULL,parameters.to.save =jags.param, n.iter=2000, n.bu
```

```
## module glm loaded
```

```
## Compiling model graph
##    Resolving undeclared variables
##    Allocating nodes
## Graph information:
##    Observed stochastic nodes: 56
##    Unobserved stochastic nodes: 58
```

```
##    Total graph size: 180
##
## Initializing model
```

##(3)

```r
op.prior <- epi.betabuster(mode = 0.25, conf = 0.95, greaterthan = F, x = 0.3)
pb.prior <- epi.betabuster(mode = 0.33, conf = 0.95, greaterthan = F, x = 0.36)
ps.prior <- epi.betabuster(mode = 0.25, conf = 0.95, greaterthan = F, x = 0.255)


capture.output(as.data.frame(op.prior, row.names = 'pior Open minded', optional = TRUE))
```

```
## [1] "                 shape1 shape2 mode    mean  median   lower   upper   variance"
## [2] "pior Open minded 58.825 174.48 0.25 0.25214 0.25143 0.19863 0.30968 0.00080481"
```

```r
capture.output(as.data.frame(pb.prior, row.names = 'Prior Psi Beliver', optional =T))
```

```
## [1] "                  shape1 shape2 mode    mean  median   lower   upper   variance"
## [2] "Prior Psi Beliver    100    202 0.33 0.33113 0.33075 0.27924 0.38513 0.00073096"
```

```r
capture.output(as.data.frame(ps.prior, row.names = 'Prior Psi skeptics', optional = T))
```

```
## [1] "                   shape1 shape2 mode    mean  median   lower   upper  variance"
## [2] "Prior Psi skeptics    100    298 0.25 0.25126 0.25084 0.20992 0.29496 0.0004715"
```

```r
df <- rbind(op.prior, pb.prior, ps.prior)
panderOptions('digits', 4)
pander(df)
```

Table 3: Table continues below

	shape1	shape2	mode	mean	median	lower	upper	variance
op.prior	58.825	174.48	0.25	0.25214	0.25143	0.19863	0.30968	0.00080481
pb.prior	100	202	0.33	0.33113	0.33075	0.27924	0.38513	0.00073096
ps.prior	100	298	0.25	0.25126	0.25084	0.20992	0.29496	0.0004715

Priors are the Beta distributions with above parameters.

##(4)

```r
jags.param <- c("alpha", "beta","mu")
jags.data = list(Y = ESP.data$hits, n = ESP.data$ï..n, N = dim(ESP.data)[1], a = op.prior$shape1,b=op.pr

jagsfit.op <- jags(data=jags.data, n.chains=5, inits=NULL,parameters.to.save =jags.param, n.iter=3000, n
```

```
## Compiling model graph
##    Resolving undeclared variables
##    Allocating nodes
## Graph information:
```

```
##    Observed stochastic nodes: 56
##    Unobserved stochastic nodes: 58
##    Total graph size: 180
##
## Initializing model
```

```
jagsfit.op$BUGSoutput$summary
```

```
##                mean        sd      2.5%       25%       50%       75%     97.5%
## alpha      9.04655  3.644474   4.19639   6.54019   8.32165  10.61217  18.46498
## beta      19.33049  7.645174   9.07227  14.11267  17.78209  22.90983  39.03660
## deviance 273.85172 10.998389 253.83401 265.99247 273.39450 281.06173 296.57446
## mu         0.31851  0.014091   0.29061   0.30926   0.31844   0.32795   0.34575
##            Rhat n.eff
## alpha    1.0064   510
## beta     1.0064   510
## deviance 1.0032  1100
## mu       1.0012  4100
```

```
jags.param <- c("alpha", "beta","mu")
jags.data = list(Y = ESP.data$hits, n = ESP.data$ï..n, N = dim(ESP.data)[1], a = pb.prior$shape1,b=pb.pr

jagsfit.pb <- jags(data=jags.data, n.chains=5, inits=NULL,parameters.to.save =jags.param, n.iter=3000, n
```

```
## Compiling model graph
##    Resolving undeclared variables
##    Allocating nodes
## Graph information:
##    Observed stochastic nodes: 56
##    Unobserved stochastic nodes: 58
##    Total graph size: 180
##
## Initializing model
```

```
jagsfit.pb$BUGSoutput$summary
```

```
##                mean        sd      2.5%       25%       50%       75%     97.5%
## alpha      9.35126  3.564911   4.53192   6.93848   8.69858  10.95789  18.12435
## beta      18.54913  7.177079   8.89970  13.66466  17.16668  21.85097  35.67414
## deviance 272.70758 10.694880 253.57577 265.29017 272.26007 279.58315 295.34388
## mu         0.33579  0.014199   0.30869   0.32621   0.33535   0.34495   0.36467
##            Rhat n.eff
## alpha    1.0046   740
## beta     1.0047   720
## deviance 1.0021  1800
## mu       1.0022  1800
```

```
jags.param <- c("alpha", "beta","mu")
jags.data = list(Y = ESP.data$hits, n = ESP.data$ï..n, N = dim(ESP.data)[1], a = ps.prior$shape1,b=ps.pr

jagsfit.ps <- jags(data=jags.data, n.chains=5, inits=NULL,parameters.to.save =jags.param, n.iter=3000, n
```

```
## Compiling model graph
##    Resolving undeclared variables
##    Allocating nodes
## Graph information:
##    Observed stochastic nodes: 56
```

```
##    Unobserved stochastic nodes: 58
##    Total graph size: 180
##
## Initializing model
```

```
jagsfit.ps$BUGSoutput$summary
```

```
##               mean        sd      2.5%       25%       50%       75%     97.5%
## alpha      8.73259  3.696050   4.04038   6.31175   8.01466  10.22420  17.92420
## beta      19.45671  8.044204   9.16774  14.19470  17.87163  22.69193  39.85249
## deviance 274.37528 11.119937 254.11952 266.69560 273.72615 281.41633 298.00059
## mu         0.30921  0.013566   0.28216   0.30004   0.30952   0.31858   0.33516
##            Rhat n.eff
## alpha    1.0018  3000
## beta     1.0017  2700
## deviance 1.0016  2900
## mu       1.0011  5000
```

Posterior of Open-mdinded: mean is 0.3189 0.2899 0.34821 0.33566 0.30754 0.36332 0.30943 0.28373 0.33596

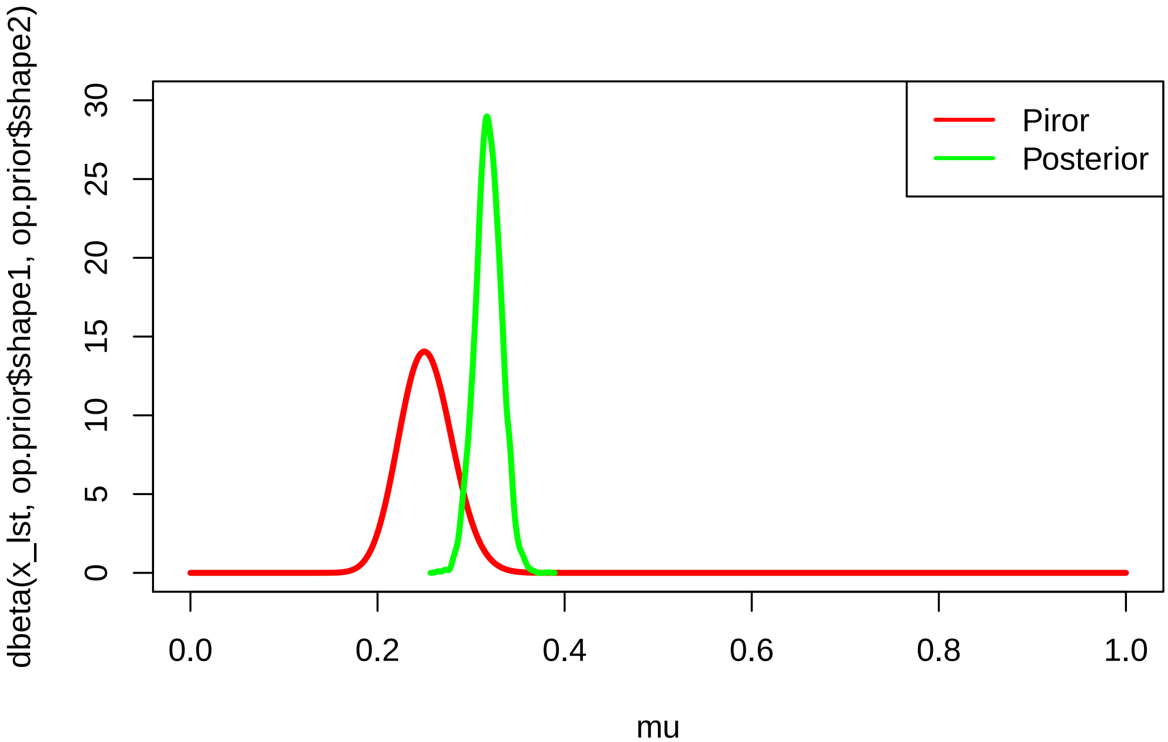
```
qtls = matrix(NA,3,3)
qtls[1,] = c(0.3189,0.2899,0.34821)
qtls[2,] = c(0.33566,0.30754,0.36332)
qtls[3,] = c(0.30943,0.28373,0.33596)
df = data.frame(qtls)
row.names(df) = c('Open-minded posterior','Psi beliver posterior','Psi skeptics posterior')
colnames(df)= c('mean','2.5%','97.5%')
knitr::kable(df, format = "markdown")
```

	mean	2.5%	97.5%
Open-minded posterior	0.31890	0.28990	0.34821
Psi beliver posterior	0.33566	0.30754	0.36332
Psi skeptics posterior	0.30943	0.28373	0.33596

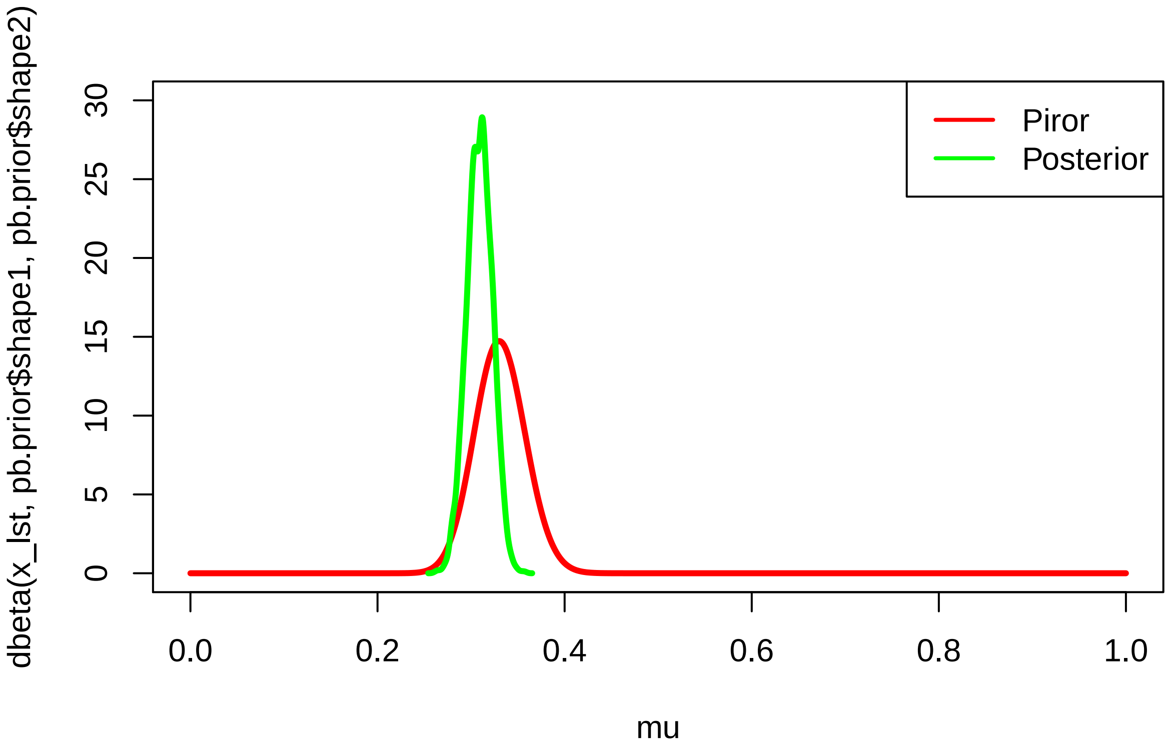
##(5)

Note that for the Posterior Distribution. The left and right curve should be converges to 0. Since we use MCMC, they don't show in those figures but does not affect the analysis much.

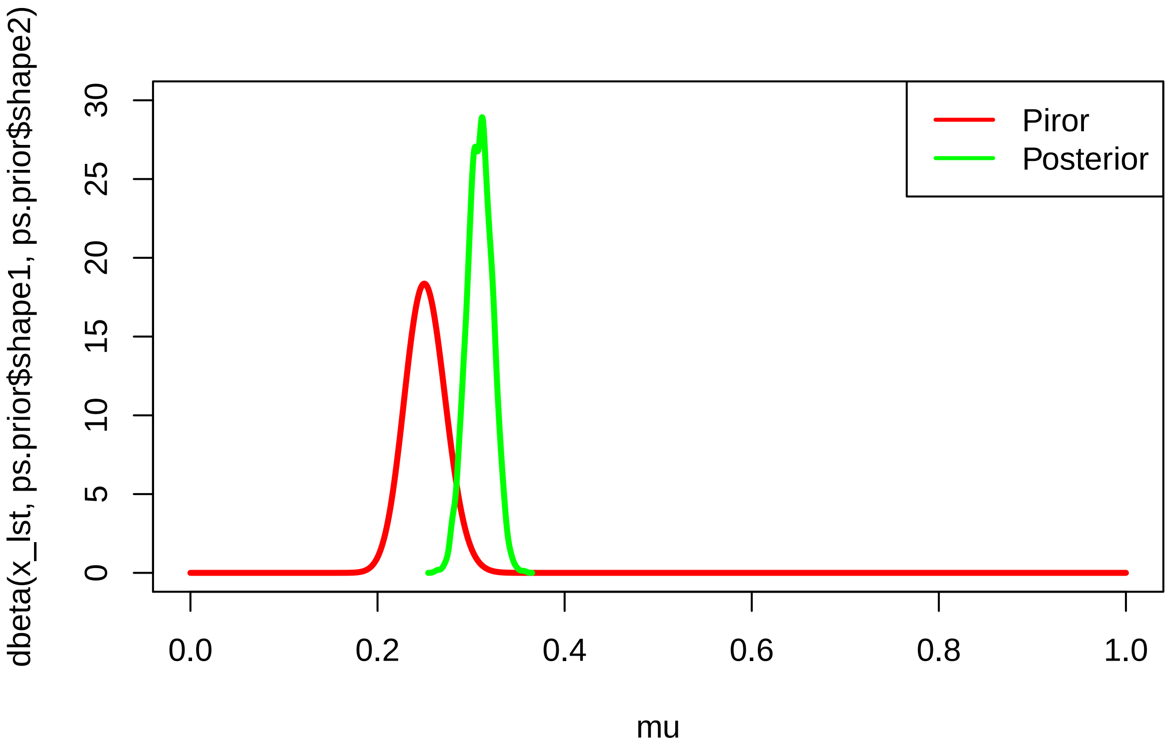
```
x_lst = seq(0,1,by=0.001)
plot(x_lst,dbeta(x_lst,op.prior$shape1,op.prior$shape2),col="red", type="l", xlab = 'mu',lwd=3,ylim = c
lines(density(jagsfit.op$BUGSoutput$sims.matrix[,4]),col="green", type="l",ylab = 'pdf',lwd=3)
legend('topright',col=c('red','green'),legend=c('Piror',"Posterior"),lwd=c(2,2))
```




```
x_lst = seq(0,1,by=0.001)
plot(x_lst,dbeta(x_lst,pb.prior$shape1,pb.prior$shape2),col="red", type="l", xlab = 'mu',lwd=3,ylim = c
lines(density(jagsfit.ps$BUGSoutput$sims.matrix[,4]),col="green", type="l",ylab = 'pdf',lwd=3)
legend('topright',col=c('red','green'),legend=c('Piror',"Posterior"),lwd=c(2,2))
```




```
x_lst = seq(0,1,by=0.001)
plot(x_lst,dbeta(x_lst,ps.prior$shape1,ps.prior$shape2),col="red", type="l", xlab = 'mu',lwd=3,ylim = c
lines(density(jagsfit.ps$BUGSoutput$sims.matrix[,4]),col="green", type="l",ylab = 'pdf',lwd=3,ylim = c(
legend('topright',col=c('red','green'),legend=c('Piror',"Posterior"),lwd=c(2,2))
```



For open-minded it's posterior mode and mean get larger with less std. THis infer that they under estimated the power of Psi.

For those Psi beliver, it's posterior mode and mean get smaller with less std. THis infer that they over estimated the power of Psi.

For those Psi Skeptic, it's posterior mode and mean get larger with less std. THis infer that they under estimated the power of Psi.

Problem 4

##(1)

$$z_{s|c} \sim Bin(N_{s|c}, \theta_{s|c}) \qquad ind$$

$$\theta_{s|c} \sim Beta(\alpha_c, \beta_c) \qquad i.i.d$$
$$\alpha_c, \beta_c \sim \mu, \eta$$
$$\mu \sim Beta(a, b)$$
$$\eta \sim LN(m, C)$$

##(2)

```
data=read.csv("./BattingAverage.csv", header=T)
head(data)
```

```
##          Player     PriPos Hits AtBats PlayerNumber PriPosNumber
## 1 Fernando Abad    Pitcher    1      7            1            1
## 2   Bobby Abreu Left Field   53    219            2            7
## 3    Tony Abreu   2nd Base   18     70            3            4
## 4 Dustin Ackley   2nd Base  137    607            4            4
## 5    Matt Adams   1st Base   21     86            5            3
## 6 Nathan Adcock    Pitcher    0      1            6            1
```

```
jags.data=list(Z = data$Hits, n = data$AtBats, Pos=data$PriPosNumber,N =nrow(data),m=0, C=3,a =23,b=77)

jags_model = "model{
  for (i in 1 : N)
    {
    Z[i] ~ dbin(theta[i], n[i])
    theta[i] ~ dbeta(alpha[Pos[i]], beta[Pos[i]])
    }
  for(j in 1:9)
    {
    alpha[j] =eta[j]*mu[j]
    beta[j] = eta[j]*(1-mu[j])
    eta[j] ~ dlnorm(m, 1/C)
    mu[j] ~ dbeta(a, b)
    }
    compare1 = theta[573]-theta[143]
    compare2 = theta[142]-theta[921]
    compare = mu[8]-mu[2]
  }"

jags.param <- c("theta", "mu","compare1","compare2","compare")
jags_fit <- jags(data = jags.data, n.chains = 5, inits = NULL, parameters.to.save = jags.param,n.iter=3
```

```
## Compiling model graph
##    Resolving undeclared variables
##    Allocating nodes
## Graph information:
##    Observed stochastic nodes: 948
##    Unobserved stochastic nodes: 966
##    Total graph size: 3847
##
## Initializing model
```

```
# jags_fit$BUGSoutput$summary
jags.mcmc = as.mcmc(jags_fit)
# mcmc_trace(jags.mcmc, pars = c("mu[2]"))
```

##(3)

```
Number_lst = unique(data$PriPosNumber)
Name_lst = unique(data$PriPos)
capture.output(as.data.frame(Number_lst))
```

```
##  [1] "  Number_lst" "1          1" "2          7" "3          4" "4          3"
##  [6] "5          5" "6          2" "7          6" "8          8" "9          9"
```

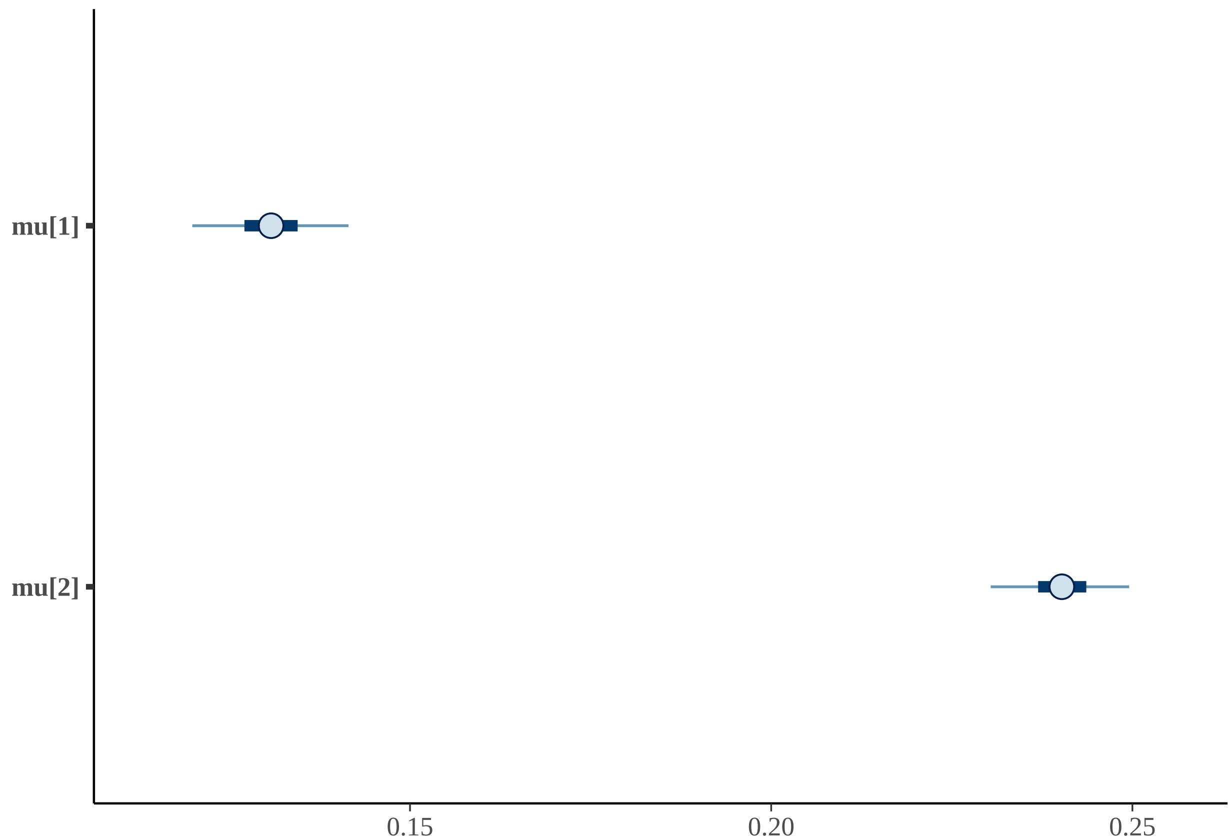
```
capture.output(as.data.frame(Name_lst))
```

```
##  [1] "      Name_lst" "1      Pitcher" "2   Left Field" "3     2nd Base"
```

```
##  [5] "4     1st Base" "5     3rd Base" "6      Catcher" "7    Shortstop"
##  [9] "8 Center Field" "9  Right Field"
```

```
# df <- rbind(Number_lst,Name_lst)
# pander(df)
```

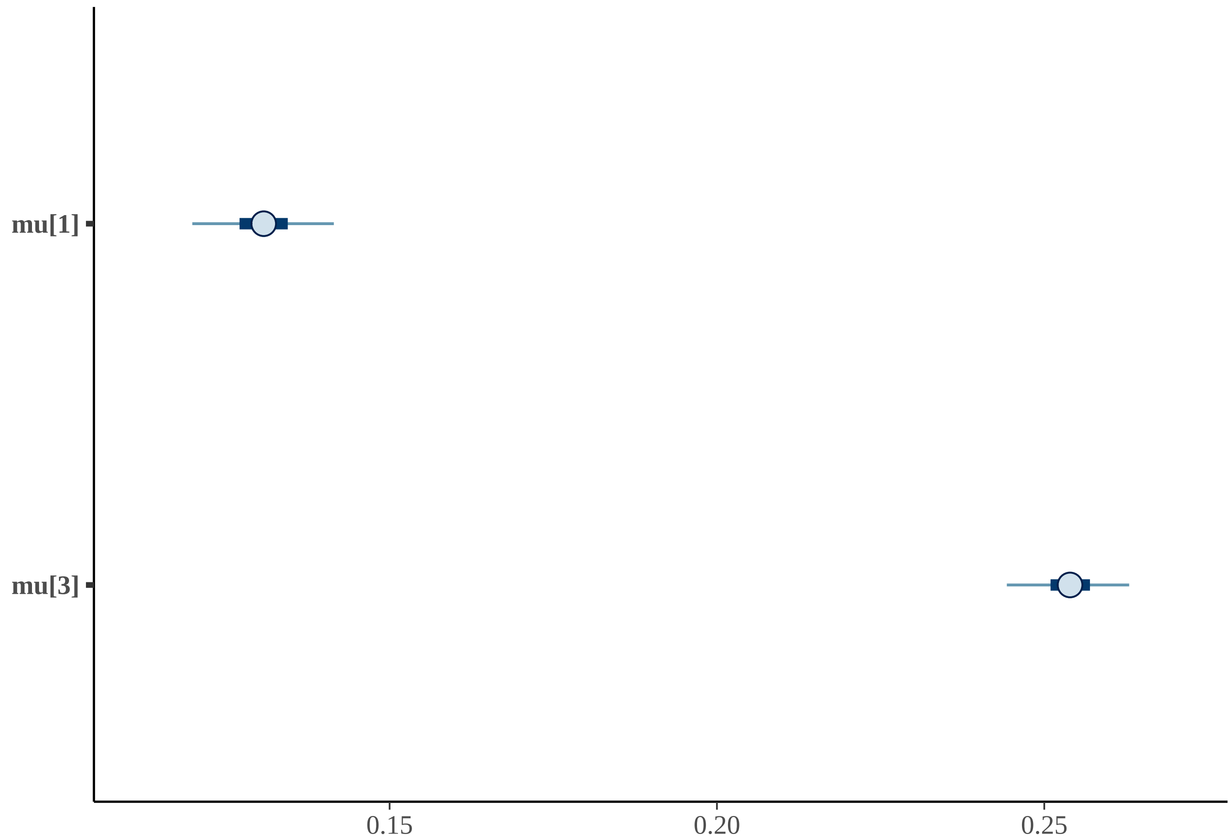
```
mcmc_intervals(jags.mcmc, pars=c("mu[1]", "mu[2]"), prob = 0.5, # 80% intervals - inner
prob_outer = 0.95, # 95% - outer
point_est = "mean")
```



By the table, mu[1] is pitcher and mu[2] is catcher.From above, we can infer that catcher (mean of at around 0.24) is better than pitcher (mean at bat around 0.13) at bat with credible interval 95%.

##(4)

```
mcmc_intervals(jags.mcmc, pars=c("mu[1]", "mu[3]"), prob = 0.5, # 80% intervals - inner
prob_outer = 0.95, # 95% - outer
point_est = "mean")
```



By the table, mu[1] is pitcher and mu[3] is First Base.From above, we can infer that First base(mean of at around 0.26) is better than pitcher (mean at bat around 0.13) at bat with credible interval 95%.

##(5)

```
which("Welington Castillo" == data$Player)
```

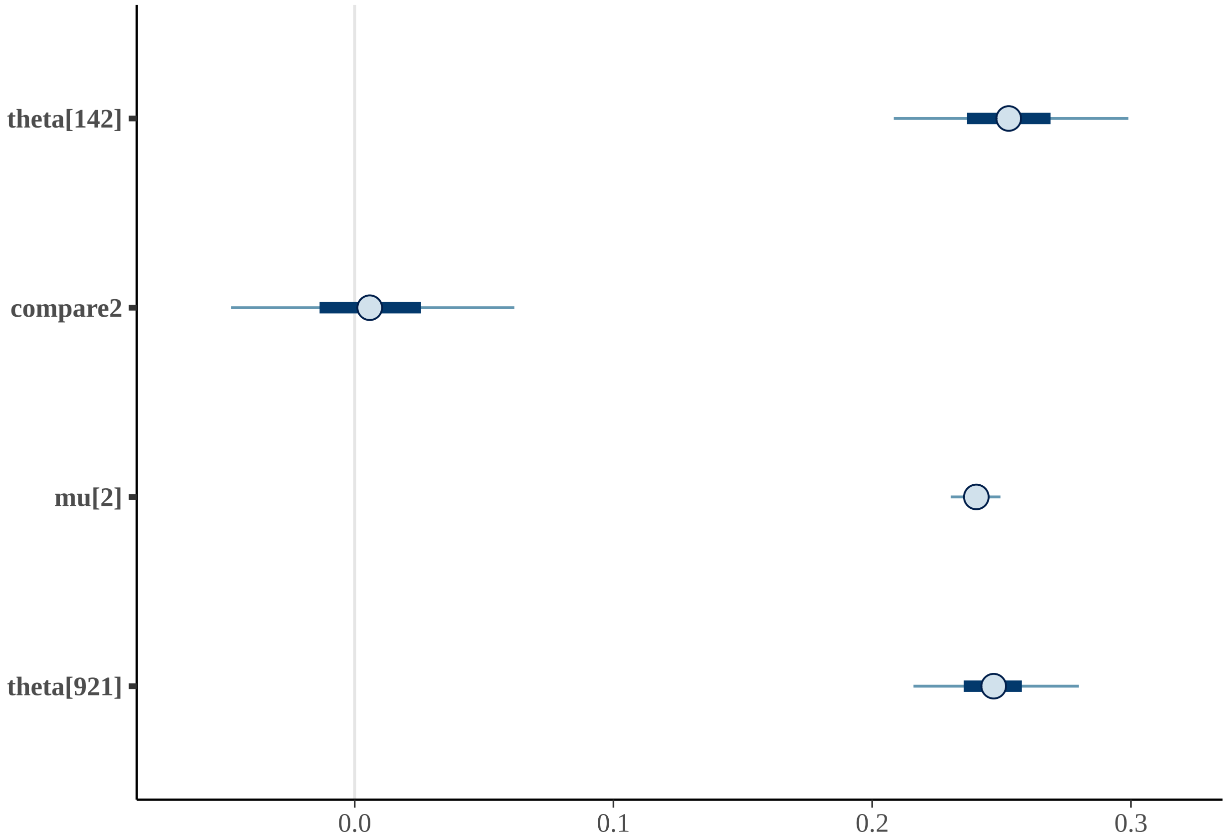
```
## [1] 142
```

```
which("Matt Wieters" == data$Player)
```

```
## [1] 921
```

142 catcher mu2 921 catcher mu2 573 central mu8 143 catcher mu2

```
mcmc_intervals(jags.mcmc, pars=c("theta[142]", "compare2","mu[2]","theta[921]"), prob = 0.5, # 80% inte
prob_outer = 0.95, # 95% - outer
point_est = "mean")
```




```
p921 = 2/8
p142 = 45/170
print(p921)
```

```
## [1] 0.25
```

```
print(p142)
```

```
## [1] 0.26471
```

By the table, theta[142] is W.C. and M.W. From individuals above, we can infer that W.C. is slightly better than M.W. for at bat by comparing mean. however, M.W. preformance is more stable than W.C. since M.W. has smaller credible interval than W.C. By individual shrikage, we see that compare between two players has mean 5.4755e-03 with 95 credible interval (-5.0248e-02 6.1392e-02) which means that two players differs only 5 at bats for 100 games. Thus, we can conclude that two players has similar abilitilies at bat.there is an evidence of shrinakge in the estimates. Since both of the mean of the at bat for players comes close to mean from their all probability of at bat.

##(6)

```
which("Andrew McCutchen" == data$Player)
```

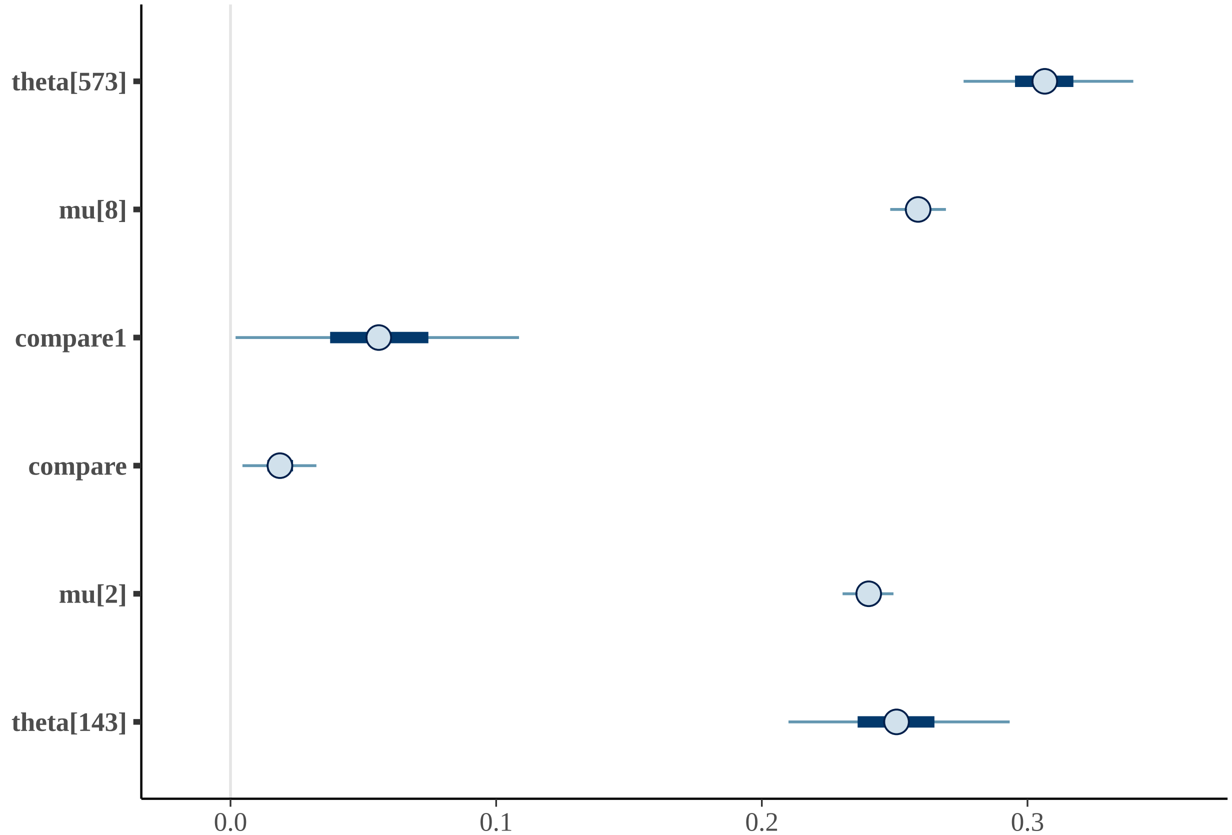
```
## [1] 573
```

```
which("Jason Castro" == data$Player)
```

```
## [1] 143
```

```
mcmc_intervals(jags.mcmc, pars=c("theta[573]", "mu[8]","compare1","compare","mu[2]","theta[143]"), prob
prob_outer = 0.95, # 95% - outer
```

```r
point_est = "mean")
```




```r
print(66/257)
```

```
## [1] 0.25681
```

```r
print(194/593)
```

```
## [1] 0.32715
```

By the table, theta[573] is J.C. and A.M. From above, we can infer that J.C. is better than A.M. for at bat by comparing mean around 0.31 to 0.25.And J.C. preformed more stable than A.M since we can see a shrinkage from distribution of J.C. than A.M. . which means J.C. can make 0 to 10 more balls average 5 balls than A.M. Thus, we conclude that J.C. is better. Thus, there is evidence of shrinakge in the estimates. Since both of the mean of the at bat for players comes close to mean from their all probability of at bat.