```
    ###################################################################
    ###################################################################
    #     This code will replicate:
    #       Table 1, Table A2, Table A3, Figure 3, and Figure A3
    ###################################################################
    ###################################################################
>
>     ### Required Adjustments
>     setwd("C:\\temp1")  # Set your working directory. All data and bugs files should be stored in this directory.
>     bugs_directory <- "C:\\WinBUGS14"  # It should be the directory of your WinBUGS program file
>
>     ### Required packages
>     library(R2WinBUGS)
Loading required package: coda
Loading required package: boot
Warning message:
package 'boot' was built under R version 4.0.5
>     library(coda)
>
>     # To create a table with results
>     bayes.easy.write <- function(id="test", stats, label, D=3){
+     M <- nrow(stats)
+     m2.sum <- rep("", 2*M)
+     m2.row <- rep("", 2*M)
+     for(i in 1:M){
+     m2.sum[2*i-1] <- paste(round(stats[i,1],digits=D), "", sep="")
```

```
+     if(max(stats[i,6],stats[i,7]) > 0 & (min(stats[i,6],stats[i,7]) > 0)) m2.sum[2*i-1] <-
paste(round(stats[i,1],digits=D), "*", sep="")

+     if(max(stats[i,6],stats[i,7]) < 0 & (min(stats[i,6],stats[i,7]) < 0)) m2.sum[2*i-1] <-
paste(round(stats[i,1],digits=D), "*", sep="")

+     if(max(stats[i,3],stats[i,4]) > 0 & (min(stats[i,3],stats[i,4]) > 0)) m2.sum[2*i-1] <-
paste(round(stats[i,1],digits=D), "**", sep="")

+     if(max(stats[i,3],stats[i,4]) < 0 & (min(stats[i,3],stats[i,4]) < 0)) m2.sum[2*i-1] <-
paste(round(stats[i,1],digits=D), "**", sep="")

+     if(max(stats[i,8],stats[i,9]) > 0 & (min(stats[i,8],stats[i,9]) > 0)) m2.sum[2*i-1] <-
paste(round(stats[i,1],digits=D), "***", sep="")

+     if(max(stats[i,8],stats[i,9]) < 0 & (min(stats[i,8],stats[i,9]) < 0)) m2.sum[2*i-1] <-
paste(round(stats[i,1],digits=D), "***", sep="")

+     m2.sum[2*i] <- paste("{", round(stats[i,2], digits=D), "}", sep="")

+     m2.row[2*i-1] <- label[i]

+     }

+     m2.table <- cbind(m2.row, m2.sum)

+     colnames(m2.table) <- c("Variable", "Coefficient")

+     return(m2.table)

+     }



    ##############################################################################################

    ##############################################################################################

    ############ Replicate Table 1

    ##############################################################################################

    ##############################################################################################



    ########################################

    # Produce the First Column of Table 1
```

```
    ########################################
>
>
>     # Micro-level data
>     Micro_Data <- read.csv("House_Micro_Data.csv", header=T)
>
>     # Congress-level data
>     Macro_Data <- read.csv("Macro_Data.csv", header=T)
>     Dem.prez <- ifelse(Macro_Data$President_party==100, 1, 0)
>
>     year.odd <- seq(from=1971, to=2015, by=2)
>     T <- length(year.odd)
>
>     # Standardize the continuous variables
>     Micro_Data$seniority_term.std  <- as.vector((Micro_Data$seniority_term - mean(Micro_Data$seniority_term,
na.rm=T))/(sd(Micro_Data$seniority_term, na.rm=T)))
>     Micro_Data$Bases.std <- as.vector((Micro_Data$Bases - mean(Micro_Data$Bases, na.rm=T))/(sd(Micro_Data$Bases,
na.rm=T)))
>     Micro_Data$LES.std <- as.vector((Micro_Data$effectiveness - mean(Micro_Data$effectiveness,
na.rm=T))/(sd(Micro_Data$effectiveness, na.rm=T)))
>     Macro_Data$GOP_Majority <- ifelse(Macro_Data$House_Majority==200, 1, 0)
>     GOP_share <- ifelse(Macro_Data$GOP_Majority==1,  Macro_Data$House_Majority_Seats/Macro_Data$House_Total_Seats,
(Macro_Data$House_Total_Seats - Macro_Data$House_Majority_Seats)/Macro_Data$House_Total_Seats)
>     GOP_share.std  <- as.vector((GOP_share - mean(GOP_share, na.rm=T))/(sd(GOP_share, na.rm=T)))
>
>     Y  <- Micro_Data$Yea_share
>     X1 <- Micro_Data$FP_mean
>     X2 <- Micro_Data$majority_member
>     X3 <- Micro_Data$seniority_term.std
>     X4 <- Micro_Data$FR_cmt
```

```
>     X5 <- Micro_Data$AS_cmt
>     X6 <- Micro_Data$veteran
>     X7 <- Micro_Data$Bases.std
>     X8 <- Micro_Data$LES.std
>     X9 <- Micro_Data$dovish_riders
>
>     Z1 <- Macro_Data$War
>     Z2 <- Macro_Data$GOP_Majority
>     Z3 <- Dem.prez
>
>     congress <- Micro_Data$congress - 91
>     N <- length(Y)
>     J <- length(Z1)
>     n.beta <- 9
>     n.gamma <- 4
>
>     data <- list("N", "J", "n.beta","n.gamma", "Y",
+                  "X1","X2","X3","X4","X5","X6","X7","X8","X9", "congress","Z1","Z2","Z3")
>
>     # setup initial values
>     init1 <- list(beta=rnorm(n.beta), sigma.y=runif(1),
+                   gamma=rnorm(n.gamma),
+                   alpha=rnorm(J), sigma.alpha=runif(1))
>     init2 <- list(beta=rnorm(n.beta), sigma.y=runif(1),
+                   gamma=rnorm(n.gamma),
+                   alpha=rnorm(J), sigma.alpha=runif(1))
>     init3 <- list(beta=rnorm(n.beta), sigma.y=runif(1),
+                   gamma=rnorm(n.gamma),
```

```
+                   alpha=rnorm(J), sigma.alpha=runif(1))
>
>     inits <- list(init1, init2, init3)
>
>     # setup parameters
>     parameters = c("beta", "sigma.y", "gamma", "alpha", "sigma.alpha")
>
>     # Test bugs.
>     Model.fit = bugs(data, inits, parameters, "model1.bug",
+                         working.directory=getwd(), bugs.directory=bugs_directory,
+                         n.chains=3, n.thin=10, n.burnin=2000, n.iter=12000)
>
>     Rhats <- Model.fit$summary[,8]
>     range(Rhats)
[1] 1.000514 1.002741
>
>     # Analyze the draws from the Posterior distribution
>
>     chain1 <- read.coda(output.file="coda1.txt", index.file="codaIndex.txt", quiet=T)
>     chain2 <- read.coda(output.file="coda2.txt", index.file="codaIndex.txt", quiet=T)
>     chain3 <- read.coda(output.file="coda3.txt", index.file="codaIndex.txt", quiet=T)
>
>     MCMC <- rbind(chain1, chain2, chain3)
>     iter <- nrow(MCMC)
>
>     alpha.finder  <- seq(from=1, to=T, by=1)
>     beta.finder   <- seq(from=(T+1), to=(T+n.beta), by=1)
>     gamma.finder  <- seq(from=(T+n.beta+2), to=(T+n.beta+1+n.gamma), by=1)
```

```
>     sigma.y.finder <- T+n.beta+1+n.gamma + 2
>
>     alpha.mc  <- MCMC[,alpha.finder]
>     beta.mc   <- MCMC[,beta.finder]
>     gamma.mc  <- MCMC[,gamma.finder]
>     sigma.y.mc<- MCMC[,sigma.y.finder]
>
>     beta.label  <- c("Hawkiness",
+                      "Majority Party",
+                      "Seniority",
+                      "Foreign Affairs",
+                      "Armed Services",
+                      "Veteran",
+                      "No of Military Bases",
+                      "Legislative Effectiveness",
+                      "Dovish Amendments")
>     gamma.label <- c("Intercept",
+                      "Major War",
+                      "GOP Majority",
+                      "Democratic Prez")
>
>
>     ### Credible Intervals
>
>     beta.CI  <- round(apply(beta.mc,  2, FUN=quantile, probs=c(0.025, 0.975, 0.5, 0.05, 0.95, 0.005, 0.995)), 3)
>     gamma.CI  <- round(apply(gamma.mc,  2, FUN=quantile, probs=c(0.025, 0.975, 0.5, 0.05, 0.95, 0.005, 0.995)), 3)
>     alpha.CI  <- round(apply(alpha.mc,  2, FUN=quantile, probs=c(0.025, 0.975, 0.5, 0.05, 0.95, 0.005, 0.995)), 3)
>
```

```
>     beta <- round(rbind(apply(beta.mc,  2, FUN=mean), apply(beta.mc,  2, FUN=sd),    beta.CI),3)
>     rownames(beta)[c(1,2)] <- c("mean", "sd")
>     beta.stats <- t(beta)
>
>     gamma <- round(rbind(apply(gamma.mc,  2, FUN=mean), apply(gamma.mc,  2, FUN=sd),    gamma.CI),3)
>     rownames(gamma)[c(1,2)] <- c("mean", "sd")
>     gamma.stats <- t(gamma)
>
>     # Summary Outputs for Table
>
>     micro <- bayes.easy.write(id="house_m1_micro_result", stats=beta.stats, label=beta.label)
>     macro <- bayes.easy.write(id="house_m1_macro_result", stats=gamma.stats, label=gamma.label)
>     print(rbind(micro, macro))
      Variable                    Coefficient
 [1,] "Hawkiness"                 "1.726***"
 [2,] ""                          "{0.313}"
 [3,] "Majority Party"            "6.017***"
 [4,] ""                          "{1.54}"
 [5,] "Seniority"                 "3.376***"
 [6,] ""                          "{0.811}"
 [7,] "Foreign Affairs"           "-3.077"
 [8,] ""                          "{1.998}"
 [9,] "Armed Services"            "-0.609"
[10,] ""                          "{1.499}"
[11,] "Veteran"                   "-0.176"
[12,] ""                          "{1.458}"
[13,] "No of Military Bases"      "0.391"
[14,] ""                          "{0.792}"
```

```
[15,] "Legislative Effectiveness" "0.951"
[16,] ""                          "{0.804}"
[17,] "Dovish Amendments"         "-2.91**"
[18,] ""                          "{1.444}"
[19,] "Intercept"                 "5.376"
[20,] ""                          "{3.459}"
[21,] "Major War"                 "1.772"
[22,] ""                          "{3.133}"
[23,] "GOP Majority"              "2.172"
[24,] ""                          "{3.182}"
[25,] "Democratic Prez"           "2.277"
[26,] ""                          "{3.12}"
>     print(Model.fit$DIC)
[1] 6933.59
>     print(N)
[1] 782
>
>     ###################################################
>     ### Second column of Table 1
>     ###################################################
>
>     Dovish_Data <- subset(Micro_Data, dovish_riders==1)
>
>     Y  <- Dovish_Data$Yea_share
>
>     # Micro-level predictors
>     X1 <- Dovish_Data$FP_mean
>     X2 <- Dovish_Data$majority_member
```

```
>     X3 <- Dovish_Data$seniority_term.std
>     X4 <- Dovish_Data$FR_cmt
>     X5 <- Dovish_Data$AS_cmt
>     X6 <- Dovish_Data$veteran
>     X7 <- Dovish_Data$Bases.std
>     X8 <- Dovish_Data$LES.std
>
>     # Macro-level predictors
>     Z1 <- Macro_Data$War
>     Z2 <- Macro_Data$GOP_Majority
>     Z3 <- Dem.prez
>
>     congress <- Dovish_Data$congress - 91
>
>     N <- length(Y)
>     J <- length(Z1)
>     n.beta <- 8
>     n.gamma <- 4
>
>     data <- list("N", "J", "n.beta","n.gamma", "Y",
+                  "X1", "X2", "X3", "X4", "X5", "X6", "X7", "X8", "congress", "Z1","Z2","Z3")
>
>     # setup initial values
>     init1 <- list(beta=rnorm(n.beta), sigma.y=runif(1),
+                   gamma=rnorm(n.gamma),
+                   alpha=rnorm(J), sigma.alpha=runif(1))
>     init2 <- list(beta=rnorm(n.beta), sigma.y=runif(1),
+                   gamma=rnorm(n.gamma),
```

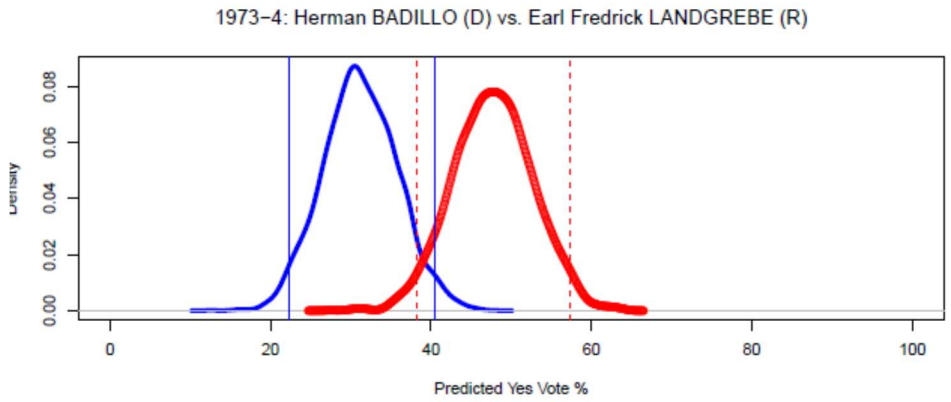
```
+                   alpha=rnorm(J), sigma.alpha=runif(1))
>     init3 <- list(beta=rnorm(n.beta), sigma.y=runif(1),
+                   gamma=rnorm(n.gamma),
+                   alpha=rnorm(J), sigma.alpha=runif(1))
>
>     inits <- list(init1, init2, init3)
>
>     # setup parameters
>     parameters = c("beta", "sigma.y", "gamma", "alpha", "sigma.alpha")
>
>     # Test bugs.
>     Model.fit = bugs(data, inits, parameters, "model2.bug",
+                         working.directory=getwd(), bugs.directory=bugs_directory,
+                         n.chains=3, n.thin=10, n.burnin=2000, n.iter=12000)
>
>     Rhats <- Model.fit$summary[,8]
>     range(Rhats)
[1] 1.000505 1.003415
>
>     # Analyze the draws from the Posterior distribution
>
>     chain1 <- read.coda(output.file="coda1.txt", index.file="codaIndex.txt", quiet=T)
>     chain2 <- read.coda(output.file="coda2.txt", index.file="codaIndex.txt", quiet=T)
>     chain3 <- read.coda(output.file="coda3.txt", index.file="codaIndex.txt", quiet=T)
>
>     MCMC <- rbind(chain1, chain2, chain3)
>     iter <- nrow(MCMC)
>
```

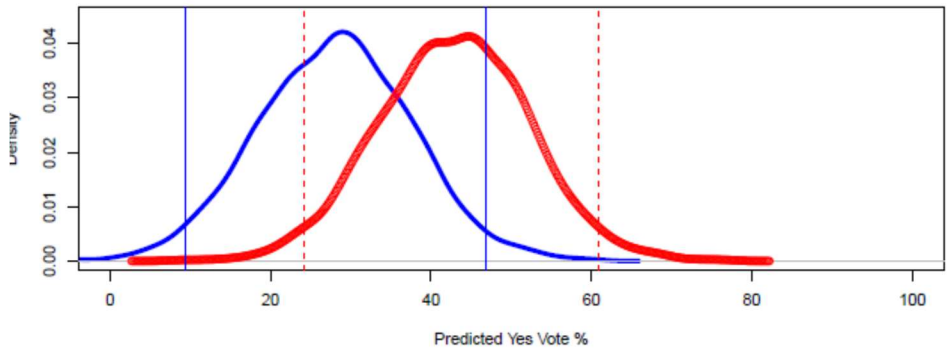
```
>     alpha.finder  <- seq(from=1, to=T, by=1)
>     beta.finder   <- seq(from=(T+1), to=(T+n.beta), by=1)
>     gamma.finder  <- seq(from=(T+n.beta+2), to=(T+n.beta+1+n.gamma), by=1)
>     sigma.y.finder <- T+n.beta+1+n.gamma + 2
>
>     alpha.mc  <- MCMC[,alpha.finder]
>     beta.mc   <- MCMC[,beta.finder]
>     gamma.mc  <- MCMC[,gamma.finder]
>     sigma.y.mc<- MCMC[,sigma.y.finder]
>
>     alpha.label <- paste(92:114, "th Congress", sep="")
>     beta.label  <- c("Hawkiness",
+                      "Majority Party",
+                      "Seniority",
+                      "Foreign Affairs",
+                      "Armed Services",
+                      "Veteran",
+                      "No of Military Bases",
+                      "Legislative Effectiveness")
>     gamma.label <- c("Intercept",
+                      "Major War",
+                      "GOP Majority",
+                      "Democratic Prez")
>
>
>     # Credible Intervals
>
>     beta.CI  <- round(apply(beta.mc,  2, FUN=quantile, probs=c(0.025, 0.975, 0.5, 0.05, 0.95, 0.005, 0.995)), 3)
```

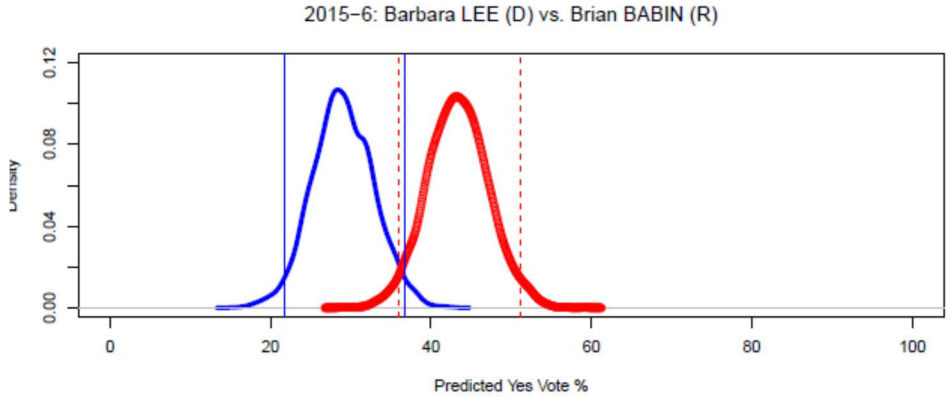
```
>     gamma.CI  <- round(apply(gamma.mc,  2, FUN=quantile, probs=c(0.025, 0.975, 0.5, 0.05, 0.95, 0.005, 0.995)), 3)
>     alpha.CI  <- round(apply(alpha.mc,  2, FUN=quantile, probs=c(0.025, 0.975, 0.5, 0.05, 0.95, 0.005, 0.995)), 3)
>
>     beta <- round(rbind(apply(beta.mc,  2, FUN=mean), apply(beta.mc,  2, FUN=sd),    beta.CI),3)
>     rownames(beta)[c(1,2)] <- c("mean", "sd")
>     beta.stats <- t(beta)
>
>     gamma <- round(rbind(apply(gamma.mc,  2, FUN=mean), apply(gamma.mc,  2, FUN=sd),    gamma.CI),3)
>     rownames(gamma)[c(1,2)] <- c("mean", "sd")
>     gamma.stats <- t(gamma)
>
>     # Summary Outputs for Table
>     micro <- bayes.easy.write(id="senate_m2_micro_result", stats=beta.stats, label=beta.label)
>     macro <- bayes.easy.write(id="senate_m2_macro_result", stats=gamma.stats, label=gamma.label)
>     print(rbind(micro, macro))
      Variable                    Coefficient
 [1,] "Hawkiness"                 "1.935***"
 [2,] ""                          "{0.367}"
 [3,] "Majority Party"            "5.354***"
 [4,] ""                          "{1.746}"
 [5,] "Seniority"                 "2.868***"
 [6,] ""                          "{0.915}"
 [7,] "Foreign Affairs"           "-4.508**"
 [8,] ""                          "{2.176}"
 [9,] "Armed Services"            "-1.246"
[10,] ""                          "{1.683}"
[11,] "Veteran"                   "-1.183"
[12,] ""                          "{1.696}"
```

```
[13,] "No of Military Bases"      "1.274"
[14,] ""                          "{0.936}"
[15,] "Legislative Effectiveness" "0.738"
[16,] ""                          "{0.883}"
[17,] "Intercept"                 "5.743*"
[18,] ""                          "{3.478}"
[19,] "Major War"                 "1.884"
[20,] ""                          "{3.094}"
[21,] "GOP Majority"              "2.193"
[22,] ""                          "{3.163}"
[23,] "Democratic Prez"           "2.506"
[24,] ""                          "{3.117}"
>     print(Model.fit$DIC)
[1] 4651.21
>     print(N)
[1] 531


    ######################################################
    # To produce the First Column of Figure 3
    # This utilizes the results of th second column of Table 1
    ######################################################
```


1973–4: Herman BADILLO (D) vs. Earl Fredrick LANDGREBE (R)
Density
0.00
0.02
0.04
0.06
0.08
0
20
40
60
80
100
Predicted Yes Vote %


2003–4: Barbara LEE (D) vs. Sam JOHNSON (R)
Density
0.00
0.01
0.02
0.03
0.04
0
20
40
60
80
100
Predicted Yes Vote %


2015–6: Barbara LEE (D) vs. Brian BABIN (R)
Density
0.00
0.04
0.08
0.12
0
20
40
60
80
100
Predicted Yes Vote %

```
>
>     ################################################
>     ### Third column of Table 1
>     ################################################
>
>     # Import the data
>
>     # Micro-level data
>     Micro_Data <- read.csv("Senate_Micro_Data.csv", header=T)
>
>     # Congress-level data
>     Macro_Data <- read.csv("Macro_Data.csv", header=T)
>     Dem.prez <- ifelse(Macro_Data$President_party==100, 1, 0)
>
>     year.odd <- seq(from=1971, to=2015, by=2)
>     T <- length(year.odd)
>
>     # Standardize the continuous variables
>     Micro_Data$Seniority_Year.std  <- as.vector((Micro_Data$Seniority_Year - mean(Micro_Data$Seniority_Year,
na.rm=T))/(sd(Micro_Data$Seniority_Year, na.rm=T)))
>     Micro_Data$Bases.std <- as.vector((Micro_Data$Bases - mean(Micro_Data$Bases, na.rm=T))/(sd(Micro_Data$Bases,
na.rm=T)))
>     Micro_Data$LES.std <- as.vector((Micro_Data$effectiveness - mean(Micro_Data$effectiveness,
na.rm=T))/(sd(Micro_Data$effectiveness, na.rm=T)))
>     Macro_Data$GOP_Majority <- ifelse(Macro_Data$Senate_Majority==200, 1, 0)
>     GOP_share <- ifelse(Macro_Data$GOP_Majority==1,  Macro_Data$Senate_Majority_Seats/Macro_Data$Senate_Total_Seats,
(Macro_Data$Senate_Total_Seats - Macro_Data$Senate_Majority_Seats)/Macro_Data$Senate_Total_Seats)
>     GOP_share.std  <- as.vector((GOP_share - mean(GOP_share, na.rm=T))/(sd(GOP_share, na.rm=T)))
```

```
>
>
>     Y  <- Micro_Data$Yea_share
>
>     # Micro-level predictors
>     X1 <- Micro_Data$FP_mean
>     X2 <- Micro_Data$majority_member
>     X3 <- Micro_Data$Seniority_Year.std
>     X4 <- Micro_Data$FR_cmt
>     X5 <- Micro_Data$AS_cmt
>     X6 <- Micro_Data$veteran
>     X7 <- Micro_Data$Bases.std
>     X8 <- Micro_Data$LES.std
>     X9 <- Micro_Data$dovish_riders
>
>     # Macro-level predictors
>     Z1 <- Macro_Data$War
>     Z2 <- Macro_Data$GOP_Majority
>     Z3 <- Dem.prez
>
>     congress <- Micro_Data$congress - 91 ### To make sure "congress" goes from 1 to M.
>
>     N <- length(Y)
>     J <- length(Z1)
>     n.beta <- 9
>     n.gamma <- 4
>
>     data <- list("N", "J", "n.beta","n.gamma", "Y",
```

```
+                 "X1", "X2", "X3", "X4", "X5", "X6", "X7", "X8", "X9", "congress", "Z1","Z2","Z3")
>
>     # setup initial values
>     init1 <- list(beta=rnorm(n.beta), sigma.y=runif(1),
+                   gamma=rnorm(n.gamma),
+                   alpha=rnorm(J), sigma.alpha=runif(1))
>     init2 <- list(beta=rnorm(n.beta), sigma.y=runif(1),
+                   gamma=rnorm(n.gamma),
+                   alpha=rnorm(J), sigma.alpha=runif(1))
>     init3 <- list(beta=rnorm(n.beta), sigma.y=runif(1),
+                   gamma=rnorm(n.gamma),
+                   alpha=rnorm(J), sigma.alpha=runif(1))
>
>     inits <- list(init1, init2, init3)
>
>     # setup parameters
>     parameters = c("beta", "sigma.y", "gamma", "alpha", "sigma.alpha")
>
>     # Test bugs.
>     Model.fit = bugs(data, inits, parameters, "model1.bug",
+                      working.directory=getwd(), bugs.directory=bugs_directory,
+                      n.chains=3, n.thin=10, n.burnin=2000, n.iter=12000)
>
>     Rhats <- Model.fit$summary[,8]
>     range(Rhats) # Rhats should be 1 or very close to 1.
[1] 1.000515 1.004542
>
>     #. Analyze the draws from the Posterior distribution
```

```
>
>     chain1 <- read.coda(output.file="coda1.txt", index.file="codaIndex.txt", quiet=T)
>     chain2 <- read.coda(output.file="coda2.txt", index.file="codaIndex.txt", quiet=T)
>     chain3 <- read.coda(output.file="coda3.txt", index.file="codaIndex.txt", quiet=T)
>
>     MCMC <- rbind(chain1, chain2, chain3)
>     iter <- nrow(MCMC)
>
>     alpha.finder  <- seq(from=1, to=T, by=1)
>     beta.finder   <- seq(from=(T+1), to=(T+n.beta), by=1)
>     gamma.finder  <- seq(from=(T+n.beta+2), to=(T+n.beta+1+n.gamma), by=1)
>     sigma.y.finder <- T+n.beta+1+n.gamma + 2
>
>     alpha.mc  <- MCMC[,alpha.finder]
>     beta.mc   <- MCMC[,beta.finder]
>     gamma.mc  <- MCMC[,gamma.finder]
>     sigma.y.mc<- MCMC[,sigma.y.finder]
>
>     beta.label  <- c("Hawkiness",
+                      "Majority Party",
+                      "Seniority",
+                      "Foreign Affairs",
+                      "Armed Services",
+                      "Veteran",
+                      "No of Military Bases",
+                      "Legislative Effectiveness",
+                      "Dovish Amendments")
>     gamma.label <- c("Intercept",
```

```
+                       "Major War",
+                       "GOP Majority",
+                       "Democratic Prez")
>
>
>     # Compute Credible Intervals
>
>     beta.CI  <- round(apply(beta.mc,  2, FUN=quantile, probs=c(0.025, 0.975, 0.5, 0.05, 0.95, 0.005, 0.995)), 3)
>     gamma.CI  <- round(apply(gamma.mc,  2, FUN=quantile, probs=c(0.025, 0.975, 0.5, 0.05, 0.95, 0.005, 0.995)), 3)
>     alpha.CI  <- round(apply(alpha.mc,  2, FUN=quantile, probs=c(0.025, 0.975, 0.5, 0.05, 0.95, 0.005, 0.995)), 3)
>
>     beta <- round(rbind(apply(beta.mc,  2, FUN=mean), apply(beta.mc,  2, FUN=sd),    beta.CI),3)
>     rownames(beta)[c(1,2)] <- c("mean", "sd")
>     beta.stats <- t(beta)
>
>     gamma <- round(rbind(apply(gamma.mc,  2, FUN=mean), apply(gamma.mc,  2, FUN=sd),    gamma.CI),3)
>     rownames(gamma)[c(1,2)] <- c("mean", "sd")
>     gamma.stats <- t(gamma)
>
>     # Summary Outputs for Table
>     micro <- bayes.easy.write(id="senate_m1_micro_result", stats=beta.stats, label=beta.label)
>     macro <- bayes.easy.write(id="senate_m1_macro_result", stats=gamma.stats, label=gamma.label)
>     print(rbind(micro, macro))
      Variable                  Coefficient
 [1,] "Hawkiness"               "1.062"
 [2,] ""                        "{0.649}"
 [3,] "Majority Party"          "1.102"
 [4,] ""                        "{1.636}"
```

```
 [5,] "Seniority"                "1.479*"
 [6,] ""                         "{0.834}"
 [7,] "Foreign Affairs"          "1.435"
 [8,] ""                         "{2.009}"
 [9,] "Armed Services"           "4.26***"
[10,] ""                         "{1.534}"
[11,] "Veteran"                  "-2.295"
[12,] ""                         "{1.69}"
[13,] "No of Military Bases"     "-0.955"
[14,] ""                         "{0.815}"
[15,] "Legislative Effectiveness" "0.51"
[16,] ""                         "{0.912}"
[17,] "Dovish Amendments"        "-2.536*"
[18,] ""                         "{1.462}"
[19,] "Intercept"                "3.873"
[20,] ""                         "{3.287}"
[21,] "Major War"                "1.428"
[22,] ""                         "{3.167}"
[23,] "GOP Majority"             "1.945"
[24,] ""                         "{3.164}"
[25,] "Democratic Prez"          "1.628"
[26,] ""                         "{3.088}"
>     print(Model.fit$DIC)
[1] 7825.49
>     print(N)
[1] 860
>
>
```

```
>     ###################################################
>     ### Forth column of Table 1
>     ###################################################
>
>     Dovish_Data <- subset(Micro_Data, dovish_riders==1)
>
>     Y  <- Dovish_Data$Yea_share
>
>     # Micro-level predictors
>     X1 <- Dovish_Data$FP_mean
>     X2 <- Dovish_Data$majority_member
>     X3 <- Dovish_Data$Seniority_Year.std
>     X4 <- Dovish_Data$FR_cmt
>     X5 <- Dovish_Data$AS_cmt
>     X6 <- Dovish_Data$veteran
>     X7 <- Dovish_Data$Bases.std
>     X8 <- Dovish_Data$LES.std
>
>     # Macro-level predictors
>     Z1 <- Macro_Data$War
>     Z2 <- Macro_Data$GOP_Majority
>     Z3 <- Dem.prez
>
>     congress <- Dovish_Data$congress - 91
>
>     N <- length(Y)
>     J <- length(Z1)
>     n.beta <- 8
```

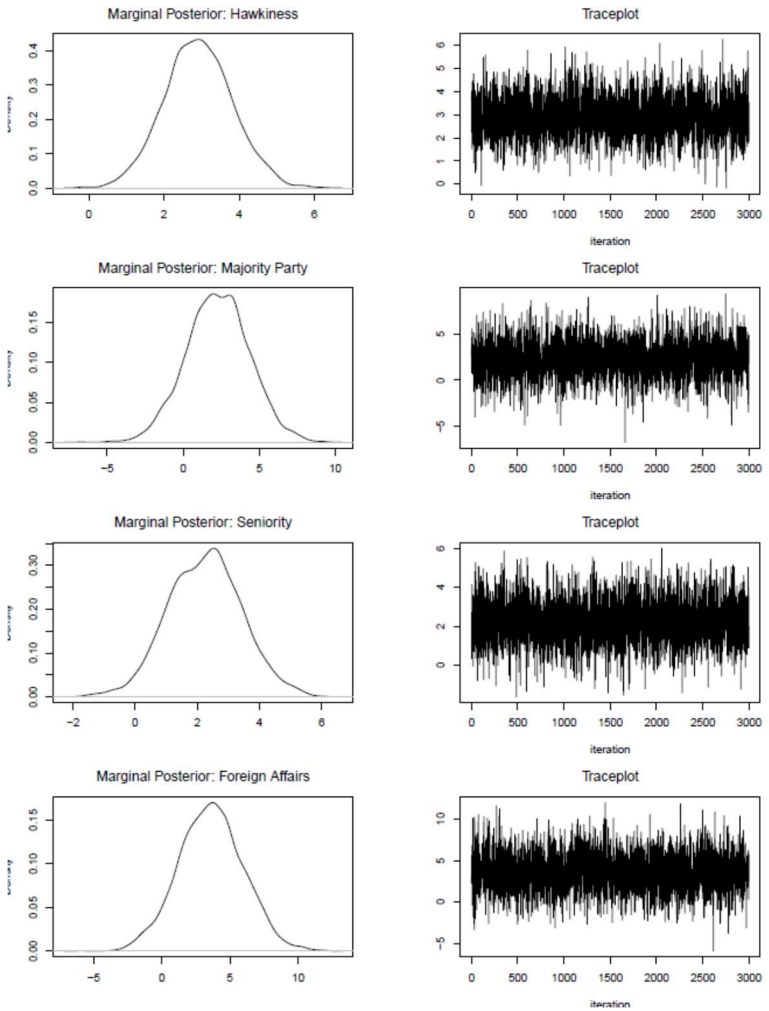
```
>     n.gamma <- 4
>
>     data <- list("N", "J", "n.beta","n.gamma", "Y",
+                  "X1", "X2", "X3", "X4", "X5", "X6", "X7", "X8", "congress", "Z1","Z2","Z3")
>
>     # setup initial values
>     init1 <- list(beta=rnorm(n.beta), sigma.y=runif(1),
+                   gamma=rnorm(n.gamma),
+                   alpha=rnorm(J), sigma.alpha=runif(1))
>     init2 <- list(beta=rnorm(n.beta), sigma.y=runif(1),
+                   gamma=rnorm(n.gamma),
+                   alpha=rnorm(J), sigma.alpha=runif(1))
>     init3 <- list(beta=rnorm(n.beta), sigma.y=runif(1),
+                   gamma=rnorm(n.gamma),
+                   alpha=rnorm(J), sigma.alpha=runif(1))
>
>     inits <- list(init1, init2, init3)
>
>     # setup parameters
>     parameters = c("beta", "sigma.y", "gamma", "alpha", "sigma.alpha")
>
>     # Test bugs.
>     Model.fit = bugs(data, inits, parameters, "model2.bug",
+                      working.directory=getwd(), bugs.directory=bugs_directory,
+                      n.chains=3, n.thin=10, n.burnin=2000, n.iter=12000)
>
>     Rhats <- Model.fit$summary[,8]
>     range(Rhats)
```

```
[1] 1.000522 1.002192
>
>     # Analyze the draws from the Posterior distribution
>
>     chain1 <- read.coda(output.file="coda1.txt", index.file="codaIndex.txt", quiet=T)
>     chain2 <- read.coda(output.file="coda2.txt", index.file="codaIndex.txt", quiet=T)
>     chain3 <- read.coda(output.file="coda3.txt", index.file="codaIndex.txt", quiet=T)
>
>     MCMC <- rbind(chain1, chain2, chain3)
>     iter <- nrow(MCMC)
>
>     alpha.finder  <- seq(from=1, to=T, by=1)
>     beta.finder   <- seq(from=(T+1), to=(T+n.beta), by=1)
>     gamma.finder  <- seq(from=(T+n.beta+2), to=(T+n.beta+1+n.gamma), by=1)
>     sigma.y.finder <- T+n.beta+1+n.gamma + 2
>
>     alpha.mc  <- MCMC[,alpha.finder]
>     beta.mc   <- MCMC[,beta.finder]
>     gamma.mc  <- MCMC[,gamma.finder]
>     sigma.y.mc<- MCMC[,sigma.y.finder]
>
>     alpha.label <- paste(92:114, "th Congress", sep="")
>     beta.label  <- c("Hawkiness",
+                      "Majority Party",
+                      "Seniority",
+                      "Foreign Affairs",
+                      "Armed Services",
+                      "Veteran",
```

```
+                     "No of Military Bases",
+                     "Legislative Effectiveness")
>     gamma.label <- c("Intercept",
+                     "Major War",
+                     "GOP Majority",
+                     "Democratic Prez")
>
>
>     # Credible Intervals
>
>     beta.CI  <- round(apply(beta.mc,  2, FUN=quantile, probs=c(0.025, 0.975, 0.5, 0.05, 0.95, 0.005, 0.995)), 3)
>     gamma.CI  <- round(apply(gamma.mc,  2, FUN=quantile, probs=c(0.025, 0.975, 0.5, 0.05, 0.95, 0.005, 0.995)), 3)
>     alpha.CI  <- round(apply(alpha.mc,  2, FUN=quantile, probs=c(0.025, 0.975, 0.5, 0.05, 0.95, 0.005, 0.995)), 3)
>
>     beta <- round(rbind(apply(beta.mc,  2, FUN=mean), apply(beta.mc,  2, FUN=sd),    beta.CI),3)
>     rownames(beta)[c(1,2)] <- c("mean", "sd")
>     beta.stats <- t(beta)
>
>     gamma <- round(rbind(apply(gamma.mc,  2, FUN=mean), apply(gamma.mc,  2, FUN=sd),    gamma.CI),3)
>     rownames(gamma)[c(1,2)] <- c("mean", "sd")
>     gamma.stats <- t(gamma)
>
>     # Summary Outputs for Table
>     micro <- bayes.easy.write(id="senate_m2_micro_result", stats=beta.stats, label=beta.label)
>     macro <- bayes.easy.write(id="senate_m2_macro_result", stats=gamma.stats, label=gamma.label)
>     print(rbind(micro, macro))
      Variable                   Coefficient
 [1,] "Hawkiness"                "2.919***"
```

```
 [2,] ""                          "{0.907}"
 [3,] "Majority Party"            "2.269"
 [4,] ""                          "{2.08}"
 [5,] "Seniority"                 "2.251*"
 [6,] ""                          "{1.176}"
 [7,] "Foreign Affairs"           "3.613"
 [8,] ""                          "{2.366}"
 [9,] "Armed Services"            "2.645"
[10,] ""                          "{1.935}"
[11,] "Veteran"                   "-2.505"
[12,] ""                          "{2.071}"
[13,] "No of Military Bases"      "0.338"
[14,] ""                          "{1.184}"
[15,] "Legislative Effectiveness" "0.855"
[16,] ""                          "{1.24}"
[17,] "Intercept"                 "4.009"
[18,] ""                          "{3.309}"
[19,] "Major War"                 "1.591"
[20,] ""                          "{3.184}"
[21,] "GOP Majority"              "2"
[22,] ""                          "{3.206}"
[23,] "Democratic Prez"           "1.678"
[24,] ""                          "{3.039}"
>     print(Model.fit$DIC)
[1] 3716.02
>     print(N)
[1] 405
```

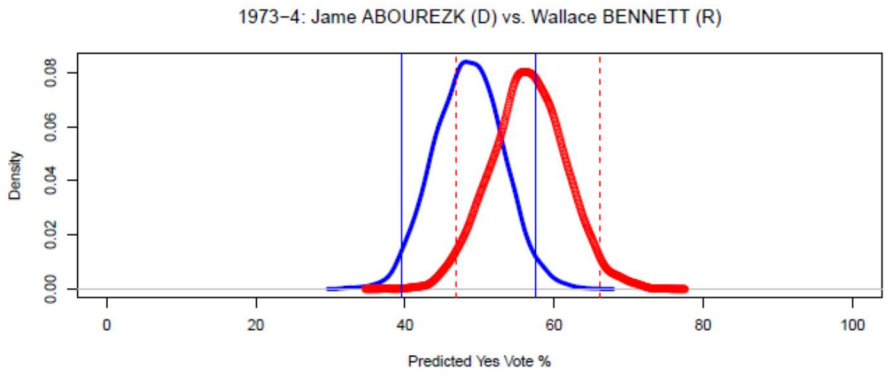
```
    ##########################################################
    # To produce Figure A3.
    # This utilizes the results of Forth column of  Table 1
    ##########################################################
>     pdf("Figure_A3.pdf", height=10, width=8)
>     par(mfrow=c(4,2), mar=c(4, 5, 3, 3)) # 'mar -> c(bottom, left, top, right)'
>     for(i in 1:length(beta.finder)){
+     plot(density(beta.mc[,i]), main="", xlab="")
+     title(main=paste("Marginal Posterior:", beta.label[i]), font.main=1)
+     plot(beta.mc[,i], type="l", xlab="iteration", ylab="")
+     title(main="Traceplot", font.main=1)
+     }
>     dev.off()
null device
```

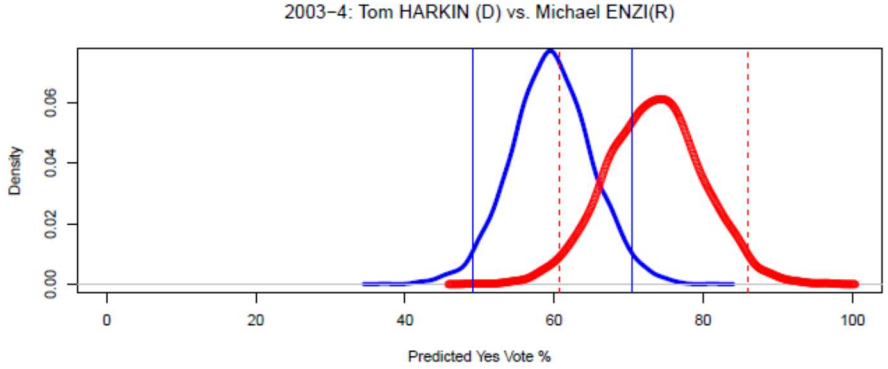


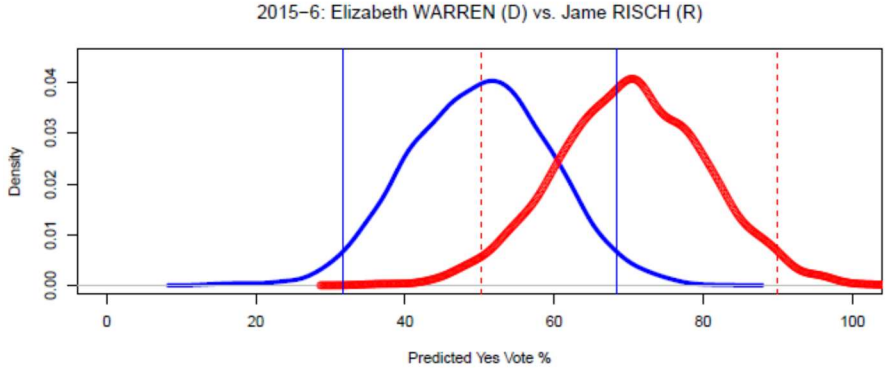

```
>
>     ################################################
>     # To produce the Second Column of Figure 3
>     ################################################
>     top1_scores <- read.csv("extreme_senators.csv", header=T)
>
>     pred.Y1 <- mu.y1 <- matrix(NA, nrow=iter, ncol=J)
>     for (t in 1:J){
```

```
+         mu.y1[,t] <- alpha.mc[,t] +
+                     beta.mc[,1]*top1_scores[t,2] +
+                     beta.mc[,2]*0 +
+                     beta.mc[,3]*mean(X3) +
+                     beta.mc[,4]*0 +
+                     beta.mc[,5]*0 +
+                     beta.mc[,6]*0 +
+                     beta.mc[,7]*mean(X7) +
+                     beta.mc[,8]*mean(X8)
+         tau.y <- sqrt(sigma.y.mc)
+         pred.Y1[,t] <- rnorm(mu.y1[,t], sigma.y.mc)
+         }
>
>     pred.Y2 <- mu.y2 <- matrix(NA, nrow=iter, ncol=J)
>     for (t in 1:J){
+         mu.y2[,t] <- alpha.mc[,t] +
+                     beta.mc[,1]*top1_scores[t,4] +
+                     beta.mc[,2]*0 +
+                     beta.mc[,3]*mean(X3) +
+                     beta.mc[,4]*0 +
+                     beta.mc[,5]*0 +
+                     beta.mc[,6]*0 +
+                     beta.mc[,7]*mean(X7) +
+                     beta.mc[,8]*mean(X8)
+         tau.y <- sqrt(sigma.y.mc)
+         pred.Y2[,t] <- rnorm(mu.y2[,t], sigma.y.mc)
+         }
>
```

```
>     mu.y1.CI  <- round(apply(mu.y1,  2, FUN=quantile, probs=c(0.025, 0.975, 0.5, 0.05, 0.95, 0.005, 0.995)), 3)
>     mu.y2.CI  <- round(apply(mu.y2,  2, FUN=quantile, probs=c(0.025, 0.975, 0.5, 0.05, 0.95, 0.005, 0.995)), 3)
>
>     pdf("Figure_3_senate.pdf", height=9, width=7)
>     par(mfcol=c(3,1), mar=c(5, 5, 4, 3)) # 'mar -> c(bottom, left, top, right)'
>     plot(density(mu.y1[,2]), lwd=3, main="", xlab="Predicted Yes Vote %", xlim=c(0,100),col='blue')
>     points(density(mu.y2[,2]), lty=1, col='red')
>     abline(v=c(mu.y1.CI[1,2], mu.y1.CI[2,2]), lty=1, col='blue')
>     abline(v=c(mu.y2.CI[1,2], mu.y2.CI[2,2]), lty=2, col='red')
>     title(main="1973-4: Jame ABOUREZK (D) vs. Wallace BENNETT (R)", font.main=1)
>
>     plot(density(mu.y1[,17]), lwd=3, main="", xlab="Predicted Yes Vote %", xlim=c(0,100), col='blue', ylim=c(0,
0.075))
>     points(density(mu.y2[,17]), col='red')
>     abline(v=c(mu.y1.CI[1,17], mu.y1.CI[2,17]), lty=1, col='blue')
>     abline(v=c(mu.y2.CI[1,17], mu.y2.CI[2,17]), lty=2, col='red')
>     title(main="2003-4: Tom HARKIN (D) vs. Michael ENZI(R)", font.main=1)
>
>     plot(density(mu.y1[,23]), lwd=3, main="", xlab="Predicted Yes Vote %", xlim=c(0,100) ,col='blue', ylim=c(0,
0.045))
>     points(density(mu.y2[,23]),col='red')
>     abline(v=c(mu.y1.CI[1,23], mu.y1.CI[2,23]), lty=1, col='blue')
>     abline(v=c(mu.y2.CI[1,23], mu.y2.CI[2,23]), lty=2, col='red')
>     title(main="2015-6: Elizabeth WARREN (D) vs. Jame RISCH (R)", font.main=1)
>     dev.off()
null device
```


1973−4: Jame ABOUREZK (D) vs. Wallace BENNETT (R)
Density
0.00
0.02
0.04
0.06
0.08
0
20
40
60
80
100
Predicted Yes Vote %


2003−4: Tom HARKIN (D) vs. Michael ENZI(R)
Density
0.00
0.02
0.04
0.06
0
20
40
60
80
100
Predicted Yes Vote %


2015−6: Elizabeth WARREN (D) vs. Jame RISCH (R)
Density
0.00
0.01
0.02
0.03
0.04
0
20
40
60
80
100
Predicted Yes Vote %

>

```
    ##########################################################################################################
    ##########################################################################################################
    ############ Replicate Table A2
    ##########################################################################################################
    ##########################################################################################################



    ########################################
    ### First column of Table A2
    ########################################
>
>     # Micro-level data
>     Micro_Data <- read.csv("House_Micro_Data.csv", header=T)
>
>     # Congress-level data
>     Macro_Data <- read.csv("Macro_Data.csv", header=T)
>     Dem.prez <- ifelse(Macro_Data$President_party==100, 1, 0)
>
>     year.odd <- seq(from=1971, to=2015, by=2)
>     T <- length(year.odd)
>
>     # Standardize the continuous variables
>     Micro_Data$seniority_term.std  <- as.vector((Micro_Data$seniority_term - mean(Micro_Data$seniority_term,
na.rm=T))/(sd(Micro_Data$seniority_term, na.rm=T)))
>     Micro_Data$Bases.std <- as.vector((Micro_Data$Bases - mean(Micro_Data$Bases, na.rm=T))/(sd(Micro_Data$Bases,
na.rm=T)))
>     Micro_Data$LES.std <- as.vector((Micro_Data$effectiveness - mean(Micro_Data$effectiveness,
na.rm=T))/(sd(Micro_Data$effectiveness, na.rm=T)))
```

```
>     Macro_Data$GOP_Majority <- ifelse(Macro_Data$House_Majority==200, 1, 0)
>     GOP_share <- ifelse(Macro_Data$GOP_Majority==1,  Macro_Data$House_Majority_Seats/Macro_Data$House_Total_Seats,
(Macro_Data$House_Total_Seats - Macro_Data$House_Majority_Seats)/Macro_Data$House_Total_Seats)
>     GOP_share.std  <- as.vector((GOP_share - mean(GOP_share, na.rm=T))/(sd(GOP_share, na.rm=T)))


>     Y  <- Micro_Data$Yea_share
>
>     X1 <- Micro_Data$FP_mean
>     X2 <- Micro_Data$majority_member
>     X3 <- Micro_Data$seniority_term.std
>     X4 <- Micro_Data$FR_cmt
>     X5 <- Micro_Data$AS_cmt
>     X6 <- Micro_Data$veteran
>     X7 <- Micro_Data$Bases.std
>     X8 <- Micro_Data$LES.std
>     X9 <- Micro_Data$dovish_riders
>
>     Z1 <- Macro_Data$Cold_War
>     Z2 <- GOP_share.std
>     Z3 <- Dem.prez
>
>     congress <- Micro_Data$congress - 91
>
>     N <- length(Y)
>     J <- length(Z1)
>     n.beta <- 10
>     n.gamma <- 4
>
```

```
>     data <- list("N", "J", "n.beta","n.gamma", "Y",
+                  "X1", "X2", "X3", "X4", "X5", "X6", "X7", "X8", "X9", "congress", "Z1","Z2","Z3")
>
>     # setup initial values
>     init1 <- list(beta=rnorm(n.beta), sigma.y=runif(1),
+                   gamma=rnorm(n.gamma),
+                   alpha=rnorm(J), sigma.alpha=runif(1))
>     init2 <- list(beta=rnorm(n.beta), sigma.y=runif(1),
+                   gamma=rnorm(n.gamma),
+                   alpha=rnorm(J), sigma.alpha=runif(1))
>     init3 <- list(beta=rnorm(n.beta), sigma.y=runif(1),
+                   gamma=rnorm(n.gamma),
+                   alpha=rnorm(J), sigma.alpha=runif(1))
>
>     inits <- list(init1, init2, init3)
>
>     # setup parameters
>     parameters = c("beta", "sigma.y", "gamma", "alpha", "sigma.alpha")
>
>     # Test bugs.
>     Model.fit = bugs(data, inits, parameters, "model1_TA2.bug",
+                          working.directory=getwd(), bugs.directory=bugs_directory,
+                          n.chains=3, n.thin=10, n.burnin=2000, n.iter=12000)
>
>     Rhats <- Model.fit$summary[,8]
>     range(Rhats)
[1] 1.000504 1.002503
>
```

```
>     # Analyze the draws from the Posterior distribution
>
>     chain1 <- read.coda(output.file="coda1.txt", index.file="codaIndex.txt", quiet=T)
>     chain2 <- read.coda(output.file="coda2.txt", index.file="codaIndex.txt", quiet=T)
>     chain3 <- read.coda(output.file="coda3.txt", index.file="codaIndex.txt", quiet=T)
>
>     MCMC <- rbind(chain1, chain2, chain3)
>     iter <- nrow(MCMC)
>
>     alpha.finder  <- seq(from=1, to=T, by=1)
>     beta.finder   <- seq(from=(T+1), to=(T+n.beta), by=1)
>     gamma.finder  <- seq(from=(T+n.beta+2), to=(T+n.beta+1+n.gamma), by=1)
>     sigma.y.finder <- T+n.beta+1+n.gamma + 2
>
>     alpha.mc  <- MCMC[,alpha.finder]
>     beta.mc   <- MCMC[,beta.finder]
>     gamma.mc  <- MCMC[,gamma.finder]
>     sigma.y.mc<- MCMC[,sigma.y.finder]
>
>     beta.label  <- c("Hawkiness",
+                      "Majority Party",
+                      "Seniority",
+                      "Foreign Affairs",
+                      "Armed Services",
+                      "Veteran",
+                      "No of Military Bases",
+                      "Legislative Effectiveness",
+                      "Dovish Amendments",
```

```
+                     "Armed Services X Veteran")
>     gamma.label <- c("Intercept",
+                     "Cold War",
+                     "GOP seatshare",
+                     "Democratic Prez")
>
>
>     ### Credible Intervals
>
>     beta.CI  <- round(apply(beta.mc,  2, FUN=quantile, probs=c(0.025, 0.975, 0.5, 0.05, 0.95, 0.005, 0.995)), 3)
>     gamma.CI  <- round(apply(gamma.mc,  2, FUN=quantile, probs=c(0.025, 0.975, 0.5, 0.05, 0.95, 0.005, 0.995)), 3)
>     alpha.CI  <- round(apply(alpha.mc,  2, FUN=quantile, probs=c(0.025, 0.975, 0.5, 0.05, 0.95, 0.005, 0.995)), 3)
>
>     beta <- round(rbind(apply(beta.mc,  2, FUN=mean), apply(beta.mc,  2, FUN=sd),    beta.CI),3)
>     rownames(beta)[c(1,2)] <- c("mean", "sd")
>     beta.stats <- t(beta)
>
>     gamma <- round(rbind(apply(gamma.mc,  2, FUN=mean), apply(gamma.mc,  2, FUN=sd),    gamma.CI),3)
>     rownames(gamma)[c(1,2)] <- c("mean", "sd")
>     gamma.stats <- t(gamma)
>
>     # Summary Outputs for Table
>     micro <- bayes.easy.write(id="house_m1_TA2_micro_result", stats=beta.stats, label=beta.label)
>     macro <- bayes.easy.write(id="house_m1_TA2_macro_result", stats=gamma.stats, label=gamma.label)
>     print(rbind(micro, macro))
     Variable                   Coefficient
 [1,] "Hawkiness"               "1.702***"
 [2,] ""                        "{0.321}"
```

```
 [3,] "Majority Party"            "6.068***"
 [4,] ""                          "{1.492}"
 [5,] "Seniority"                 "3.411***"
 [6,] ""                          "{0.809}"
 [7,] "Foreign Affairs"           "-3.161"
 [8,] ""                          "{1.977}"
 [9,] "Armed Services"            "0.41"
[10,] ""                          "{1.679}"
[11,] "Veteran"                   "0.656"
[12,] ""                          "{1.615}"
[13,] "No of Military Bases"      "0.502"
[14,] ""                          "{0.781}"
[15,] "Legislative Effectiveness" "0.989"
[16,] ""                          "{0.804}"
[17,] "Dovish Amendments"         "-2.886**"
[18,] ""                          "{1.406}"
[19,] "Armed Services X Veteran"  "-2.66"
[20,] ""                          "{2.1}"
[21,] "Intercept"                 "5.219"
[22,] ""                          "{3.343}"
[23,] "Cold War"                  "2.188"
[24,] ""                          "{3.166}"
[25,] "GOP seatshare"             "0.456"
[26,] ""                          "{2.973}"
[27,] "Democratic Prez"           "2.321"
[28,] ""                          "{3.154}"
>     print(Model.fit$DIC)
[1] 6932.26
```

```
>     print(N)
[1] 782
>


>     ########################################
>     ### Second column of Table A2
>     ########################################
>
>     Y  <- Dovish_Data$Yea_share
>
>     X1 <- Dovish_Data$FP_mean
>     X2 <- Dovish_Data$majority_member
>     X3 <- Dovish_Data$seniority_term.std
>     X4 <- Dovish_Data$FR_cmt
>     X5 <- Dovish_Data$AS_cmt
>     X6 <- Dovish_Data$veteran
>     X7 <- Dovish_Data$Bases.std
>     X8 <- Dovish_Data$LES.std
>
>     Z1 <- Macro_Data$Cold_War
>     Z2 <- GOP_share.std
>     Z3 <- Dem.prez
>
>     congress <- Dovish_Data$congress - 91
>
>     N <- length(Y)
>     J <- length(Z1)
>     n.beta <- 9
```

```
>     n.gamma <- 4
>
>     data <- list("N", "J", "n.beta","n.gamma", "Y",
+                  "X1", "X2", "X3", "X4", "X5", "X6", "X7", "X8", "congress", "Z1","Z2","Z3")
>
>     # setup initial values
>     init1 <- list(beta=rnorm(n.beta), sigma.y=runif(1),
+                   gamma=rnorm(n.gamma),
+                   alpha=rnorm(J), sigma.alpha=runif(1))
>     init2 <- list(beta=rnorm(n.beta), sigma.y=runif(1),
+                   gamma=rnorm(n.gamma),
+                   alpha=rnorm(J), sigma.alpha=runif(1))
>     init3 <- list(beta=rnorm(n.beta), sigma.y=runif(1),
+                   gamma=rnorm(n.gamma),
+                   alpha=rnorm(J), sigma.alpha=runif(1))
>
>     inits <- list(init1, init2, init3)
>
>     # setup parameters
>     parameters = c("beta", "sigma.y", "gamma", "alpha", "sigma.alpha")
>
>     # Test bugs.
>     Model.fit = bugs(data, inits, parameters, "model2_TA2.bug",
+                      working.directory=getwd(), bugs.directory="C:\\WinBUGS14",
+                      n.chains=3, n.thin=10, n.burnin=2000, n.iter=12000)
>
>     Rhats <- Model.fit$summary[,8]
>     range(Rhats)
```

```
[1] 1.000522 1.003356
>
>     # Analyze the draws from the Posterior distribution
>
>     chain1 <- read.coda(output.file="coda1.txt", index.file="codaIndex.txt", quiet=T)
>     chain2 <- read.coda(output.file="coda2.txt", index.file="codaIndex.txt", quiet=T)
>     chain3 <- read.coda(output.file="coda3.txt", index.file="codaIndex.txt", quiet=T)
>
>     MCMC <- rbind(chain1, chain2, chain3)
>
>     iter <- nrow(MCMC)
>
>     alpha.finder  <- seq(from=1, to=T, by=1)
>     beta.finder   <- seq(from=(T+1), to=(T+n.beta), by=1)
>     gamma.finder  <- seq(from=(T+n.beta+2), to=(T+n.beta+1+n.gamma), by=1)
>     sigma.y.finder <- T+n.beta+1+n.gamma + 2
>
>     alpha.mc  <- MCMC[,alpha.finder]
>     beta.mc   <- MCMC[,beta.finder]
>     gamma.mc  <- MCMC[,gamma.finder]
>     sigma.y.mc<- MCMC[,sigma.y.finder]
>
>     alpha.label <- paste(92:114, "th Congress", sep="")
>     beta.label  <- c("Hawkiness",
+                      "Majority Party",
+                      "Seniority",
+                      "Foreign Affairs",
+                      "Armed Services",
```

```
+                     "Veteran",
+                     "No of Military Bases",
+                     "Legislative Effectiveness",
+                     "Armed Services X Veteran")
>     gamma.label <- c("Intercept",
+                     "Cold War",
+                     "GOP seatshare",
+                     "Democratic Prez")
>
>
>     ### Credible Intervals
>
>     beta.CI  <- round(apply(beta.mc,  2, FUN=quantile, probs=c(0.025, 0.975, 0.5, 0.05, 0.95, 0.005, 0.995)), 3)
>     gamma.CI  <- round(apply(gamma.mc,  2, FUN=quantile, probs=c(0.025, 0.975, 0.5, 0.05, 0.95, 0.005, 0.995)), 3)
>     alpha.CI  <- round(apply(alpha.mc,  2, FUN=quantile, probs=c(0.025, 0.975, 0.5, 0.05, 0.95, 0.005, 0.995)), 3)
>
>     beta <- round(rbind(apply(beta.mc,  2, FUN=mean), apply(beta.mc,  2, FUN=sd),    beta.CI),3)
>     rownames(beta)[c(1,2)] <- c("mean", "sd")
>     beta.stats <- t(beta)
>
>     gamma <- round(rbind(apply(gamma.mc,  2, FUN=mean), apply(gamma.mc,  2, FUN=sd),    gamma.CI),3)
>     rownames(gamma)[c(1,2)] <- c("mean", "sd")
>     gamma.stats <- t(gamma)
>
>     # Summary Outputs for Table
>     micro <- bayes.easy.write(id="house_m2_TA2_micro_result", stats=beta.stats, label=beta.label)
>     macro <- bayes.easy.write(id="house_m2_TA2_macro_result", stats=gamma.stats, label=gamma.label)
>     print(rbind(micro, macro))
```

```
      Variable                    Coefficient
 [1,] "Hawkiness"                 "1.92***"
 [2,] ""                          "{0.364}"
 [3,] "Majority Party"            "5.223***"
 [4,] ""                          "{1.691}"
 [5,] "Seniority"                 "2.907***"
 [6,] ""                          "{0.929}"
 [7,] "Foreign Affairs"           "-4.61**"
 [8,] ""                          "{2.176}"
 [9,] "Armed Services"            "0.196"
[10,] ""                          "{1.851}"
[11,] "Veteran"                   "-0.084"
[12,] ""                          "{1.778}"
[13,] "No of Military Bases"      "1.461"
[14,] ""                          "{0.93}"
[15,] "Legislative Effectiveness" "0.847"
[16,] ""                          "{0.874}"
[17,] "Armed Services X Veteran"  "-4.015*"
[18,] ""                          "{2.22}"
[19,] "Intercept"                 "5.709"
[20,] ""                          "{3.521}"
[21,] "Cold War"                  "2.563"
[22,] ""                          "{3.134}"
[23,] "GOP seatshare"             "0.284"
[24,] ""                          "{2.938}"
[25,] "Democratic Prez"           "2.522"
[26,] ""                          "{3.159}"
>     print(Model.fit$DIC)
```

```
[1] 4646.95
>     print(N)
[1] 531


>
>     ##################################################
>     ### Third column of Table A2
>     ##################################################
>
>     # Import the data
>
>     # Micro-level data
>     Micro_Data <- read.csv("Senate_Micro_Data.csv", header=T)
>
>     # Congress-level data
>     Macro_Data <- read.csv("Macro_Data.csv", header=T)
>     Dem.prez <- ifelse(Macro_Data$President_party==100, 1, 0)
>
>     year.odd <- seq(from=1971, to=2015, by=2)
>     T <- length(year.odd)
>
>     # Standardize the continuous variables
>     Micro_Data$Seniority_Year.std  <- as.vector((Micro_Data$Seniority_Year - mean(Micro_Data$Seniority_Year,
na.rm=T))/(sd(Micro_Data$Seniority_Year, na.rm=T)))
>     Micro_Data$Bases.std <- as.vector((Micro_Data$Bases - mean(Micro_Data$Bases, na.rm=T))/(sd(Micro_Data$Bases,
na.rm=T)))
>     Micro_Data$LES.std <- as.vector((Micro_Data$effectiveness - mean(Micro_Data$effectiveness,
na.rm=T))/(sd(Micro_Data$effectiveness, na.rm=T)))
>     Macro_Data$GOP_Majority <- ifelse(Macro_Data$Senate_Majority==200, 1, 0)
```

```
>     GOP_share <- ifelse(Macro_Data$GOP_Majority==1,  Macro_Data$Senate_Majority_Seats/Macro_Data$Senate_Total_Seats,
(Macro_Data$Senate_Total_Seats - Macro_Data$Senate_Majority_Seats)/Macro_Data$Senate_Total_Seats)
>     GOP_share.std  <- as.vector((GOP_share - mean(GOP_share, na.rm=T))/(sd(GOP_share, na.rm=T)))
>
>     Y  <- Micro_Data$Yea_share
>
>     X1 <- Micro_Data$FP_mean
>     X2 <- Micro_Data$majority_member
>     X3 <- Micro_Data$Seniority_Year.std
>     X4 <- Micro_Data$FR_cmt
>     X5 <- Micro_Data$AS_cmt
>     X6 <- Micro_Data$veteran
>     X7 <- Micro_Data$Bases.std
>     X8 <- Micro_Data$LES.std
>     X9 <- Micro_Data$dovish_riders
>
>     Z1 <- Macro_Data$Cold_War
>     Z2 <- GOP_share.std
>     Z3 <- Dem.prez
>
>     congress <- Micro_Data$congress - 91
>
>     N <- length(Y)
>     J <- length(Z1)
>     n.beta <- 10
>     n.gamma <- 4
>
>     data <- list("N", "J", "n.beta","n.gamma", "Y",
```

```
+                 "X1", "X2", "X3", "X4", "X5", "X6", "X7", "X8", "X9", "congress", "Z1","Z2","Z3")
>
>     # setup initial values
>     init1 <- list(beta=rnorm(n.beta), sigma.y=runif(1),
+                   gamma=rnorm(n.gamma),
+                   alpha=rnorm(J), sigma.alpha=runif(1))
>     init2 <- list(beta=rnorm(n.beta), sigma.y=runif(1),
+                   gamma=rnorm(n.gamma),
+                   alpha=rnorm(J), sigma.alpha=runif(1))
>     init3 <- list(beta=rnorm(n.beta), sigma.y=runif(1),
+                   gamma=rnorm(n.gamma),
+                   alpha=rnorm(J), sigma.alpha=runif(1))
>
>     inits <- list(init1, init2, init3)
>
>     # setup parameters
>     parameters = c("beta", "sigma.y", "gamma", "alpha", "sigma.alpha")
>
>     # Test bugs.
>     Model.fit = bugs(data, inits, parameters, "model1_TA2.bug",
+                      working.directory=getwd(), bugs.directory=bugs_directory,
+                      n.chains=3, n.thin=10, n.burnin=2000, n.iter=12000)
>
>     Rhats <- Model.fit$summary[,8]
>     range(Rhats)
[1] 1.000510 1.003307
>
>     # Analyze the draws from the Posterior distribution
```

```
>
>     chain1 <- read.coda(output.file="coda1.txt", index.file="codaIndex.txt", quiet=T)
>     chain2 <- read.coda(output.file="coda2.txt", index.file="codaIndex.txt", quiet=T)
>     chain3 <- read.coda(output.file="coda3.txt", index.file="codaIndex.txt", quiet=T)
>
>     MCMC <- rbind(chain1, chain2, chain3)
>     iter <- nrow(MCMC)
>
>     alpha.finder  <- seq(from=1, to=T, by=1)
>     beta.finder   <- seq(from=(T+1), to=(T+n.beta), by=1)
>     gamma.finder  <- seq(from=(T+n.beta+2), to=(T+n.beta+1+n.gamma), by=1)
>     sigma.y.finder <- T+n.beta+1+n.gamma + 2
>
>     alpha.mc  <- MCMC[,alpha.finder]
>     beta.mc   <- MCMC[,beta.finder]
>     gamma.mc  <- MCMC[,gamma.finder]
>     sigma.y.mc<- MCMC[,sigma.y.finder]
>
>     alpha.label <- paste(92:114, "th Congress", sep="")
>     beta.label  <- c("Hawkiness",
+                      "Majority Party",
+                      "Seniority",
+                      "Foreign Affairs",
+                      "Armed Services",
+                      "Veteran",
+                      "No of Military Bases",
+                      "Legislative Effectiveness",
+                      "Dovish Amendments",
```

```
+                     "Armed Services X Veteran")
>     gamma.label <- c("Intercept",
+                     "Cold War",
+                     "GOP seatshare",
+                     "Democratic Prez")
>
>
>     # Credible Intervals
>
>     beta.CI  <- round(apply(beta.mc,  2, FUN=quantile, probs=c(0.025, 0.975, 0.5, 0.05, 0.95, 0.005, 0.995)), 3)
>     gamma.CI  <- round(apply(gamma.mc,  2, FUN=quantile, probs=c(0.025, 0.975, 0.5, 0.05, 0.95, 0.005, 0.995)), 3)
>     alpha.CI  <- round(apply(alpha.mc,  2, FUN=quantile, probs=c(0.025, 0.975, 0.5, 0.05, 0.95, 0.005, 0.995)), 3)
>
>     beta <- round(rbind(apply(beta.mc,  2, FUN=mean), apply(beta.mc,  2, FUN=sd),    beta.CI),3)
>     rownames(beta)[c(1,2)] <- c("mean", "sd")
>     beta.stats <- t(beta)
>
>     gamma <- round(rbind(apply(gamma.mc,  2, FUN=mean), apply(gamma.mc,  2, FUN=sd),    gamma.CI),3)
>     rownames(gamma)[c(1,2)] <- c("mean", "sd")
>     gamma.stats <- t(gamma)
>
>     # Summary Outputs for Table
>     micro <- bayes.easy.write(id="senate_m1_TA2_micro_result", stats=beta.stats, label=beta.label)
>     macro <- bayes.easy.write(id="senate_m1_TA2_macro_result", stats=gamma.stats, label=gamma.label)
>     print(rbind(micro, macro))
      Variable                  Coefficient
 [1,] "Hawkiness"               "1.076"
 [2,] ""                        "{0.678}"
```

```
 [3,] "Majority Party"            "1.041"
 [4,] ""                          "{1.625}"
 [5,] "Seniority"                 "1.468*"
 [6,] ""                          "{0.825}"
 [7,] "Foreign Affairs"           "1.43"
 [8,] ""                          "{1.96}"
 [9,] "Armed Services"            "4.367**"
[10,] ""                          "{1.914}"
[11,] "Veteran"                   "-2.198"
[12,] ""                          "{1.735}"
[13,] "No of Military Bases"      "-0.898"
[14,] ""                          "{0.837}"
[15,] "Legislative Effectiveness" "0.531"
[16,] ""                          "{0.882}"
[17,] "Dovish Amendments"         "-2.534*"
[18,] ""                          "{1.437}"
[19,] "Armed Services X Veteran"  "-0.249"
[20,] ""                          "{2.157}"
[21,] "Intercept"                 "3.865"
[22,] ""                          "{3.246}"
[23,] "Cold War"                  "1.612"
[24,] ""                          "{3.102}"
[25,] "GOP seatshare"             "0.381"
[26,] ""                          "{3.129}"
[27,] "Democratic Prez"           "1.796"
[28,] ""                          "{3.119}"
>     print(Model.fit$DIC)
[1] 7825.56
```

```
>     print(N)
[1] 860
>
>
>
>     ########################################
>     ### Forth column of Table A2
>     ########################################
>
>     Y  <- Dovish_Data$Yea_share
>
>     X1 <- Dovish_Data$FP_mean
>     X2 <- Dovish_Data$majority_member
>     X3 <- Dovish_Data$Seniority_Year.std
>     X4 <- Dovish_Data$FR_cmt
>     X5 <- Dovish_Data$AS_cmt
>     X6 <- Dovish_Data$veteran
>     X7 <- Dovish_Data$Bases.std
>     X8 <- Dovish_Data$LES.std
>
>     Z1 <- Macro_Data$Cold_War
>     Z2 <- GOP_share.std
>     Z3 <- Dem.prez
>
>     congress <- Dovish_Data$congress - 91
>
>     N <- length(Y)
>     J <- length(Z1)
```

```
>     n.beta <- 9
>     n.gamma <- 4
>
>     data <- list("N", "J", "n.beta","n.gamma", "Y",
+                  "X1", "X2", "X3", "X4", "X5", "X6", "X7", "X8", "congress", "Z1","Z2","Z3")
>
>     # setup initial values
>     init1 <- list(beta=rnorm(n.beta), sigma.y=runif(1),
+                   gamma=rnorm(n.gamma),
+                   alpha=rnorm(J), sigma.alpha=runif(1))
>     init2 <- list(beta=rnorm(n.beta), sigma.y=runif(1),
+                   gamma=rnorm(n.gamma),
+                   alpha=rnorm(J), sigma.alpha=runif(1))
>     init3 <- list(beta=rnorm(n.beta), sigma.y=runif(1),
+                   gamma=rnorm(n.gamma),
+                   alpha=rnorm(J), sigma.alpha=runif(1))
>
>     inits <- list(init1, init2, init3)
>
>     # setup parameters
>     parameters = c("beta", "sigma.y", "gamma", "alpha", "sigma.alpha")
>
>     # Test bugs.
>     Model.fit = bugs(data, inits, parameters, "model2_TA2.bug",
+                      working.directory=getwd(), bugs.directory="C:\\WinBUGS14",
+                      n.chains=3, n.thin=10, n.burnin=2000, n.iter=12000)
>
>     Rhats <- Model.fit$summary[,8]
```

```
>     range(Rhats)
[1] 1.000534 1.003829
>
>     # Analyze the draws from the Posterior distribution
>
>     chain1 <- read.coda(output.file="coda1.txt", index.file="codaIndex.txt", quiet=T)
>     chain2 <- read.coda(output.file="coda2.txt", index.file="codaIndex.txt", quiet=T)
>     chain3 <- read.coda(output.file="coda3.txt", index.file="codaIndex.txt", quiet=T)
>
>     MCMC <- rbind(chain1, chain2, chain3)
>
>     iter <- nrow(MCMC)
>
>     alpha.finder  <- seq(from=1, to=T, by=1)
>     beta.finder   <- seq(from=(T+1), to=(T+n.beta), by=1)
>     gamma.finder  <- seq(from=(T+n.beta+2), to=(T+n.beta+1+n.gamma), by=1)
>     sigma.y.finder <- T+n.beta+1+n.gamma + 2
>
>     alpha.mc  <- MCMC[,alpha.finder]
>     beta.mc   <- MCMC[,beta.finder]
>     gamma.mc  <- MCMC[,gamma.finder]
>     sigma.y.mc<- MCMC[,sigma.y.finder]
>
>     alpha.label <- paste(92:114, "th Congress", sep="")
>     beta.label  <- c("Hawkiness",
+                      "Majority Party",
+                      "Seniority",
+                      "Foreign Affairs",
```

```
+                     "Armed Services",
+                     "Veteran",
+                     "No of Military Bases",
+                     "Legislative Effectiveness",
+                     "Armed Services X Veteran")
>     gamma.label <- c("Intercept",
+                     "Cold War",
+                     "GOP seatshare",
+                     "Democratic Prez")
>
>
>     ### Credible Intervals
>
>     beta.CI  <- round(apply(beta.mc,  2, FUN=quantile, probs=c(0.025, 0.975, 0.5, 0.05, 0.95, 0.005, 0.995)), 3)
>     gamma.CI  <- round(apply(gamma.mc,  2, FUN=quantile, probs=c(0.025, 0.975, 0.5, 0.05, 0.95, 0.005, 0.995)), 3)
>     alpha.CI  <- round(apply(alpha.mc,  2, FUN=quantile, probs=c(0.025, 0.975, 0.5, 0.05, 0.95, 0.005, 0.995)), 3)
>
>     beta <- round(rbind(apply(beta.mc,  2, FUN=mean), apply(beta.mc,  2, FUN=sd),    beta.CI),3)
>     rownames(beta)[c(1,2)] <- c("mean", "sd")
>     beta.stats <- t(beta)
>
>     gamma <- round(rbind(apply(gamma.mc,  2, FUN=mean), apply(gamma.mc,  2, FUN=sd),    gamma.CI),3)
>     rownames(gamma)[c(1,2)] <- c("mean", "sd")
>     gamma.stats <- t(gamma)
>
>     # Summary Outputs for Table
>     micro <- bayes.easy.write(id="senate_m2_TA2_micro_result", stats=beta.stats, label=beta.label)
>     macro <- bayes.easy.write(id="senate_m2_TA2_macro_result", stats=gamma.stats, label=gamma.label)
```

```
>     print(rbind(micro, macro))
      Variable                     Coefficient
 [1,] "Hawkiness"                  "2.904***"
 [2,] ""                           "{0.934}"
 [3,] "Majority Party"             "2.216"
 [4,] ""                           "{2.084}"
 [5,] "Seniority"                  "2.273*"
 [6,] ""                           "{1.183}"
 [7,] "Foreign Affairs"            "3.634"
 [8,] ""                           "{2.34}"
 [9,] "Armed Services"             "2.144"
[10,] ""                           "{2.195}"
[11,] "Veteran"                    "-2.859"
[12,] ""                           "{2.132}"
[13,] "No of Military Bases"       "0.296"
[14,] ""                           "{1.217}"
[15,] "Legislative Effectiveness"  "0.863"
[16,] ""                           "{1.289}"
[17,] "Armed Services X Veteran"   "1.104"
[18,] ""                           "{2.389}"
[19,] "Intercept"                  "4.106"
[20,] ""                           "{3.28}"
[21,] "Cold War"                   "1.775"
[22,] ""                           "{3.197}"
[23,] "GOP seatshare"              "0.652"
[24,] ""                           "{3.065}"
[25,] "Democratic Prez"            "1.675"
[26,] ""                           "{3.164}"
```

```
>     print(Model.fit$DIC)
[1] 3716.87
>     print(N)
[1] 405
>


    ##############################################################################################################
    ##############################################################################################################
    ############ Replicate Table A3
    ##############################################################################################################
    ##############################################################################################################

    ########################################
    ### First column of Table A3
    ########################################
>
>     # Micro-level data
>     Micro_Data <- read.csv("House_Micro_Data.csv", header=T)
>
>     # Congress-level data
>     Macro_Data <- read.csv("Macro_Data.csv", header=T)
>     Dem.prez <- ifelse(Macro_Data$President_party==100, 1, 0)
>
>     year.odd <- seq(from=1971, to=2015, by=2)
>     T <- length(year.odd)
>
>     # Standardize the continuous variables
```

```
>     Micro_Data$seniority_term.std  <- as.vector((Micro_Data$seniority_term - mean(Micro_Data$seniority_term,
na.rm=T))/(sd(Micro_Data$seniority_term, na.rm=T)))

>     Micro_Data$Bases.std <- as.vector((Micro_Data$Bases - mean(Micro_Data$Bases, na.rm=T))/(sd(Micro_Data$Bases,
na.rm=T)))

>     Micro_Data$LES.std <- as.vector((Micro_Data$effectiveness - mean(Micro_Data$effectiveness,
na.rm=T))/(sd(Micro_Data$effectiveness, na.rm=T)))

>     Macro_Data$GOP_Majority <- ifelse(Macro_Data$House_Majority==200, 1, 0)

>     GOP_share <- ifelse(Macro_Data$GOP_Majority==1,  Macro_Data$House_Majority_Seats/Macro_Data$House_Total_Seats,
(Macro_Data$House_Total_Seats - Macro_Data$House_Majority_Seats)/Macro_Data$House_Total_Seats)

>     GOP_share.std  <- as.vector((GOP_share - mean(GOP_share, na.rm=T))/(sd(GOP_share, na.rm=T)))

>     Y  <- Micro_Data$Yea_share

>

>     # Micro-level predictors

>     X1 <- Micro_Data$majority_member

>     X2 <- Micro_Data$seniority_term.std

>     X3 <- Micro_Data$FR_cmt

>     X4 <- Micro_Data$AS_cmt

>     X5 <- Micro_Data$veteran

>     X6 <- Micro_Data$Bases.std

>     X7 <- Micro_Data$LES.std

>     X8 <- Micro_Data$dovish_riders

>

>     # Macro-level predictors

>     Z1 <- Macro_Data$War

>     Z2 <- Macro_Data$GOP_Majority

>     Z3 <- Dem.prez

>

>     congress <- Micro_Data$congress - 91

>

>     N <- length(Y)
```

```
>     J <- length(Z1)
>     n.beta <- 8
>     n.gamma <- 4
>
>     data <- list("N", "J", "n.beta","n.gamma", "Y",
+                  "X1", "X2", "X3", "X4", "X5", "X6", "X7", "X8", "congress", "Z1","Z2","Z3")
>
>     # setup initial values
>     init1 <- list(beta=rnorm(n.beta), sigma.y=runif(1),
+                   gamma=rnorm(n.gamma),
+                   alpha=rnorm(J), sigma.alpha=runif(1))
>     init2 <- list(beta=rnorm(n.beta), sigma.y=runif(1),
+                   gamma=rnorm(n.gamma),
+                   alpha=rnorm(J), sigma.alpha=runif(1))
>     init3 <- list(beta=rnorm(n.beta), sigma.y=runif(1),
+                   gamma=rnorm(n.gamma),
+                   alpha=rnorm(J), sigma.alpha=runif(1))
>
>     inits <- list(init1, init2, init3)
>
>     # setup parameters
>     parameters = c("beta", "sigma.y", "gamma", "alpha", "sigma.alpha")
>
>     # Test bugs.
>     Model.fit = bugs(data, inits, parameters, "model_TA3.bug",
+                      working.directory=getwd(), bugs.directory=bugs_directory,
+                      n.chains=3, n.thin=10, n.burnin=2000, n.iter=12000)
>
```

```
>     Rhats <- Model.fit$summary[,8]
>     range(Rhats)
[1] 1.000499 1.004199
>
>     ### 4. Analyze the draws from the Posterior distribution #######
>
>     chain1 <- read.coda(output.file="coda1.txt", index.file="codaIndex.txt", quiet=T)
>     chain2 <- read.coda(output.file="coda2.txt", index.file="codaIndex.txt", quiet=T)
>     chain3 <- read.coda(output.file="coda3.txt", index.file="codaIndex.txt", quiet=T)
>
>     MCMC <- rbind(chain1, chain2, chain3)
>     iter <- nrow(MCMC)
>
>     alpha.finder  <- seq(from=1, to=T, by=1)
>     beta.finder   <- seq(from=(T+1), to=(T+n.beta), by=1)
>     gamma.finder  <- seq(from=(T+n.beta+2), to=(T+n.beta+1+n.gamma), by=1)
>     sigma.y.finder <- T+n.beta+1+n.gamma + 2
>
>     alpha.mc  <- MCMC[,alpha.finder]
>     beta.mc   <- MCMC[,beta.finder]
>     gamma.mc  <- MCMC[,gamma.finder]
>     sigma.y.mc<- MCMC[,sigma.y.finder]
>
>     beta.label  <- c("Majority Party",
+                      "Seniority",
+                      "Foreign Affairs",
+                      "Armed Services",
+                      "Veteran",
```

```
+                     "No of Military Bases",
+                     "Legislative Effectiveness",
+                     "Dovish Amendments")
>     gamma.label <- c("Intercept",
+                     "Major War",
+                     "GOP Majority",
+                     "Democratic Prez")
>
>
>     ### Credible Intervals
>
>     beta.CI  <- round(apply(beta.mc,  2, FUN=quantile, probs=c(0.025, 0.975, 0.5, 0.05, 0.95, 0.005, 0.995)), 3)
>     gamma.CI  <- round(apply(gamma.mc,  2, FUN=quantile, probs=c(0.025, 0.975, 0.5, 0.05, 0.95, 0.005, 0.995)), 3)
>     alpha.CI  <- round(apply(alpha.mc,  2, FUN=quantile, probs=c(0.025, 0.975, 0.5, 0.05, 0.95, 0.005, 0.995)), 3)
>
>     beta <- round(rbind(apply(beta.mc,  2, FUN=mean), apply(beta.mc,  2, FUN=sd),    beta.CI),3)
>     rownames(beta)[c(1,2)] <- c("mean", "sd")
>     beta.stats <- t(beta)
>
>     gamma <- round(rbind(apply(gamma.mc,  2, FUN=mean), apply(gamma.mc,  2, FUN=sd),    gamma.CI),3)
>     rownames(gamma)[c(1,2)] <- c("mean", "sd")
>     gamma.stats <- t(gamma)
>
>     # Summary Outputs for Table
>     micro <- bayes.easy.write(id="house_TA3_micro_result", stats=beta.stats, label=beta.label)
>     macro <- bayes.easy.write(id="house_TA3_macro_result", stats=gamma.stats, label=gamma.label)
>     print(rbind(micro, macro))
      Variable                   Coefficient
```

```
 [1,] "Majority Party"            "5.564***"
 [2,] ""                          "{1.527}"
 [3,] "Seniority"                 "2.557***"
 [4,] ""                          "{0.832}"
 [5,] "Foreign Affairs"           "-2.233"
 [6,] ""                          "{2.055}"
 [7,] "Armed Services"            "0.297"
 [8,] ""                          "{1.497}"
 [9,] "Veteran"                   "0.932"
[10,] ""                          "{1.491}"
[11,] "No of Military Bases"      "0.967"
[12,] ""                          "{0.793}"
[13,] "Legislative Effectiveness" "1.165"
[14,] ""                          "{0.831}"
[15,] "Dovish Amendments"         "-3.328**"
[16,] ""                          "{1.464}"
[17,] "Intercept"                 "5.631"
[18,] ""                          "{3.497}"
[19,] "Major War"                 "1.994"
[20,] ""                          "{3.138}"
[21,] "GOP Majority"              "2.298"
[22,] ""                          "{3.162}"
[23,] "Democratic Prez"           "2.497"
[24,] ""                          "{3.217}"
>     print(Model.fit$DIC)
[1] 6964.51
>     print(N)
[1] 782
```

```
>     ##################################################
>     ### Second column of Table A3
>     ##################################################
>
>     # Import the data
>
>     # Micro-level data
>     Micro_Data <- read.csv("Senate_Micro_Data.csv", header=T)
>
>     # Congress-level data
>     Macro_Data <- read.csv("Macro_Data.csv", header=T)
>     Dem.prez <- ifelse(Macro_Data$President_party==100, 1, 0)
>
>     year.odd <- seq(from=1971, to=2015, by=2)
>     T <- length(year.odd)
>
>     # Standardize the continuous variables
>     Micro_Data$Seniority_Year.std  <- as.vector((Micro_Data$Seniority_Year - mean(Micro_Data$Seniority_Year,
na.rm=T))/(sd(Micro_Data$Seniority_Year, na.rm=T)))
>     Micro_Data$Bases.std <- as.vector((Micro_Data$Bases - mean(Micro_Data$Bases, na.rm=T))/(sd(Micro_Data$Bases,
na.rm=T)))
>     Micro_Data$LES.std <- as.vector((Micro_Data$effectiveness - mean(Micro_Data$effectiveness,
na.rm=T))/(sd(Micro_Data$effectiveness, na.rm=T)))
>     Macro_Data$GOP_Majority <- ifelse(Macro_Data$Senate_Majority==200, 1, 0)
>     GOP_share <- ifelse(Macro_Data$GOP_Majority==1,  Macro_Data$Senate_Majority_Seats/Macro_Data$Senate_Total_Seats,
(Macro_Data$Senate_Total_Seats - Macro_Data$Senate_Majority_Seats)/Macro_Data$Senate_Total_Seats)
>     GOP_share.std  <- as.vector((GOP_share - mean(GOP_share, na.rm=T))/(sd(GOP_share, na.rm=T)))
>
>
>     Y  <- Micro_Data$Yea_share
```

```
>
>     # Micro-level predictors
>     X1 <- Micro_Data$majority_member
>     X2 <- Micro_Data$Seniority_Year.std
>     X3 <- Micro_Data$FR_cmt
>     X4 <- Micro_Data$AS_cmt
>     X5 <- Micro_Data$veteran
>     X6 <- Micro_Data$Bases.std
>     X7 <- Micro_Data$LES.std
>     X8 <- Micro_Data$dovish_riders
>
>     # Macro-level predictors
>     Z1 <- Macro_Data$War
>     Z2 <- Macro_Data$GOP_Majority
>     Z3 <- Dem.prez
>
>     congress <- Micro_Data$congress - 91
>
>     N <- length(Y)
>     J <- length(Z1)
>     n.beta <- 8
>     n.gamma <- 4
>
>     data <- list("N", "J", "n.beta","n.gamma", "Y",
+                  "X1", "X2", "X3", "X4", "X5", "X6", "X7", "X8", "congress", "Z1","Z2","Z3")
>
>     # setup initial values
>     init1 <- list(beta=rnorm(n.beta), sigma.y=runif(1),
```

```
+                   gamma=rnorm(n.gamma),
+                   alpha=rnorm(J), sigma.alpha=runif(1))
>     init2 <- list(beta=rnorm(n.beta), sigma.y=runif(1),
+                   gamma=rnorm(n.gamma),
+                   alpha=rnorm(J), sigma.alpha=runif(1))
>     init3 <- list(beta=rnorm(n.beta), sigma.y=runif(1),
+                   gamma=rnorm(n.gamma),
+                   alpha=rnorm(J), sigma.alpha=runif(1))
>
>     inits <- list(init1, init2, init3)
>
>     # setup parameters
>     parameters = c("beta", "sigma.y", "gamma", "alpha", "sigma.alpha")
>
>     # Test bugs.
>     Model.fit = bugs(data, inits, parameters, "model_TA3.bug",
+                         working.directory=getwd(), bugs.directory="C:\\WinBUGS14",
+                         n.chains=3, n.thin=10, n.burnin=2000, n.iter=12000)
>
>     Rhats <- Model.fit$summary[,8]
>     range(Rhats)
[1] 1.000526 1.003510
>
>     # Analyze the draws from the Posterior distribution
>
>     chain1 <- read.coda(output.file="coda1.txt", index.file="codaIndex.txt", quiet=T)
>     chain2 <- read.coda(output.file="coda2.txt", index.file="codaIndex.txt", quiet=T)
>     chain3 <- read.coda(output.file="coda3.txt", index.file="codaIndex.txt", quiet=T)
```

```
>
>     MCMC <- rbind(chain1, chain2, chain3)
>     iter <- nrow(MCMC)
>
>     alpha.finder  <- seq(from=1, to=T, by=1)
>     beta.finder   <- seq(from=(T+1), to=(T+n.beta), by=1)
>     gamma.finder  <- seq(from=(T+n.beta+2), to=(T+n.beta+1+n.gamma), by=1)
>     sigma.y.finder <- T+n.beta+1+n.gamma + 2
>
>     alpha.mc  <- MCMC[,alpha.finder]
>     beta.mc   <- MCMC[,beta.finder]
>     gamma.mc  <- MCMC[,gamma.finder]
>     sigma.y.mc<- MCMC[,sigma.y.finder]
>
>     alpha.label <- paste(92:114, "th Congress", sep="")
>     beta.label  <- c("Majority Party",
+                      "Seniority",
+                      "Foreign Affairs",
+                      "Armed Services",
+                      "Veteran",
+                      "No of Military Bases",
+                      "Legislative Effectiveness",
+                      "Dovish Amendments")
>     gamma.label <- c("Intercept",
+                      "Major War",
+                      "GOP Majority",
+                      "Democratic Prez")
>
```

```
>
>     ### Credible Intervals
>
>     beta.CI  <- round(apply(beta.mc,  2, FUN=quantile, probs=c(0.025, 0.975, 0.5, 0.05, 0.95, 0.005, 0.995)), 3)
>     gamma.CI  <- round(apply(gamma.mc,  2, FUN=quantile, probs=c(0.025, 0.975, 0.5, 0.05, 0.95, 0.005, 0.995)), 3)
>     alpha.CI  <- round(apply(alpha.mc,  2, FUN=quantile, probs=c(0.025, 0.975, 0.5, 0.05, 0.95, 0.005, 0.995)), 3)
>
>     beta <- round(rbind(apply(beta.mc,  2, FUN=mean), apply(beta.mc,  2, FUN=sd),    beta.CI),3)
>     rownames(beta)[c(1,2)] <- c("mean", "sd")
>     beta.stats <- t(beta)
>
>     gamma <- round(rbind(apply(gamma.mc,  2, FUN=mean), apply(gamma.mc,  2, FUN=sd),    gamma.CI),3)
>     rownames(gamma)[c(1,2)] <- c("mean", "sd")
>     gamma.stats <- t(gamma)
>
>     # Summary Outputs for Table
>     micro <- bayes.easy.write(id="senate_TA3_micro_result", stats=beta.stats, label=beta.label)
>     macro <- bayes.easy.write(id="senate_TA3_macro_result", stats=gamma.stats, label=gamma.label)
>     print(rbind(micro, macro))
     Variable                   Coefficient
 [1,] "Majority Party"          "1.428"
 [2,] ""                        "{1.57}"
 [3,] "Seniority"               "1.458*"
 [4,] ""                        "{0.834}"
 [5,] "Foreign Affairs"         "1.185"
 [6,] ""                        "{2.011}"
 [7,] "Armed Services"          "4.444***"
 [8,] ""                        "{1.52}"
```

```
 [9,] "Veteran"                   "-2.034"
[10,] ""                          "{1.616}"
[11,] "No of Military Bases"      "-0.757"
[12,] ""                          "{0.812}"
[13,] "Legislative Effectiveness" "0.364"
[14,] ""                          "{0.874}"
[15,] "Dovish Amendments"         "-2.661*"
[16,] ""                          "{1.418}"
[17,] "Intercept"                 "3.84"
[18,] ""                          "{3.22}"
[19,] "Major War"                 "1.501"
[20,] ""                          "{3.164}"
[21,] "GOP Majority"              "1.95"
[22,] ""                          "{3.14}"
[23,] "Democratic Prez"           "1.631"
[24,] ""                          "{3.067}"
>     print(Model.fit$DIC)
[1] 7826.06
>     print(N)
[1] 860
>
>
```