

Varying-Intercept Modeling of Congressional Bipartisanship

Dalia Habiby

2022-12-07

Data and Methods

The model of interest in this paper measures the strength of bipartisanship in Congress by evaluating observations about individual votes on various bills. I am utilizing the data set “Important Congressional Votes, 1991-2020” for this model, created by Jordan Tama for the purpose of investigating consequential voting patterns and political polarization. The data are collected from the Congressional Quarterly Almanac, defining “important” votes in Congress as those appearing in the CQ Almanac’s yearly “Key Votes” report, which are the most major issues upon which Congress acted. The data set has 59 variables measuring different aspects of the vote on a specific legislation, including tallies for the Democratic and Republican party votes, ratios of bipartisanship and consensus, the type of vote, and other identifying information. There are 3,445 observations in total from 1991 to 2020, however they are not all complete cases. Many variables were missing 1551 observations, including the strength of bipartisanship measure, while a few others were missing over 2800 observations. Therefore, I utilized multiple imputation with 5 iterations, running the model 5 times on separate imputations and then averaging the results together.

This paper investigates the question: Is there a correlation between strength of Congressional bipartisanship and year of vote, congress number, whether the vote was on domestic policy, and whether the vote happened during an election year. I am using a varying-intercept multilevel model in JAGS with the form $y_i = \alpha_j[i] + \beta_1 x_i + \beta_2 x_i + \beta_3 x_i + \epsilon_i$. I decided to use a varying intercept model in order to explore the aforementioned relationship while investigating different intercepts for each Congress number. I hypothesize that there will be bigger differences between Congresses than within each formation of Congress. Furthermore, I expect that there will be less strength of bipartisan as the year increases, less bipartisanship during an election year, and less bipartisanship when the bill is domestic policy. These expectations stem from the growing discourse about political polarization in the news media as well as in academia. Bipartisanship has been often striven for, yet rarely achieved in the recent election cycles, which has halted many bills from making it to the President’s desk. Additionally, I expect Congress to be more unified on foreign affairs than domestic policy, and that there would be less agreement during a congressional election year since Congressmembers might try to accentuate their partisanship in the public eye.

Below, I have summarized the data set limited to the variables that are most pertinent to my research question. These variables are as follows: “Strengthofbipartisanship” measures bipartisanship given the following equation: 1 - (the absolute value of (the share of Republicans voting in favor of the legislation - the share of Democrats voting in favor of the legislation)). “Year” describes the year in which the vote happened. “Electionyear” is an indicator variable with 0 given that the vote did not happen in an election year and 1 if it did. “Domesticpolicy” is also an indicator variable, with 0 given that the bill was not domestic policy and 1 when it was. “Congressnumber” ranges from 102-116, representing the number of the unique makeup of Congress between elections. Due to the nature of the JAGS model process, I recoded the “congressnumber” variable to span from 1 to 15 (where 1 is 102 and 15 is 116) to facilitate the subsetting of this variable to create distinct intercepts for each Congress. Similarly, I recoded the “year” variable to range from 1 to 30 (where 1 is 1991 and 30 is 2020) in order to rescale the outputs. The vectors of these two variables appear in descending order, as this is their order within the dataset.

Model Code

```
## strengthofbipartisanship electionyear domesticpolicy year
## Min. :0.0000      Min. :0.0000      Min. :0.0000      Min. :1991
## 1st Qu.:0.0300    1st Qu.:0.0000    1st Qu.:0.0000    1st Qu.:1995
## Median :0.1200    Median :1.0000    Median :1.0000    Median :2000
## Mean   :0.3388    Mean   :0.5615    Mean   :0.6191    Mean   :2003
## 3rd Qu.:0.6900    3rd Qu.:1.0000    3rd Qu.:1.0000    3rd Qu.:2013
## Max.  :1.0000    Max.  :1.0000    Max.  :1.0000    Max.  :2020
## congressnumber
## Min.  :102.0
## 1st Qu.:104.0
## Median :106.0
## Mean   :107.8
## 3rd Qu.:113.0
## Max.  :116.0

jags.out.mean <- jags.out.se <- jags.out.2.5 <- jags.out.25 <- jags.out.50<- jags.out.75 <- jags.out.97
asap.mat.list<-list()
for (i in 1:5) {
current.votes.dat <- data.frame(votes.array[, , i])
names(current.votes.dat) <- names(votes)
counts<- current.votes.dat%>%
  count(congressnumber)%>%
  arrange(desc(congressnumber))

conrep<- vector()
connum_vec<- list()
for(i in 1:15){
  n<- counts$n[i]
  yr<- c(15:1)
  conrep<- rep(yr[i], n)
  connum_vec[i]<- list(conrep)
}

counts2<- current.votes.dat%>%
  count(year)%>%
  arrange(desc(year))

yrrep<- vector()
yr_vec<- list()
for(i in 1:30){
  n<- counts2$n[i]
  yr<- c(30:1)
  yrrep<- rep(yr[i], n)
  yr_vec[i]<- list(yrrep)
}

y<- as.numeric(current.votes.dat$strengthofbipartisanship)
N<- length(y)
J<- length(unique(current.votes.dat$congressnumber))
elecyr<- as.numeric(current.votes.dat$electionyear)
domestic<- as.numeric(current.votes.dat$domesticpolicy)
year_vec<- as.numeric(unlist(yr_vec))
```

```

congressnum<-as.numeric(unlist(connum_vec))

votes.model <- function() {
  for (i in 1:N){
    y[i] ~ dnorm(y.hat[i], tau.y)
    y.hat[i] <- a[congressnum[i]] + b1*elecyr[i] + b2*domestic[i] + b3*year_vec[i]
  }
  b1 ~ dnorm(0, .0001)
  b2 ~ dnorm(0, .0001)
  b3 ~ dnorm(0, .0001)
  tau.y <- pow(sigma.y, -2)
  sigma.y ~ dunif (0, 100)
  for (j in 1:J){
    a[j] ~ dnorm (mu.a, tau.a)
  }
  mu.a ~ dnorm(0, 0.0001)
  tau.a <- pow(sigma.a, -2)
  sigma.a ~ dunif (0, 100)
}

votes.data <- list("N", "J", "y", "congressnum", "elecyr", "domestic", "year_vec")
votes.inits <- function(){
  list(a=rnorm(J), b1=rnorm(1), b2=rnorm(1), b3=rnorm(1),
       sigma.y=runif(1), sigma.a=runif(1))}
votes.params <- c("a", "b1", "b2", "b3", "sigma.y", "sigma.a", "mu.a")

write.model(votes.model, "votes.model.rjags")

set.seed(12)
votes.jags.out <- jags(data=votes.data, inits=votes.inits,
parameters.to.save=votes.params,
model="votes.model.rjags", n.chains=3, n.iter=400000, DIC=F)
set.seed(12)
update.out<- update(votes.jags.out, n.iter=400000, n.chains=3)

jags.out.mean <- cbind(jags.out.mean, update.out$BUGSoutput$summary[c(1:3, 13:21),1])
jags.out.se <- cbind(jags.out.se, update.out$BUGSoutput$summary[c(1:3, 13:21),2])
jags.out.2.5 <- cbind(jags.out.2.5, update.out$BUGSoutput$summary[c(1:3, 13:21),3])
jags.out.25 <- cbind(jags.out.25, update.out$BUGSoutput$summary[c(1:3, 13:21),4])
jags.out.50<- cbind(jags.out.50, update.out$BUGSoutput$summary[c(1:3, 13:21),5])
jags.out.75 <- cbind(jags.out.75, update.out$BUGSoutput$summary[c(1:3, 13:21),6])
jags.out.97.5<- cbind(jags.out.97.5, update.out$BUGSoutput$summary[c(1:3, 13:21),7])
jags.out.Rhat <- cbind(jags.out.Rhat, update.out$BUGSoutput$summary[c(1:3, 13:21),8])
jags.out.n.eff<- cbind(jags.out.n.eff, update.out$BUGSoutput$summary[c(1:3, 13:21),9])

asap.mat.a1<- cbind(asap.mat.a1, update.out$BUGSoutput$sims.array[,1,1])
asap.mat.a2<- cbind(asap.mat.a2, update.out$BUGSoutput$sims.array[,1,2])
asap.mat.a3<- cbind(asap.mat.a3, update.out$BUGSoutput$sims.array[,1,3])
asap.mat.a4<- cbind(asap.mat.a4, update.out$BUGSoutput$sims.array[,1,4])
asap.mat.a5<- cbind(asap.mat.a5, update.out$BUGSoutput$sims.array[,1,5])
asap.mat.a6<- cbind(asap.mat.a6, update.out$BUGSoutput$sims.array[,1,6])
asap.mat.a7<- cbind(asap.mat.a7, update.out$BUGSoutput$sims.array[,1,7])
asap.mat.a8<- cbind(asap.mat.a8, update.out$BUGSoutput$sims.array[,1,8])

```

```

asap.mat.a9<- cbind(asap.mat.a9, update.out$BUGSoutput$sims.array[,1,9])
asap.mat.a10<- cbind(asap.mat.a10, update.out$BUGSoutput$sims.array[,1,10])
asap.mat.a11<- cbind(asap.mat.a11, update.out$BUGSoutput$sims.array[,1,11])
asap.mat.a12<- cbind(asap.mat.a12, update.out$BUGSoutput$sims.array[,1,12])
asap.mat.a13<- cbind(asap.mat.a13, update.out$BUGSoutput$sims.array[,1,13])
asap.mat.a14<- cbind(asap.mat.a14, update.out$BUGSoutput$sims.array[,1,14])
asap.mat.a15<- cbind(asap.mat.a15, update.out$BUGSoutput$sims.array[,1,15])
asap.mat.a16<- cbind(asap.mat.a16, update.out$BUGSoutput$sims.array[,1,16])
asap.mat.a17<- cbind(asap.mat.a17, update.out$BUGSoutput$sims.array[,1,17])
asap.mat.a18<- cbind(asap.mat.a18, update.out$BUGSoutput$sims.array[,1,18])
asap.mat.a19<- cbind(asap.mat.a19, update.out$BUGSoutput$sims.array[,1,19])
asap.mat.a20<- cbind(asap.mat.a20, update.out$BUGSoutput$sims.array[,1,20])
asap.mat.a21<- cbind(asap.mat.a21, update.out$BUGSoutput$sims.array[,1,21])
}

```

```

## module glm loaded

## module dic loaded

avg.output<- data.frame(mean = rowMeans(jags.out.mean), sd = rowMeans(jags.out.se), "2.5%"= rowMeans(jags.out.ci[,1]))
names(avg.output)<- c("mean", "sd", "2.5%", "25%", "75%", "97.5%", "Rhat", "n.eff")
knitr::kable(avg.output,
digits = 3, caption = "Multiple Imputed Varying Intercepts Model, Abbreviated")

```

Table 1: Multiple Imputed Varying Intercepts Model, Abbreviated

	mean	sd	2.5%	25%	75%	97.5%	Rhat	n.eff
a[1]	0.585	0.016	0.553	0.574	0.596	0.617	1.001	305200
a[2]	0.662	0.028	0.608	0.643	0.681	0.717	1.001	261120
a[3]	0.659	0.025	0.610	0.642	0.676	0.710	1.001	30980
a[13]	0.658	0.097	0.474	0.594	0.718	0.858	1.001	16100
a[14]	0.582	0.102	0.386	0.514	0.646	0.793	1.001	16100
a[15]	0.603	0.115	0.384	0.528	0.675	0.839	1.001	15180
b1	0.022	0.013	-0.003	0.014	0.031	0.047	1.001	247200
b2	-0.119	0.011	-0.141	-0.126	-0.111	-0.097	1.001	888000
b3	-0.005	0.004	-0.013	-0.007	-0.002	0.003	1.001	14040
mu.a	0.664	0.067	0.535	0.621	0.706	0.800	1.001	20720
sigma.a	0.128	0.032	0.081	0.105	0.144	0.206	1.001	101820
sigma.y	0.331	0.004	0.323	0.328	0.333	0.338	1.001	908000

```

asap.mat<- data.frame(a1= rowMeans(asap.mat.a1), a2 = rowMeans(asap.mat.a2), a3 = rowMeans(asap.mat.a3))
names(asap.mat) <- names(as.data.frame(update.out$BUGSoutput$sims.array[,1,]))
asap.mcmc<- mcmc(as.matrix(asap.mat))
convergence<- superdiag(asap.mcmc, burnin = 100000)
convergence$geweke

```

```
## $windows
```

```

##           From start From stop
## chain 1      0.1       0.5
##
## $chains
##           a[1]      a[2]      a[3]      a[4]      a[5]      a[6]
## chain 1 -0.2066944 -0.1585019 -0.082031 -0.146332 -0.1567613 -0.09985812
##           a[7]      a[8]      a[9]      a[10]     a[11]     a[12]
## chain 1 -0.1553056 -0.1860236 -0.177345 -0.1875873 -0.1436373 -0.1644589
##           a[13]     a[14]     a[15]      b1       b2       b3      mu.a
## chain 1 -0.1584535 -0.195152 -0.2493973 -1.165141  1.357242  0.2520616 -0.1988207
##           sigma.a   sigma.y
## chain 1 0.06193647 0.8457891

```

```
convergence$heidel
```

```

## $heidel.params
##           epsilon alpha
## chain 1      0.1  0.05
##
## $heidel.list
## $heidel.list[[1]]
##
##           Stationarity start      p-value
##           test        iteration
## a[1]    passed      1       0.1575
## a[2]    passed      1       0.2273
## a[3]    passed      1       0.2860
## a[4]    passed      1       0.3059
## a[5]    passed      1       0.2744
## a[6]    passed      1       0.2771
## a[7]    passed      1       0.3058
## a[8]    passed      1       0.2608
## a[9]    passed      1       0.2575
## a[10]   passed      1       0.2662
## a[11]   passed      1       0.2608
## a[12]   passed      1       0.2612
## a[13]   passed      1       0.2676
## a[14]   passed      1       0.2728
## a[15]   passed      1       0.3010
## b1      passed      1       0.0873
## b2      passed      1       0.5212
## b3      passed      1       0.2867
## mu.a    passed      1       0.2775
## sigma.a passed      1       0.0774
## sigma.y passed      1       0.3870
##
##           Halfwidth Mean      Halfwidth
##           test
## a[1]    passed    0.58527 2.59e-04
## a[2]    passed    0.66245 5.37e-04
## a[3]    passed    0.65938 8.13e-04
## a[4]    passed    0.75014 1.24e-03
## a[5]    passed    0.52010 1.45e-03
## a[6]    passed    0.75541 1.72e-03

```

```

## a[7]    passed    0.70728 2.13e-03
## a[8]    passed    0.66063 2.32e-03
## a[9]    passed    0.76625 2.58e-03
## a[10]   passed    0.75179 2.96e-03
## a[11]   passed    0.64209 3.17e-03
## a[12]   passed    0.66912 3.44e-03
## a[13]   passed    0.65950 3.81e-03
## a[14]   passed    0.58367 4.08e-03
## a[15]   passed    0.60521 4.58e-03
## b1      passed    0.02208 4.92e-05
## b2      passed    -0.11859 4.56e-05
## b3      passed    -0.00505 1.60e-04
## mu.a   passed    0.66516 2.36e-03
## sigma.a passed    0.12813 3.54e-04
## sigma.y passed    0.33060 8.65e-06

```

```
convergence$raftery
```

```

## $raftery.params
##           Quantile (q) Accuracy (r) Probability (s) Convergence eps
## chain 1          0.025        0.005          0.95        0.001
##
## $raftery.list
## $raftery.list[[1]]
##
## Quantile (q) = 0.025
## Accuracy (r) = +/- 0.005
## Probability (s) = 0.95
##
##           Burn-in  Total  Lower bound Dependence
##           (M)     (N)    (Nmin)   factor (I)
## a[1]      33    43274  3746       11.60
## a[2]      54    71928  3746       19.20
## a[3]     184   246790  3746       65.90
## a[4]     210   252126  3746       67.30
## a[5]     205   230379  3746       61.50
## a[6]     246   236242  3746       63.10
## a[7]     258   299753  3746       80.00
## a[8]     264   297660  3746       79.50
## a[9]     312   354900  3746       94.70
## a[10]    276   305026  3746       81.40
## a[11]    288   285984  3746       76.30
## a[12]    266   303392  3746       81.00
## a[13]    336   395892  3746      106.00
## a[14]    312   339066  3746       90.50
## a[15]    344   405017  3746      108.00
## b1      9     12879   3746       3.44
## b2      9     12720   3746       3.40
## b3     605   630575  3746      168.00
## mu.a   220   265320  3746       70.80
## sigma.a 14   27769   3746       7.41
## sigma.y 4    7838    3746       2.09

```

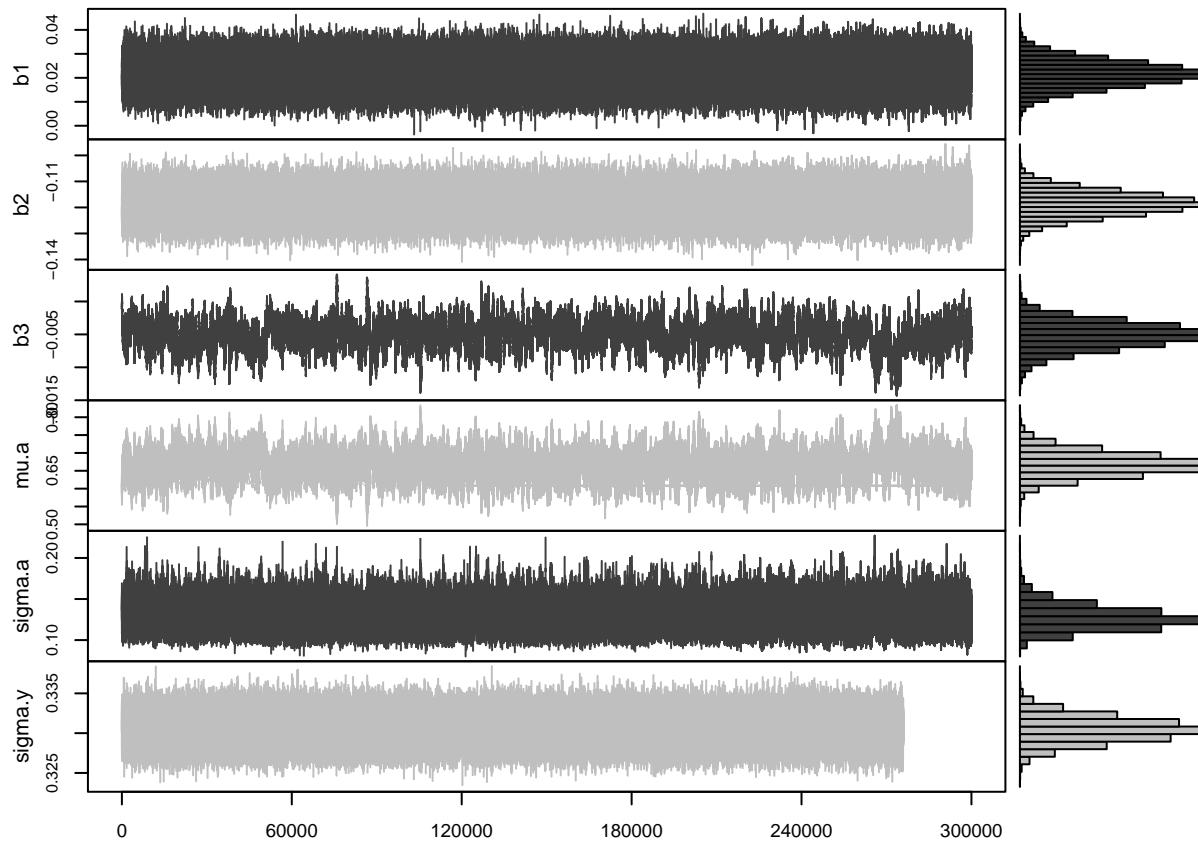
```

convergence$hellinger

## $within
## $within$chain1
##          60000     120000     180000     240000
## a[1]    0.020793219 0.037746433 0.015122944 0.036509587
## a[2]    0.023073285 0.037279732 0.011919587 0.040814618
## a[3]    0.037103186 0.051430348 0.021209822 0.065430569
## a[4]    0.039685916 0.057144458 0.023012615 0.072961567
## a[5]    0.035872866 0.052313022 0.020113147 0.067850614
## a[6]    0.047876536 0.063413176 0.026267824 0.078240893
## a[7]    0.047606642 0.062893760 0.028290202 0.081534251
## a[8]    0.047897675 0.059943366 0.027848181 0.080959734
## a[9]    0.053142364 0.066960530 0.033975132 0.087040733
## a[10]   0.051670318 0.066183250 0.032985281 0.087409023
## a[11]   0.051004466 0.063257327 0.030060649 0.085075780
## a[12]   0.053648641 0.066620405 0.034458750 0.089239968
## a[13]   0.056310929 0.067605324 0.035010052 0.089461929
## a[14]   0.057354320 0.068577031 0.038661971 0.091306959
## a[15]   0.053509083 0.067204739 0.034973473 0.089808394
## b1      0.011977805 0.013070412 0.008404970 0.009634769
## b2      0.012442597 0.008832159 0.009256043 0.008487833
## b3      0.060160188 0.072641453 0.040754085 0.093372166
## mu.a    0.041633433 0.056510091 0.023241464 0.076935149
## sigma.a 0.028294330 0.011947195 0.011413393 0.030181437
## sigma.y 0.009713741 0.009506167 0.011064733 0.008450942
## attr(),"class")
## [1] "hellinger.diag"

plot.mat<- asap.mat[c(16:21)]
plot.mcmc<- mcmc(as.matrix(plot.mat))
convergence.plot<- superdiag(plot.mcmc, burnin = 100000, plot = TRUE)

```



Results

The table above shows a summarized version of the model results, with the first three and last three intercepts, as well as the three predictor coefficients, mean of the intercepts, the standard deviation of y and the standard deviation of alpha. The results show that the estimates on the group-level parameters vary, with an average intercept of 0.664 that is precisely estimated with a standard error of 0.067. This is what I expected, as given the changing political climate each iteration of congress will have a slightly different starting point for their bipartisanship measure. This may be due to the previous Congressional group setting a precedent for the political climate and the newer members adopting the resulting partisan outlook each time. However, contrary to my prediction, the estimated within-Congress standard deviation, 0.331 or σ_{y} , is larger than the estimated between-Congress standard deviation, 0.128 or σ_a . This means that there is less variation between distinct Congresses than there is within each one. In other words, the strength of bipartisanship regarding “important” or key legislation varies more within each Congress group than it does across groups, and thus the Congresses are more similar than I had assumed.

The estimate of the parameter on whether or not the vote was held during a Congressional election year is 0.022. It is precise with a standard error of 0.013 and statistically distinguishable from zero at the 25%-75% interval, but not at the 2.5%-97.5% interval, as seen in the quantile values. This suggests that on average with all other predictors held constant, when a vote occurs during an election year, there is weak evidence that the percentage strength of bipartisanship increases by about 2.2 percentage points. This result is also unexpected, as I had predicted election years to make votes more contentious. This finding suggests that perhaps bipartisanship increases in an effort from incumbent candidates to show voters that they are effective.

The estimate of the domestic policy indicator is -0.119 with a standard error of 0.011 showing precision,

and it is statistically distinguishable from zero as represented by the quantile values. This estimate suggests that on average with all other predictors held constant, when a vote is on a domestic policy bill, the strength of bipartisanship decreases by about 11.9 percentage points, or 0.119. This finding was expected, because domestic policy tends to be more controversial and involve more topic areas with clear partisan divides and strong party platforms. This also upholds that foreign policy votes are more bipartisan, which could be explained by national views on other countries being more unified.

The estimate of the year parameter is -0.005 with a precise standard error of 0.004 and it is also statistically distinguishable from zero at the 25%-75% interval, but not at the 2.5%-97.5% interval given the quantile values. This estimate suggests that when all other variables are held constant, we have weak evidence that on average a one unit increase in year corresponds with a decrease of 0.009 or 0.9 percentage points in strength of bipartisanship in Congress. This variable was of particular interest regarding the research question. It is often claimed and debated that political polarization has increased in the past few decades, and this model supports that claim. It is unclear why strength of bipartisanship has decreased over time, but some prominent arguments include social media and the widespread access to political discourse with people across the country, partisan redistricting practices, and major party platforms becoming more distinguishable.

In exploring the convergence and robustness of this model, I took various precautions and executed multiple evaluations. I initially ran the model with case-wise deleted data knowing that it would be biased. The case-wise deleted model resulted in the three predictor estimates and the intercepts all having slightly higher magnitude than those in this multiple imputed model, and all of them being strongly statistically distinguishable from zero. Furthermore, the case-wise deleted model showed less variation between Congresses (σ_{a}) than this model does.

The first attempt at this multiple imputation model ran 200,000 iterations and 3 chains, though the convergence test results and trace plots indicated that the model had not converged. My next attempt with 300,000 iterations and 3 chains was better, though not optimal in terms of the R.hat and N.eff column values as well as the trace plots. The final model presented above ran 400,000 iterations and 3 chains. The resulting tests, trace plots, R.hat values and n.eff values indicated that the model had converged and reached the stationary distribution.

By looking at the output table R.hat and n.eff columns, we can see strong evidence of convergence. The R.hat values of 1.001 illustrate clearly that the algorithm converged, and the n.eff values suggest that the parameter estimates are precise because there are enough effective simulations. The trace plots displayed above indicate little to no pattern or wavering for each parameter of interest, which clearly shows convergence. They also depict that the first 100,000 iterations were not graphed or convergence tested through the use of the “burnin” argument. Furthermore, the model passed the Heidelberger-Welch, Raftery-Lewis, and Hellinger distance diagnostics, indicating that there was no evidence of nonconvergence.

This multilevel model concluded that although the strength of bipartisanship ratio in Congress varies between each unique Congress group, this variability does not outweigh the variation within each Congress itself. Therefore, it is only slightly beneficial to vary the intercepts by Congress number. However, I would argue that the variation it explains at the group level is beneficial enough to remain in the model, keeping the σ_{a} of 0.128 from adding onto the σ_{y} of 0.331. Furthermore, the model indicates correlations of differing strengths between congressional election years and increased strength of bipartisanship, domestic policy and a decrease in strength of bipartisanship, as well as a decrease in strength of bipartisanship as the year in which voting occurs increases. The reasons behind these relationships are yet unclear, but these findings lay a foundation for future research investigating what may have caused them.