Modern Bayesian Methods and Computing for Survey Research
Part 2: Fitting Realistic Bayesians Modesl with Survey Data

JEFF GILL
Washington University in St. Louis
Bureaucratic Politics Example

- Contains *every* federal political appointee to full-time positions requiring Senate confirmation from November, 1964 through December, 1984 (collected by Mackenzie and Light, ICPSR Study Number 8458, Spring 1987).

- The survey queries various aspects of the Senate confirmation process, acclamation to running an agency or program, and relationships with other functions of government.

- See Gill and Casella (JASA 2009) for more details.
Bureaucratic Politics Example

- **Outcome Variable:** stress as a surrogate measure for self-perceived effectiveness and job-satisfaction, measured as a five-point scale from “not stressful at all” to “very stressful.”

- **Explanatory Variables:**
  - Government Experience,
  - Ideology,
  - Committee Relationship,
  - Career.Exec-Compet,
  - Career.Exec-Liaison/Bur,
  - Career.Exec-Liaison/Cong,
  - Career.Exec-Day2day,
  - Career.Exec-Diff,
  - Confirmation Preparation,
  - Hours/Week,
  - President Orientation.
Ordered Logit Model

- A Bayesian random effects specification for ordered survey outcomes, so latent variable thresholds for $Y$ are assumed on the ordering:

  $$ U_i : \theta_0 \overset{c=1}{\leftrightarrow} \theta_1 \overset{c=2}{\leftrightarrow} \theta_2 \overset{c=3}{\leftrightarrow} \cdots \theta_{C-1} \overset{c=C}{\leftrightarrow} \theta_C $$

- The vector of (unseen) utilities across individuals in the sample, $U$, is determined by a linear additive specification of explanatory variables: $U = -X'\gamma + \eta$, where $\gamma = [\gamma_1, \gamma_2, \ldots, \gamma_p]$ does not depend on the $\theta_j$, and $\eta \sim F_{\eta}$.

- This means that the probability that individual $i$ in the sample is observed to be in category $r$ or lower is:

  $$ P(Y_i \leq r|X_i) = P(U_i \leq \theta_r) = P(\eta \leq \theta_r + X_i'\gamma) = F_{\eta_i}(\theta_r + X_i'\gamma). $$

- Specifying a logistic distributional assumption on the errors and adding the random effect term produces this logistic cumulative specification for the whole sample:

  $$ F_{\eta}(\theta_r + X'\gamma + b) = P(Y \leq r|X) = [1 + \exp(-\theta_r - X'\gamma + b)]^{-1} $$
Prior distributions are either semi-informed or skeptical:

\[
p(\gamma_k) \sim \mathcal{N}(\mu_{\gamma_k}, \sigma_{\gamma}^2), \ k = 1, \ldots, p
\]
for each of the \( p \) explanatory variables,

\[
p(\theta_j) \sim \mathcal{N}(0, \sigma_{\theta}^2), \ j = 1, \ldots, C - 1
\]
for the four latent variable thresholds,

\[
b_i \sim \mathcal{N}(0, \tau)
\]
for the random effects term,

\[
\tau \sim \mathcal{IG}(\delta_1, \delta_2)
\]
for the random effects hyperprior,
All this produces a posterior distribution according to:

$$
\pi(\gamma, \theta|X, Y) \propto L(\gamma, \theta|X, Y)p(\theta)p(\gamma)p(b|\tau)p(\tau)
$$

$$
\propto \prod_{i=1}^{n} \prod_{j=1}^{C-1} \prod_{k=1}^{p} [\Lambda(\theta_j - X_i'\gamma + b_i) - \Lambda(\theta_{j-1} - X_i'\gamma + b_i)]^{z_{ij}}
$$

$$
\times \exp \left( -\frac{(\gamma_k - \mu_{\gamma_k})^2}{2\sigma_{\gamma}^2} - \frac{\theta_j^2}{2\sigma_{\theta}^2} - \frac{b_i^2}{2\tau_2^2} - \frac{\delta_2^2}{\tau} \right) \tau^{-(\delta_1+1)}
$$

which is kind of ugly (and hard marginalize).

Solution: Gibbs sampling, a type of MCMC.
Posterior Summary, Model for Survey of Political Executives

Table 3: Posterior Summary, Model for Survey of Political Executives

<table>
<thead>
<tr>
<th>Explanatory Variables:</th>
<th>Mean</th>
<th>Std.Err.</th>
<th>95% HPD Intervals</th>
</tr>
</thead>
<tbody>
<tr>
<td>Constant Term</td>
<td>1.20215</td>
<td>2.24169</td>
<td></td>
</tr>
<tr>
<td>Government Experience</td>
<td>-0.65500</td>
<td>0.60664</td>
<td></td>
</tr>
<tr>
<td>Ideology</td>
<td>-0.49140</td>
<td>0.32964</td>
<td></td>
</tr>
<tr>
<td>Committee Relationship</td>
<td>1.14550</td>
<td>0.39131</td>
<td></td>
</tr>
<tr>
<td>Career.Exec-Compet</td>
<td>1.73300</td>
<td>0.85088</td>
<td></td>
</tr>
<tr>
<td>Career.Exec-Liaison/Bur</td>
<td>-2.81800</td>
<td>0.55620</td>
<td></td>
</tr>
<tr>
<td>Career.Exec-Liaison/Cong</td>
<td>1.03675</td>
<td>0.49193</td>
<td></td>
</tr>
<tr>
<td>Career.Exec-Day2day</td>
<td>-0.76595</td>
<td>0.38019</td>
<td></td>
</tr>
<tr>
<td>Career.Exec-Diff</td>
<td>0.27780</td>
<td>0.29838</td>
<td></td>
</tr>
<tr>
<td>Confirmation Preparation</td>
<td>1.02440</td>
<td>0.45753</td>
<td></td>
</tr>
<tr>
<td>Hours/Week</td>
<td>-0.72720</td>
<td>0.42732</td>
<td></td>
</tr>
<tr>
<td>President Orientation</td>
<td>1.94950</td>
<td>0.86787</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Threshold Intercepts:</th>
<th>Mean</th>
<th>Std.Err.</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>None</td>
<td>Little</td>
<td>-5.93500</td>
<td>1.85782</td>
</tr>
<tr>
<td>Little</td>
<td>Some</td>
<td>-2.69250</td>
<td>1.59126</td>
</tr>
<tr>
<td>Some</td>
<td>Significant</td>
<td>1.19300</td>
<td>1.52653</td>
</tr>
<tr>
<td>Significant</td>
<td>Extreme</td>
<td>8.40450</td>
<td>2.10379</td>
</tr>
</tbody>
</table>

\[ \hat{\sigma} = 6.04350 (1.20325), \text{ dashed vertical line at zero.} \]
Core BUGS Code for this Model

for (i in 1:N) {
  b[i] ~ dnorm(0.0, tau)
  mu[i] <- theta[1]
  + theta[2]*previous.EMPL[i] + theta[3]*ideology[i]
  + theta[4]*senate.rel[i] + theta[5]*confirm.prep[i]
  + theta[6]*hours.week[i] + theta[7]*career.exec.compet[i]
  + theta[8]*career.exec.liason.bur[i] + theta[9]*career.exec.liason.cong[i]
  + theta[10]*career.exec.day2day[i] + theta[11]*career.exec.diff[i]
  + theta[12]*president.orient[i]
  for (j in 1:Ncut) { logit(Q[i, j]) <- -(k[j] + mu[i] - b[i]) } # CUM PROB LOWER THAN j
  # NOW DO PROBABILITY THAT RESPONSE = j IN THREE STAGES: LOWEST, MIDDLES, HIGHEST
  p[i, 1] <- max( min(1 - Q[i, 1], 1), 0)
  for (j in 2 : Ncut) { p[i,j] <- max( min(Q[i, j - 1] - Q[i, j],1), 0) }
  p[i, (Ncut+1)] <- max( min(Q[i, Ncut], 1), 0)
  # DRAW STRESS VALUE FROM p[]
  stress[i] ~ dcat(p[i,])
}
What is a Stochastic Process?

◊ A type of stochastic process that will help us estimate posterior quantities.

◊ A *stochastic process* is a consecutive set of random quantities defined on some known state space, \( \Theta \), indexed so that the order is known: \( \{\theta^t: t \in T\} \).

◊ Frequently, but not necessarily, \( T \) is the set of positive integers implying consecutive, even-spaced time intervals: \( \{\theta^{t=0}, \theta^{t=1}, \theta^{t=2}, \ldots\} \).

◊ A stochastic process must also be defined with respect to a *state space*, \( \Theta \), which identifies the range of possible values of \( \theta \). This state space is either discrete or continuous depending on how the variable of interest is measured.
What is a Markov Chain?

◊ A *Markov chain* is a stochastic process with the property that any specified state in the series, $\theta^{[t]}$, is dependent only the previous value of the chain, $\theta^{[t-1]}$.

◊ Therefore values are *conditionally* independent of all other previous values: $\theta^{[0]}, \theta^{[1]}, \ldots, \theta^{[t-2]}$.

◊ Formally:

$$P(\theta^{[t]} \in A | \theta^{[0]}, \theta^{[1]}, \ldots, \theta^{[t-1]}) = P(\theta^{[t]} \in A | \theta^{[t-1]})$$

where $A$ is any identified set (an event or range of events) on the complete state space. (We will use this $A$ notation extensively.)

◊ Colloquially:

“A Markov chain wanders around the state space remembering only where it has been in the last period.”
Why Is this Useful?

◊ This “short-term” memory property is very useful because when the chain eventually finds the region of the state space with highest density, it will wander around there producing a sample that is only modestly nonindependent.

◊ If this is the posterior region, then we can use these “empirical” values as legitimate posterior sample values.

◊ Thus difficult posterior calculations can be done with MCMC by letting the chain wander around “sufficiently long”, thus producing summary statistics from recorded values.

◊ Sounds simple.
How Does it Move?

- How does the Markov chain decide to move?

- Define the *transition kernel*, $K$, as a general mechanism for describing the probability of moving to some other specified state based on the current chain status.

- $K(\theta, A)$ is a defined probability measure for all $\theta$ points in the state space to the set $A \in \Theta$.

- So $K(\theta, A)$ maps potential transition events to their probability of occurrence.
What is a Discrete Space Markov Chain Kernel?

◊ When the state space is discrete, \( K \) is a matrix mapping, \( k \times k \) for \( k \) discrete elements in \( A \), where each cell defines the probability of a state transition from the first term to all possible states:

\[
P_A = \begin{bmatrix}
p(\theta_1, \theta_1) & p(\theta_1, \theta_2) & \cdots & p(\theta_1, \theta_{k-1}) & p(\theta_1, \theta_k) \\
p(\theta_2, \theta_1) & p(\theta_2, \theta_2) & \cdots & p(\theta_2, \theta_{k-1}) & p(\theta_2, \theta_k) \\
\vdots & \vdots & \ddots & \vdots & \vdots \\
p(\theta_k, \theta_1) & p(\theta_k, \theta_2) & \cdots & p(\theta_k, \theta_{k-1}) & p(\theta_k, \theta_k)
\end{bmatrix}
\]

where the row indicates where the chain is at this period and the column indicates where the chain is going in the next period.

◊ Each matrix element is a well-behaved probability, \( p(\theta_i, \theta_j) \geq 0, \forall i, j \in A \). When the state space is continuous, then \( K \) is a conditional PDF: \( f(\theta|\theta_i) \).

◊ Rows of \( P_A \) sum to one and define a conditional PMF since they are all specified for the same starting value and cover each possible destination in the state space: for row \( i: \sum_{j=1}^{k} p(\theta_i, \theta_j) \).
What is a Continuous Space Markov Chain Kernel?

◊ When the state space is continuous, then $K$ is a conditional PDF: $f(\theta|\theta_i)$.

◊ $K$ is a conditional PDF: $f(\theta|\theta_i)$, meaning a properly defined probability statement for all $\theta \in A$, given some given current state $\theta_i$.

◊ Continuous state space Markov chains have more involved theory; so it’s often convenient to think about discrete Markov chains at first.
A Two State Markov Chain

♦ A two-dimensional state space: a discrete vote choice between two political parties, a commercial purchase decision between two brands, etc.

♦ Voters/consumers/predators/viruses/etc. who normally select $\theta_1$ have an 80% chance of continuing to do so, and voters/consumers who normally select $\theta_2$ have only a 40% chance of continuing to do so.

♦ The transition matrix $P$:

\[
\begin{pmatrix}
\theta_1 & \theta_2 \\
\theta_1 & 0.8 & 0.2 \\
\theta_2 & 0.6 & 0.4
\end{pmatrix}
\]
A Two State Markov Chain

◊ Assign a starting point:

\[ S_0 = \begin{bmatrix} 0.5 & 0.5 \end{bmatrix}; \]

◊ To get to the first state, we simply multiply the initial state by the transition matrix:

\[ S_1 = \begin{bmatrix} 0.5 & 0.5 \end{bmatrix} \begin{bmatrix} 0.8 & 0.2 \\ 0.6 & 0.4 \end{bmatrix} = \begin{bmatrix} 0.7 & 0.3 \end{bmatrix} = S_1. \]
A Two State Markov Chain

◊ This series continues multiplicatively as long as we like:

Second state: \( S_2 = \begin{bmatrix} 0.7 & 0.3 \\ 0.6 & 0.4 \end{bmatrix} \begin{bmatrix} 0.8 & 0.2 \\ 0.2 & 0.4 \end{bmatrix} = \begin{bmatrix} 0.74 & 0.26 \end{bmatrix} \)

Third state: \( S_3 = \begin{bmatrix} 0.74 & 0.26 \\ 0.6 & 0.4 \end{bmatrix} \begin{bmatrix} 0.8 & 0.2 \\ 0.2 & 0.4 \end{bmatrix} = \begin{bmatrix} 0.748 & 0.252 \end{bmatrix} \)

Fourth state: \( S_4 = \begin{bmatrix} 0.748 & 0.252 \\ 0.6 & 0.4 \end{bmatrix} \begin{bmatrix} 0.8 & 0.2 \\ 0.2 & 0.4 \end{bmatrix} = \begin{bmatrix} 0.7496 & 0.2504 \end{bmatrix} \).
A Two State Markov Chain

◊ The **Big Picture**: Imagine that this stationary distribution was the articulation of some PMF or PDF that we could not analytically describe but would like to. If we could run this Markov chain sufficiently long we would eventually get the stationary distribution *for any point in the state space*.

Actually, for this simple example we could solve directly for the steady state $S = [s_1, s_2]$ by stipulating:

$$
\begin{bmatrix}
  s_1 & s_2 \\
  0.8 & 0.2 \\
  0.6 & 0.4 \\
\end{bmatrix}
= 
\begin{bmatrix}
  s_1 \\
  s_2 \\
\end{bmatrix},
$$

and solving the resulting two equations for the two unknowns.

◊ This operation of running a Markov chain until it reaches its stationary distribution is exactly the process employed in MCMC.
Another Simple Example

- Define the intended algorithm as some means of shuffling cards.

- The objective (stationary distribution) is a uniformly random distribution in the deck: for any given order to the stack, the probability of any one card occupying any one position is 1/52.

- Algorithm: take the top card and insert it uniformly randomly at some other point in the deck, continue.

- Is this a Markov chain?

- What is the limiting distribution?

- See BTW, Bayer and Diaconis (1992).
Another Simple Example (cont.)

- For simplicity (without loss of generality), set $n = 3$ cards.

- Sample space (3! elements): $\{[1, 2, 3], [1, 3, 2], [2, 1, 3], [2, 3, 1], [3, 1, 2], [3, 2, 1]\}$.

- Transition kernel:

$$
P = \begin{bmatrix}
    1/3 & 0 & 1/3 & 1/3 & 0 & 0 \\
    0 & 1/3 & 0 & 0 & 1/3 & 1/3 \\
    1/3 & 1/3 & 1/3 & 0 & 0 & 0 \\
    0 & 0 & 0 & 1/3 & 1/3 & 1/3 \\
    1/3 & 1/3 & 0 & 0 & 1/3 & 0 \\
    0 & 0 & 1/3 & 1/3 & 0 & 1/3 \\
\end{bmatrix}
$$

- Starting point: $[1, 2, 3]$.

- Assume no periodicity, obviously it is irreducible and closed (therefore positive recurrent), therefore this transition kernel defines an ergodic Markov chain.
Another Simple Example (cont.)

R code:

```r
P <- matrix(c(1/3,0,1/3,0,0,1/3,1/3,0,1/3,0,1/3,0,1/3,0,0,1/3,
              1/3,0,0,1/3,0,1/3,0,1/3,1/3,1/3,0,0,1/3,0,1/3,0,1/3),nrow=6)
P
```

```
[1,] 0.3333333 0.0000000 0.3333333 0.3333333 0.0000000 0.0000000
[2,] 0.0000000 0.3333333 0.0000000 0.0000000 0.3333333 0.3333333
[3,] 0.3333333 0.3333333 0.3333333 0.0000000 0.0000000 0.0000000
[4,] 0.0000000 0.0000000 0.0000000 0.3333333 0.3333333 0.3333333
[5,] 0.3333333 0.3333333 0.0000000 0.0000000 0.3333333 0.0000000
[6,] 0.0000000 0.0000000 0.3333333 0.3333333 0.0000000 0.3333333
```
Another Simple Example (cont.)

MC.multiply <- function(P.in, N) {
    S <- c(1,0,0,0,0,0)%*%P.in
    for (i in 2:N) {
        S <- S%*%P.in
        print(S)
    }
}

MC.multiply(P, 15)

[1,]  0.2222222 0.1111111 0.2222222 0.2222222 0.1111111 0.1111111

[1,]  0.1851852 0.1481481 0.1851852 0.1851852 0.1481481 0.1481481

[1,]  0.1728395 0.1604938 0.1728395 0.1728395 0.1604938 0.1604938
<table>
<thead>
<tr>
<th></th>
<th>,1</th>
<th>,2</th>
<th>,3</th>
<th>,4</th>
<th>,5</th>
<th>,6</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0.1687243</td>
<td>0.1646091</td>
<td>0.1687243</td>
<td>0.1687243</td>
<td>0.1646091</td>
<td>0.1646091</td>
</tr>
<tr>
<td>0</td>
<td>0.1673525</td>
<td>0.1659808</td>
<td>0.1673525</td>
<td>0.1673525</td>
<td>0.1659808</td>
<td>0.1659808</td>
</tr>
<tr>
<td>0</td>
<td>0.1668953</td>
<td>0.1664380</td>
<td>0.1668953</td>
<td>0.1668953</td>
<td>0.1664380</td>
<td>0.1664380</td>
</tr>
<tr>
<td>0</td>
<td>0.1667429</td>
<td>0.1665905</td>
<td>0.1667429</td>
<td>0.1667429</td>
<td>0.1665905</td>
<td>0.1665905</td>
</tr>
<tr>
<td>0</td>
<td>0.1666921</td>
<td>0.1666413</td>
<td>0.1666921</td>
<td>0.1666921</td>
<td>0.1666413</td>
<td>0.1666413</td>
</tr>
<tr>
<td>0</td>
<td>0.1666751</td>
<td>0.1666582</td>
<td>0.1666751</td>
<td>0.1666751</td>
<td>0.1666582</td>
<td>0.1666582</td>
</tr>
<tr>
<td>0</td>
<td>0.1666695</td>
<td>0.1666638</td>
<td>0.1666695</td>
<td>0.1666695</td>
<td>0.1666638</td>
<td>0.1666638</td>
</tr>
<tr>
<td>0</td>
<td>0.1666676</td>
<td>0.1666657</td>
<td>0.1666676</td>
<td>0.1666676</td>
<td>0.1666657</td>
<td>0.1666657</td>
</tr>
<tr>
<td>0</td>
<td>0.1666670</td>
<td>0.1666640</td>
<td>0.1666670</td>
<td>0.1666670</td>
<td>0.1666640</td>
<td>0.1666640</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td>[1,]</td>
<td>0.1666667</td>
<td>0.1666666</td>
<td>0.1666667</td>
<td>0.1666667</td>
<td>0.1666666</td>
<td></td>
</tr>
<tr>
<td>[1,]</td>
<td>0.1666667</td>
<td>0.1666666</td>
<td>0.1666667</td>
<td>0.1666667</td>
<td>0.1666666</td>
<td></td>
</tr>
<tr>
<td>[1,]</td>
<td>0.1666667</td>
<td>0.1666666</td>
<td>0.1666667</td>
<td>0.1666667</td>
<td>0.1666666</td>
<td></td>
</tr>
</tbody>
</table>
Marginal Distributions

We want the *marginal* distribution at some step $m$ th from the transition kernel.

For the discrete case the marginal distribution of the chain at the $m$ step is obtained by inserting the current value of the chain, $\theta_i^{[m]}$, into the row of the transition kernel for the $m^{th}$ step, $p^m$:

$$
\pi^m(\theta) = [p^m(\theta_1), p^m(\theta_2), \ldots, p^m(\theta_k)].
$$

So the marginal distribution at the first step of discrete Markov chain is given by:

$$
\pi^1(\theta) = p^1 \pi^0(\theta),
$$

where $\pi^0$ is the initial starting value assigned to the chain and $p^1 = p$ is a transition matrix.
Markov Chain Properties: Homogeneity

- **Homogeneity**

  - A Markov chain is said to be **homogeneous** at step \( m \) if the transition probabilities at this step do not depend on the value of \( m \).

  - With non-homogeneous Markov chains the transition matrix is not constant but a function of time.

  - These are useful for modeling probabilistic decay processes, but not for our purposes.

  - The samplers that we use/pick are known to obtain the property, subject to machine implementation.

  - Counter-example: at the starting point a chain cannot be homogeneous since the marginal distribution for the first step is clearly not independent of the initial values that are hand-picked.
Markov Chain Properties: States

Some Characteristics of States:

▷ A state is absorbing if once the chain enters this state it cannot leave: \(p(A, A^c) = 0\).

▷ A state is transient if the probability of the chain not returning to this state is non-zero: \(1 - p(A, A) > 0\). Also, phrases as a finite number of visits in infinite time.

▷ State \(A\) is closed to state \(B\) if a Markov chain on \(A\) cannot reach \(B\): \(p(A, B) = 0\). State \(A\) is closed in the general sense if it is absorbing.
Markov Chain Properties: Irreducibility

- Irreducibility

  ▶ A state is irreducible if for every $\theta_i$ and $\theta_j$ in this state, the two sub-states “communicate.”

  ▶ A Markov chain is irreducible on $A$ if every reached point or collection of points (necessarily a subspace in the continuous case) can be reached from every other reached point or collection of points:

  $$p(\theta_i, \theta_j) \neq 0, \forall \theta_i, \theta_j \in A.$$  

  ▶ A convenient way to remember the principle behind irreducibility is the notion that you could reduce the set if you wanted to, but that you do not want to because then there will be points that cannot be reached from other points.
Markov Chain Properties: Recurrence

▸ Recurrence

▷ If a state is closed, discrete or continuous but finite, and irreducible, then this state and all sub-spaces within this subspace are recurrent.

▷ Markov chains operating on recurrent state spaces are recurrent.

▷ Recurrence is good!

▷ Formal definition:

A irreducible Markov chain is called *recurrent* with regard to a given state, A, which is a single point or a defined collection of points (required for the bounded-continuous case), if the probability that the chain occupies A infinitely often over unbounded time is nonzero.
Markov Chain Properties: Recurrence

► **Recurrence**

▷ Colloquial definition:

   When a chain moves into a recurrent state, it stays there forever and visits every subspace infinitely often.

▷ A Markov chain is *positive recurrent* if the mean time to return to \( A \) is bounded.

▷ Otherwise it is called *null recurrent*.

▷ Note: irreducible chains are either recurrent or transient.
Markov Chain Properties: Harris Recurrence

- **Harris Recurrence**
  - If we only had to deal with discrete or finite state spaces, then standard recurrence would be enough.
  - With unbounded-continuous state spaces it is necessary to have a stricter definition that guarantees that the probability of visiting subspace $A$ infinitely often in the limit is still one.
  - Define $\eta_A$ as the number of visits to $A$ in the limit.
  - Require $P(B) > 0 \forall B \subset A$.
  - The set $A$ is Harris recurrent if $P(\eta_B = \infty) = 1$, $B \in A$.
  - An irreducible Markov chain is Harris recurrent if it every possible subspace is Harris recurrent.
  - An aperiodic, irreducible chain with an invariant distribution on an unbounded continuous state space that is *not* Harris recurrent has a positive probability of getting stuck forever in an area bounded away from convergence, given a starting point there.
Markov Chain Properties: Linkages

► Recurrence and Irreducibility

▷ The union of a set of recurrent states (nonempty, and bounded or countable) or Harris recurrent states is a new state that is closed and irreducible.

▷ The linkage between recurrence and irreducibility is important in defining a subspace that captures a Markov chain and at the same time assures that this Markov chain will explore all of the subspace.

▷ Important result:
  • Given a Markov chain on continuous state space,
  • when the chain wanders into a closed, irreducible set of Harris recurrent states,
  • it stays there forever and visits every single sub-state (region) with probability one.
Markov Chain Properties: Stationarity

- **Stationarity**

  - Define $\pi(\theta)$ as the stationary distribution of the Markov chain for $\theta$ on the state space $A$.

  - Recall that $p(\theta_i, \theta_j)$ is the probability that the chain will move from $\theta_i$ to $\theta_j$ at some arbitrary step $t$,

  - and $\pi^t(\theta)$ is the corresponding marginal distribution.

  - The stationary distribution satisfies:

    \[
    \sum_{\theta_i} \pi^t(\theta_i)p(\theta_i, \theta_j) = \pi^{t+1}(\theta_j) \quad \text{Discrete case}
    \]

    \[
    \int \pi^t(\theta_i)p(\theta_i, \theta_j)d\theta_i = \pi^{t+1}(\theta_j) \quad \text{Continuous case.}
    \]

  meaning that application of the kernel to the current distribution probabilities returns the same probabilities.
Markov Chain Properties: Periodicity

- **Periodicity**

- The *period* of a Markov chain is the length of time required to repeat an identical cycle of chain values.

- Actually, periodicity is bad and we want an *aperiodic* Markov chain: the repeating length is the trivial value of one.

- The problem with periodicity is that it destroys the probabilistic property of the transition kernel.

- Update of our status: we have recurrence to make sure that we revisit states infinitely often in the limit, we have stationarity to make sure that eventually we will obtain constancy of the probability structure, but we don’t have any guarantee about what kind of distribution we obtain in this limit.

- Caveat: all Markov chains run on computers are actually periodic (“Anyone who considers arithmetical methods of producing random digits is, of course, in a state of sin.” –John von Neumann [1903-1957])
Markov Chain Properties: Ergodicity

If a chain is positive Harris recurrent (unbounded continuous case) or recurrent (discrete/bounded continuous case), and aperiodic, then we call it ergodic. Ergodic Markov chains have the property:

$$\lim_{n \to \infty} K^n(\theta_i, \theta_j) = \pi(\theta_j),$$

for all $\theta_i$, and $\theta_j$ in the subspace

What this means: once a specified chain is determined to have reached its ergodic state, sample values behave as if they were produced by the posterior of interest from the model.

The ergodic theorem is the equivalent of the strong law of large numbers but for Markov chains, since it states that any specified function of the posterior distribution can be estimated with samples from a Markov chain in its ergodic state because averages of sample values give strongly consistent parameter estimates.
Markov Chain Properties

- Consequences of Ergodicity

  - Suppose \( \theta_{i+1}, \ldots, \theta_{i+n} \) are \( n \) (not necessarily consecutive) values from a Markov chain that has reached its ergodic distribution, a statistic of interest, \( h(\theta) \), can be calculated empirically:

    \[
    \hat{h}(\theta_i) = \frac{1}{n} \sum_{j=i+1}^{i+n} h(\theta_j) = \int h(\theta) d\pi(\theta)
    \]

    This is the ergodic theorem.

  - For finite quantities this converges almost surely:

    \[
    p[\hat{h}(\theta_i) \to h(\theta), \text{ as } n \to \infty] = 1
    \]

  - Even though Markov chain values, by their very definition, have serial dependence, the mean of the chain values provides a strongly consistent estimate of the true parameter.
Consequences...

- No matter which rate of ergodicity applies, we get the important theoretical result from the CLT.

- For a given empirical estimator \( \hat{h}(\theta_i) \) with bounded limiting variance, we get:

\[
\sqrt{n} \frac{\hat{h}(\theta_i) - h(\theta)}{\sqrt{Var(\hat{h}(\theta_i))}} \rightarrow \mathcal{N}(0, 1) \quad n \to \infty
\]

since \( \hat{h}(\theta_i) = \frac{1}{n} \sum_{j=i+1}^{i+n} h(\theta_j) \) is really a mean.

- If we want to be even more simple, but less general:

\[
\sqrt{n} \frac{\hat{\theta}_i - \theta}{\sqrt{Var(\hat{\theta}_i)}} \rightarrow \mathcal{N}(0, 1) \quad n \to \infty
\]
The Gibbs Sampler

◊ Recall the conditional exponential pdfs:

\[
f(x|y) \propto y \exp[-yx], \quad f(y|x) \propto x \exp[-xy], \quad 0 < x, y < B < \infty.
\]

where we want to describe the marginal distributions of \( x \) and \( y \) (Casella and George 1992).

◊ For two parameters, \( x \) and \( y \), this involves a starting point, \([x_0, y_0]\), and the cycles defined by drawing random values from the conditionals according to:

\[
\begin{align*}
x_1 & \sim f(x|y_0), & y_1 & \sim f(y|x_1) \\
x_2 & \sim f(x|y_1), & y_2 & \sim f(y|x_2) \\
x_3 & \sim f(x|y_2), & y_3 & \sim f(y|x_3) \\
\vdots & & \vdots \\
x_m & \sim f(x|y_{m-1}), & y_m & \sim f(y|x_m).
\end{align*}
\]
The Gibbs Sampler, Conditional Exponential Distributions

B <- 5; k <- 15; m <- 500; x <- NULL; y <- NULL
while (length(x) < m) {
    x.val <- c(runif(1,0,B),rep((B+1),length=k))
    y.val <- c(runif(1,0,B),rep((B+1),length=k))
    for (j in 2:(k+1)) {
        while(x.val[j] > B) x.val[j] <- rexp(1,y.val[j-1])
        while(y.val[j] > B) y.val[j] <- rexp(1,x.val[j])
    }
    x <- c(x,x.val[(k+1)])
    y <- c(y,y.val[(k+1)])
}
The Gibbs Sampler (cont.)
The Gibbs Sampler (cont.)

◊ Steps:

1. Choose starting values: $\theta^{[0]} = [\theta_1^{[0]}, \theta_2^{[0]}, \ldots, \theta_k^{[0]}]$.

2. At the $j^{th}$ iteration, $j = 1, \ldots m$ complete the single cycle by drawing values from the $k$ distributions given by:

\[
\begin{align*}
\theta_1^{[j]} & \sim \pi(\theta_1 | \theta_2^{[j-1]}, \theta_3^{[j-1]}, \theta_4^{[j-1]}, \ldots, \theta_{k-1}^{[j-1]}, \theta_k^{[j-1]}) \\
\theta_2^{[j]} & \sim \pi(\theta_2 | \theta_1^{[j]}, \theta_3^{[j-1]}, \theta_4^{[j-1]}, \ldots, \theta_{k-1}^{[j-1]}, \theta_k^{[j-1]}) \\
\theta_3^{[j]} & \sim \pi(\theta_3 | \theta_1^{[j]}, \theta_2^{[j]}, \theta_4^{[j-1]}, \ldots, \theta_{k-1}^{[j-1]}, \theta_k^{[j-1]}) \\
& \vdots \\
\theta_{k-1}^{[j]} & \sim \pi(\theta_{k-1} | \theta_1^{[j]}, \theta_2^{[j]}, \theta_3^{[j]}, \ldots, \theta_{k-2}^{[j]}, \theta_k^{[j-1]}) \\
\theta_k^{[j]} & \sim \pi(\theta_k | \theta_1^{[j]}, \theta_2^{[j]}, \theta_3^{[j]}, \ldots, \theta_{k-2}^{[j]}, \theta_{k-1}^{[j]})
\end{align*}
\]

3. Increment $j$ and repeat until convergence.
Gibbs Sampler Theory

Properties of the Gibbs sampler:

- Since the Gibbs sampler conditions only on values from the last iteration of its chain values, it clearly has the Markovian property.

- The Gibbs sampler has the true posterior distribution of the parameter vector as its limiting distribution: \( \theta^{[i]} \xrightarrow{d} \theta \sim \pi(\theta) \).

- The Gibbs sampler is a homogeneous Markov chain: the consecutive probabilities are independent of \( n \), the current length of the chain.

- The Gibbs sampler converges at a geometric rate: the total variation distance between an arbitrary time and the point of convergence decreases at a geometric rate in time \( t \).

- The Gibbs sampler is ergodic.
Consider the following economic data from Organization for Economic Cooperation and Development (OECD) that highlight the relationship between commitment to employment protection measured on an interval scale (0 to 4) indicating the quantity and extent of national legislation to protect jobs, and the total factor productivity difference in growth rates between 1980-1990 and 1990-1998 (see *The Economist*, September 23, 2000 for a discussion).
Linear Example Data

The actual data:

<table>
<thead>
<tr>
<th>Country</th>
<th>Prot</th>
<th>Prod</th>
<th>Country</th>
<th>Prot</th>
<th>Prod</th>
</tr>
</thead>
<tbody>
<tr>
<td>United States</td>
<td>0.2</td>
<td>0.5</td>
<td>Canada</td>
<td>0.6</td>
<td>0.6</td>
</tr>
<tr>
<td>Australia</td>
<td>1.1</td>
<td>1.3</td>
<td>New Zealand</td>
<td>1.0</td>
<td>0.4</td>
</tr>
<tr>
<td>Ireland</td>
<td>1.0</td>
<td>0.1</td>
<td>Denmark</td>
<td>2.0</td>
<td>0.9</td>
</tr>
<tr>
<td>Finland</td>
<td>2.2</td>
<td>0.7</td>
<td>Austria</td>
<td>2.4</td>
<td>-0.1</td>
</tr>
<tr>
<td>Belgium</td>
<td>2.5</td>
<td>-0.4</td>
<td>Japan</td>
<td>2.6</td>
<td>-0.4</td>
</tr>
<tr>
<td>Sweden</td>
<td>2.9</td>
<td>0.5</td>
<td>Netherlands</td>
<td>2.8</td>
<td>-0.5</td>
</tr>
<tr>
<td>France</td>
<td>2.9</td>
<td>-0.9</td>
<td>Germany</td>
<td>3.2</td>
<td>-0.2</td>
</tr>
<tr>
<td>Greece</td>
<td>3.6</td>
<td>-0.3</td>
<td>Portugal</td>
<td>3.9</td>
<td>0.3</td>
</tr>
<tr>
<td>Italy</td>
<td>3.8</td>
<td>-0.3</td>
<td>Spain</td>
<td>3.5</td>
<td>-1.5</td>
</tr>
</tbody>
</table>
Data In **BUGS** Format

```r
list(x= c(0.20, 0.60, 1.10, 1.00, 1.00, 2.00, 2.20, 2.40, 2.50, 2.82, 2.90,
         2.80, 2.90, 3.20, 3.60, 3.90, 3.90, 3.50),
y= c(0.5, 0.6, 1.3, 0.4, 0.1, 0.9, 0.7, -0.1, -0.4, -0.4, 0.5, -0.6,
    -0.9, -0.2, -0.3, 0.3, -0.3, -1.5), N=18)
```
Data In JAGS Format

```r
oecd.dat <- list(x = c(0.20, 0.60, 1.10, 1.00, 1.00, 2.00, 2.20, 2.40, 2.50, 2.82, 2.90, 2.80, 2.90, 3.20, 3.60, 3.90, 3.90, 3.50),
                 y = c(0.5, 0.6, 1.3, 0.4, 0.1, 0.9, 0.7, -0.1, -0.4, -0.4, 0.5, -0.6, -0.9, -0.2, -0.3, 0.3, -0.3, -1.5),
                 N = 18)
```
Linear Example Comparison

► We know from Gauss-Markov theory that the posterior distribution of both the intercept and the slope coefficients is student’s-t with \( n - k - 1 = 17 \) degrees of freedom.

► So why are we running **BUGS** on a linear model? Consider how different the estimation process really is here:

\[ \hat{b} = (X'X)^{-1}X'y \]

versus

\[
\begin{align*}
\alpha_1 & \sim f(\alpha|\beta_0), \\
\beta_1 & \sim f(\beta|\alpha_1) \\
\alpha_2 & \sim f(\alpha|\beta_1), \\
\beta_2 & \sim f(\beta|\alpha_2) \\
& \vdots
\end{align*}
\]

\[
\begin{align*}
\alpha_m & \sim f(\alpha|\beta_{m-1}), \\
\beta_m & \sim f(\beta|\alpha_m).
\end{align*}
\]

not to mention *priors*!
Linear Example Code

- First define the *statistical structure* of the model:

  \[
  \mu[i] \leftarrow \alpha + \beta \times x[i]; \\
  y[i] \sim \text{dnorm}(\mu[i], \tau);
  \]

  Note that we are indexing across the data here (not chaining!).

- Now define the variables in the model and their distributional assumptions:

  \[
  \alpha \sim \text{dnorm}(0.0, 0.001); \\
  \beta \sim \text{dnorm}(0.0, 0.001); \\
  \tau \sim \text{dgamma}(1, 0.1);
  \]

  The second normal parameter is a *precision* not a variance, by convention.
So the Full Model *In An ASCII File* Looks Like This

```r
model {
  for (i in 1:n) {
    mu[i] <- alpha + beta*x[i]
    y[i] ~ dnorm(mu[i], tau)
  }
  alpha ~ dnorm(0.0, 0.001)
  beta ~ dnorm(0.0, 0.001)
  tau ~ dgamma(1, 0.1)
}
```
And the Full Model in $R$ Looks Like This

```r
oecd.jags <- function() {
  for (i in 1:N) {
    mu[i] <- alpha + beta*x[i]
    y[i] ~ dnorm(mu[i],tau)
  }
  alpha ~ dnorm(0.0,0.001)
  beta ~ dnorm(0.0,0.001)
  tau ~ dgamma(1,0.1)
}
```
Running the Sampler from Within R

```r
lapply(c("rjags","arm","coda","superdiag","R2jags"),library, character.only=TRUE)

ocecd.inits <- function(){ list("alpha" = 0, "beta" = 0, "tau" = 1) }
ocecd.params <- c("alpha","beta","tau")

ocecd.model <- jags(oecd.jags, data=ocecd.dat, inits=ocecd.inits, parameters.to.save=ocecd.params, n.iter=50000)
```
Results

eocd.model

Inference for Bugs model at "/var/folders/...", fit using jags,
3 chains, each with 50000 iterations (first 25000 discarded), n.thin = 25
n.sims = 3000 iterations saved

mu.vect sd.vect 2.5% 25% 50% 75% 97.5% Rhat n.eff
alpha 0.859 0.323 0.203 0.652 0.860 1.078 1.480 1.001 3000
beta -0.349 0.124 -0.596 -0.430 -0.352 -0.267 -0.092 1.001 3000
tau 3.259 1.051 1.492 2.508 3.170 3.904 5.611 1.001 2700
deviance 32.054 2.444 29.328 30.315 31.424 33.079 38.290 1.002 1700

For each parameter, n.eff is a crude measure of effective sample size,
and Rhat is the potential scale reduction factor (at convergence, Rhat=1).

DIC info (using the rule, pD = var(deviance)/2) pD = 3.3 and DIC = 35.5
DIC is an estimate of expected predictive error (lower deviance is better).
Model Comparison

- Run for 50,000 iterations of the Markov chain, discard first 25,000.

- Using the posterior mean as a point estimate, we can compare with \texttt{lm}:

<table>
<thead>
<tr>
<th></th>
<th>OLS Model</th>
<th>MCMC Posterior</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Estimate</td>
<td>Std. Error</td>
</tr>
<tr>
<td>(Intercept)</td>
<td>0.859</td>
<td>0.317</td>
</tr>
<tr>
<td>Slope</td>
<td>-0.349</td>
<td>0.121</td>
</tr>
</tbody>
</table>

- Observations?
Linear Example

- The following general steps are given for a **JAGS** implementation where each command corresponds to a specific button in **WinBUGS**.

  ```
  model in "oecd.bug"
data in "oecd-data.R"
compile
inits in "oecd-init.R"
initialize
update 1000
monitor set alpha
monitor set beta
monitor set tau
update 50000
coda *
exi
  ```
A Hierarchical Model of Lobbying Influence in the US States


- The American State Administrator’s Project (ASAP) survey asks administrators about the influ¬
  ence of a variety of external political actors including “clientele groups” in their agencies.

- Clientele group is arguably not perfectly synonymous with interest group, but previous studies
  have used these terms interchangably (Kelleher and Yackee 2009).
A Hierarchical Model of Lobbying Influence in the US States

- Consider a $713 \times 22$ matrix $X$ with a leading column of 1’s for individual level explanatory variables, and a $50 \times 3$ matrix $Z$ for state-level explanatory variables.

- Our outcome variable (group influence) measures the respondents’ perception of interest groups’ influence on total budget, special budgets, and general public policies.

- We relate these variables through the linear hierarchical model:

  $$Y_i \sim N(\alpha_{j[i]} + \beta X_i, \sigma_y^2), \quad \text{for } i = 1, \ldots, 713$$

  $$\alpha_j \sim N(\gamma Z, \sigma_{\alpha}^2), \quad \text{for } j = 1, \ldots, 50,$$

  using hierarchical notation from Gelman and Hill (2007) to indicate that the $i$th respondent is nested in the $j$th state: $\alpha_{j[i]}$ to get a state-specific random intercept.

- This random intercept is the parameterized at the second level by the three explanatory variables in $Z$ and their corresponding estimated coefficients, $\gamma$. 
A Hierarchical Model of Lobbying Influence in the US States

- Note the two different variances accounted for in this specification. The term $\sigma_y^2$ measures the within-state variance of the outcome variables, whereas the term $\sigma_\alpha^2$ gives the variance of the mean estimates between-states.

- We choose to use informed versions of the prior distributions for the unknown parameters since a high-quality source exists.

- Our prior distributions are diffuse normals centered at the point estimates from Kelleher and Yackee (2009), Model 3 (page 593).

- We deviate from their values in only two ways: we substitute the variable elected/board for their variable Merit Position, and since we are using 2008 data only there cannot be a dummy variable for the year 2004.
A Hierarchical Model of Lobbying Influence in the US States

Thus:

\[ \beta \sim N(\beta_{ky}, \Sigma_\beta) \]
\[ \gamma \sim N(\gamma_{ky}, \Sigma_\gamma), \]

where the \( \Sigma_\beta \) and \( \Sigma_\gamma \) matrices are diagonal forms with large variances relative to the Kelleher and Yackee point estimates.

Get the data and code for this example at:
Using rjags from R (same model)

# DEFINE THE MODEL, NOTICE THE ASSIGNMENT STATEMENT
asap.model2.rjags <- function() {
  for (i in 1:SUBJECTS) {
    mu[i] <- alpha[state.id[i]]
    + beta[4]*elect.board[i] + beta[5]*years.tenure[i] + beta[6]*education[i]
    + beta[7]*party.ID[i] + beta[8]*category2[i] + beta[9]*category3[i]
    + beta[13]*category7[i] + beta[14]*category8[i] + beta[15]*category9[i]
    + beta[16]*category10[i] + beta[17]*category11[i] + beta[18]*category12[i]
    + beta[19]*med.time[i] + beta[20]*medt.contr[i]
    grp.influence[i] ~ dnorm(mu[i],tau.y)
  }
}
for (j in 1:STATES) {
  alpha[j] ~ dnorm(eta[j],tau.alpha)
}

beta[1] ~ dnorm(0.070,1); # PRIOR MEANS FROM KELLEHER AND YACKEE 2009, MODEL 3
beta[2] ~ dnorm(-0.054,1);
beta[3] ~ dnorm(0.139,1);
beta[4] ~ dnorm(0.051,1);
beta[5] ~ dnorm(0.017,1);
beta[6] ~ dnorm(0.056,1);
beta[7] ~ dnorm(0.039,1);
beta[8] ~ dnorm(0.0,1); # DIFFUSE PRIORS
beta[9] ~ dnorm(0.0,1);
beta[10] ~ dnorm(0.0,1);
beta[11] ~ dnorm(0.0,1);
beta[12] ~ dnorm(0.0,1);
beta[13] ~ dnorm(0.0,1);
beta[14] ~ dnorm(0.0,1);
\begin{verbatim}

beta[15] ~ dnorm(0.0,1);
beta[16] ~ dnorm(0.0,1);
beta[17] ~ dnorm(0.0,1);
beta[18] ~ dnorm(0.0,1);
beta[19] ~ dnorm(0.184,1);  # PRIOR MEANS FROM KELLEHER AND YACKEE 2009, MODEL 3
beta[20] ~ dnorm(0.156,1);
gamma[1] ~ dnorm(0.0,1);    # DIFFUSE PRIORS
gamma[2] ~ dnorm(0.0,1);
gamma[3] ~ dnorm(0.0,1);
tau.y ~ dgamma(1.0,1);
tau.alpha ~ dgamma(1.0,1);
\end{verbatim}
Using `rjags` from R, Data Handling

```r
# LOAD LIBRARIES
lapply(c("rjags","arm","coda","superdiag","R2jags"),library, character.only=TRUE)

# LOAD 2 DATA FILES (DIFFERENT SIZES DUE TO HIERARCHY)
asap.individual.data <- read.table("Article.JPART/asap.individual.dat",header=TRUE)
asap.group.data <- read.table("Article.JPART/asap.group.dat",header=TRUE)
```
Using `rjags` from R (same model)

# SAVE MODEL TO A FILE
```r
write.model(asap.model2.rjags, "Article.JPART/asap.model2.rjags")
```

# RUN THE SAMPLER AND COLLECT coda SAMPLES
```r
asap2.model <- jags(file="Article.JPART/asap.model2.rjags",
                    data=asap.jags.list, n.chains=3, n.adapt=1000)
update(asap2.model, n.iter=2500)
model.names <- c("beta","gamma","tau.y","tau.alpha")
asap2.mcmc <- coda.samples(model=asap2.model,
                           variable.names=model.names,n.iter=2500)
summary(asap2.mcmc)
```
A Hierarchical Model of Lobbying Influence in the US States, Results

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Post.Mean</th>
<th>Post.SE</th>
<th>95%.Lower.CI</th>
<th>95%.Upper.CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>α mean(1:50)</td>
<td>1.3905</td>
<td>0.7037</td>
<td>0.0112</td>
<td>2.7698</td>
</tr>
<tr>
<td>contracting</td>
<td>0.1987</td>
<td>0.0963</td>
<td>0.0099</td>
<td>0.3874</td>
</tr>
<tr>
<td>gov.influence</td>
<td>0.0481</td>
<td>0.0367</td>
<td>-0.0239</td>
<td>0.1202</td>
</tr>
<tr>
<td>leg.influence</td>
<td>0.3519</td>
<td>0.0397</td>
<td>0.2741</td>
<td>0.4297</td>
</tr>
<tr>
<td>elect.board</td>
<td>1.3436</td>
<td>0.3546</td>
<td>0.6486</td>
<td>2.0386</td>
</tr>
<tr>
<td>years.tenure</td>
<td>0.0347</td>
<td>0.0233</td>
<td>-0.0110</td>
<td>0.0804</td>
</tr>
<tr>
<td>education</td>
<td>0.1249</td>
<td>0.1217</td>
<td>-0.1136</td>
<td>0.3634</td>
</tr>
<tr>
<td>party.ID</td>
<td>-0.0046</td>
<td>0.0845</td>
<td>-0.1703</td>
<td>0.1611</td>
</tr>
</tbody>
</table>
A Hierarchical Model of Lobbying Influence in the US States, Results

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Post.Mean</th>
<th>Post.SE</th>
<th>95%.Lower.CI</th>
<th>95%.Upper.CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>category2</td>
<td>-0.4282</td>
<td>0.5423</td>
<td>-1.4912</td>
<td>0.6348</td>
</tr>
<tr>
<td>category3</td>
<td>-0.0596</td>
<td>0.5885</td>
<td>-1.2131</td>
<td>1.0938</td>
</tr>
<tr>
<td>category4</td>
<td>1.5501</td>
<td>0.4571</td>
<td>0.6541</td>
<td>2.4461</td>
</tr>
<tr>
<td>category5</td>
<td>-0.5473</td>
<td>0.5010</td>
<td>-1.5292</td>
<td>0.4347</td>
</tr>
<tr>
<td>category6</td>
<td>0.9227</td>
<td>0.5395</td>
<td>-0.1348</td>
<td>1.9801</td>
</tr>
<tr>
<td>category7</td>
<td>1.7014</td>
<td>0.4353</td>
<td>0.8482</td>
<td>2.5546</td>
</tr>
<tr>
<td>category8</td>
<td>1.0013</td>
<td>0.4986</td>
<td>0.0240</td>
<td>1.9785</td>
</tr>
<tr>
<td>category9</td>
<td>0.9412</td>
<td>0.4860</td>
<td>-0.0115</td>
<td>1.8938</td>
</tr>
<tr>
<td>category10</td>
<td>0.6157</td>
<td>0.4634</td>
<td>-0.2925</td>
<td>1.5239</td>
</tr>
<tr>
<td>category11</td>
<td>-0.1264</td>
<td>0.4265</td>
<td>-0.9624</td>
<td>0.7096</td>
</tr>
<tr>
<td>category12</td>
<td>-0.1592</td>
<td>0.5727</td>
<td>-1.2816</td>
<td>0.9632</td>
</tr>
</tbody>
</table>
A Hierarchical Model of Lobbying Influence in the US States, Results

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Post.Mean</th>
<th>Post.SE</th>
<th>95%.Lower.CI</th>
<th>95%.Upper.CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>med.time</td>
<td>1.1435</td>
<td>0.3587</td>
<td>0.4405</td>
<td>1.8465</td>
</tr>
<tr>
<td>medt.contr</td>
<td>-0.0869</td>
<td>0.1372</td>
<td>-0.3559</td>
<td>0.1821</td>
</tr>
<tr>
<td>gov.ideology</td>
<td>0.0182</td>
<td>0.0062</td>
<td>0.0060</td>
<td>0.0303</td>
</tr>
<tr>
<td>lobbyists</td>
<td>0.0007</td>
<td>0.0008</td>
<td>-0.0007</td>
<td>0.0022</td>
</tr>
<tr>
<td>nonprofits</td>
<td>-0.0217</td>
<td>0.1267</td>
<td>-0.2701</td>
<td>0.2266</td>
</tr>
<tr>
<td>τ.y</td>
<td>0.0763</td>
<td>0.0042</td>
<td>0.0682</td>
<td>0.0845</td>
</tr>
<tr>
<td>τα</td>
<td>3.1021</td>
<td>1.3523</td>
<td>0.4517</td>
<td>5.7525</td>
</tr>
</tbody>
</table>
A Hierarchical Model of Lobbying Influence in the US States, Convergence Plots
Using \texttt{rjags} from \texttt{R} (same model)

\begin{verbatim}
# CHECK CONVERGENCE
sink(asap2.diags)
superdiag(asap2.mcmc)
sink()

# GET THE DEVIANCE AND THE DIC
asap2.dic <- dic.samples(asap2.model, n.iter=2500, type="pD")
Mean deviance: 3861
penalty 34.2
Penalized deviance: 3895
\end{verbatim}
The Deviance Information Criterion is an alternative to AIC that explicitly accounts for the different role that parameters can play in a multilevel model (parent nodes, descendent nodes, etc.).


A rough approximation to theoretical out-of-sample predictive error.

Note: make sure the chains have converged first.

Two objectives: describing model complexity, giving model fit comparisons.

Calculated for you by WinBUGS and JAGS, so you do not have to do the hard work.
The Deviance Information Criterion (DIC)

- Model under consideration, defined by the likelihood \( p(y|\theta) \) for data \( y \) and parameter vector \( \theta \).

- A measure of “Bayesian deviance” is given by:

\[
D(\theta) = -2 \log[p(y|\theta)] + 2 \log[f(y)]
\]

where \( f(y) \) is some function of just the data, the “standardizing factor.”

- Note the similarity to the AIC:

\[
AIC = -2\ell(\hat{\theta}|y) + 2p
\]  

(1)

- Spiegelhalter et al. de-emphasize \( f(y) \) for model comparison and even suggest using \( f(y) = 1 \) (giving zero contribution above) since this term must be identical for both model calculations and therefore cancels out.

- So if you had a point estimate of \( \theta \), then \( D(\theta) \) would be the deviance from the saturated model.
The Deviance Information Criterion (DIC)

We can use the deviance in explicitly posterior terms by inserting a condition on the data and taking an expectation over $\theta$, to get the Expected Bayesian Deviance:

$$D(\theta) = \mathbb{E}_\theta[-2 \log[p(y|\theta)|y] + 2 \log[f(y)]$$

$$= -2 \int_\Theta (\log[p(y|\theta)] - \log[f(y)]) d\theta,$$

Now we do not need a single point estimate of $\theta$.

This is a posterior mean difference that now gives a measure of Bayesian model fit.

The conditioning on the data is easy, but the expectation might be hard analytically.

This is usually easy in an MCMC context since we get the expectation empirically from the samples.
The Deviance Information Criterion (DIC), $p_D$

- Define $\tilde{\theta}$ as any posterior estimate of $\theta$, which can be the posterior mean or some other easily produced value.

- We can also insert $\tilde{\theta}$ into $D(\theta)$ to get the “plug-in” deviance.

- The effective dimension of the model is now defined by:

$$p_D = D(\theta) - D(\tilde{\theta}),$$

where this is the “mean deviance minus the deviance of the means” computed by JAGS as the (variance of the deviance)/2 from MCMC samples.

- $p_D$ is the weighted sum of the parameters accounting for the hierarchy:
  - $\omega_p = 1$ for parameters unconstrained by prior information
  - $\omega = 0$ for parameters completely specified (fixed) by prior information
  - $\omega \in [0:1]$ for parameters with specific dependencies on the data or priors.

- The DIC is just $DIC = \overline{D(\theta)} + p_D = 2\overline{D(\theta)} - D(\tilde{\theta})$. 
Using `rjags` from R, Repeat Steps for Null Model

```r
asap.null.rjags <- function() {
  for (i in 1:SUBJECTS) {
    nu[i] <- alpha[state.id[i]] +
            beta[4]*elect.board[i] + beta[5]*years.tenure[i] + beta[6]*education[i] +
            beta[7]*party.ID[i] + beta[8]*category2[i] + beta[9]*category3[i] +
            beta[13]*category7[i] + beta[14]*category8[i] + beta[15]*category9[i] +
            beta[16]*category10[i] + beta[17]*category11[i] + beta[18]*category12[i] +
            beta[19]*med.time[i] + beta[20]*medt.contr[i]
    mu[i] <- alpha[state.id[i]]
    grp.influence[i] ~ dnorm(mu[i], tau.y)
  }
  for (j in 1:STATES) {
    alpha[j] ~ dnorm(0, tau.alpha)
  }
}
```
Using `rjags` from R (same model)

# SAVE MODEL TO A FILE
write.model(asap.null.rjags, "Article.JPART/asap.null.rjags")

# RUN THE SAMPLER AND COLLECT coda SAMPLES
asap.null.model <- jags.model(file="asap.null.rjags",
   data=asap.jags.list, n.chains=3, n.adapt=5000)
update(asap.null.model, n.iter=2500)
asap.null.mcmc <- coda.samples(model=asap.null.model,
   variable.names=model.names,n.iter=2500)

# CHECK CONVERGENCE
superdiag(asap.null.mcmc)

# GET THE DEVIANCE AND THE DIC
asap.null.dic <- dic.samples(asap.null.model, n.iter=2500, type="pD")
**Deviance Analysis, $\chi^2$ Statistics**

<table>
<thead>
<tr>
<th>Model</th>
<th>Deviance</th>
<th>Difference</th>
<th>DF</th>
<th>tail value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Null</td>
<td>3963.8</td>
<td>103.7</td>
<td>24</td>
<td>6.9787e-12</td>
</tr>
<tr>
<td>Estimated</td>
<td>3861.1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Saturated</td>
<td>0.0</td>
<td>3861.1</td>
<td>689</td>
<td>&lt;1.0e-300</td>
</tr>
</tbody>
</table>
Model: Education Policy

- General Problems/Challenges:
  - Controversy about the “education production function.”
  - A “bottom-less pit”?
  - Demographic constraints.

- Methodological Issues:
  - The data sucks.
  - Multilevel nature of the application.
  - Seemingly marginal contribution of theoretically important factors.
The Data

- 1998 STAR Program in California at the district level.

- Outcome Variable: ninth grade reading, $y_i$ successes out of $n_i$ trials.

- Explanatory Variables:
  
  Location Specification, $X_{n \times 7}$:
  - Percent Low Income
  - Percent Asian
  - Percent Black
  - Percent Hispanic
  - Mean Teacher Experience
  - Per-Pupil Spending
  - Percent UC/CSU.

Variance Specification, $Z_{n \times 3}$:
- Percent Minority Teachers
- Class Size
- Percent Year-Round.
Harvey Heteroscedastic Model Specification

stipulate:

\[ y_i \sim \text{binomial}(n_i, p_i) \]

\[ \logit(p_i) = \frac{X\beta}{e^{Z\gamma}} + b_i \]

\[ \beta_j \sim \mathcal{N}(0, k_j) \]
\[ \gamma_\ell \sim \mathcal{N}(0, k_\ell) \]
\[ b_i \sim \mathcal{N}(0, \tau) \]
\[ \tau \sim \text{gamma}(\delta_1, \delta_2) \]

meaning:

\[ P(y|n, p) = \prod_{i=1}^{N} \left( \frac{n_i}{y_i} \right) p_i^{y_i} (1 - p_i)^{n_i-y_i} \]

\[ P(b|\tau) = \prod_{i=1}^{N} (2\pi\tau)^{-\frac{1}{2}}\exp\left[ -\frac{1}{2\tau} b_i^2 \right] \]

\[ P(\beta_j|k_j) = (2\pi k_j)^{-\frac{1}{2}}\exp\left[ -\frac{1}{2k_j} \beta_j^2 \right] \]

\[ P(\gamma_\ell|k_\ell) = (2\pi k_\ell)^{-\frac{1}{2}}\exp\left[ -\frac{1}{2k_\ell} \gamma_\ell^2 \right] \]

\[ P(\tau|\delta_1, \delta_2) = \frac{1}{\Gamma(\delta_1)\delta_2^{\delta_1}}\tau^{\delta_1-1}\exp\left[ -\frac{1}{\delta_2} \tau \right] \]
Obtaining the Posterior

\[
P(p, b, \tau | y, n, \delta_1, \delta_2) \propto P(y | n, p)P(b | \tau)P(\tau | \delta_1, \delta_2)
\]

\[
= (2\pi \tau)^{-\frac{N}{2}} \prod_{i=1}^{N} \frac{\Gamma(n_i + 1)}{\Gamma(n_i - y_i + 1) \Gamma(y_i + 1) \Gamma(\delta_1)} \times p_i^{y_i} (1 - p_i)^{n_i - y_i} \delta_2^{-\delta_1 \tau} \delta_1^{-1} \\
\times \exp \left[ -\frac{1}{2\tau} b_i^2 - \frac{1}{\delta_2} \right] \\
= (2\pi \tau)^{-\frac{N}{2}} \prod_{i=1}^{N} \frac{\Gamma(n_i + 1)}{\Gamma(n_i - y_i + 1) \Gamma(y_i + 1) \Gamma(\delta_1)} \\
\times (1 + \exp(X_i \beta/\exp(Z_i \gamma + b))^{-1})^{y_i} \\
\times (1 - (1 + \exp(X_i \beta/\exp(Z_i \gamma + b))^{-1}))^{n_i - y_i} \\
\times \delta_2^{-\delta_1 \tau} \delta_1^{-1} \exp \left[ -\frac{1}{2\tau} b_i^2 - \frac{1}{\delta_2} \tau - \sum_{j=1}^{8} \left( \frac{1}{2k_j} \right) \beta_j^2 - \sum_{\ell=1}^{3} \left( \frac{1}{k_\ell} \gamma_\ell^2 \right) \right]
\]
Obtaining the Education Policy Model Posterior Through Simulation: Gibbs Sampling

► PARAMETERS:

▷ Define $\phi = [\boldsymbol{\beta}, \gamma, \tau]$ as the $12 \times 1$ vector of unknown parameters ($b$ a nuisance parameter).

▷ Chain values use $\phi_{-i}^k$: removing the $i^{th}$ parameter from $I = 12$ at the $k^{th}$ MCMC step.

► STEPS:

1. Specify $\phi^{[0]} = [\boldsymbol{\beta}_0, \gamma_0, \tau_0]$.

2. At the $k^{th}$ step draw $\phi_i|\phi_{-i}$ according to:

\[
\begin{align*}
\phi_1^{[k]} &\sim \pi(\beta_1|\beta_2^{[k-1]}, \beta_3^{[k-1]}, \beta_4^{[k-1]}, \beta_5^{[k-1]}, \beta_6^{[k-1]}, \beta_7^{[k-1]}, \beta_8^{[k-1]}, \gamma_1^{[k-1]}, \gamma_2^{[k-1]}, \gamma_3^{[k-1]}, \tau^{[k-1]}) \\
\phi_2^{[k]} &\sim \pi(\beta_2|\beta_2^{[k]}, \beta_3^{[k-1]}, \beta_4^{[k-1]}, \beta_5^{[k-1]}, \beta_6^{[k-1]}, \beta_7^{[k-1]}, \beta_8^{[k-1]}, \gamma_1^{[k-1]}, \gamma_2^{[k-1]}, \gamma_3^{[k-1]}, \tau^{[k-1]}) \\
\phi_3^{[k]} &\sim \pi(\beta_3|\beta_2^{[k]}, \beta_3^{[k-1]}, \beta_4^{[k]}, \beta_5^{[k-1]}, \beta_6^{[k-1]}, \beta_7^{[k-1]}, \beta_8^{[k-1]}, \gamma_1^{[k-1]}, \gamma_2^{[k-1]}, \gamma_3^{[k-1]}, \tau^{[k-1]}) \\
\phi_4^{[k]} &\sim \pi(\beta_4|\beta_2^{[k]}, \beta_3^{[k-1]}, \beta_4^{[k-1]}, \beta_5^{[k]}, \beta_6^{[k-1]}, \beta_7^{[k-1]}, \beta_8^{[k-1]}, \gamma_1^{[k-1]}, \gamma_2^{[k-1]}, \gamma_3^{[k-1]}, \tau^{[k-1]}) \\
\phi_5^{[k]} &\sim \pi(\beta_5|\beta_2^{[k]}, \beta_3^{[k-1]}, \beta_4^{[k-1]}, \beta_5^{[k-1]}, \beta_6^{[k]}, \beta_7^{[k-1]}, \beta_8^{[k-1]}, \gamma_1^{[k-1]}, \gamma_2^{[k-1]}, \gamma_3^{[k-1]}, \tau^{[k-1]}) \\
\phi_6^{[k]} &\sim \pi(\beta_6|\beta_2^{[k]}, \beta_3^{[k-1]}, \beta_4^{[k-1]}, \beta_5^{[k-1]}, \beta_6^{[k-1]}, \beta_7^{[k-1]}, \beta_8^{[k-1]}, \gamma_1^{[k-1]}, \gamma_2^{[k-1]}, \gamma_3^{[k-1]}, \tau^{[k-1]}) \\
\phi_7^{[k]} &\sim \pi(\beta_7|\beta_2^{[k]}, \beta_3^{[k-1]}, \beta_4^{[k-1]}, \beta_5^{[k-1]}, \beta_6^{[k-1]}, \beta_7^{[k]}, \beta_8^{[k-1]}, \gamma_1^{[k-1]}, \gamma_2^{[k-1]}, \gamma_3^{[k-1]}, \tau^{[k-1]}) \\
\phi_8^{[k]} &\sim \pi(\beta_8|\beta_2^{[k]}, \beta_3^{[k-1]}, \beta_4^{[k-1]}, \beta_5^{[k-1]}, \beta_6^{[k-1]}, \beta_7^{[k-1]}, \beta_8^{[k]}, \gamma_1^{[k-1]}, \gamma_2^{[k-1]}, \gamma_3^{[k-1]}, \tau^{[k-1]}) \\
\phi_9^{[k]} &\sim \pi(\gamma_1|\beta_2^{[k]}, \beta_3^{[k-1]}, \beta_4^{[k-1]}, \beta_5^{[k-1]}, \beta_6^{[k-1]}, \beta_7^{[k-1]}, \beta_8^{[k-1]}, \gamma_1^{[k-1]}, \gamma_2^{[k-1]}, \gamma_3^{[k-1]}, \tau^{[k-1]}) \\
\phi_{10}^{[k]} &\sim \pi(\gamma_2|\beta_2^{[k]}, \beta_3^{[k-1]}, \beta_4^{[k-1]}, \beta_5^{[k-1]}, \beta_6^{[k-1]}, \beta_7^{[k-1]}, \beta_8^{[k-1]}, \gamma_1^{[k-1]}, \gamma_2^{[k-1]}, \gamma_3^{[k-1]}, \tau^{[k-1]}) \\
\phi_{11}^{[k]} &\sim \pi(\gamma_3|\beta_2^{[k]}, \beta_3^{[k-1]}, \beta_4^{[k-1]}, \beta_5^{[k-1]}, \beta_6^{[k-1]}, \beta_7^{[k-1]}, \beta_8^{[k-1]}, \gamma_1^{[k-1]}, \gamma_2^{[k-1]}, \gamma_3^{[k-1]}, \tau^{[k-1]}) \\
\phi_{12}^{[k]} &\sim \pi(\tau|\beta_2^{[k]}, \beta_3^{[k-1]}, \beta_4^{[k-1]}, \beta_5^{[k-1]}, \beta_6^{[k-1]}, \beta_7^{[k-1]}, \beta_8^{[k-1]}, \gamma_1^{[k-1]}, \gamma_2^{[k-1]}, \gamma_3^{[k-1]}, \tau^{[k-1]}) 
\end{align*}
\]

3. Repeat: 10,000 times for “burn-in,” 40,000 times for convergence.
JAGS Code

star.model <- function()
{
  for (j in 1:K) {
    phi[j] ~ dnorm(0.0, 1.0E-6)I(-4,4);
  }
  tau ~ dgamma(1.0E-3, 1.0E-3);  # 1/sigma^2
  sigma <- 1.0/sqrt(tau);
  for (i in 1:N) {
    b[i] ~ dnorm(0.0, tau);
    logit(p[i]) <- (phi[1] + phi[2]*LOWINC[i] + phi[3]*PERASIAN[i]
                   + phi[4]*PERBLACK[i] + phi[5]*PERHISP[i]
                   + phi[6]*AVYRSEXP[i] + phi[7]*PERSPEN[i]
                   + phi[8]*PCTAF[i])/exp(1 + phi[9]*PERMI[i]
                   + phi[10]*PTRATIO[i] + phi[11]*PCTYRRND[i]) + b[i];
    READPASS[i] ~ dbin(p[i], READTOT[i]);
  }
}

### Model Results

<table>
<thead>
<tr>
<th></th>
<th>95% HPD Regions</th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Untreated</td>
<td>SIR and Untreated Rao-Blackwellized</td>
<td>Geweke Convergence Diagnostic</td>
<td></td>
</tr>
<tr>
<td>( \tau )</td>
<td>[0.001: 0.002]</td>
<td>[0.000: 0.001]</td>
<td></td>
<td>-2.338</td>
</tr>
<tr>
<td>( \beta ) Intercept</td>
<td>[5.804: 8.584]</td>
<td>[5.836: 8.388]</td>
<td></td>
<td>-1.698</td>
</tr>
<tr>
<td>Percent Asian</td>
<td>[0.097: 0.139]</td>
<td>[0.114: 0.127]</td>
<td></td>
<td>0.860</td>
</tr>
<tr>
<td>Percent Black</td>
<td>[-0.008: 0.063]</td>
<td>[-0.055:-0.003]</td>
<td></td>
<td>1.567</td>
</tr>
<tr>
<td>Percent Hispanic</td>
<td>[-0.154:-0.062]</td>
<td>[-0.144:-0.069]</td>
<td></td>
<td>1.773</td>
</tr>
<tr>
<td>Mean Teacher Experience</td>
<td>[0.033: 0.037]</td>
<td>[0.034: 0.037]</td>
<td></td>
<td>2.296</td>
</tr>
<tr>
<td>Per-Pupil Spending</td>
<td>[-0.096: 0.133]</td>
<td>[-0.008: 0.028]</td>
<td></td>
<td>-1.201</td>
</tr>
<tr>
<td>Percent UC/CSU</td>
<td>[0.014: 0.017]</td>
<td>[0.015: 0.016]</td>
<td></td>
<td>-0.515</td>
</tr>
<tr>
<td>( \gamma ) Percent Minority Teachers</td>
<td>[0.069: 0.102]</td>
<td>[0.077: 0.093]</td>
<td></td>
<td>-0.988</td>
</tr>
<tr>
<td>Class Size</td>
<td>[0.001: 0.001]</td>
<td>[0.001: 0.001]</td>
<td></td>
<td>-0.177</td>
</tr>
<tr>
<td>Percent Year-Round</td>
<td>[-0.003: 0.045]</td>
<td>[0.015: 0.028]</td>
<td></td>
<td>-1.217</td>
</tr>
</tbody>
</table>
R Function

mcmc.table <- function(n.chains=1,burnin=1/2,alpha=0.05) {
  lapply(c("coda","xtable"),library, character.only=TRUE)
  print(paste("current working directory: ",getwd()))
  for (i in 1:n.chains) {
    full.out <- read.coda(paste("CODAchain",i,".txt",sep=""),
      "CODAindex.txt",quiet=TRUE)
    start <- nrow(full.out)*burnin; stop <- nrow(full.out)
    chain.mean <- apply(full.out[start:stop,],2,mean)
    chain.se <- apply(full.out[start:stop,],2,sd)
    full.tab <- cbind(chain.mean,chain.se,
      chain.mean-qnorm(1-alpha/2)*chain.se,
      chain.mean-qnorm(1-alpha/2)*chain.se)
    print(xtable(full.tab,digits=4))
  }
}
The 1960 U.S. presidential election between John Kennedy and Richard Nixon was one of the closest contests in national history, with Kennedy’s margin of victory less than one percent of the popular vote.

This example uses the 1960 American National Election Study (subsetted in the R library BaM) to explore the link between personal characteristics and perception of closeness of the impending election.

The outcome variable has four ordered categories: [one candidate will win by a lot], [one candidate will win by quite a bit], [this will be a close race–fairly even], [this will be a very close race].
The specified explanatory variables are:

- **education**, 1=8th grade or lower (233 cases), 2=highschool (428 cases), 3=some college or more (185 cases).
- **sex**, 1=male (412 cases), 2=female (434 cases).
- **seedebates**, 1=no (141 cases), 2=yes (660 cases).
- **importance**, 1=care very much (270 cases), 2=care pretty much (308 cases), 3=pro-con/depends (6 cases), 4=don’t care very much (155 cases), 5=don’t care at all (85 cases).
- **involvement**, 8 categories from low to high with the distribution of cases: (158, 165, 245, 101, 78, 55, 4, 39).
- **catholic**, 0=no (623 cases), 1=yes (179 cases).
- **partyid**, 1=strong Democrat (198 cases), 2=not very strong Democrat (201 cases), 3=indepen-dent closer to Democrats (52 cases), 4=independent (67 cases), 5=independent closer to Republicans (59 cases), 6=not very strong Republican (117 cases), 7=strong Republican (139 cases).
anes <- function() {
  for (i in 1:N) {
    mu[i] <- theta[1]*education[i] + theta[2]*sex[i] + theta[3]*seede debates[i]
    + theta[4]*importance[i] + theta[5]*involvement[i] + theta[6]*importance.involvement
    + theta[7]*catholic[i] + theta[8]*partyid[i]
    ### LIKE R: logit P(Y <= k|x) = zeta_k - X*beta ###
    for (j in 1:(Ncat-1)) { logit(Q[i,j]) <- cut[j] - mu[i] }
    p[i,1] <- Q[i,1]
    for (j in 2:(Ncat-1)) { p[i,j] <- Q[i,j] - Q[i,(j-1)] }
    p[i,Ncat] <- 1 - Q[i,(Ncat-1)]
    close[i] ~ dcat(p[i,1:Ncat])
    E.y[i] <- close[i] - mu[i]
  }
}
American National Election Study, Code

sd.y <- sd(E.y[])
for (k in 1:Nvar) { theta[k] ~ dt(0,1,5) }
for (k in 1:(Ncat-1)) { cut[k] ~ dt(0,1,5) }
cut.sort[1:(Ncat-1)] <- sort(cut)
ANES MODEL

# SETUP
lapply(c("MASS","rjags","R2jags","R2WinBUGS","coda","superdiag"),library,
       character.only=TRUE)

# GET FILE FROM https://pages.wustl.edu/jgill/courses FIRST
source("writeDatafileR.R")

# JAGS DATA
anes.dat <- read.table("Book.Bayes.III/Example.ANES/anes.clean.dat")
anes.dat <- list("education"=anes.df[,5], "sex"=anes.df[,15], "seedebates"=anes.df[,7],
                 "importance"=anes.df[,12], "involvement"=anes.df[,9], "catholic"=anes.df[,22],
                 "partyid"=anes.df[,13], "close"=anes.df[,11],
                 "importance.involvement"=anes.df[,12]*anes.df[,9], "N"=846, "Ncat"=4, "Nvar" = 8)
# SETUP AND MODEL

anes.inits <- function(){ list("cut" = c(-5,0,5), "theta" = c(1,1,1,1,0,0,0,0)) }
anes.params <- c("cut","theta")
anes.out <- jags(anes, data=anes.dat, inits=anes.inits, parameters.to.save=anes.params,
                 n.burnin = 10000, n.iter=50000, n.chains=3, n.thin=1)
anes.mcmc <- as.mcmc(anes.out)
superdiag(anes.mcmc)
ANES Model Results

anes.out
model3a603690c52a.txt", fit using jags,
3 chains, each with 50000 iterations (first 10000 discarded)
n.sims = 120000 iterations saved

<table>
<thead>
<tr>
<th></th>
<th>mu.vect</th>
<th>sd.vect</th>
<th>2.5%</th>
<th>25%</th>
<th>50%</th>
<th>75%</th>
<th>97.5%</th>
<th>Rhat</th>
<th>n.eff</th>
</tr>
</thead>
<tbody>
<tr>
<td>cut[1]</td>
<td>-1.840</td>
<td>0.492</td>
<td>-2.818</td>
<td>-2.169</td>
<td>-1.837</td>
<td>-1.507</td>
<td>-0.886</td>
<td>1.002</td>
<td>1900</td>
</tr>
<tr>
<td>cut[2]</td>
<td>-0.418</td>
<td>0.475</td>
<td>-1.357</td>
<td>-0.739</td>
<td>-0.412</td>
<td>-0.098</td>
<td>0.502</td>
<td>1.002</td>
<td>1800</td>
</tr>
<tr>
<td>theta[1]</td>
<td>0.288</td>
<td>0.106</td>
<td>0.083</td>
<td>0.216</td>
<td>0.287</td>
<td>0.359</td>
<td>0.497</td>
<td>1.002</td>
<td>2300</td>
</tr>
<tr>
<td>theta[2]</td>
<td>0.370</td>
<td>0.141</td>
<td>0.096</td>
<td>0.275</td>
<td>0.369</td>
<td>0.464</td>
<td>0.650</td>
<td>1.001</td>
<td>20000</td>
</tr>
<tr>
<td>theta[3]</td>
<td>0.402</td>
<td>0.177</td>
<td>0.059</td>
<td>0.283</td>
<td>0.401</td>
<td>0.520</td>
<td>0.755</td>
<td>1.002</td>
<td>1400</td>
</tr>
<tr>
<td>theta[4]</td>
<td>-0.058</td>
<td>0.121</td>
<td>-0.291</td>
<td>-0.139</td>
<td>-0.059</td>
<td>0.023</td>
<td>0.181</td>
<td>1.004</td>
<td>680</td>
</tr>
<tr>
<td>theta[5]</td>
<td>-0.310</td>
<td>0.105</td>
<td>-0.514</td>
<td>-0.380</td>
<td>-0.311</td>
<td>-0.240</td>
<td>-0.105</td>
<td>1.005</td>
<td>470</td>
</tr>
<tr>
<td>theta[6]</td>
<td>0.056</td>
<td>0.028</td>
<td>0.001</td>
<td>0.038</td>
<td>0.057</td>
<td>0.075</td>
<td>0.110</td>
<td>1.006</td>
<td>400</td>
</tr>
<tr>
<td>theta[7]</td>
<td>0.213</td>
<td>0.174</td>
<td>-0.128</td>
<td>0.096</td>
<td>0.213</td>
<td>0.330</td>
<td>0.557</td>
<td>1.001</td>
<td>39000</td>
</tr>
</tbody>
</table>
For each parameter, n.eff is a crude measure of effective sample size, and Rhat is the potential scale reduction factor (at convergence, Rhat=1).

DIC info (using the rule, pD = var(deviance)/2)
pD = 10.7 and DIC = 1593.2
DIC is an estimate of expected predictive error (lower deviance is better).
Comments on Burn-in

- The burn-in period is the initial time that is considered to be pre-stationarity.

- We run the chain for some time after the starting point and throw away the values.

- Convergence assessment is essential (diagnostics to come).

- It pays to be conservative in deciding the length of the burn-in period.

- There is no “golden rule” here.
MCMC Convergence

- Empirical summaries from a given MCMC analysis are not reliable until the chain has reached its stationary distribution and had time to sufficiently mix throughout.

- Until $X_t$ converges at time $t$ ($X \sim \pi$), it is not possible to rely upon the effect of any variant of the CLT.

- Therefore it is necessary to convince yourself and your readers that the Markov chain has mixed throughout its stationary distribution.
MCMC Convergence (cont.)

Three primary problems:

▷ There is no *general* proof of convergence for the Gibbs sampler or the more broad Metropolis-Hastings algorithm.

▷ There is no way to guarantee that a Markov chain will explore all of the areas of the target distribution in finite time.

▷ For a given Markov chain at a given time $t$, there is no absolute assurance that the chain is currently in the target distribution.

And Gelman (1996) gives three additional mechanical level worries: an inappropriately specified model, errors in programming the Markov chain (stationary distribution of the chain may not be the desired target), and slow convergence. Markov chain has mixed throughout its stationary distribution.
The good news:

- All ergodic Markov chains are guaranteed to converge asymptotically.
- It is often quite easy to determine if a given chain has \textit{not} converged.
- We have lots of tests.
MCMC Convergence (cont.)

- Convergence monitoring:
  - Stop the chain intermittently and apply empirical diagnostics for nonconvergence.
  - A somewhat subjective, ad-hoc process.

- **Important principle**: these are indicators of nonconvergence. So failing to find evidence of nonconvergence with these procedures is not evidence of convergence.

- Careful practitioners will use more than one tool here.
Correlation and Autocorrelation

- High correlation *between* the parameters of a chain tends to give slow convergence.

- Whereas high correlation within a single parameter (autocorrelation) chain leads to slow mixing and possibly individual *nonconvergence* to the limiting distribution.
Traceplots and Autocorrelations

Traceplots

Autocorrelations

alpha.mu

beta.mu

tau.alpha

tau.beta

tau.c

0K 20K 40K 60K 80K 100K 0 10 20 30 40 50
Correlation and Autocorrelation (cont.)

▶ Notes:

▷ Traceplots are more informative if the burn-in period is omitted.

▷ Things to look for in traceplots: *trends* and *snaking*.

▷ Autocorrelation: or a series of length $n$, the lag $k$ autocorrelation is the sum of $n - k$ correlations according to:

$$\rho_k = \sum_{i=1}^{n-k} (x_i - \bar{x})(x_{i+k} - \bar{x}) / \sum_{i=1}^{n} (x_i - \bar{x})^2$$

▷ Usually its not necessary to look beyond lag 50.

▷ Autocorrelation problems can often be helped with reparameterization.
Correlation and Autocorrelation with \texttt{coda}

\begin{verbatim}
library(coda)
codamenu()  
CODA startup menu

1: Read BUGS output files
2: Use an mcmc object
3: Quit

Selection: 2

Enter name of saved object (or type "exit" to quit)
1:
anes.mcmc

Checking effective sample size ... OK
\end{verbatim}
Correlation and Autocorrelation with coda

CODA Main Menu
1: Output Analysis
2: Diagnostics
3: List/Change Options
4: Quit

Selection: 2

CODA Diagnostics Menu

1: Geweke
2: Gelman and Rubin
3: Raftery and Lewis
4: Heidelberger and Welch
5: Autocorrelations
6: Cross-Correlations
7: List/Change Options
8: Return to Main Menu
Correlation and Autocorrelation with coda

Selection: 5

AUTOCORRELATIONS WITHIN EACH CHAIN:

===================================

Iterations used = 10001:50000

Thinning interval = 1

Sample size per chain = 40000

, , theta[8]


<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0.218</td>
<td>0.227</td>
<td>0.228</td>
<td>0.0170</td>
<td>-0.1428</td>
<td>-0.04112</td>
<td>0.0357</td>
<td>-0.00842</td>
<td>0.0133</td>
<td>0.0316</td>
<td>0.2478</td>
<td>1.0000</td>
</tr>
<tr>
<td>1</td>
<td>0.217</td>
<td>0.228</td>
<td>0.229</td>
<td>0.0196</td>
<td>-0.1373</td>
<td>-0.03488</td>
<td>0.0528</td>
<td>-0.00223</td>
<td>0.0307</td>
<td>0.0576</td>
<td>0.3165</td>
<td>0.8021</td>
</tr>
<tr>
<td>5</td>
<td>0.226</td>
<td>0.236</td>
<td>0.231</td>
<td>0.0196</td>
<td>-0.0988</td>
<td>0.00627</td>
<td>0.1460</td>
<td>0.01729</td>
<td>0.0815</td>
<td>0.0213</td>
<td>0.1650</td>
<td>0.3619</td>
</tr>
<tr>
<td>10</td>
<td>0.224</td>
<td>0.234</td>
<td>0.232</td>
<td>0.0107</td>
<td>-0.0549</td>
<td>0.04995</td>
<td>0.1961</td>
<td>0.03637</td>
<td>0.1079</td>
<td>-0.0514</td>
<td>0.0445</td>
<td>0.1388</td>
</tr>
<tr>
<td>50</td>
<td>0.148</td>
<td>0.153</td>
<td>0.151</td>
<td>-0.0042</td>
<td>0.0413</td>
<td>0.06040</td>
<td>0.0893</td>
<td>0.05515</td>
<td>0.0540</td>
<td>-0.0558</td>
<td>0.0093</td>
<td>0.0191</td>
</tr>
</tbody>
</table>
Correlation and Autocorrelation with \textit{coda}

Autocorrelation Plots Menu
1: Plot autocorrelations
2: Return to Diagnostics Menu

Selection: 1
Hit <Return> to see next plot:
Other Graphical Diagnostics

- Running mean smoother: run after burn-in, look for trends
- Smoothed density estimates: pick some later period. Here last 1000 values for smoother compared with HPD for full 100,000...
Geweke Time-Series Diagnostic

- General idea: test based on comparing some proportion of the early era of the chain after the burn-in period with some nonoverlapping proportion of the late era of the chain.

- This suggest a difference of means test using an asymptotic approximation for the standard error of the difference.

- Values that are atypical of a standard normal distribution provide evidence that the two selected portions of the chain differ reasonably (in the first moment).
Geweke Time-Series Diagnostic (cont.)

- Preselect two nonoverlapping window proportions, one early in the chain and one later in the chain: \( \theta_1 \) of length \( n_1 \) and \( \theta_2 \) of length \( n_2 \) along with some function of interest \( g() \).

- Diagnostic is given by:

\[
G = \frac{\bar{g}(\theta_1) - \bar{g}(\theta_2)}{\sqrt{\frac{s_1(0)}{n_1} + \frac{s_2(0)}{n_2}}}.
\]

and \( s_1(0) \) and \( s_2(0) \) are standard errors from the symmetric spectral density functions: the uncorrelated contribution from the individual values to the total variance.

- Typical test is for \( g() \) to be the mean:

\[
\bar{g}(\theta_1) = \frac{1}{n_1} \sum_{i=1}^{n_1} g(\theta_i)/n,
\]

\[
\bar{g}(\theta_2) = \frac{1}{n_2} \sum_{i=1}^{n_2} g(\theta_i)/n,
\]
Geweke Time-Series Diagnostic (cont.)

- Geweke suggests using the ratios $n_1/n = 0.1$ and $n_2/n = 0.5$.

- If these proportions are held fixed as the chain grows in length, then the central limit theorem applies and $G$ converges in distribution to standard normal.

- Geweke’s idea is that more can be learned by one very long chain since it will end up exploring areas where humans might not think to send it.

- Unfortunately the window proportions can greatly affect the value of $G$, and it is therefore important not to only use the defaults: $(0.1/0.5)$. 
### Geweke Diagnostic for the ANES Model (from superdiag)

********** The Geweke diagnostic: **********

#### Z-scores:

<table>
<thead>
<tr>
<th></th>
<th>chain1</th>
<th>chain 2</th>
<th>chain 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>cut[1]</td>
<td>1.9330</td>
<td>-0.6608</td>
<td>0.2307</td>
</tr>
<tr>
<td>cut[2]</td>
<td>1.9192</td>
<td>-0.5751</td>
<td>0.2424</td>
</tr>
<tr>
<td>cut[3]</td>
<td>2.1074</td>
<td>-0.5900</td>
<td>0.2738</td>
</tr>
<tr>
<td>deviance</td>
<td>0.4164</td>
<td>-1.3127</td>
<td>0.4130</td>
</tr>
<tr>
<td>theta[1]</td>
<td>2.5207</td>
<td>-1.3391</td>
<td>-0.6455</td>
</tr>
<tr>
<td>theta[2]</td>
<td>1.5526</td>
<td>1.3243</td>
<td>0.8913</td>
</tr>
<tr>
<td>theta[3]</td>
<td>0.9665</td>
<td>0.4562</td>
<td>0.9086</td>
</tr>
<tr>
<td>theta[4]</td>
<td>1.3304</td>
<td>-1.6041</td>
<td>-0.2237</td>
</tr>
<tr>
<td>theta[5]</td>
<td>1.7348</td>
<td>-1.8337</td>
<td>-0.0839</td>
</tr>
<tr>
<td>theta[6]</td>
<td>-1.3182</td>
<td>1.5621</td>
<td>0.1016</td>
</tr>
<tr>
<td>theta[7]</td>
<td>-0.6710</td>
<td>0.4272</td>
<td>-1.1397</td>
</tr>
<tr>
<td>theta[8]</td>
<td>-0.8419</td>
<td>-0.3750</td>
<td>-1.1057</td>
</tr>
</tbody>
</table>

**Window From Start** 0.1000 0.8230 0.6473  
**Window From Stop** 0.5000 0.1592 0.3328
Gelman and Rubin’s Multiple Sequence Diagnostic

- An ANOVA type test based on multiple chains.

- Steps:

1. Run $m \geq 2$ chains of length $2n$ from overdispersed starting points, $(1)$, $(2)$, \ldots, $(m)$:

$$
\theta_{(1)}^{[0]}, \theta_{(1)}^{[1]}, \ldots, \theta_{(1)}^{[2n-1]}, \theta_{(1)}^{[2n]},
\theta_{(2)}^{[0]}, \theta_{(2)}^{[1]}, \ldots, \theta_{(2)}^{[2n-1]}, \theta_{(2)}^{[2n]},
\vdots
\theta_{(m)}^{[0]}, \theta_{(m)}^{[1]}, \ldots, \theta_{(m)}^{[2n-1]}, \theta_{(m)}^{[2n]},
$$

The starting points should be determined by overdispersing around suspected or known modes. Discard the first $n$ chain iterations.
Continuing steps:

2. For each parameter of interest calculate the following:

   • **Within chain variance:**

     \[
     W = \frac{1}{m(n-1)} \sum_{j=1}^{m} \sum_{i=1}^{n} (\hat{\theta}^{[i]}_{(j)} - \bar{\theta}_{(j)})^2
     \]

     where \( \bar{\theta}_{(j)} \) is the mean of the \( n \) values for the \( j^{th} \) chain.

   • **Between chain variance:**

     \[
     B = \frac{n}{m-1} \sum_{j=1}^{m} (\bar{\theta}_{(j)} - \bar{\theta})^2
     \]

     where \( \bar{\theta} \) is the grand mean (mean of means since each subchain is of equal length).

   • **Estimated variance:**

     \[
     \hat{Var}(\theta) = (1 - 1/n)W + (1/n)B.
     \]
Gelman and Rubin’s Multiple Sequence Diagnostic (cont.)

Continuing steps:

3. Calculate the convergence diagnostic, a single scalar value, called the *estimated scale reduction* (or shrink factor):

\[
\sqrt{\hat{R}} = \sqrt{\frac{\hat{\text{Var}}(\theta)}{W}}.
\]

Values of \(\sqrt{\hat{R}}\) near 1 are evidence that the \(m\) chains are all operating on the same distribution (in practice values less than roughly 1.1 or 1.2 are acceptable).

Unless one is very worried about multimodality of the prior, the number of separate chains need only be about 5 to 10.

It is easy to monitor \(\sqrt{\hat{R}}\) as the Markov chain runs and move on to other diagnostics when we are happy.
Gelman and Rubin’s Multiple Sequence Diagnostic Notes

- Underlying principle:
  - Before convergence $W$ underestimates total posterior variation in $\theta$ because the chains have not fully explored the target distribution.
  - $W$ is therefore based on smaller differences early in the chain.
  - Also before convergence $\hat{V}ar(\theta)$ overestimates total posterior variance because the starting points are intentionally overdispersed relative to the target.
  - Once the chains have converged, the difference should be incidental since the chains are exploring the same region and are therefore overlapping.
Problems/Challenges:

- It is not always easy to obtain suitably overdispersed starting points, since determining their position requires some knowledge of the target distribution to begin with.

- This is especially a problem in higher dimensions.

- Actually this point is critical since the test relies upon the underdispersed/overdispersed distributional contrast. EM can help!

- G&R also suggest starting with a mixture of normals centered at known nodes and improving the quality of density estimate with importance sampling.

- Normal assumptions may not always be supportable (somewhat mitigated by the Brooks-Gelman modification [1998]; accommodating finite-sample variability in the variance calculations for $B$ and $W$) by calculating $\sqrt{\hat{R}}$ for $\alpha$ trimmed intervals.
## Gelman and Rubin Diagnostic for the ANES Model (from superdiag)

********** The Gelman-Rubin diagnostic: **********

Potential scale reduction factors:

<table>
<thead>
<tr>
<th></th>
<th>Point est.</th>
<th>Upper C.I.</th>
</tr>
</thead>
<tbody>
<tr>
<td>cut[1]</td>
<td>1.01</td>
<td>1.02</td>
</tr>
<tr>
<td>cut[2]</td>
<td>1.01</td>
<td>1.03</td>
</tr>
<tr>
<td>cut[3]</td>
<td>1.01</td>
<td>1.02</td>
</tr>
<tr>
<td>deviance</td>
<td>1.00</td>
<td>1.00</td>
</tr>
<tr>
<td>theta[1]</td>
<td>1.00</td>
<td>1.01</td>
</tr>
<tr>
<td>theta[2]</td>
<td>1.00</td>
<td>1.01</td>
</tr>
<tr>
<td>theta[3]</td>
<td>1.01</td>
<td>1.03</td>
</tr>
<tr>
<td>theta[4]</td>
<td>1.01</td>
<td>1.05</td>
</tr>
<tr>
<td>theta[5]</td>
<td>1.03</td>
<td>1.09</td>
</tr>
<tr>
<td>theta[6]</td>
<td>1.03</td>
<td>1.10</td>
</tr>
<tr>
<td>theta[7]</td>
<td>1.00</td>
<td>1.00</td>
</tr>
<tr>
<td>theta[8]</td>
<td>1.00</td>
<td>1.01</td>
</tr>
</tbody>
</table>
Heidelberger and Welch Diagnostic

- Approach originally applied in OR for individual parameters in single chains.

- Based on a Brownian Bridge (Wiener Process) over the interval \([0, 1]\), in which the starting and stopping points are tied to the endpoints 0 and 1 and the “string” in between varies.

- Like the Geweke diagnostic, it has an inherently time-series orientation and uses the spectral density estimate.
Heidelberger and Welch Diagnostic

General procedure:

1. Specify a number of iterations to consider $N$, an accuracy ($\epsilon$), and an alpha level for the test.

2. The null hypothesis is that the chain is currently in the stationary distribution and the test starts with the full set of iterations. If the test rejects the null, the first 10% of the iterations are discarded and the test is run again.

3. This continues until either 50% of the data have been dismissed or the test fails to reject the null with the remaining iterations.

4. If some proportion of the data are found to be consistent with stationarity, then the \textit{halfwidth} analysis part of the diagnostic is performed.
Heidelberger and Welch Diagnostic (cont.)

- Actually a variant of the Kolmogorov-Smirnov nonparametric test for large sample sizes referred to as the Cramér-von Mises test after the form of the test statistic.

- A parameter, $s$, is defined as the proportion of the continuing sum of the chain, ranging from zero to one.

- We are interested in the difference between the sum of the first $sT$ of the chain (where $T$ is the total length of the chain), and the mean of the complete set of chain values scaled by $sT$.

- If the chain were in its stationary distribution, then this difference would be negligible. For the disparity between empirical and theoretical CDFs, the Cramér-von Mises test statistic is

\[
C_n(F) = \int_X [F_n(x) - F(x)]^2dF(x),
\]

with a predefined rejection region $C_n(F) > c$, established by normal tail values under asymptotic assumptions and the hypothesis that $F_n(x) \neq F(x)$. 

Heidelberger and Welch Diagnostic (cont.)

Some details:

- $T$ = the total length of the “accepted” chain after the discards.
- $s \in [0;1] =$ the test chain proportion.
- $T[sT] = \sum_{i=1}^{[sT]} \theta_i =$ the sum of the chain values from one to the integer value just below $sT$.
- $[sT] \bar{\theta} =$ the chain mean times the integer value just below $sT$.
- $s(0) =$ the spectral density of the chain.
Heidelberger and Welch Diagnostic

► Using these quantities, for any given $s$, we can construct the test statistic:

$$B_T(s) = \frac{T\lfloor sT \rfloor - \lfloor sT \rfloor \bar{\theta}}{\sqrt{Ts(0)}},$$

which is the Cramér-von Mises test statistic for sums as cumulative values scaled by the spectral density.

► Now $B_T(s)$ can be treated as an approximate Brownian bridge and tested using normal tail values to decide on the 10% discards (where some adjustments are necessary because $s(0)$ is estimated rather than known).
Heidelberger and Welch Diagnostic for the ANES Model (from superdiag)

********** The Heidelberger-Welch diagnostic: **********

Chain 1, epsilon=0.1, alpha=0.05

<table>
<thead>
<tr>
<th>Stationarity start</th>
<th>p-value</th>
</tr>
</thead>
<tbody>
<tr>
<td>test</td>
<td>iteration</td>
</tr>
<tr>
<td>cut[1] passed</td>
<td>1</td>
</tr>
<tr>
<td>cut[2] passed</td>
<td>1</td>
</tr>
<tr>
<td>cut[3] passed</td>
<td>1</td>
</tr>
<tr>
<td>deviance passed</td>
<td>1</td>
</tr>
<tr>
<td>theta[1] passed</td>
<td>1</td>
</tr>
<tr>
<td>theta[2] passed</td>
<td>1</td>
</tr>
<tr>
<td>theta[3] passed</td>
<td>1</td>
</tr>
<tr>
<td>theta[4] passed</td>
<td>1</td>
</tr>
<tr>
<td>theta[5] passed</td>
<td>1</td>
</tr>
<tr>
<td>theta[6] passed</td>
<td>1</td>
</tr>
<tr>
<td>theta[7] passed</td>
<td>1</td>
</tr>
<tr>
<td>theta[8] passed</td>
<td>1</td>
</tr>
</tbody>
</table>
Using BOA

library(boa)
boa.menu()

Bayesian Output Analysis Program (BOA)
Version 1.0.0 for UNIX R
Copyright (c) 2001 Brian J. Smith <brian-j-smith@uiowa.edu>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License or any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.
For a copy of the GNU General Public License write to the Free Software Foundation, Inc., 59 Temple Place - Suite 330, Boston, MA 02111-1307, USA, or visit their web site at http://www.gnu.org/copyleft/gpl.html

NOTE: if the menu unexpectedly terminates, type "boa.menu(recover= TRUE)" to restart and recover your work

BOA MAIN MENU

*************

1:File  >>
2:Data  >>
3:Analysis >>
4:Plot  >>
5:Options >>
6:Window >>
FILE MENU
=========
1: Import Data >>
2: Load Session
3: Save Session
4: Return to Main Menu
5: Exit BOA
Selection: 1

IMPORT DATA MENU
-----------------
1: BUGS Output File
2: Flat ASCII File
3: Data Matrix Object
4: View Format Specifications
5: Options...
6: Back
7: Return to Main Menu
Selection: 1

Enter filename prefix without the .ind or .out extension [Working Directory: ""]
1: Article.P-Agent/exec.short

Read 1 items
Read 18 records

Read 180000 records
+++ Data successfully imported +++

IMPORT DATA MENU
-----------------
1: BUGS Output File
2: Flat ASCII File
3: Data Matrix Object
4: View Format Specifications
5: Options...
6: Back
7: Return to Main Menu
Selection: 7

BOA MAIN MENU
*************
1: File >>
2: Data >>
3: Analysis >>
4: Plot >>
5: Options >>
6: Window >>
Selection: 3

ANALYSIS MENU
==============
1: Descriptive Statistics >>
2: Convergence Diagnostics >>
3: Options...
4: Return to Main Menu
Selection: 1

DESCRIPTIVE STATISTICS MENU
-------------------------------
1: Autocorrelations
2: Correlation Matrix
3: Highest Probability Density Intervals
4: Summary Statistics
5: Back
6: Return to Main Menu
Selection: 1
Selection: 3

HIGHEST PROBABILITY DENSITY INTERVALS:
======================================
Alpha level = 0.05
Chain: Article.P-Agent/exec.short

<table>
<thead>
<tr>
<th>Variable</th>
<th>Lower Bound</th>
<th>Upper Bound</th>
</tr>
</thead>
<tbody>
<tr>
<td>k[1]</td>
<td>-10.05000</td>
<td>-0.88130</td>
</tr>
<tr>
<td>k[2]</td>
<td>-5.93900</td>
<td>1.43900</td>
</tr>
<tr>
<td>k[3]</td>
<td>-1.82500</td>
<td>4.34400</td>
</tr>
<tr>
<td>sigma</td>
<td>3.21600</td>
<td>8.75000</td>
</tr>
<tr>
<td>tau</td>
<td>0.00945</td>
<td>0.06781</td>
</tr>
<tr>
<td>theta[10]</td>
<td>0.22970</td>
<td>1.33000</td>
</tr>
<tr>
<td>theta[11]</td>
<td>-1.08300</td>
<td>0.05925</td>
</tr>
<tr>
<td>theta[12]</td>
<td>0.62890</td>
<td>3.93200</td>
</tr>
<tr>
<td>theta[1]</td>
<td>-4.13900</td>
<td>2.87200</td>
</tr>
<tr>
<td>theta[2]</td>
<td>-1.86500</td>
<td>0.55190</td>
</tr>
<tr>
<td>theta[3]</td>
<td>-0.83480</td>
<td>0.17190</td>
</tr>
<tr>
<td>theta[4]</td>
<td>0.36910</td>
<td>2.05700</td>
</tr>
</tbody>
</table>
theta[5] 0.28810 3.25600
theta[7] 0.54550 2.15500
theta[8] -1.66400 -0.35180
theta[9] -0.14270 0.86110

Selection: 4

SUMMARY STATISTICS:
===================

Bin size for calculating Batch SE and (Lag 1) ACF = 50

Chain: Article.P-Agent/exec.short
----------------------------------
<table>
<thead>
<tr>
<th>Mean</th>
<th>SD</th>
<th>Naive SE</th>
<th>MC Error</th>
<th>Batch SE</th>
</tr>
</thead>
<tbody>
<tr>
<td>k[1]</td>
<td>-5.59806203</td>
<td>2.42271566</td>
<td>0.0242271566</td>
<td>0.1223327834</td>
</tr>
<tr>
<td>k[2]</td>
<td>-2.42940786</td>
<td>1.94031614</td>
<td>0.0194031614</td>
<td>0.0968573426</td>
</tr>
<tr>
<td>k[3]</td>
<td>1.35981060</td>
<td>1.60391981</td>
<td>0.0160391981</td>
<td>0.0778199101</td>
</tr>
<tr>
<td>k[4]</td>
<td>8.40854860 1.96700713 0.0196700713 0.1001023043 0.134728021</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>------</td>
<td>---------------------------------------------------------------</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>sigma</td>
<td>5.81126050 1.40852457 0.0140852457 0.0750363453 0.097579773</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>tau</td>
<td>0.03498487 0.01690473 0.0001690473 0.008895715 0.001164072</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>theta[10]</td>
<td>0.76229775 0.29844260 0.0029844260 0.0153456628 0.020729837</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>theta[11]</td>
<td>-0.47943636 0.29944492 0.0029944492 0.0145505332 0.020824001</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>theta[12]</td>
<td>2.12008306 0.82920162 0.0082920162 0.0403465492 0.049794374</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>theta[1]</td>
<td>-0.44837512 2.13613478 0.0213613478 0.1104139879 0.150234893</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>theta[2]</td>
<td>-0.64348071 0.63055720 0.0063055720 0.0309518537 0.038675137</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>theta[3]</td>
<td>-0.31625976 0.25289936 0.0025289936 0.0131013569 0.017111244</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>theta[4]</td>
<td>1.11830299 0.44912150 0.0044912150 0.0226997128 0.031597119</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>theta[5]</td>
<td>1.71032372 0.77381701 0.0077381701 0.0352462425 0.045846571</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>theta[6]</td>
<td>-2.67945860 0.58558035 0.0058558035 0.0312714378 0.041294397</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>theta[7]</td>
<td>1.18996756 0.47287189 0.0047287189 0.0219656503 0.033222243</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>theta[8]</td>
<td>-0.85445191 0.33927737 0.0033927737 0.018183079 0.023634776</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>theta[9]</td>
<td>0.34877322 0.24568023 0.0024568023 0.0128376865 0.016650594</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Batch ACF** | 0.025 | 0.5 | 0.975 MinIter MaxIter Sample
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>k[1]</td>
<td>0.9075014</td>
<td>-9.87607500</td>
<td>-5.73300</td>
</tr>
<tr>
<td>k[2]</td>
<td>0.8830128</td>
<td>-5.8130000</td>
<td>-2.56200</td>
</tr>
<tr>
<td>k[3]</td>
<td>0.8511580</td>
<td>-1.8807500</td>
<td>1.40200</td>
</tr>
<tr>
<td>k[4]</td>
<td>0.8960185</td>
<td>4.4370000</td>
<td>8.49550</td>
</tr>
<tr>
<td>sigma</td>
<td>0.9565878</td>
<td>3.6239500</td>
<td>5.70100</td>
</tr>
<tr>
<td>tau</td>
<td>0.9305307</td>
<td>0.0114950</td>
<td>0.03077</td>
</tr>
<tr>
<td>theta[10]</td>
<td>0.9185856</td>
<td>0.2317600</td>
<td>0.75560</td>
</tr>
<tr>
<td>theta[11]</td>
<td>0.9200565</td>
<td>-1.0352500</td>
<td>-0.47600</td>
</tr>
<tr>
<td>theta[12]</td>
<td>0.5477688</td>
<td>0.5545000</td>
<td>2.07700</td>
</tr>
<tr>
<td>theta[1]</td>
<td>0.9651961</td>
<td>-4.0380000</td>
<td>-0.26025</td>
</tr>
<tr>
<td>theta[2]</td>
<td>0.6337737</td>
<td>-1.9870750</td>
<td>-0.57785</td>
</tr>
<tr>
<td>theta[3]</td>
<td>0.8160511</td>
<td>-0.8309000</td>
<td>-0.30690</td>
</tr>
<tr>
<td>theta[4]</td>
<td>0.9594892</td>
<td>0.4183000</td>
<td>1.04000</td>
</tr>
<tr>
<td>theta[5]</td>
<td>0.6044240</td>
<td>0.3126275</td>
<td>1.67100</td>
</tr>
<tr>
<td>theta[6]</td>
<td>0.9783000</td>
<td>-3.8020000</td>
<td>-2.58350</td>
</tr>
<tr>
<td>theta[7]</td>
<td>0.9560934</td>
<td>0.5226000</td>
<td>1.09300</td>
</tr>
<tr>
<td>theta[8]</td>
<td>0.9260070</td>
<td>-1.7630000</td>
<td>-0.77130</td>
</tr>
<tr>
<td>theta[9]</td>
<td>0.8242078</td>
<td>-0.0836812</td>
<td>0.32735</td>
</tr>
</tbody>
</table>
Consider a model of support for abortion under different scenarios using survey data from Britain in consecutive years from 1983 to 1986 (Gill 2014).

The panel data for 264 respondents is collected annually by McGrath and Waterton (1986) where seven scenarios are provided and these respondents have the option of expressing support or disagreement for abortion.

The full collection of these seven queries do not fall into an obvious ordinal scale, so we will treat them here as nominal and judge total support for abortion as a binomial test for each respondent at each wave of the panel.

The scenarios are: (1) the woman decides on her own that she does not wish to have the child, (2) the couple agree that they do not wish to have the child, (3) the woman is not married and does not wish to marry the man, (4) the couple cannot afford any more children, (5) there a strong chance that the baby has a biological defect, (6) the woman’s health is seriously endangered by the pregnancy, and (7) the woman became pregnant as a result of rape.
Predictive Checks

► The model is specified for \( i = 1, \ldots, 264 \) respondents across \( j = 1, \ldots, 4 \) panel waves:

\[
y_{ij} \sim \mathcal{BN}(n_i, p_{ij})
\]

\[
\logit(p_{ij}) = \beta_{0,j} + \beta_{1,i}X_{1,i}
\]

\[
\beta_{0,j} \sim \mathcal{N}(\mu_0, \tau_0) \quad \beta_{1,i} \sim \mathcal{N}(\mu_1, \tau_1)
\]

\[
\mu_0 \sim \mathcal{N}(0, 100) \quad \mu_1 \sim \mathcal{N}(0, 100)
\]

\[
\tau_0 \propto \mathcal{C}_{\text{half}}(25) \quad \tau_1 \propto \mathcal{C}_{\text{half}}(25)
\]

► The second term in the normals is a variance and half-Cauchy priors are positive-support, zero-centralized forms with a scale term equal to \( A = 25, f(\tau) = (1 + \tau/A)^{-1}, \tau > 0 \).

► Here \( X_{1,i} \) is the \( i \)th person’s self-identified religion: (1) Catholic, (2) Protestant, (3) Other, and (4) No Religion, plus \( n_i = 7 \) at each wave for each person.

► \( X_{1,i} \) is indexed by 1 to infer that more explanatory variables can be specified, and \( n_i = 7 \) is indexed by \( i \).
Panel Model JAGS Code

model {
  for (i in 1:N) {
    for (j in 1:P) {
      logit(p[i,j]) <- b0[j] + b1[i]*x1[i]
      r[i,j] ~ dbin(p[i,j], n[i])
      b0[j] ~ dnorm(mu0, nu0)
    }
    b1[i] ~ dnorm(mu1, nu1)
  }
  mu0 ~ dnorm(0.0,1.0E-3)
  mu1 ~ dnorm(0.0,1.0E-3)
  tau0 ~ dt(0,25,1)T(0,)
  tau1 ~ dt(0,25,1)T(0,)
}

©JEFF GILL, AAPOR Workshop 2016 [132]
Model Summary, Traceplots

```
par(mfrow=c(2,2),mar=c(3,3,3,1),oma=c(1,1,1,1),col.axis="white",col.lab="white",
   col.sub="white",col="white",bg="slategray")
plot(out.table[1:264,1],type="l",col="mistyrose2")
abline(h=mean(out.table[1:264,1]),lwd=2,col="thistle4")
mtext(side=3,"Wave 1")
plot(out.table[(264+1):(2*264),1],type="l",col="navajowhite2")
abline(h=mean(out.table[265:(2*264),1]),lwd=2,col="skyblue1")
mtext(side=3,"Wave 2")
plot(out.table[((2*264+1):(3*264),1],type="l",col="yellowgreen")
abline(h=mean(out.table[((2*264+1):(3*264),1]),lwd=2,col="indianred2")
mtext(side=3,"Wave 3")
plot(out.table[((3*264+1):(4*264),1],type="l",col="blanchedalmond")
abline(h=mean(out.table[((3*264+1):(4*264),1]),lwd=2,col="chocolate1")
mtext(side=3,"Wave 4")
```
Model Summary, Abortion Attitudes in Britain, 20,000 Iterations, 10,000 Burnin

Model Summary

<table>
<thead>
<tr>
<th></th>
<th>Posterior Quantiles</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0.025</td>
</tr>
<tr>
<td>$\mu_0$</td>
<td>-0.6708</td>
</tr>
<tr>
<td>$\tau_0$</td>
<td>0.0283</td>
</tr>
<tr>
<td>$\mu_1$</td>
<td>0.3029</td>
</tr>
<tr>
<td>$\tau_1$</td>
<td>0.0303</td>
</tr>
</tbody>
</table>

- Draw values from these posterior distributions, insert them into the model statement with data and produce $\hat{y}$ values.

- For example, pick the 4th (last) panel wave with 7 outcomes...
Outcome Comparison: Observed Versus Simulated

# FIRST GET DATA FROM: https://pages.wustl.edu/jgill/courses
abortion <- read.spss("abortion_cluster.sav")
names(abortion)
                "/Users/jgill/Book.Bayes.II/Example.Abortion/CODAindex.txt")
dim(ab)
[1] 10000 1060

N <- 264; P <- 4 # SUBJECTS AND WAVE
ab.index <- sample(1:nrow(ab),9) # SAMPLE 9 RESPONDENTS
ab.index
[1] 4580 9198 9936 4893 194 6387 1547 5891 4858
Outcome Comparison: Observed Versus Simulated

# p = POSTERIOR PROBABILITIES, r = POSTERIOR PREDICTIONS
N <- 264; P <- 4 # SUBJECTS AND WAVE
ab.index <- sample(1:nrow(ab),9) # SAMPLE 9 RESPONDENTS
par(mfrow=c(3,3),mar=c(0,0,0,0),oma=c(8,8,1,1))
for (k in 1:length(ab.index)) {
  p <- matrix(ab[ab.index[k],1:(N*P)],nrow=N,ncol=P,byrow=FALSE)
  r <- matrix(NA,nrow=N,ncol=P)
  for (i in 1:N) { for (j in 1:P) { r[i,j] <- rbinom(n=1,size=7,prob=p[i,j]) } }
  plot( jitter(abortion$RESPON_C), jitter(r[,4]),
       pch="+",xaxt="n",yaxt="n", xlab="", ylab="", xlim=c(0.5,7.5),ylim=c(0.5,7.5))
  text(7,1,ab.index[k],cex=2.0)
  if(k %% 3 == 1) { axis(2,labels=1:7,at=1:7) }
  if(k > 6) { axis(1,labels=1:7,at=1:7) }
  mtext(side=2,line=4.5,outer=TRUE,cex=1.5,"Iterative Simulation Values")
  mtext(side=1,line=4.5,outer=TRUE,cex=1.5,"4th Panel Wave")
Outcome Comparison: Observed Versus Simulated