Bayesian Analysis for Political Science Workshop

Sarah Hunter

University of Colorado, Boulder

sarah.hunter@colorado.edu

Day 4

Overview

The Gibbs Sampler

Defining Convergence

How do you know your MCMC chain is done?

Definition

When our sampler reaches a stationary distribution (i.e. does not move outside the distribution)

- You cannot "prove" convergence
- You can provide evidence against nonconvergence
- Convergence is important because if your model is not converged, it is worthless

MLE v. OLS

- What is the main difference between GLM and LM?
- How do you think Bayes copes with the differences?

GLM with Bayes

The major change is the **distribution** and the parameters that describe Y.

- If a linear model starts with a Normal Distribution to model the DV, which distribution would we use for that of a logit model?
- What about for a count model?

GLM with Bayes

Other Considerations of GLM Models:

- Link functions
- Coefficients tend to be smaller
- Bounded quantities (i.e. probabilities)
- Measures of fit (i.e. Log Likelihood)

Useful Distributions

The Model Code

```
logit.model.jags<-function(){
    for(i in 1:N){
        protectionist[i]~ dbern(p.bound[i]);
        p.bound[i]<-max(0, min(1,p[i]))
        logit(p[i])<- mu[i]
        mu[i]<-beta1
            +beta2*age[i]
            +beta3*female[i]
            +beta4*tumember[i]
            +beta5*partyid[i]
            +beta6*ideology[i]
            +beta7*schooling[i]

            llh[i]<-protectionist[i]*log(p[i])+
            (1-protectionist[i])*log(1-p[i])
}
sumllh<-sum(llh[])

beta1~dnorm(0.0,0.1)
beta2~dnorm(0.0,0.1)
beta3~dnorm(0.0,0.1)
beta4~dnorm(0.0,0.1)
beta5~dnorm(0.0,0.1)
beta6~dnorm(0.0,0.1)
beta7~dnorm(0.0,0.1)
}
```

The Priors and Posteriors of Bayesian GLMs

- How does the choice of prior change with the GLM?
- How does the estimation of the posterior change with the GLM?

Implementing Bayesian GLMs

1. Write your model function.
2. Choose your priors.
3. Convert your data to list format.
4. Choose the parameters you want to monitor.
5. Estimate the model.
6. Conduct convergence diagnostics.
7. Model Presentation.

What do you mean by "Multilevel"?

"Multilevel" refers to the *structure* of data:

- Nested groups
- Clustered data
- Repeated measures
- Panel Data

This hierarchy can be:

- Substantive (Think nested models)
- Methodological (Think priors)
- Both

Why Use MLM?

- More accurately represents the data generating process
- More accurately tests (many) theories
- Makes a huge difference when there is real group-level variation:
  - E.g.: Think about European Parliament members: differences in parities/interests based on countries.

Jargon issues

MLMs can allow us to account for individual and group-level differences:

- Group-level regression coefficients (intercepts or regression coefficients)
- Individual regression coefficients(intercepts or regression coefficients)

But what do we call these "effects"?

- Fixed Effects
- Random Effects

Fixed v. Random Effects

What are Fixed and Random Effects? Well, there is some confusion based on field of study:

TSCS World

- Fixed Effects: within-unit estimator
- Random Effects: Weighted average of within and between unit estimators

Kreft and De Leeuw

- Fixed Effects: constant across all individuals
- Random Effects: Vary across all individuals

Green and Tukey

- Fixed Effects: When a sample exhausts the entire population
- Random Effects: when the sample is a small part of the population

Snijders and Bosker

- Fixed Effects: Conditional on unique groups, and interest in with-in group differences
- Random Effects: When groups are regarded a sample of the population

Bayesian Effects

The definitions of fixed and random effects has no impact on the Bayesian models for multilevel data.

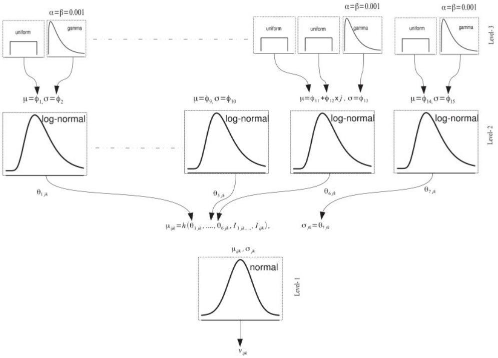
- Everything in a Bayesian model is a **random effect**.
- Because we assume that the data are fixed and the parameters are **random**.

Bayesian MLM

```
state.mod <- function() {
  for (i in 1:n){
    y[i] ~ dbin (p.bound[i], 1)
    p.bound[i] <- max(0, min(1, p[i]))
    logit(p[i]) <- Xbeta[i]
    Xbeta[i] <- b.female*female[i] + b.black*black[i] +
      b.age*age[i] + b.edu*edu[i]
      + b.state[state[i]]
  }
  b.female ~ dnorm (0, .01)
  b.black ~ dnorm (0, .01)
  b.age ~ dnorm (0, .01)
  b.edu ~ dnorm (0, .01)

  for (j in 1:n.state){
    b.state[j] ~ dnorm(b.state.hat[j], tau.state)
    b.state.hat[j] <- b.state0 + b.v.prev*v.prev[j]
    }
    b.state.hat.mu<-mean(b.state.hat[])
    b.v.prev ~ dnorm(0, .01)
    b.state0 ~ dnorm(0, .01)
    tau.state <- pow(sigma.state, -2)
    sigma.state ~ dunif (0, 100)
}
```

Tips for Coding MLMs in JAGS



Source: (Mirsha, Martinsson, Rantatalo, and Goebel 2018)

- Be very careful with your indices
- Spacing and tabs can be your friend
- Hyperpriors can actually be used as random effects

More advanced application of Bayesian Analysis

- Text Classification
- Data Classification
- Sentiment Analysis
- Network Analysis
- Neural Networks
- Deep Learning

More advanced application of Bayesian Analysis

- **Text Classification**
- Data Classification
- **Sentiment Analysis**
- Network Analysis
- Neural Networks
- Deep Learning

Text Analysis

One of the fastest growing analytic tools in political science is text analysis. Recent studies have analyzed:

- Tweets
- Trade Agreements
- Party Manifestos
- Political Speeches

Bayes is the perfect tool for text classification.

Bayes' Rule and Classification

Recall Bayes' Rule:

Bayes' Law

$$P(A|B) = \frac{P(B|A)*P(A)}{P(B)}$$

What is the goal of text classification?

- To find the probability of the document belonging to class j, given the document

  OR
- $P(c|d)$

Bayesian Text Classification

$P(c|d)$ looks a lot like a posterior that want to derive from Bayesian models

Bayes' Law tells us:

$$p(c|d) = \frac{p(d|c) * p(c}{p(d)}$$

To classify the text, we need to find the *most likely category*, which requires maximizing the Bayes function:

$$c_{MAP} = \operatorname*{argmax}_{c \in C} \frac{p(d|c) * p(c)}{p(d)}$$

How it Works

Documents can be seen as a collection of words, which gives us (dropping the denominator):

$$c_{MAP} = \operatorname*{argmax}_{c \in C} p(X_1, X_2, X_3, ..., X_n | c) * p(c)$$

TO get the Naive Bayes Classifier:

Naive Bayes

$$c_{NB} = \operatorname*{argmax}_{c \in C} p(c_j) \prod_{x \in X} p(x|c)$$

Bag of Works Assumption

Bag of words (BoW)

Very good drama although it appeared to have a few blank areas leaving the viewers to fill in the action for themselves. I can imagine life being this way for someone who can neither read nor write. This film simply smacked of the real world: the wife who is suddenly the sole supporter, the live-in relatives and their quarrels, the troubled child who gets knocked up and then, typically, drops out of school, a jackass husband who takes the nest egg and buys beer with it. 2 thumbs up... very very very good movie.

('the', 8),
(',', 5),
('very', 4),
('.', 4),
('who', 4),
('and', 3),
('good', 2),
('it', 2),
('to', 2),
('a', 2),
('for', 2),
('can', 2),
('this', 2),
('of', 2),
('drama', 1),
('although', 1),
('appeared', 1),
('have', 1),
('few', 1),
('blank', 1)
.....

If the Position of the Word Matters...

Using Word Positions

$$c_{NB} = \underset{c_j \in c_j}{\operatorname{argmax}} p(c_j) \prod_{i \in Positions} p(x_i | c_j)$$

Steps in Text Classification

- Clean the Text
- Train your model
- Test your model
- Evaluate your model

Training the Model with bag of Words

These models are usually *supervised learning*. To train a model,

- Extract *Vocabulary*
- Calculate $p(c_j)$

$$p(c_j) = \frac{|docs_j|}{total\#documents}$$

- Calculate $p(w_k)|c_j)$:

$$p(w_k|c_j) = \frac{n_k+\alpha}{n+\alpha|Vocabulary|}$$

R Code

```
#Training the Model
trainedNBclassifier<-naiveBayes(x=fword_train,
    y=factor(trainY))

#Predict

NBClasifier_test_pred<-predict(trainedNBclassifier,
    newdata=fword_test)
```

Until Next Time