

# 450D Political Methodology IV

## TA Section 3

Vincent Bauer

October 13, 2017

# Goal Today

- ▶ Demonstrate more (but still not entirely) realistic examples of MCMC.

Road map to those goals:

1. Metropolis Sampler using Poisson data
2. Stan example with same data
3. Help setting up Stan

## Poisson regression example

- ▶ The example for today comes from Simon Jackman's book *Bayesian Analysis for the Social Sciences*.

## Poisson regression example

- ▶ The example for today comes from Simon Jackman's book *Bayesian Analysis for the Social Sciences*. The book is available for *free* from Stanford.
- ▶ In Chapter 5, (pg 209) he sets up an example modeling the number of published articles produced by 915 graduate students in Biochemistry.

## Poisson regression example

- ▶ The example for today comes from Simon Jackman's book *Bayesian Analysis for the Social Sciences*. The book is available for *free* from Stanford.
- ▶ In Chapter 5, (pg 209) he sets up an example modeling the number of published articles produced by 915 graduate students in Biochemistry.
- ▶ We know that the Gamma distribution is conjugate for a univariate Poisson distribution

## Poisson regression example

- ▶ The example for today comes from Simon Jackman's book *Bayesian Analysis for the Social Sciences*. The book is available for *free* from Stanford.
- ▶ In Chapter 5, (pg 209) he sets up an example modeling the number of published articles produced by 915 graduate students in Biochemistry.
- ▶ We know that the Gamma distribution is conjugate for a univariate Poisson distribution
- ▶ but there is no conjugate prior for generalized linear model,

## Poisson regression example

- ▶ The example for today comes from Simon Jackman's book *Bayesian Analysis for the Social Sciences*. The book is available for *free* from Stanford.
- ▶ In Chapter 5, (pg 209) he sets up an example modeling the number of published articles produced by 915 graduate students in Biochemistry.
- ▶ We know that the Gamma distribution is conjugate for a univariate Poisson distribution
- ▶ but there is no conjugate prior for generalized linear model, so we have to use MCMC.

## Poisson regression example

- ▶ The example for today comes from Simon Jackman's book *Bayesian Analysis for the Social Sciences*. The book is available for *free* from Stanford.
- ▶ In Chapter 5, (pg 209) he sets up an example modeling the number of published articles produced by 915 graduate students in Biochemistry.
- ▶ We know that the Gamma distribution is conjugate for a univariate Poisson distribution
- ▶ but there is no conjugate prior for generalized linear model, so we have to use MCMC.

There is not really a benefit to using a Bayesian approach in this example but its a stepping stone to more complicated models.



## Poisson regression example

We need six pieces of information to set up our MCMC algorithm

## Poisson regression example

We need six pieces of information to set up our MCMC algorithm

1. The data (obviously)

## Poisson regression example

We need six pieces of information to set up our MCMC algorithm

1. The data (obviously)
2. Proposal distribution

## Poisson regression example

We need six pieces of information to set up our MCMC algorithm

1. The data (obviously)
2. Proposal distribution
3. Prior distribution

## Poisson regression example

We need six pieces of information to set up our MCMC algorithm

1. The data (obviously)
2. Proposal distribution
3. Prior distribution
4. Likelihood function

## Poisson regression example

We need six pieces of information to set up our MCMC algorithm

1. The data (obviously)
2. Proposal distribution
3. Prior distribution
4. Likelihood function
5. A function to compute the posterior distribution

## Poisson regression example

We need six pieces of information to set up our MCMC algorithm

1. The data (obviously)
2. Proposal distribution
3. Prior distribution
4. Likelihood function
5. A function to compute the posterior distribution
6. Metropolis sampler

## Poisson regression example

We need six pieces of information to set up our MCMC algorithm

1. The data (obviously)
2. Proposal distribution
3. Prior distribution
4. Likelihood function
5. A function to compute the posterior distribution
6. Metropolis sampler



## Poisson regression example

We need six pieces of information to set up our MCMC algorithm

1. The data (obviously)
2. Proposal distribution
3. Prior distribution
4. Likelihood function
5. A function to compute the posterior distribution
6. Metropolis sampler

Because we're using the Metropolis algorithm, we will use the multivariate normal distribution as our *proposal distribution*.

## Poisson regression example

We need six pieces of information to set up our MCMC algorithm

1. The data (obviously)
2. Proposal distribution
3. Prior distribution
4. Likelihood function
5. A function to compute the posterior distribution
6. Metropolis sampler

Because we're using the Metropolis algorithm, we will use the multivariate normal distribution as our *proposal distribution*. We get our *posterior distribution* by multiplying the prior and the likelihood.

## Poisson regression example

We need six pieces of information to set up our MCMC algorithm

1. The data (obviously)
2. Proposal distribution
3. Prior distribution
4. Likelihood function
5. A function to compute the posterior distribution
6. Metropolis sampler

Because we're using the Metropolis algorithm, we will use the multivariate normal distribution as our *proposal distribution*. We get our *posterior distribution* by multiplying the prior and the likelihood. We will use a normal distribution with  $\mu = 0$  and  $\text{sd} = 10,000$  as our prior for the regression coefficients.

## Poisson regression example

We need six pieces of information to set up our MCMC algorithm

1. The data (obviously)
2. Proposal distribution
3. Prior distribution
4. Likelihood function
5. A function to compute the posterior distribution
6. Metropolis sampler

Because we're using the Metropolis algorithm, we will use the multivariate normal distribution as our *proposal distribution*. We get our *posterior distribution* by multiplying the prior and the likelihood. We will use a normal distribution with  $\mu = 0$  and  $\text{sd} = 10,000$  as our prior for the regression coefficients. Our metropolis sampler is the same as in the previous Section.

```
library(mvtnorm)
library(pscl)

#get the data and clean
data(bioChemists)
bioChemists$fem <- ifelse(bioChemists$fem=="Men", 0,1)
bioChemists$mar <- ifelse(bioChemists$mar=="Single", 0,1)

k <- ncol(bioChemists) #k - dv + intercept
y <- bioChemists$art #outcome values
x <- cbind(1, as.matrix(bioChemists)[,-1]) #predictor values
```

# Proposal density

Multivariate normal distribution.

```
#jackson uses step size equal to MLE coefficients
stepsize <- solve(vcov(glm(y ~ x - 1,
                           family="poisson", data=bioChemists)))

#I will use stepsize 1/100 to prove that the MLE is not key
stepsize <- matrix(100, ncol=ncol(x), nrow=ncol(x))

#proposal density
proposal <- function(beta.cur){

  #parameters from jackman
  T <- diag(1.1,k)
  prior.sigma <- diag(10^4,k)
  P <- T %*% solve(prior.sigma + stepsize) %*% T

  #calculations
  out <- rmvnorm(1, mean=beta.cur, sigma=P)
  return(out)
}
```

## Posterior distribution

```
log.post <- function(param, y, x){  
  #looks the same as here  
  #MCMCpack:::logpost.poisson  
  
  #parameters from jackman  
  prior.mean <- rep(0, k)  
  prior.sigma <- diag(10^4,k)  
  
  #log likelihood  
  loglk <- sum(dpois(y, exp(x %*% t(param)), log=TRUE))  
  
  #log prior  
  logprior <- dmvnorm(param, mean=prior.mean,  
                       sigma=prior.sigma, log=TRUE)  
  
  #"multiply" together  
  return(loglk + logprior)  
}
```

# Metropolis Sampler

```
metro.sampler<-function (n, init.mean){
  out <- list() #output vector
  beta.cur <- init.mean #initial values
  out[[1]] <- beta.cur #store value

  #now iterate
  for (i in 2:n) {
    if(i %% 1000 == 0){cat(paste(i,"... "))} #report progress

    beta.can <- proposal(beta.cur) #make a proposal

    #prob of update
    aprob <- min(1, exp(log.post(beta.can, y, x) -
                        log.post(beta.cur, y, x)))

    #update with probability aprob
    if(runif(1) < aprob){beta.cur <- beta.can}
    out[[i]] <- beta.cur
  }
  cat("done!\n"); return(out)
}
```

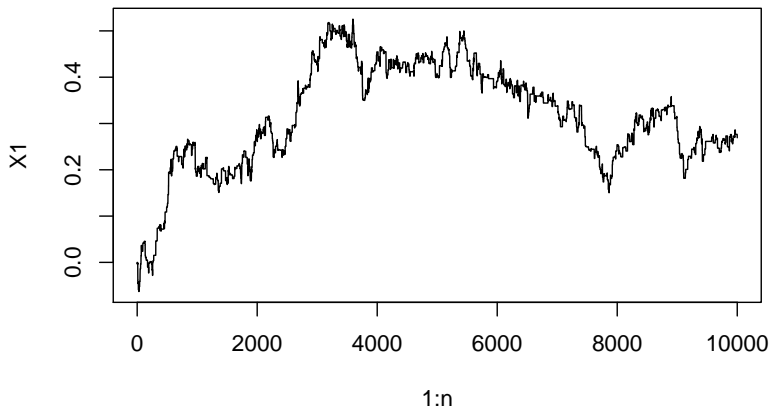


# Computation

```
init.mean <- t(matrix(rep(0, 6)))  
  
n <- 10000  
  
system.time(  
  prob <- metro.sampler(n=n, init.mean=init.mean)  
)  
  
## 1000 ... 2000 ... 3000 ... 4000 ... 5000 ... 6000 ... 7000 ..  
##      user  system elapsed  
##      8.848   0.074   9.045
```

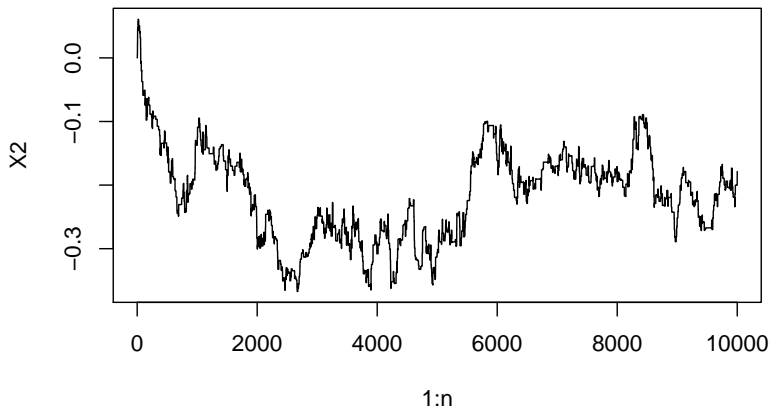
## Diagnostics

Autocorrelation is pretty high, a bigger step size would probably help.



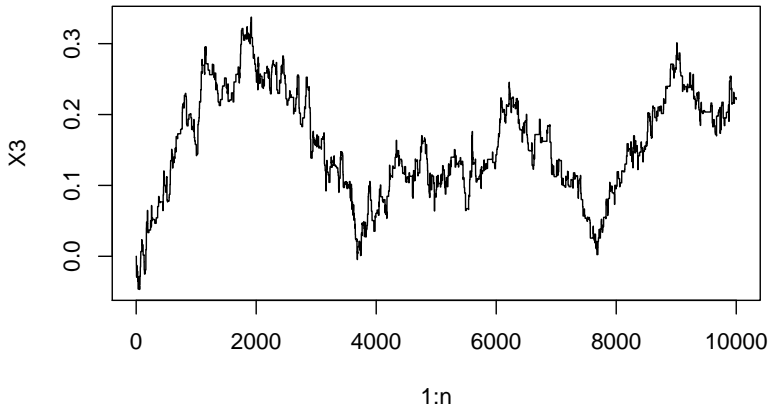
## Diagnostics

Autocorrelation is pretty high, a bigger step size would probably help.



## Diagnostics

Autocorrelation is pretty high, a bigger step size would probably help.



## Validity

More draws would have gotten us closer to the GLM estimate.

```
library(xtable)
est <- apply(do.call(rbind, prob[5000:n]), 2, median)
names(est) <- colnames(x)

xtable(cbind("MCMC"=est,
"GLM"=glm(y ~ x - 1, family="poisson")$coefficients))
```

	MCMC	GLM
	0.32	0.30
fem	-0.19	-0.22
mar	0.15	0.16
kid5	-0.20	-0.18
phd	0.00	0.01
ment	0.03	0.03

## Stan Example

Next I'll walk through exactly the same analysis in Stan.

## Stan Example

Next I'll walk through exactly the same analysis in Stan. Most models you set up will be more complicated than this example.

## Stan Example

Next I'll walk through exactly the same analysis in Stan. Most models you set up will be more complicated than this example.

We only need three pieces of information for the Stan model,



## Stan Example

Next I'll walk through exactly the same analysis in Stan. Most models you set up will be more complicated than this example.

We only need three pieces of information for the Stan model, that's half as many as before.

1. The data

## Stan Example

Next I'll walk through exactly the same analysis in Stan. Most models you set up will be more complicated than this example.

We only need three pieces of information for the Stan model, that's half as many as before.

1. The data
2. Prior distribution

## Stan Example

Next I'll walk through exactly the same analysis in Stan. Most models you set up will be more complicated than this example.

We only need three pieces of information for the Stan model, that's half as many as before.

1. The data
2. Prior distribution
3. Likelihood function

## Stan Example

Next I'll walk through exactly the same analysis in Stan. Most models you set up will be more complicated than this example.

We only need three pieces of information for the Stan model, that's half as many as before.

1. The data
2. Prior distribution
3. Likelihood function

We don't need

1. A proposal distribution

## Stan Example

Next I'll walk through exactly the same analysis in Stan. Most models you set up will be more complicated than this example.

We only need three pieces of information for the Stan model, that's half as many as before.

1. The data
2. Prior distribution
3. Likelihood function

We don't need

1. A proposal distribution
2. To write a sampler

## Stan Example

Next I'll walk through exactly the same analysis in Stan. Most models you set up will be more complicated than this example.

We only need three pieces of information for the Stan model, that's half as many as before.

1. The data
2. Prior distribution
3. Likelihood function

We don't need

1. A proposal distribution
2. To write a sampler
3. A function for the posterior distribution

## Stan Example

```
data {  
  int<lower=1> N;           //number of observations  
  int<lower=1> K;           //number of variables  
  matrix[K, N] X;         //predictors matrix  
  int Y[N];               //outcome variable  
}  
  
parameters {  
  row_vector[K] betas;    //coefficients to estimate  
}  
  
model {  
  row_vector[N] lambda;   //does not return this  
  betas ~ normal(0, sqrt(10^4)); //non-informative priors  
  
  //likelihood function  
  lambda = multiply(betas, X);  
  Y ~ poisson(exp(lambda));  
}
```

## Stan Example

```
library(rstan)
rstan_options(auto_write = TRUE)
options(mc.cores = parallel::detectCores())
parallel::detectCores()

data <- list(X=t(x), Y=y, N = nrow(x), K = ncol(x))

stan.model <- "~/Documents/Graduate School/Teaching/450D_Ba
fit <- stan(file=stan.model,
           model_name = "poisson",
           data = data,
           iter = 1000,
           chains = 2)

stan <- apply(extract(fit, "betas")$betas, 2, mean)
```



# Stan Example

```
print(fit, "betas")
```

```
## Inference for Stan model: poisson.
```

```
## 2 chains, each with iter=1000; warmup=500; thin=1;
```

```
## post-warmup draws per chain=500, total post-warmup draws=1000.
```

```
##
```

```
##           mean se_mean   sd  2.5%  25%  50%  75% 97.5% n_eff Rhat
```

```
## betas[1]  0.30    0.01 0.11  0.07  0.23  0.30  0.37  0.49   431 1.00
```

```
## betas[2] -0.22    0.00 0.06 -0.32 -0.26 -0.22 -0.19 -0.10   561 1.00
```

```
## betas[3]  0.16    0.00 0.06  0.04  0.12  0.16  0.20  0.29   608 1.00
```

```
## betas[4] -0.18    0.00 0.04 -0.26 -0.21 -0.18 -0.16 -0.10   652 1.01
```

```
## betas[5]  0.01    0.00 0.03 -0.04 -0.01  0.01  0.03  0.06   520 1.00
```

```
## betas[6]  0.03    0.00 0.00  0.02  0.02  0.03  0.03  0.03  1000 1.00
```

```
##
```

```
## Samples were drawn using NUTS(diag_e) at Thu Oct 19 23:05:25 2017.
```

```
## For each parameter, n_eff is a crude measure of effective sample size
```

```
## and Rhat is the potential scale reduction factor on split chains (at
```

```
## convergence, Rhat=1).
```

## Results

Stan's sampler is much better than the one I programmed above, using many fewer iterations.

	STAN	MCMC	GLM
	0.30	0.32	0.30
fem	-0.22	-0.19	-0.22
mar	0.16	0.15	0.16
kid5	-0.18	-0.20	-0.18
phd	0.01	0.00	0.01
ment	0.03	0.03	0.03

Stan setup help