www.ICA2014.org

LEARN
INTERACT
GROW

ICA 2014 MCMC Workshop

Glenn Meyers

April 3, 2014

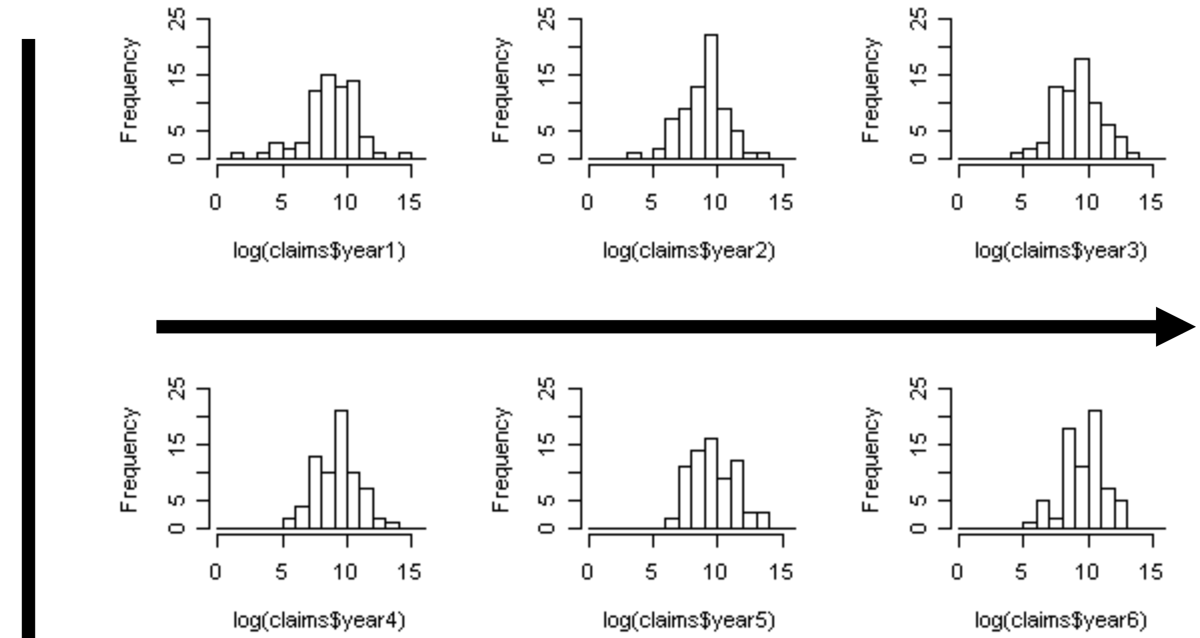
Get Ready to Run

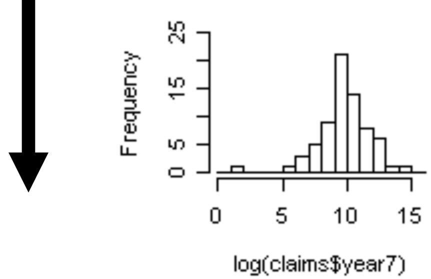
- Download and Install
  - R at www.r-project.org/
  - JAGS v 3.3 or higher at http://mcmc-jags.sourceforge.net/
  - RStudio at http://www.rstudio.com/
- In R, install the packages “actuar”, “runjags”, “ChainLadder” and ”coda.”
- Copy the distributed folder into your workshop folder.
- Make your workshop folder the default working directory in RStudio preferences
  - Or use the “setwd()” function in your R scripts.

Introductory Example COTOR Challenge Round 3

- Seven consecutive years of claims have been drawn at random from a heavy tailed distribution by our loss distribution expert, Stuart Klugman. A total of 490 claims have been sampled, split amongst the 7 years. Each year of claims is drawn from the same distribution, except that the scale parameter changes each year as due to inflation.
- The challenge is to estimate the mean severity and calculate a 95 percent confidence interval for the $500,000 excess of $500,000 layer for the next (eighth) year. We are asking that you estimate and provide confidence intervals for both the true severity and the actual severity (based on 70 claims). The first one concerns the accuracy of the premium to be assessed while the second concerns accuracy relative to the actual realized losses experienced in year eight.
- Full details at http://www.casact.org/research/cotor/index.cfm?fa=round3

Histograms of log(Claims) by Year





Observations

- Claim severity increasing with time.
- Distribution of log(claims) symmetric

Model Description

- Claim ~ lognormal(μ – ltrend*(8 – Year), σ)
- μ ~ Uniform(0,20)
- σ ~ Uniform(1,4)
- ltrend ~ Normal(0,1)
- Go though R and JAGS script to
  - Get a sample from the posterior distribution of μ, σ and ltrend.
  - Translate the parameters into statistics of interest.

Run Script

1. Load “COTOR Challenge Intro.R” into Rstudio
2. Be sure that “cotorchallenge3.csv” data is in your working directory.
3. Run script by clicking on “Source” in Rstudio
   - Parameters of interest from JAGS
   - Layer Expected Value calculated in R
   - Simulated outcomes done in R

- This illustrates what MCMC can do for actuarial problems.
- Gary will now explain the underlying theory of Bayesian MCMC.

Lognormal Example

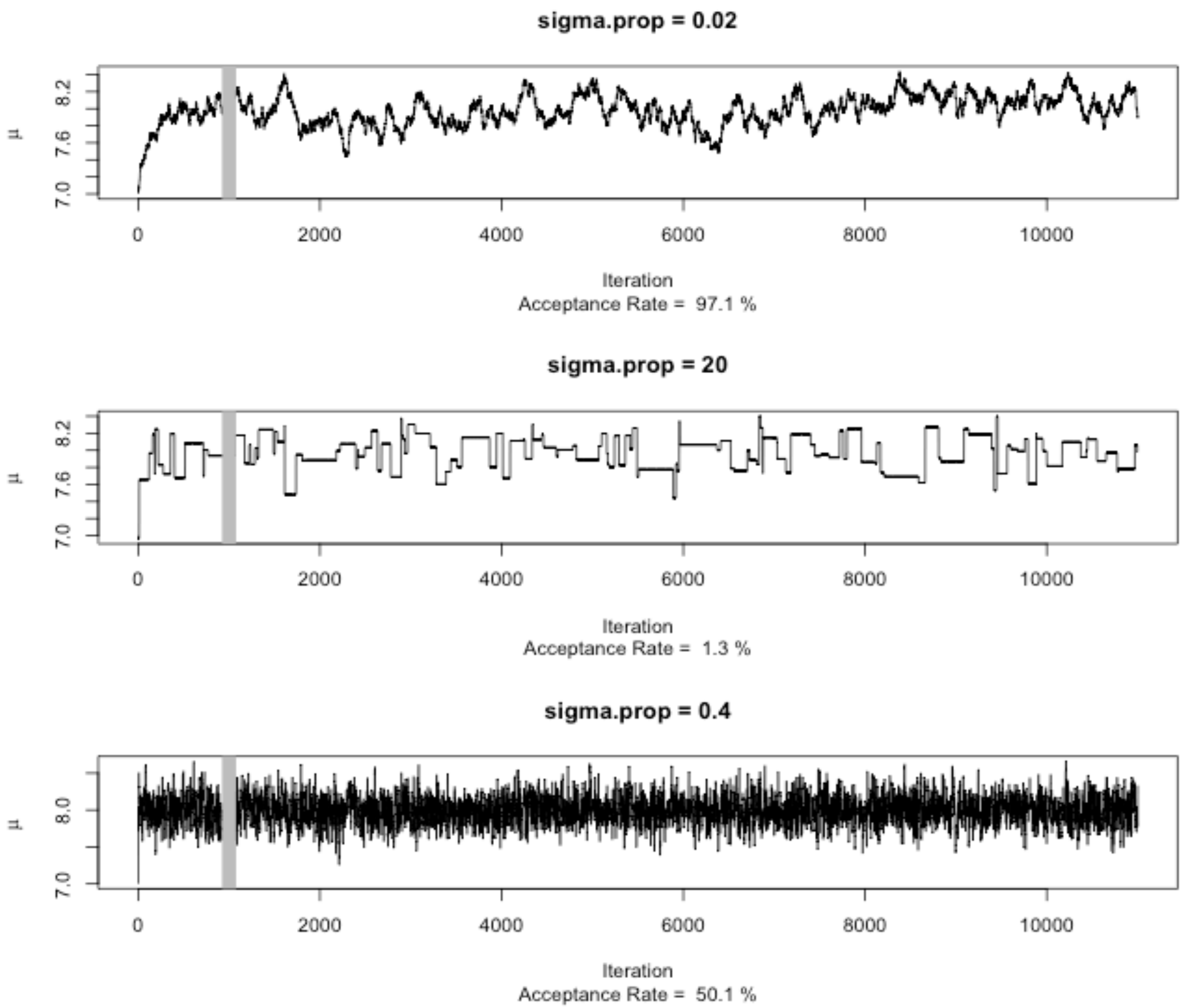
- Given the data below estimate the cost of a 15,000 xs of 10,000 layer.
  - Find predictive distribution of losses in that layer
- Fit a lognormal distribution with
  - log(mean) = μ
  - Log(standard deviation) = 1.

484	1,407	2,262	5,015	6,500
603	1,565	2,654	5,354	6,747
631	1,894	2,672	5,464	9,143
1,189	2,140	4,019	5,598	12,782
1,229	2,244	4,318	6,060	18,349

Lognormal Example

- Open Rstudio
- Load the script “MH Scaling with Lognormal.R”
  - Needs “actuar” package installed.
- Run script by clicking on “Source.”
- Try different parameters
  - Different values of “sigma.prop” on line 28
  - Different values of “thin” on line 29

Lognormal Example Trace Plots with Different Proposal Scalings



Adjustments to Get a Representative Sample of the Posterior Distribution

- Adaptation – Scaling the proposal distribution to get a representative sample in as few iterations as possible.
  - Acceptance rate goes from 50% for 1 parameter, down to 25% for many parameters.
- Thinning – When adaption does not work well.
  - Take every *n*th iteration.

Posterior Distributions of Statistics of Interest

- Script generates 10,000 μ parameters
  - Of no particular interest.
- Of possible interest
  - Expected value of 15,000 xs of 10,000 layer
    - Calculate for each μ using “levlnorm” function in “actuar” package.
  - Predictive distribution of losses in the 15,000 xs of 10,000 layer.
    - Simulate once for each μ using the “rlnorm” function in R and adjust for the layer.

Continue with COTOR Challenge Example

- Load "COTOR Challenge.R" into Rstudio
- Claim ~ lognormal(μ – ltrend*(8 – Year), σ)
- μ ~ Uniform(0,20)
- σ ~ Uniform(1,4)
- ltrend ~ Normal(0,1)

COTOR Challenge Example Discussion Highlights

- Look at JAGS script
- Run 4 independent chains
- Trace Plots – The 4 chains should overlap
- Gelman – Rubin convergence diagnostics
- Predictive distribution of parameters and some statistics of interest.

Gelman-Rubin Convergence Diagnostics

- $\sqrt{\hat{R}}$ = Potential Scale Reduction Factor – PSRF
  - Should approach 1 as number of iterations increase.
  - Gelman and Rubin recommend 1.1 as being OK
- $\hat{W}$ = Within chain variance
- $\hat{B}$ = Between chain variance

$$\sqrt{\hat{R}} = \sqrt{\frac{\hat{W} + \hat{B}}{\hat{W}}} \rightarrow 1$$

Collective Risk Model (Tweedie Distribution)

- Given the data below
  - Predict the expected cost for losses in excess of 1,000 for an insured with exposure = 1000

Exposure	Y	Exposure	Y	Exposure	Y
51	27	226	204	368	468
66	67	231	238	374	434
119	168	254	255	377	327
125	194	255	244	381	313
131	108	258	427	392	436
152	129	268	346	444	394
196	189	279	276	449	532
197	250	295	269	478	569
225	248	340	325	484	459
225	171	364	321	495	458

Model for Tweedie Example

$$N \sim Poisson(\lambda)$$

$$Y \sim Gamma(\theta, N \cdot \alpha)$$

$$\lambda = k \cdot Exposure$$

$$k \sim Uniform(0.05, 0.15)$$

$$\alpha \sim Uniform(0.1, 10)$$

$$\theta \sim Uniform(5, 200)$$

Run Script for Tweedie Example

- Load “Tweedie Example.R” into Rstudio
- Run script by clicking on “Source” in Rstudio

Discussion Points for Tweedie Example

- Initial trace plots show lack of convergence
- Effect of thinning
- Gelman plots and convergence diagnostics
- Correlation between parameters
  - Is this a bad thing?

Final Example
Stochastic Loss Reserving

- Data source – CAS Loss Reserve Database

  http://www.casact.org/research/index.cfm?fa=loss_reserves_data

- Contains hundreds of 10 x 10 triangles taken from American Schedule P.
- Triangles are completed with data from subsequent annual statements.

The Correlated Chain Ladder (CCL) Model

- $\mu_{1,d} = \alpha_1 + \beta_d$
- $C_{1,d} \sim \text{lognormal}(\boldsymbol{\mu}_{\boldsymbol{1,d}}, \sigma_d)$
- $\mu_{w,d} = \alpha_w + \beta_d + \rho \cdot(\log(C_{w-1,d}) - \boldsymbol{\mu}_{\boldsymbol{w-1,d}})$ for $w = 1,\ldots,10$
- $C_{w,d} \sim \text{lognormal}(\boldsymbol{\mu}_{\boldsymbol{w,d}}, \sigma_d)$
- $\rho \sim U(-1,1)$
- $\alpha_w \sim \text{normal}(\log(\text{Premium}_w) + \log\text{ELR},10)$, $\log\text{ELR} \sim \text{uniform}(-1.5,0.5)$
- $\beta_d \sim \text{uniform}(-5,5)$ for $d = 1,\ldots,9$, $\beta_{10} = 0$
- $\sigma_d = \sum_{i=d}^{10} a_i$ $\quad a_i \sim U(0,1)$ Forces σ_d to decrease as d increases
- Estimate distribution of $\sum_{w=1}^{10} C_{w,10}$

Run Script for CCL Model

- Load “CCL Model.R”
- Be sure “Sample CAS Data.csv” is in your working directory.
- Run script by clicking on “Source” in Rstudio
- Compare with “Mack Model.R” script.

My Current Favorite References

- Jackman - 2009
  - *Bayesian Analysis for the Social Sciences*
- Brooks, Gelman, Jones and Meng - 2011
  - *Handbook of Markov Chain Monte Carlo*

What do you do next?

- Go home and start a project to do with MCMC.
- Suggestion
  - Replace the Poisson assumption in the "Collective Risk Model" example with a negative binomial distribution

Do Not Delay!