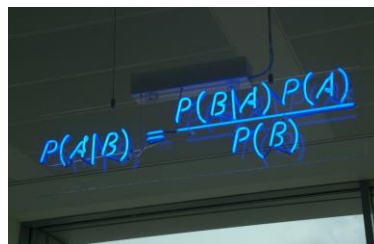


Stochastic Loss Reserving With Bayesian MCMC Models


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

CAS Loss Reserving Seminar
Boston
September 15, 2013

Glenn Meyers – Actuary at Large
James Guszczka – Deloitte Consulting LLP

Antitrust Notice

The Casualty Actuarial Society is committed to adhering strictly to the letter and spirit of the antitrust laws. Seminars conducted under the auspices of the CAS are designed solely to provide a forum for the expression of various points of view on topics described in the programs or agendas for such meetings.

Under no circumstances shall CAS seminars be used as a means for competing companies or firms to reach any understanding – expressed or implied – that restricts competition or in any way impairs the ability of members to exercise independent business judgment regarding matters affecting competition.

It is the responsibility of all seminar participants to be aware of antitrust regulations, to prevent any written or verbal discussions that appear to violate these laws, and to adhere in every respect to the CAS antitrust compliance policy.

3

Today's Agenda

- Morning – Introduction to Bayesian Data Analysis
 - Session 1: Bayesian concepts, computation (MCMC), and software (JAGS)
 - Session 2: Bayesian case studies

- Afternoon – Bayesian data analysis for loss reserving
 - Session 3: Retrospective Testing of Stochastic Loss Reserve Models
 - Session 4: The Correlated Chain Ladder and Correlated Incremental Trend models

4

Agenda – Morning

- Preamble
- Bayesian Concepts
- Bayesian Computation: Markov Chain Monte Carlo
- Software: R and JAGS
- Simple Case Studies
 - Loss Distribution Analyses
 - Bayesian Regression and GLM
 - Adding autoregressive structure to a regression
 - Simple Bayesian Chain Ladder Analysis
- Nonlinear Hierarchical Bayes Loss Reserving Model

5

Agenda – Afternoon

- How to Validate Stochastic Loss Reserving Methodologies
- Data: The CAS Loss Reserve Database
- Validating the Mack and England-Verrall Models
- Searching for stochastic models that do validate
- Correlated Chain Ladder (CCL) Model
- Bayesian Loss Reserving Models for Incremental Paid Loss Data
 - The problem of negative incremental losses
 - The skew normal distribution
 - The Correlated Incremental Trend (CIT) Model
- Conclusions and Open Discussion

6

Preamble

Why Stochastic Loss Reserving

- Much everyday loss reserving practice is “pre-theoretical” in nature: based on spreadsheet projection methods originating before the availability of cheap computing power.
- Advantages:
 - Flexible
 - Easy to learn/explain
 - Places appropriate emphasis on the need for expert judgment and knowledge of the business context behind the data
 - Avoids common pitfall of model complexity for the sake of model complexity
- Disadvantages:
 - Prone to over-fit small datasets.
 - No concept of “model criticism”
 - Some procedures are equivalent to statistical procedures that might seem arbitrary when assumptions are viewed in the light of day
 - Produce point estimates... but we are ultimately interested in predictive distributions of ultimate losses.
 - (“No probabilities in, no probabilities out.”)

The Ultimate Issue

- “Given any value (estimate of future payments) and our current state of knowledge, what is the probability that the final payments will be no larger than the given value?”
 - Casualty Actuarial Society
 - Working Party on Quantifying Variability in Reserve Estimates, 2004
- This can be read as a request for a Bayesian analysis.
- We ultimately would like to estimate a posterior probability distribution of the aggregate future payments random variable.
- Premise: not all stochastic reserving frameworks are created equal.
 - We want to avoid overly “procedural” data analytic approaches to stochastic loss reserving.
 - Simply moving from “methods” to “models” is not the answer.
 - We want a “*modeling methodology*” that offers a formal framework for (a) modeling the data-generating process and (b) incorporating prior knowledge into the analysis.
 - **Enter modern Bayesian data analysis.**

9

Why Bayes, Why Now

From John Kruschke, Indiana University:

“An open letter to Editors of journals, Chairs of departments, Directors of funding programs, Directors of graduate training, Reviewers of grants and manuscripts, Researchers, Teachers, and Students”:

Statistical methods have been evolving rapidly, and many people think it's time to adopt modern Bayesian data analysis as standard procedure in our scientific practice and in our educational curriculum. Three reasons:

1. Scientific disciplines from astronomy to zoology are moving to Bayesian data analysis. **We should be leaders of the move, not followers.**
2. Modern Bayesian methods provide richer information, with greater flexibility and broader applicability than 20th century methods. Bayesian methods are intellectually coherent and intuitive. **Bayesian analyses are readily computed with modern software and hardware.**
3. Null-hypothesis significance testing (NHST), with its reliance on p values, has many problems. **There is little reason to persist with NHST now that Bayesian methods are accessible to everyone.**

My conclusion from those points is that we should do whatever we can to encourage the move to Bayesian data analysis.

(I couldn't have said it better myself...)

10

Bayesian Concepts

Vocabulary – Preview

These are some of the concepts we will discuss and illustrate as the day progresses.

- “Evidential” (“subjective”) probability vs limiting relative frequency
- Credible intervals vs confidence intervals (informal discussion)
- Posterior and predictive distributions
- Shrinkage / Credibility
- Hierarchical models
- “Borrowing strength”
- Markov Chain Monte Carlo Simulation

How Frequentist and Bayesian Inference Differs

- The methodological differences between frequentists and Bayesians emanate from the philosophical difference about the interpretation of probability.
- As an example – consider the statement: “the probability that a tossed coin will land heads is $\frac{1}{2}$.”
- **Frequentists:** the “true probability of heads” is a fact about the world that is manifested in relative frequencies in repeated tosses.
 - The outcome of (say) 3 heads in 12 tosses is one of many possible outcomes of sampling from the “true distribution in the sky”.
 - **Probability is assigned to the data... not to model parameters**
- **Bayesians:** the data is a fact in the world. We assign probabilities to quantities we are uncertain about...
 - Probabilities are not assigned to data (although we can incorporate observation errors/sampling mechanisms in a model).
 - Rather, **probabilities are assigned to model parameters** which we do not know with certainty.
 - “Evidential probability” (aka “subjective probability”)

15

Updating Subjective Probability

- Bayes’ **Theorem** (a mathematical fact):

$$\Pr(H | E) = \frac{\Pr(H \wedge E)}{\Pr(E)} = \frac{\Pr(E | H) \Pr(H)}{\Pr(E)}$$

- Bayes’ **updating rule** (a methodological premise):
- Let $P(H)$ represents our belief in hypothesis H before receiving evidence E .
- Let $P^*(H)$ represent our belief about H after receiving evidence E .
- **Bayes Rule:** $P^*(H) = \Pr(H|E)$

$$\Pr(H) \xrightarrow{E} \Pr(H | E)$$

16

Bayesian Computation

Why Isn't Everyone a Bayesian?

Why Isn't Everyone a Bayesian?

B. EFRON*

Originally a talk delivered at a conference on Bayesian statistics, this article attempts to answer the following question: why is most scientific data analysis carried out in a non-Bayesian framework? The argument consists mainly of some practical examples of data analysis, in which the Bayesian approach is difficult but Fisherian/frequentist solutions are relatively easy. There is a brief discussion of objectivity in statistical analyses and of the difficulties of achieving objectivity within a Bayesian framework. The article ends with a list of practical advantages of Fisherian/frequentist methods, which so far seem to have outweighed the philosophical superiority of Bayesianism.

*B. Efron is Professor, Department of Statistics, Stanford University, Stanford, CA 94305.

© 1986 American Statistical Association

Why Isn't Everyone a Bayesian?

- Given that the Bayesian framework is so great, why isn't it used more in practice?
- **Answer 1:** Actually, it is... things have changed rapidly.
- **Answer 2:** Thoughts on why frequentism has been dominant.
 - (Jim's speculation): Cognitive biases... failures of probabilistic reasoning
 - E.g. the Monty Hall problem, the prosecutor's fallacy, Kahneman's blue taxis
 - Much of classical statistics is "automatic" in ways that can be programmed into canned software packages (PROCs).
 - Argument that Bayesian statistics is "subjective" and science isn't "subjective".
 - **Bayesian computation has traditionally been very difficult.**
 - Pre-1990s: Bayesian practice was largely limited to ad hoc credibility formulas and conjugate prior relationships.

19

Why Bayesian Computation is Difficult

- Remember Bayes' Theorem:

$$f(\theta | X) = \frac{f(X | \theta)\pi(\theta)}{\int f(X | \theta)\pi(\theta)d\theta}$$

The great virtue of the Bayesian framework:

- It enables us to calculate a **predictive distribution** for future outcomes Y given past outcomes X : $f(Y|X)$
 - E.g. in loss reserving, we can get a predictive distribution of future claim payments Y given a loss triangle of past payments X .

$$f(Y | X) = \int f(Y | \theta)f(\theta | X)d\theta = \int f(Y | \theta) \left(\frac{f(X | \theta)\pi(\theta)}{\int f(X | \theta)\pi(\theta)d\theta} \right) d\theta$$

- But in practice all of this integration is intractable... impasse.

20

A New World Order

- This impasse came to an end ~1990 when a simulation-based approach to estimating posterior probabilities was introduced.
 - (Circa the fall of the Soviet empire and Francis Fukuyama's "end of history")

Sampling-Based Approaches to Calculating Marginal Densities

ALAN E. GELFAND AND ADRIAN F. M. SMITH*

© 1990 American Statistical Association
Journal of the American Statistical Association
June 1990, Vol. 85, No. 410, Theory and Methods

21

What is Markov Chain Monte Carlo?

- **Markov chain:** a type of stochastic process in which each future state is independent of each past state, conditional upon the present state.
 - Intuitively: once you know the present state, information about past states contain no additional information useful for predicting the future.
 - For us the space of states will be a parameter space
 - We will construct Markov chains that will wander around parameter space....
 - ... and use these chains to do Monte Carlo simulation
- **Monte Carlo:** stochastic simulation
- Monte Carlo simulation is already familiar, so let's discuss these concepts in reverse order.

22

Why Traditional Monte Carlo Isn't Enough

- Monte Carlo simulation is all well and good when we can write down the probability distribution in a computer program.
 - It enables to generate iid draws from the distribution of interest...
 - ... and the Strong Law of Large Numbers implies that the Monte Carlo estimate will converge to the true value of the integral with probability 1.
- But the problem in Bayesian computation is that **we generally can't write down an expression for the posterior probability distribution**.
- Specifically: the integral in the denominator gets very nasty very quickly... especially when θ is a vector of parameters...

$$f(\theta | X) = \frac{f(X | \theta)\pi(\theta)}{\int f(X | \theta)\pi(\theta)d\theta}$$

- We therefore turn to the theory stochastic processes.
- This will enable us to bypass the independence requirement of MC integration.

23

Markov Chains – Definitions

- **Stochastic process**: a time-indexed set of random variables $\{X_i\}$ defined on a space of states $\Omega = \{x_1, x_2, \dots\}$.
 - For us Ω will be a parameter space.
- **Markov chain**: is a stochastic process that satisfies:

$$\Pr(X_t = y | X_{t-1} = x, \dots, X_1 = x_1) = \Pr(X_t = y | X_{t-1} = x) \equiv P(x, y)$$

- In words: the probability of an event in the chain depends only on the immediately previous event.
- P is called a *transition matrix* and represents the Markov chain
- P gives the probability of moving from each possible state at time t to each possible state at time $t+1$.
 - If the state space has a finite number k values, then P is a k -by- k matrix of transition probabilities

$$P_{i,j} = \Pr(X_t = i | X_{t-1} = j)$$

24

Illustration of Metropolis-Hastings Sampling

A Random Walk Down Parameter Lane

- Recall: we can't do Monte Carlo because in general we can't write down the posterior probability density $f(\theta|X)$.
- But what if we could set up a random walk through our parameter space that... in the limit... passes through each point in the probability space in proportion to the posterior probability density.
- **If we could**, then we could just use the most recent $\times 1000$ steps of that random walk as a good approximation of the posterior density...
- **Yes we can!**



Chains We Can Believe In

- The **Metropolis-Hastings sampler** generates a **Markov chain** $\{\theta_1, \theta_2, \theta_3, \dots\}$ in the following way:

- Time $t=1$: select a random initial position θ_1 in parameter space.
- Select a **proposal distribution** $p(\theta)$ that we will use to select proposed random steps away from our current position in parameter space.
- Starting at time $t=2$: repeat the following until you get convergence:
 - At step t , generate a proposed $\theta^* \sim p(\theta)$
 - Also generate $u \sim \text{unif}(0,1)$
 - If $u < R$ then $\theta_t = \theta^*$. Else, $\theta_t = \theta_{t-1}$.

$$R = \frac{f(\theta^* | X) \cdot p(\theta_{t-1} | \theta^*)}{f(\theta_{t-1} | X) \cdot p(\theta^* | \theta_{t-1})}$$

(R is known as the **acceptance ratio**.)

- Step 3c) implies that at step t , we accept the proposed step θ^* with probability $\min(1, R)$.

27

Making Bayesian Computation Practical

- At each step we flip a coin with probability of heads $\min(1, R)$ and accept θ^* if the coin lands heads.
 - Otherwise reject θ^* and stay put at θ_{t-1} .
- But why is this any easier? R contains the dreaded posterior density $f(\theta|X)$ that we can't write down.

$$R = \frac{f(\theta^* | X) \cdot p(\theta_{t-1} | \theta^*)}{f(\theta_{t-1} | X) \cdot p(\theta^* | \theta_{t-1})}$$

28

Making Bayesian Computation Practical

- At each step we flip a coin with probability of heads $\min(1, R)$ and accept θ^* if the coin lands heads.
 - Otherwise reject θ^* and stay put at θ_{t-1} .
- But why is this any easier? R contains the dreaded posterior density $f(\theta|X)$ that we can't write down.

$$R = \frac{f(\theta^* | X)}{f(\theta_{t-1} | X)} \cdot \frac{p(\theta_{t-1} | \theta^*)}{p(\theta^* | \theta_{t-1})}$$

- Here's why:

$$R = \frac{\frac{f(X | \theta^*)\pi(\theta^*)}{\int f(X | \vartheta)\pi(\vartheta)d\vartheta}}{\frac{f(X | \theta_{t-1})\pi(\theta_{t-1})}{\int f(X | \vartheta)\pi(\vartheta)d\vartheta}} \cdot \frac{p(\theta_{t-1} | \theta^*)}{p(\theta^* | \theta_{t-1})}$$

29

Making Bayesian Computation Practical

- At each step we flip a coin with probability of heads $\min(1, R)$ and accept θ^* if the coin lands heads.
 - Otherwise reject θ^* and stay put at θ_{t-1} .
- But why is this any easier? R contains the dreaded posterior density $f(\theta|X)$ that we can't write down.

$$R = \frac{f(\theta^* | X)}{f(\theta_{t-1} | X)} \cdot \frac{p(\theta_{t-1} | \theta^*)}{p(\theta^* | \theta_{t-1})}$$

- Here's why:

The integrals in the denominator of Bayes theorem cancel out... they are functions only of the data X , not the parameters θ .

We have re-written R in terms of the likelihood function $f(X|\theta)$, and the prior $\pi(\theta)$.

$$R = \frac{\frac{f(X | \theta^*)\pi(\theta^*)}{\int f(X | \vartheta)\pi(\vartheta)d\vartheta}}{\frac{f(X | \theta_{t-1})\pi(\theta_{t-1})}{\int f(X | \vartheta)\pi(\vartheta)d\vartheta}} \cdot \frac{p(\theta_{t-1} | \theta^*)}{p(\theta^* | \theta_{t-1})}$$

$$= \frac{f(X | \theta^*)\pi(\theta^*)}{f(X | \theta_{t-1})\pi(\theta_{t-1})} \cdot \frac{p(\theta_{t-1} | \theta^*)}{p(\theta^* | \theta_{t-1})}$$

30

Now We Can Go to the Metropolis

- So now we have something we can easily program into a computer.
- At each step, give yourself a coin with probability of heads $\min(1, R)$ and flip it.

$$R = \frac{f(X | \theta^*) \pi(\theta^*)}{f(X | \theta_{t-1}) \pi(\theta_{t-1})} \cdot \frac{p(\theta_{t-1} | \theta^*)}{p(\theta^* | \theta_{t-1})}$$

- If the coin lands heads move from θ_{t-1} to θ^*
- Otherwise, stay put.
- The result is a Markov chain (step t depends only on step $t-1$... not on prior steps). And it converges on the posterior distribution.

31

Simple Illustration

- Let's illustrate MH via a simple example.
- "Target" density that we wish to simulate: the lognormal.

$$f(x | \mu, \sigma) = \frac{1}{x\sigma\sqrt{2\pi}} \exp\left(-\frac{z^2}{2}\right), \quad z = \frac{\ln(x) - \mu}{\sigma}$$

- We take logs so that we add/subtract rather than multiply/divide

- "Target" "density":

- As noted before, we can eliminate terms that cancel out

$$\log f(x, \mu, \sigma) = -\ln(\sigma) - 0.5 \left(\frac{\log(x) - \mu}{\sigma} \right)^2$$

- Proposal densities:

- The proposal (μ^*, σ) is a standard normal step away from the current location.

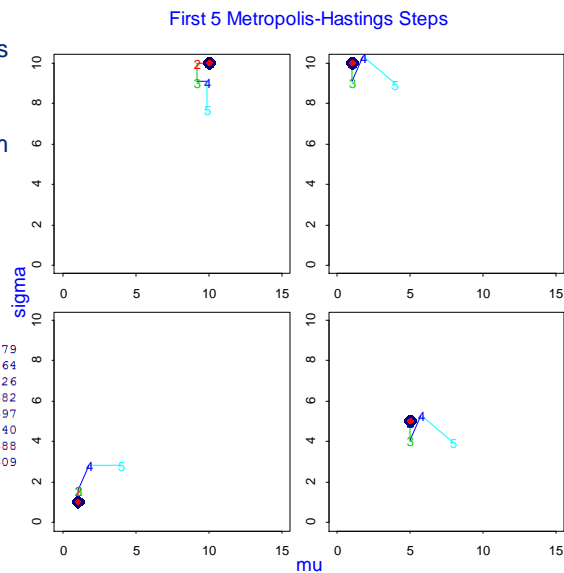
$$p(\mu^* | \mu_{t-1}) = N(\mu_{t-1}, 1) \quad ; \quad p(\sigma^* | \sigma_{t-1}) = N(\sigma_{t-1}, 1)$$

32

Random Walks with 4 Different Starting Points

- We estimate the lognormal density using 4 separate sets of starting values.
- Data: 50 random draws from $\text{lognormal}(9,2)$.

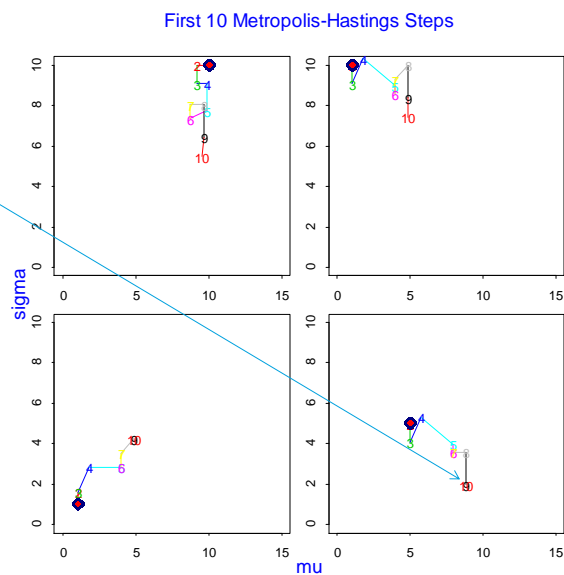
```
> round(xx) [order(xx)]
[1] 50 210 443 561 596 779
[7] 1037 1544 2365 2480 2749 2764
[13] 2865 2947 3007 3440 3599 4226
[19] 4348 4770 4962 5411 6438 6682
[25] 7128 7612 8555 9260 9697 9697
[31] 10486 11380 13630 17910 19014 25840
[37] 28737 35448 38379 50122 60746 78688
[43] 94977 97028 98491 139625 143219 199609
[49] 494979 662527
```



33

Random Walks with 4 Different Starting Points

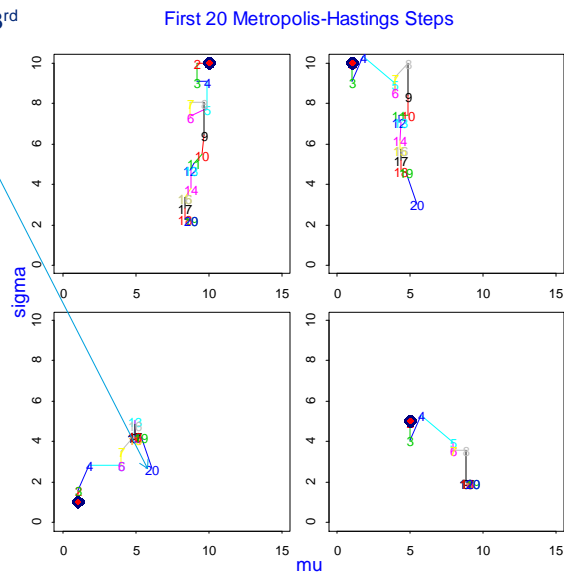
- After 10 iterations, the lower right chain is already in the right neighborhood.



34

Random Walks with 4 Different Starting Points

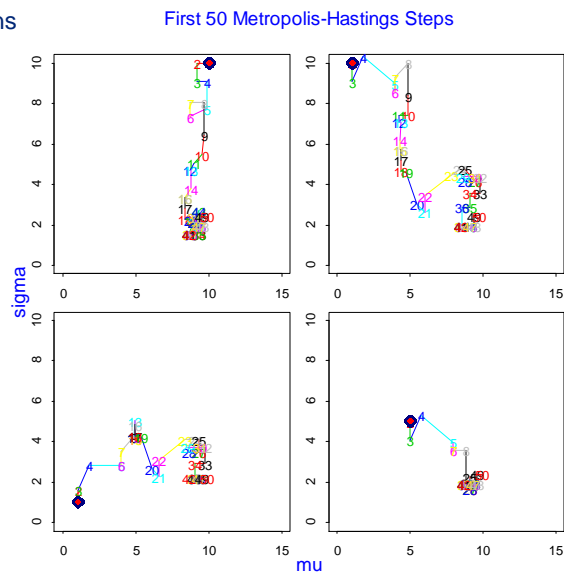
- After 20 iterations, only the 3rd chain is still in the wrong neighborhood.



35

Random Walks with 4 Different Starting Points

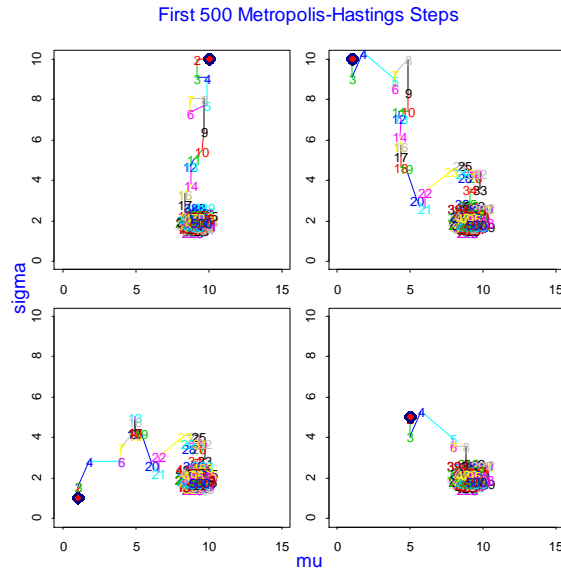
- After 50 iterations, all 4 chains have arrived in the right neighborhood.



36

Random Walks with 4 Different Starting Points

- By 500 chains, it appears that the *burn-in* has long since been accomplished.
- The chain continues to wander.
- The time the chain spends in a neighborhood approximates the posterior probability that (μ, σ) lies in this neighborhood.

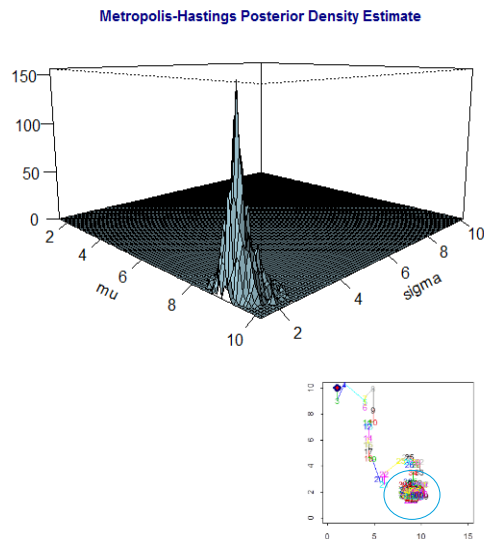


37

In 3D

- The true lognormal parameters are: $\mu=9$ and $\sigma=2$
- The MH algorithm yields an estimate of the posterior density:

$$f(\mu, \sigma | X_1, X_2, \dots, X_{50})$$
- This density results from a diffuse prior
- It is based on the information available in the data.



38

Metropolis-Hastings Results

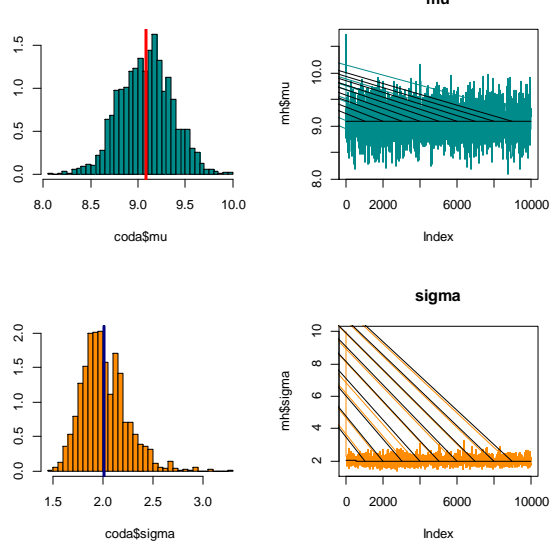
- The true lognormal parameters are:
 $\mu=9$ and $\sigma=2$
- The MH simulation is gives consistent results:

```
> apply(coda, 2, mean)
      mu      sigma
9.077489 2.007377
> apply(coda, 2, sd)
      mu      sigma
0.2741341 0.2247070
```

- Only the final 5000 of the 10000 MH iterations were used to estimate μ, σ
 - (This motivates the use of the musical term "coda")

39

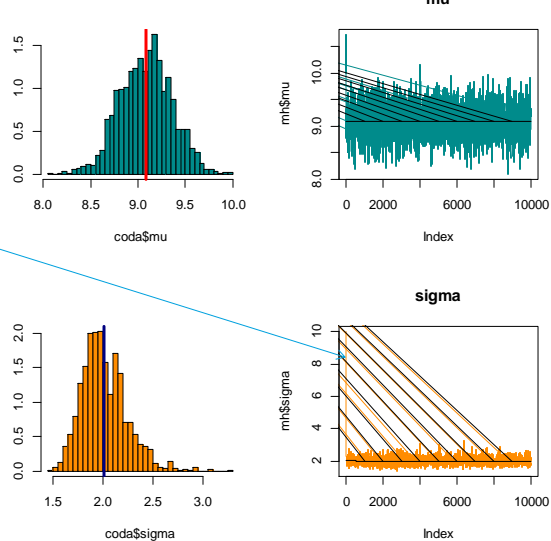
Metropolis-Hastings Simulation of Lognormal(9,2)



Metropolis-Hastings Results

- The true lognormal parameters are:
 $\mu=9$ and $\sigma=2$
- Note the very rapid convergence despite unrealistic initial values.

Metropolis-Hastings Simulation of Lognormal(9,2)



40

Some MCMC Intuition

Metropolis-Hastings Intuition

- Let's take a step back and remember why we've done all of this.
- In ordinary Monte Carlo integration, we take a large number of independent draws from the probability distribution of interest and let the sample average of $\{g(\theta_i)\}$ approximate the expected value $E[g(\theta)]$.

$$\frac{1}{N} \sum_{i=1}^N g(\theta^{(i)}) \xrightarrow{N \rightarrow \infty} \int g(\theta) \pi(\theta) d\theta = E[g(\theta)]$$

- The Strong Law of Large Numbers justifies this approximation.
- But: when estimating Bayesian posteriors, we are generally not able to take independent draws from the distribution of interest.
- Results from the theory of stochastic processes tell us that suitably well-behaved Markov Chains can *also* be used to perform Monte Carlo integration.

Some Facts from Markov Chain Theory

How do we know this algorithm yields reasonable approximations?

- Suppose our Markov chain $\theta_1, \theta_2, \dots$ with transition matrix P satisfies some “reasonable conditions”:
 - Aperiodic, irreducible, positive recurrent (more on these in a moment)
 - Chains generated by the M-H algorithm satisfy these conditions
- **Fact #1 (convergence theorem):** P has a unique stationary (“equilibrium”) distribution, π . (i.e. $\pi = \pi P$). Furthermore, the chain converges to π .
 - Implication: We can start anywhere in the sample space so long as we through out a sufficiently long “burn-in”.
- **Fact #2 (Ergodic Theorem):** suppose $g(\theta)$ is some function of θ . Then:

$$\frac{1}{N} \sum_{i=1}^N g(\theta^{(i)}) \xrightarrow{N \rightarrow \infty} \int g(\theta) \pi(\theta) d\theta = E[g(\theta)]$$

- Implication: After a suitable burn-in, perform Monte Carlo integration by averaging over a suitably well-behaved Markov chain.
- The values of the chain are *not* independent, as required by the SLLN.
- But the Ergodic Theorem says we’re close enough to independence to get what we need.

43

Conditions for Ergodicity

More on those “reasonable conditions” on Markov chains:

- **Aperiodic:** The chain does *not* regularly return to any value θ in the state space in multiples of some $k > 1$.
- **Irreducible:** It is possible to go from any state θ_i to any other state θ_j in some finite number of steps.
- **Positive recurrent:** The chain will return to any particular state θ with probability 1, and expected return time finite.
- **Intuition:**
 - *The Ergodic Theorem tells us that (in the limit) the amount of time the chain spends in a particular region of state space equals the probability assigned to that region.*
 - *This won’t be true if (for example) the chain gets trapped in a loop, or won’t visit certain parts of the space in finite time.*
- **The practical problem:** use the Markov chain to select a representative sample from the distribution π , expending a minimum amount of computer time.

44

Tuning the Metropolis Hastings Algorithm

A Tweedie Example

- $E[X] = \mu$, $Var[X] = \phi \cdot \mu^p$
- We are given that $\phi = 1$, $p = 1.5$ and μ is unknown
- Given the data:

Loss Amount	0	1	2	3	5	8	10	12	16
Number	8	6	2	2	2	1	1	1	2

- Find the predictive distribution of μ and X

The Metropolis-Hastings Algorithm

1. Select a starting value μ_1
2. For $t = 2, \dots$, select a candidate value, μ^* , at random from the proposal density distribution.

$$p(\mu | \mu_{t-1}, \alpha) = \Gamma(\mu | \mu_{t-1} / \alpha, \alpha)$$

$$\text{Note } E[\mu] = \mu_{t-1} \text{ and } CV[\mu] = \frac{1}{\sqrt{\alpha}}$$

3. Calculate the ratio

$$R = \frac{f(\mathbf{x} | \mu^*) \cdot \pi(\mu^*)}{f(\mathbf{x} | \mu_{t-1}) \cdot \pi(\mu_{t-1})} \cdot \frac{p(\mu_{t-1} | \mu^*, \alpha)}{p(\mu^* | \mu_{t-1}, \alpha)}$$

$\pi(\mu)$ is a Γ distribution with mean = 5 and standard deviation = 5

$$f(\mathbf{x} | \mu) = \prod_{i=1}^{25} \text{tweedie}(x_i | \mu, \rho, \phi)$$

4. Select the value, U , at random from a uniform distribution.
5. If $U < R$ then $\mu_t = \mu^*$, else $\mu_t = \mu_{t-1}$

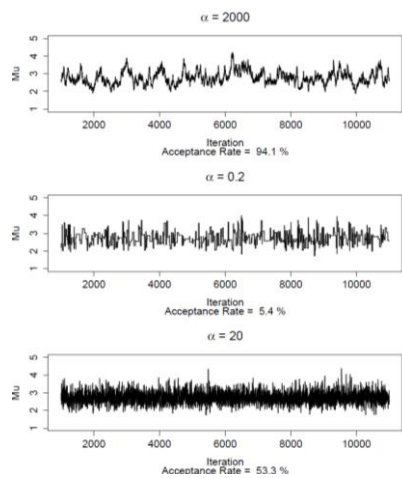
47

Tuning the Metropolis Hastings Algorithm

- Run the “MH Example Tuning.R” script
- Choose “burn in” period = 1,000 iterations
- Run 10,000 additional iterations
- Choose α ranging from 0.2 to 2000
 - Large α means that μ^* is “close” to μ_{t-1} , so R is “close” to 1
 - Acceptance ($\mu_t = \mu^*$) is likely
 - Small α means that μ^* could be “far” from μ_{t-1} , so R could be less than 1
 - Rejection ($\mu_t = \mu_{t-1}$) is likely
- There are “optimal” rejection rates
 - 50% for one parameter, and decreasing to 25% for many parameters

48

Trace plots for different values of α

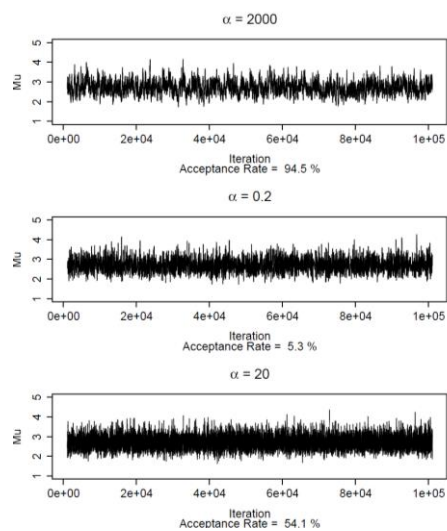


- Tuning by trial and error – this example
- Mechanical or “adaptive” tuning – JAGS

49

When Tuning Doesn't Work - Thinning

- Run longer chain and take every k^{th} iteration
- Our example with $k = 10$

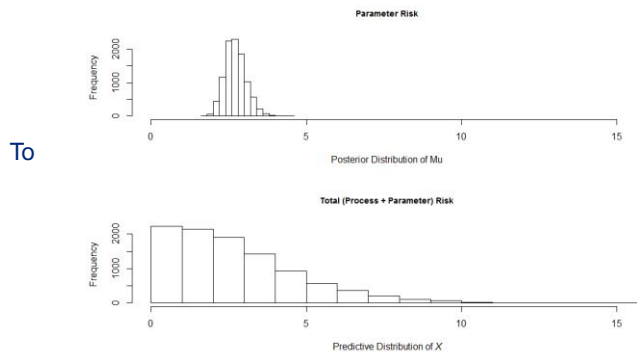


50

Back to the problem – Predictive distributions of μ and X

- MH Algorithm produces a sample from the posterior distribution of μ
- For each μ in the sample, simulate a random variable, x , from a Tweedie distribution with parameters $\phi = 1$, $\rho = 1.5$ and mean μ .

From	Loss Amount	0	1	2	3	5	8	10	12	16
	Number	8	6	2	2	2	1	1	1	2



51

A JAGS Example Adaptive Tuning and Convergence Testing

A Simple Example with JAGS

Predict the Distribution of the Outcomes of a Time Series of Loss Ratios for the Next 5 Years

The Data

Accident Year	1	2	3	4	5	6	7	8	9	10
Loss Ratio	0.685	0.762	0.737	0.735	0.848	0.665	0.545	0.644	0.557	0.671

The Model

- $LR_t \sim \text{normal}(ELR_t, \sigma)$
 - $ELR_1 \sim \text{uniform}(0.5, 1.5)$
 - $ELR_t = z \cdot LR_{t-1} + (1-z) \cdot ELR_{t-1}$
 - $z \sim \text{uniform}(0, 1)$
 - $\sigma \sim \text{uniform}(0, 0.25)$
-
- True parameters – ELR_1, z, σ (i.e. those parameters with prior distributions)
 - Derived parameters – ELR_2, \dots, ELR_{10}

53

General Structure of an R/JAGS Script Created by Meyers

1. Get data
2. Create JAGS object – calls a separate text file with JAGS script
 - Specify data
 - Specify (adaptive) tuning period
 - Thinning parameter
 - Setting a fixed random number seed
 - Specify the number of chains (Why does this matter?)
3. Update the JAGS object (Burning Period)
 - Burn until chain converges
 - Question – What do we mean by “converge?”
4. Take the sample
 - I use the “coda” package (distributed with “rjags”)
5. Construct statistics of interest and produce output

Comment – No unique way to do these analyses. My approach is to find something that “appears” to work and focus on problems of interest to actuaries.

54

In Rstudio - Open “ELR JAGS Example.R” Script

- Run the script and explore output
 - Run with “n.adapt = 10”
 - Discuss “convergence” - I use the Gelman-Rubin convergence diagnostic.
1. Run multiple chains in JAGS
 2. Estimate the average within-chain variability, W
 3. Estimate the between-chain variability, B
 4. Calculate the “Potential Scale Reduction Factor” or PSRF

$$\sqrt{R} = \sqrt{\frac{W+B}{W}} \rightarrow 1 \quad \text{Gelman and Rubin } < 1.2 \text{ is OK.}$$

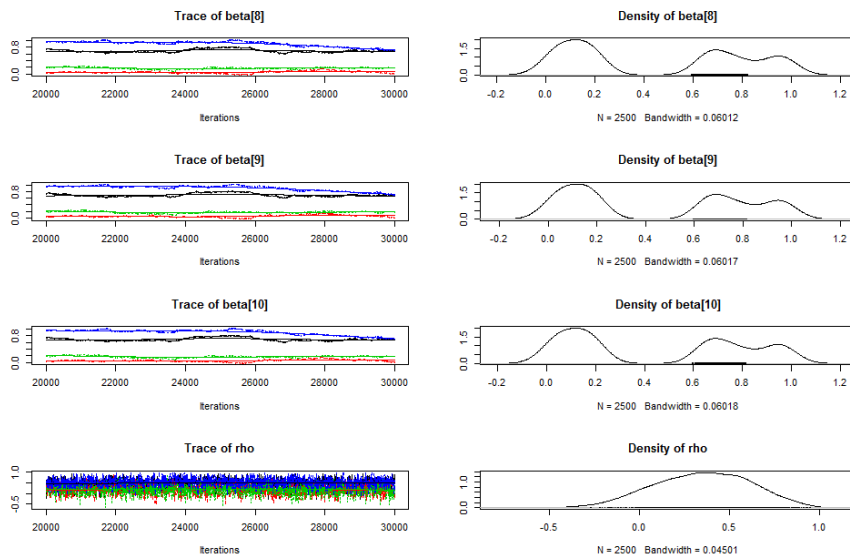
Brooks and Gelman “General Methods for Monitoring Convergence of Iterative Simulations” describe a “Multivariate PSRF.” < 1.2 is OK

Gelman Plots

1. PSRF for iterations 1-50
2. PSRF for iterations 1-100
3. Etc.

55

In practice, bad results can happen – MPSRF = 7.88 A preview of things to come.



Case Studies

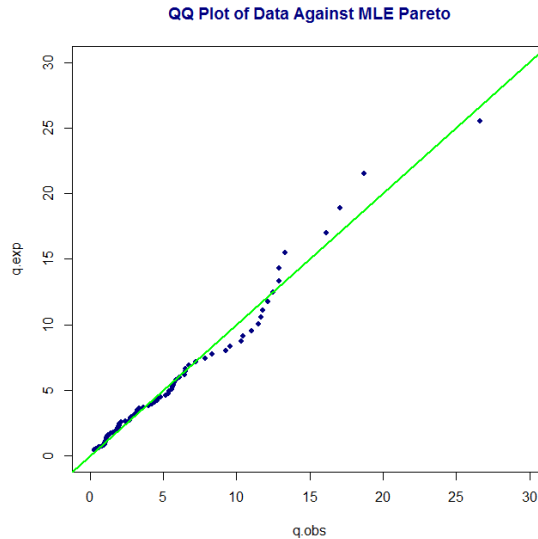
Case Study #1

Loss Models

Exploratory Data Analysis

- Motivated by the two outliers in the Gamma analysis, let's fit a Pareto.
- The fit is still ambiguous, but the heavier tailed Pareto seems more consistent with the data.

```
> p0 <- c(1,1)
> f <- function(x, p) p[1]*p[2]^p[1] / (x+p[2])^(p[1]+1)
> loglik <- function(x, p) -sum(log(f(x,p)))
> MLE <- optim(p0, loglik, x=x)
> MLE[[1]]
[1] 3.720533 13.719413
```



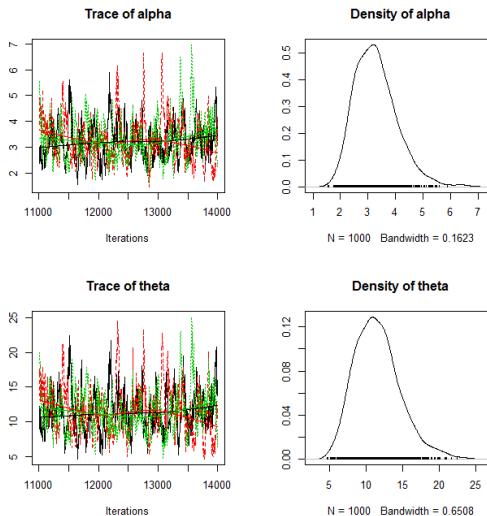
61

Bayesian Analysis

- We will assume that the data is Pareto distributed.
- Given this assumption, what can we infer about {} given the data?
- Technical note: JAGS provides only a 1-parameter Pareto function (dpar). We therefore use the fact that a Pareto is a gamma mixture of exponentials.

```
model {
  for (i in 1:n) {
    x[i] ~ dgamma(1, lambda[i])
    lambda[i] ~ dgamma(alpha, theta)
  }
  alpha ~ dunif(0, 100)
  theta ~ dgamma(10, 1)
}
```

	2.5%	25%	50%	75%	97.5%
alpha	2.043	2.720	3.218	3.737	5.073
theta	6.377	9.288	11.270	13.368	18.993

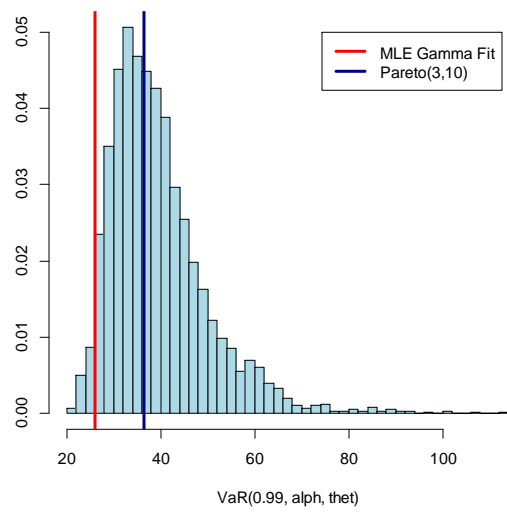


Posterior Distribution VaR₉₉ Estimates

```
round(x) [order(x)]
[1] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1
[26] 1 1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 3 3
[51] 3 3 3 3 3 3 3 3 3 3 4 4 4 5 5 5 5 5 6 6 6 6 6 6 6 6 6 6
[76] 7 7 7 7 8 8 9 9 10 10 11 12 12 12 12 13 13 13 16 17 18 27 30 31
```

- If we had settled for our initial Gamma MLE fit, our estimate would have likely been way too low.
- Just reporting the VaR for a Pareto(3,10) fit doesn't tell the whole story either.
 - Parameter uncertainty results in widely divergent VaR estimates.
 - In real life, the next step would be to specify more informative priors...

Estimated Bayesian Posterior Distribution of 99% VaR



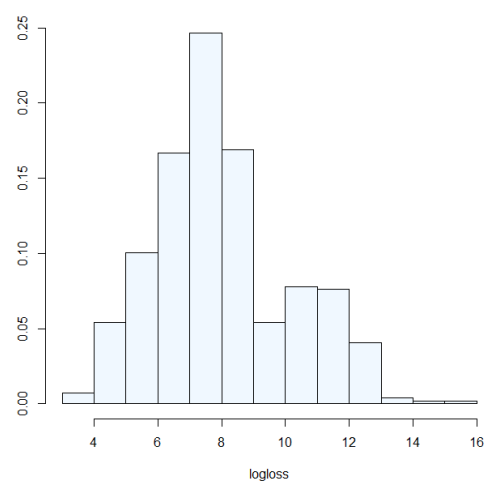
63

Loss Model Case Study #1b: Finite Mixtures

- Actual Project data:
- We are given 539 size-of-loss observations.
 - Distribution of logged losses plotted to right.

Min.	: 22
1st Qu.:	695
Median :	2174
Mean :	38027
3rd Qu.:	8995
Max. :	5007232

- What can we say about the distribution of these observations?



64

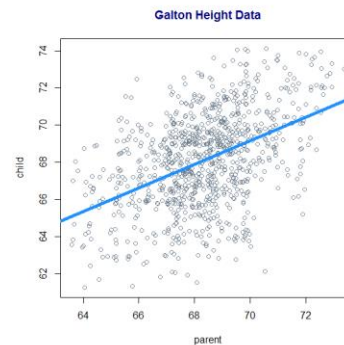
Case Study #2

Bayesian Regression Modeling

Bayesian Regression Case Study

- The classic dataset used to introduce ordinary least squares [OLS] regression is the Galton height data.
- We predict the height of the child using the height of the parent.
- Let's fit a Bayesian regression model to this data.

```
> summary(dat)
      child      parent
Min.   :61.26  Min.   :63.58
1st Qu.:66.40  1st Qu.:67.24
Median :68.17  Median :68.35
Mean   :68.09  Mean   :68.31
3rd Qu.:69.74  3rd Qu.:69.49
Max.   :74.13  Max.   :73.38
```

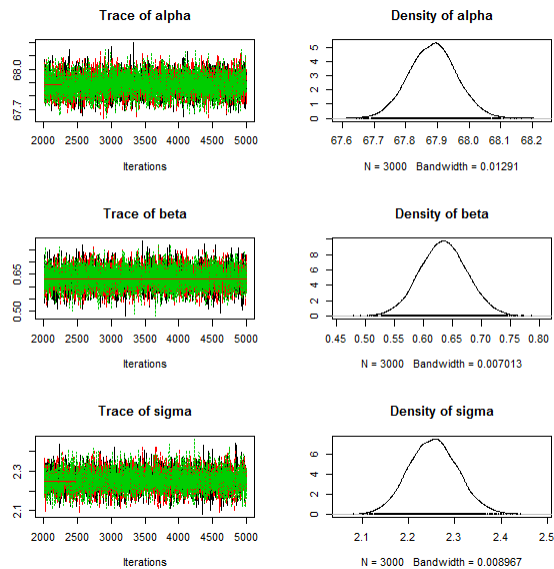


Results

- Bayesian posterior density estimate is well behaved and consistent with classical regression.

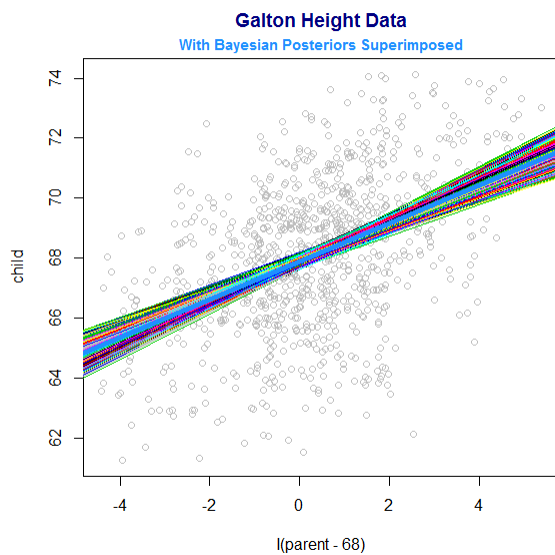
```
> summary(chains)[[1]][,1:2]
      Mean      SD
alpha 67.8856020 0.07526188
beta  0.6346204 0.04105279
sigma 2.2541855 0.05226225
> r2 <- lm(child ~ I(parent-68) , dat)
> summary(r2)$coefficients
              Estimate Std. Error
(Intercept)  67.8899738  0.07495994
I(parent - 68) 0.6342877  0.04067201
> summary(r2)$sigma
[1] 2.250903
```

67



Results

- We superimpose our draws from the simulated posterior on the original data.



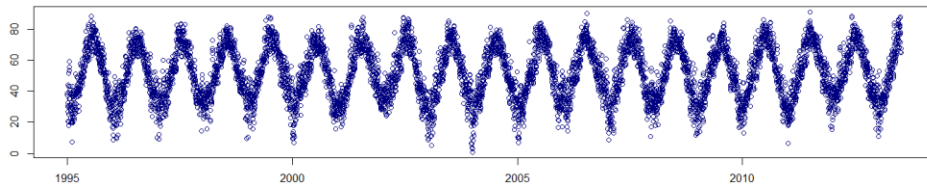
68

Bayesian Non-Linear Regression Case Study

- Data: daily 1995-2013 Boston average temperature observations
- Let's fit a non-linear Bayesian model on the data < 2011, test on remaining data.

```
> dim(dat)
[1] 6783 4
> summary(dat)
      m           d           y           avg.temp
Min.   : 1.000   Min.   : 1.00   Min.   :1995   Min.   : -99.00
1st Qu.: 3.000   1st Qu.: 8.00   1st Qu.:1999   1st Qu.: 38.30
Median : 6.000   Median :16.00   Median :2004   Median : 51.70
Mean   : 6.444   Mean   :15.72   Mean   :2004   Mean   : 51.29
3rd Qu.: 9.000   3rd Qu.:23.00   3rd Qu.:2008   3rd Qu.: 66.00
Max.   :12.000   Max.   :31.00   Max.   :2013   Max.   : 90.70
> DATE <- paste(dat$m, dat$d, dat$y, sep="/")
> dat$date <- as.Date(DATE, "%m/%d/%Y")
> summary(dat$date)
      Min.      1st Qu.      Median      Mean      3rd Qu.      Max.
"1995-01-01" "1999-08-23" "2004-04-14" "2004-04-14" "2008-12-04" "2013-07-27"
```

Daily Boston Average Temperature Measurements



Nonlinear Bayesian Model

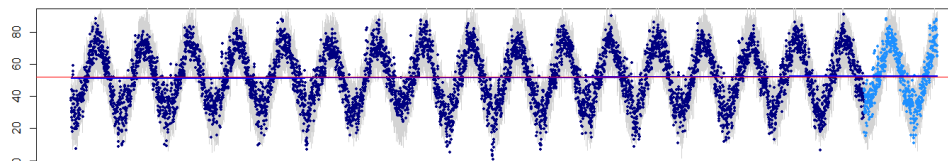
- Our model assumes normal dispersion around an underlying pattern that includes both a linear trend as well as seasonal variation.
 - The beta1 parameter is interesting.

```
model <- "model{
  for( i in 1:N ) {
    y[i] ~ dnorm( mu[i], 1/sigma^2)
    mu[i] <- beta0 + beta1*t[i] + alpha*cos(omega*t[i] + theta)
  }
  beta0 ~ dnorm(0, 0.0001)
  beta1 ~ dnorm(0, 0.0001)
  alpha ~ dunif(0, 100)
  omega ~ dunif(2*3.1415 * 0.95, 2*3.1415 * 1.05)
  theta ~ dunif(-3.1415, 3.1415)
  sigma ~ dunif(0, 100)
}"
```

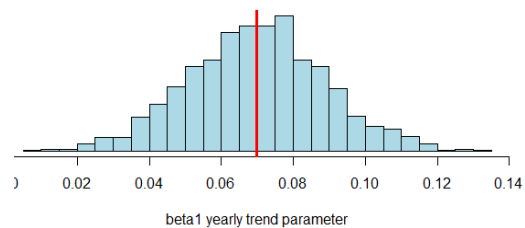
- t: number of years from arbitrary origin (April 1, 2004)
- alpha: amplitude of seasonal component
- omega: frequency (presumably 2π)
- theta: phase shift

Nonlinear Bayesian Model

- Grey lines: 20 draws from the posterior predictive distribution
- Dark blue dots: data used to fit the model
- Light blue dots: holdout data to test the model's predictions.



- The posterior distribution of β_1 suggests a gradual rise in temperature since 1995.



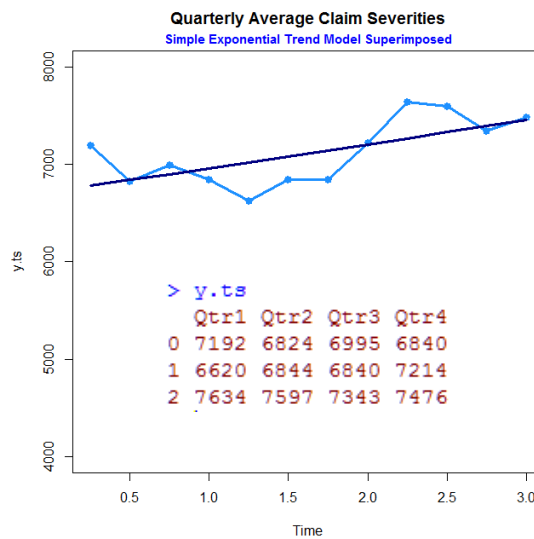
71

Case Study #3

Trend Analysis with Autocorrelation

Trend Analysis with Autocorrelation

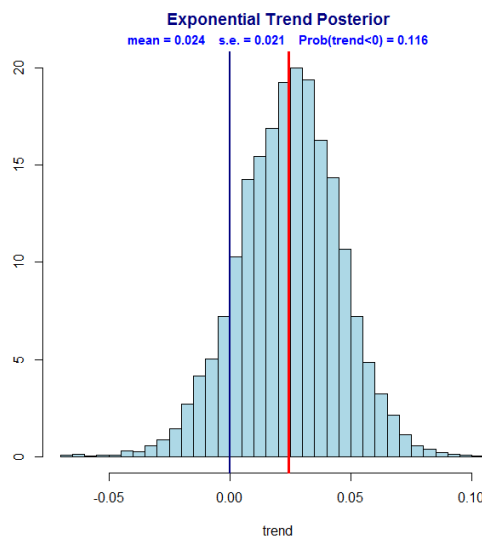
- Average claim severity time series from Dave Clark via Glenn Meyers' *Brainstorms* column.
- Let's build a Bayesian exponential trend model, incorporating autocorrelation.



73

Bayesian Trend Analysis with Autocorrelation

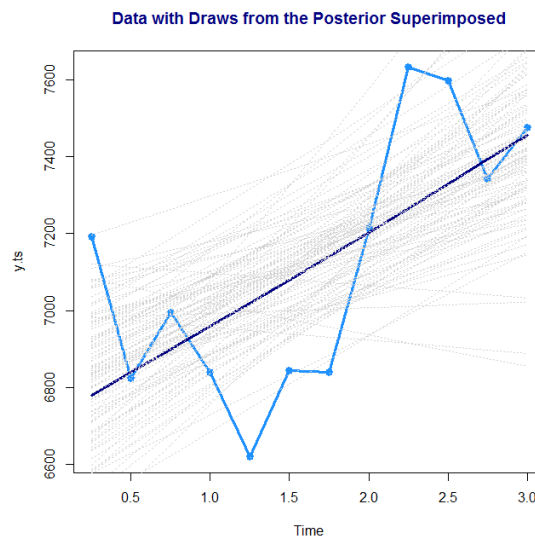
- Posterior Density estimate of the trend parameter.



74

Bayesian Trend Analysis with Autocorrelation

- We re-plot the data and superimpose 100 draws from the posterior.



75

Case Study #4

Bayesian Poisson Regression (Loss Reserving Warm-up)

Bayesian Poisson Regression

- To demonstrate Bayesian GLM, we will construct a Bayesian analog of the over-dispersed Poisson [ODP] model outlined in England-Verrall [2002]
- The ODP model is mathematically equivalent to the type of model commonly used in contingency table analysis.
- A over-dispersed Poisson GLM model with 20 covariates
 - One indicator variable for each accident year
 - One indicator variable for each development period
 - No intercept term
- Reserve variability can be estimated by bootstrapping residuals and re-running the model on the resulting pseud-datasets

77

Case Study Data

- A garden-variety Workers Comp Schedule P loss triangle:

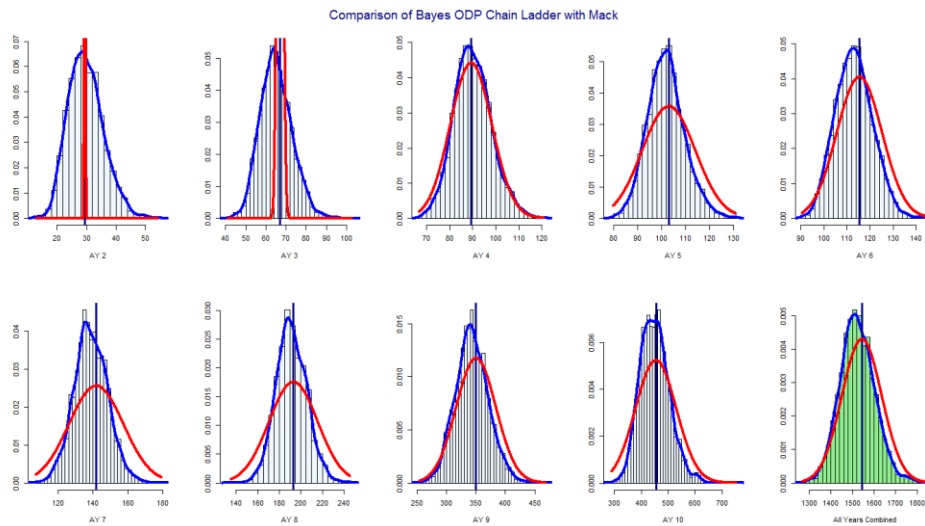
Cumulative Losses in 1000's														
AY	premium	12	24	36	48	60	72	84	96	108	120	CL Ult	CL LR	CL res
1988	2,609	404	986	1,342	1,582	1,736	1,833	1,907	1,967	2,006	2,036	2,036	0.78	0
1989	2,694	387	964	1,336	1,580	1,726	1,823	1,903	1,949	1,987		2,017	0.75	29
1990	2,594	421	1,037	1,401	1,604	1,729	1,821	1,878	1,919			1,986	0.77	67
1991	2,609	338	753	1,029	1,195	1,326	1,395	1,446				1,535	0.59	89
1992	2,077	257	569	754	892	958	1,007					1,110	0.53	103
1993	1,703	193	423	589	661	713						828	0.49	115
1994	1,438	142	361	463	533							675	0.47	142
1995	1,093	160	312	408								601	0.55	193
1996	1,012	131	352									702	0.69	350
1997	976	122										576	0.59	454
chain link		2.365	1.354	1.164	1.090	1.054	1.038	1.026	1.020	1.015	1.000	12,067		1,543
chain ldf		4.720	1.996	1.473	1.266	1.162	1.102	1.062	1.035	1.015	1.000			
growth curve		21.2%	50.1%	67.9%	79.0%	86.1%	90.7%	94.2%	96.6%	98.5%	100.0%			

- Let's model this as a longitudinal dataset.
- Grouping dimension: Accident Year (AY)
- We can build a parsimonious non-linear model that uses random effects to allow the model parameters to vary by accident year.

78

Results: Bayesian Poisson Regression

- Blue densities are density estimates of Bayesian MCMC posteriors
- Red densities are normal with mean, s.d. taken from Mack model results



Case Study #5

Bayesian Hierarchical Poisson Regression

Ratemaking Example

Data and Problem

- We have 7 years of Workers Comp data
 - For each of 7 years we are given payroll and claim count by class.
 - Let's build a Bayesian hierarchical Poisson GLM model on years 1-6 and compare the result with the actual claim counts from year 7.
 - Data is from Start Klugman 1992 book on Bayesian Statistics for actuarial science.

```

> dim(dat)
[1] 893 5
> round(nrow(dat)/7)
[1] 128
> summary(dat)
  class      year      payroll      clmcnt
Min.   : 1.00   Min.   :1.000   Min.   : 0.201   Min.   : 0.00
1st Qu.: 35.00   1st Qu.:2.000   1st Qu.: 75.521   1st Qu.: 1.00
Median : 69.00   Median :4.000   Median : 188.862   Median : 7.00
Mean   : 67.96   Mean   :4.009   Mean   : 713.064   Mean   : 17.49
3rd Qu.:101.00   3rd Qu.:6.000   3rd Qu.: 602.841   3rd Qu.: 21.00
Max.   :133.00   Max.   :7.000   Max.   :21163.600   Max.   :228.00

```

81

Exploratory Data Analysis

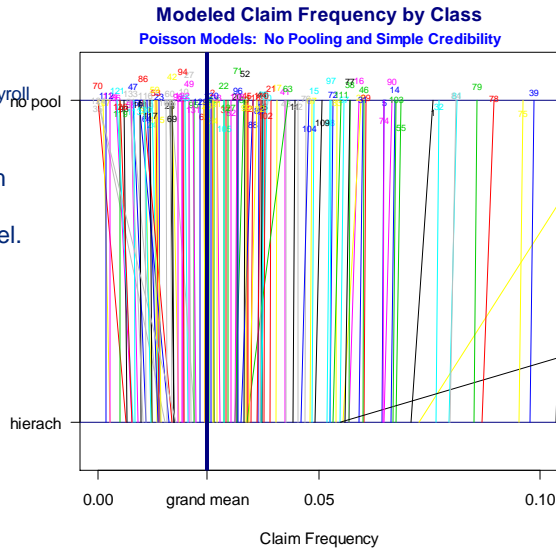
- The endgame is to build a Bayesian hierarchical GLM model.
- But in the spirit of data exploration, it makes sense to built empirical Bayes models first.
 - This is essentially a Bühlmann-Straub type credibility model.
 - This will help us get a feel for how much "shrinkage" (credibility-weighting) is called for.
 - Compare credibility weighted result with simply calculating empirical 6-year claim frequency by class.

$$\begin{aligned}
 clmcnt_i &\sim Poi(\text{payroll}_i \lambda_{j[i]}) \\
 \lambda_j &\sim N(\mu_\lambda, \sigma_\lambda^2)
 \end{aligned}$$

82

Shrinkage Effect of Hierarchical Model

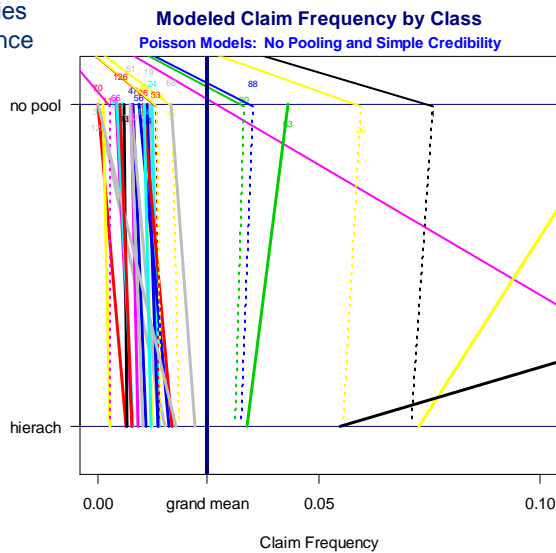
- Top row: estimated claim frequencies from un-pooled model.
 - Separately calculate #claims/payroll by class
- Bottom row: estimated claim frequencies from Poisson hierarchical (credibility) model.
- Credibility estimates are “shrunk” towards the grand mean.



83

Shrinkage Effect of Hierarchical Model

- Let's plot the claim frequencies only for classes that experience a shrinkage effect is 5% or greater.
 - Dotted line: shrinkage between 5=10%.
 - Solid line: shrinkage > 10%

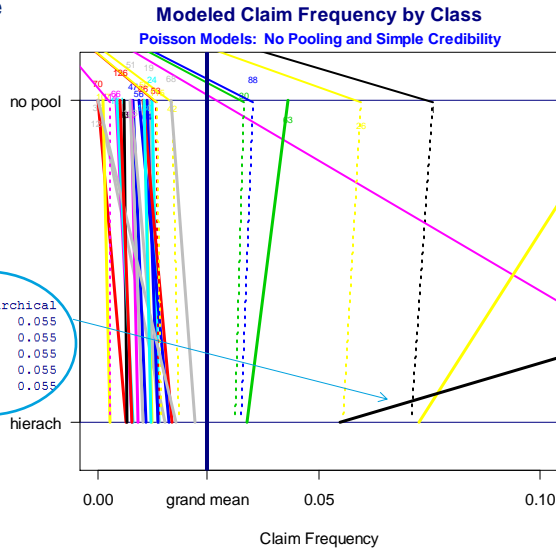


84

Shrinkage Effect of Hierarchical Model

- The most extreme shrinkage occurs for class 61.
 - Only 1 claim in years 3-6.
 - But very low payroll results in a large pre-shrunk estimated frequency.

class	year	payroll	clmcnt	freq	noPool	hierarchical
61	3	0.288	0	0.000	0.303	0.055
61	4	0.433	1	2.303	0.303	0.055
61	5	1.312	0	0.000	0.303	0.055
61	6	1.268	0	0.000	0.303	0.055
61	7	0.806	0	0.000	0.303	0.055

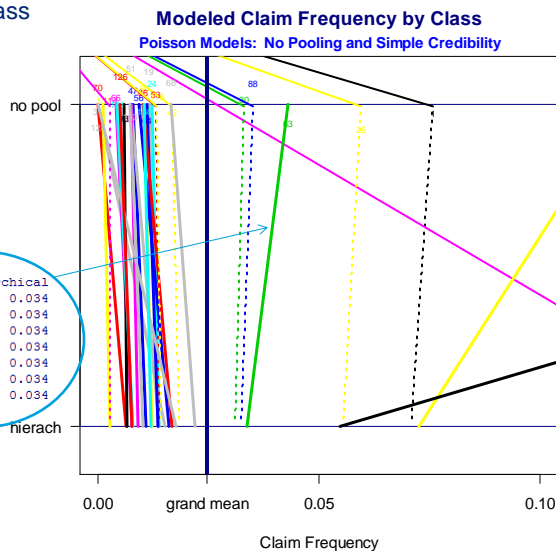


85

Shrinkage Effect of Hierarchical Model

- Shrinkage also occurs for class 63.
 - More payroll than class 61 but similar logic.

class	year	payroll	clmcnt	freq	noPool	hierarchical
63	1	3.119	0	0.0	0.043	0.034
63	2	3.685	0	0.0	0.043	0.034
63	3	3.764	0	0.0	0.043	0.034
63	4	3.831	0	0.0	0.043	0.034
63	5	4.993	1	0.2	0.043	0.034
63	6	3.780	0	0.0	0.043	0.034
63	7	2.618	0	0.0	0.043	0.034



86

Now Specify a Fully Bayesian Model

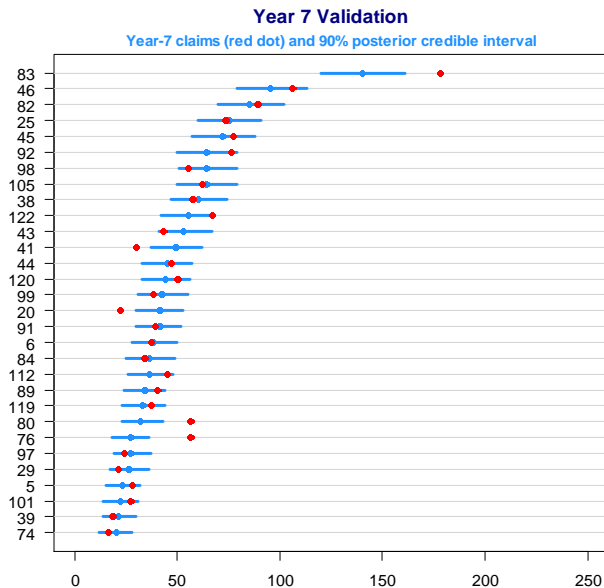
- Here we specify a fully Bayesian model.
 - Still Poisson regression with an offset ($y[i]$ is claim count)
 - Replace year-7 actual values with missing values so that we model the year-7 results and can compare actual with posterior credible interval.
 - Let's run and then criticize the model.

```
### Bayesian hierarchical model - first try
model = "model {
  for (i in 1:n) {
    y[i] ~ dpois( lambda[i] )
    log(lambda[i]) <- alpha[class[i]] + offset[i]
    offset[i] <- log(w[i])
  }
  for (j in 1:J) {
    alpha[j] ~ dnorm(mu.class, 1/sigma.class^2)
    theta[j] <- exp(alpha[j])
  }
  mu.class ~ dnorm(0, 0.0001)
  sigma.class ~ dunif(0, 100)
  for (k in 1:n.new) { yhat[k] <- y[new[k] ] }
}"
writeLines(model, con="JAGStemp.txt")
data.list <- list("y"=infile$y, "n"=nrow(infile), "w"=infile$payroll
, "class"=class, "J"=length(unique(class))
, "new"=holdout, "n.new"=length(holdout))
parms <- c("theta", "sigma.class", "yhat")
```

87

First Model: Validation

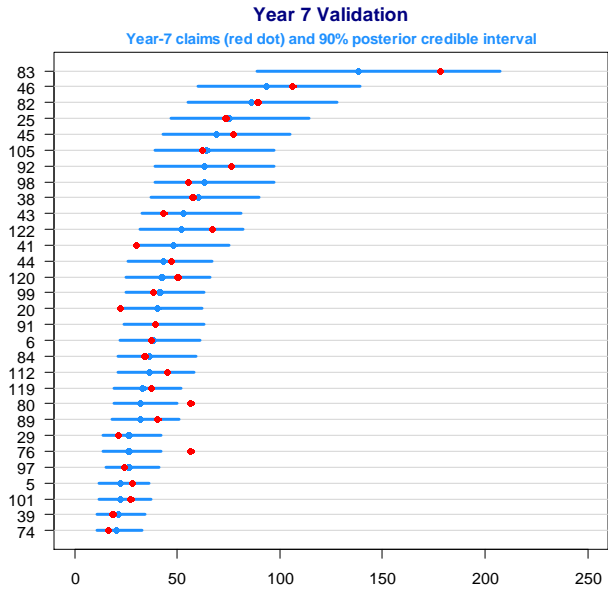
- Does model seem realistic?
- What change should we make?



88

Second Model: Validation

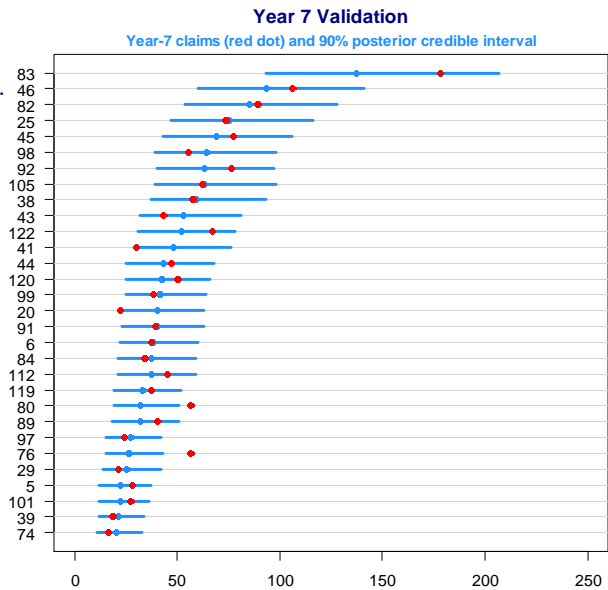
- Now roughly 90% of the year-7 claims fall within the 90% credible interval.



89

Third Model: Validation

- Only a minor difference.



90

Case Study #6

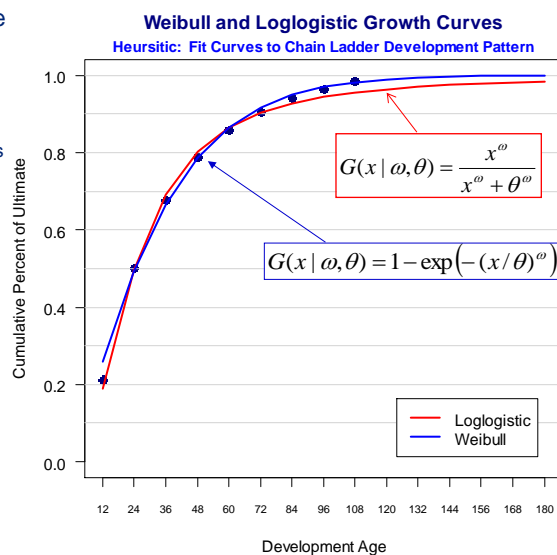
Bayesian Nonlinear Hierarchical Model

References:

Wayne Zhang, Vanja Dukic, James Guszczka: "A Bayesian Nonlinear Model for Forecasting Insurance Loss Payments", *Journal of the Royal Statistical Society, Series A*, 175, 637-56.
 James Guszczka, "Hierarchical Growth Curves Models for Loss Reserving", *CAS Forum*, 2008.

Growth Curves – At the Heart of the Model

- We want our model to reflect the **non-linear** nature of loss development.
 - GLM shows up a lot in the stochastic loss reserving literature...
 - ... but are GLMs natural models for loss triangles?
- Growth curves (Clark 2003)
 - γ = ultimate loss ratio
 - θ = scale
 - ω = shape ("warp")
- Heuristic idea
 - We judgmentally select a growth curve form
 - Let γ vary by year (hierarchical)
 - Add priors to the hyperparameters (Bayesian)



An Exploratory Non-Bayesian Hierarchical Model

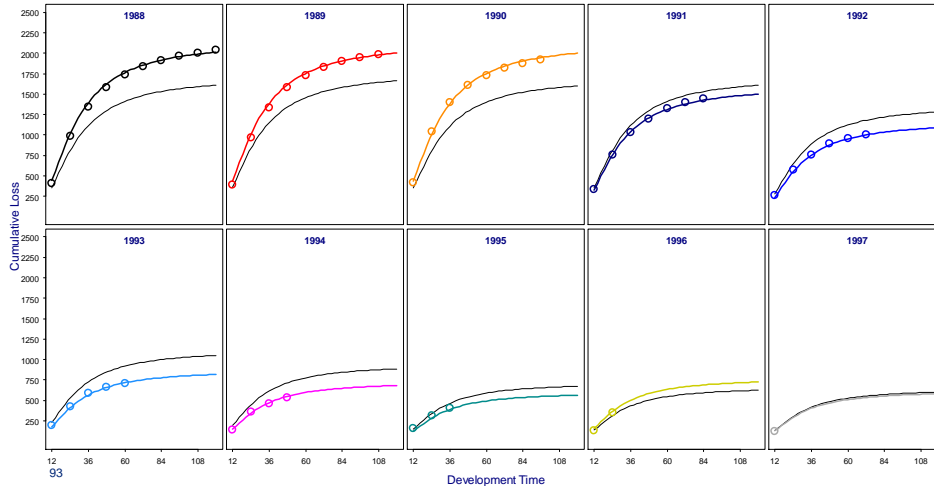
- It is easy to fit non-Bayesian hierarchical models as a data exploration step.

$$y_i(t_j) = \gamma_i * p_i * \left(\frac{t^{\omega}}{t^{\omega} + \theta^{\omega}} \right) + \varepsilon_i(t_j)$$

$$\gamma_i \sim N(\gamma, \sigma_{\gamma}^2)$$

$$\varepsilon_i(t_j) = \rho \varepsilon_i(t_{j-1}) + \delta_i(t_j)$$

Log-Logistic Hierarchical Model (non-Bayesian)

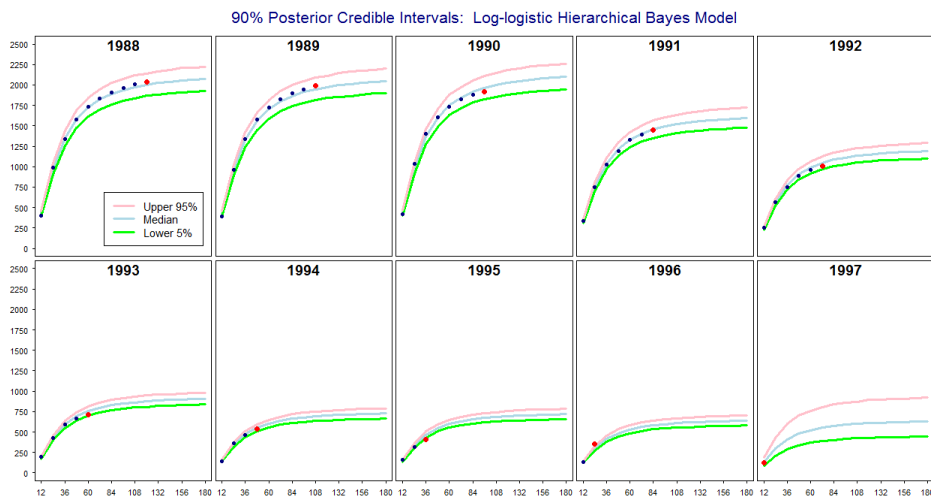


Adding Bayesian Structure

- Our hierarchical model is “half-way Bayesian”
 - On the one hand, we place probability sub-models on certain parameters
 - But on the other hand, various (hyper)parameters are estimated directly from the data.
- To make this fully Bayesian, we need to put probability distributions on **all** quantities that are uncertain.
- We then employ Bayesian updating: the model (“likelihood function”) together with the prior results in a posterior probability distribution over **all** uncertain quantities.
 - Including ultimate loss ratio parameters and hyperparameters!
 - We are directly modeling the ultimate quantity of interest.
- Before this morning this might have sounded impossible.
 - JAGS to the rescue

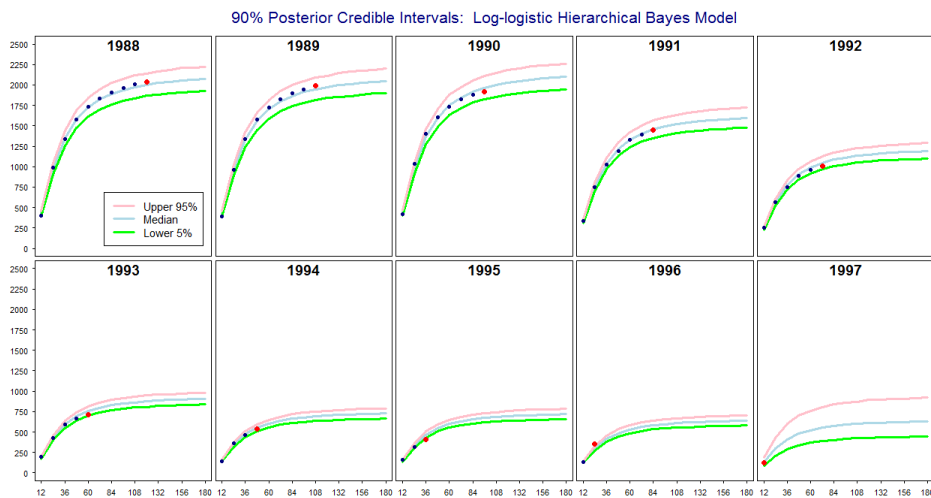
Results

- Now we fit a fully Bayesian version of the model by providing **prior distributions** for all of the model hyperparameters, and simulating the posterior distribution.



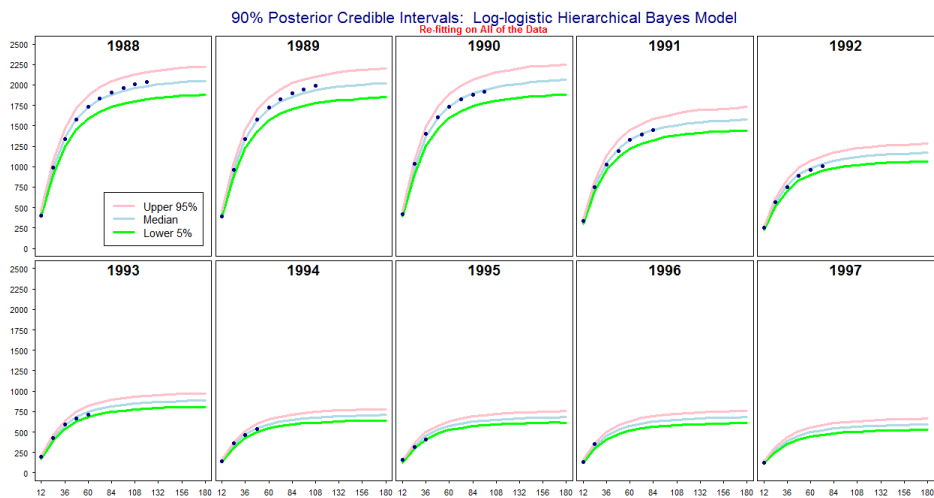
Results

- Here we are using the most recent Calendar Year (red) as a holdout sample.
- The model fits the holdout well.



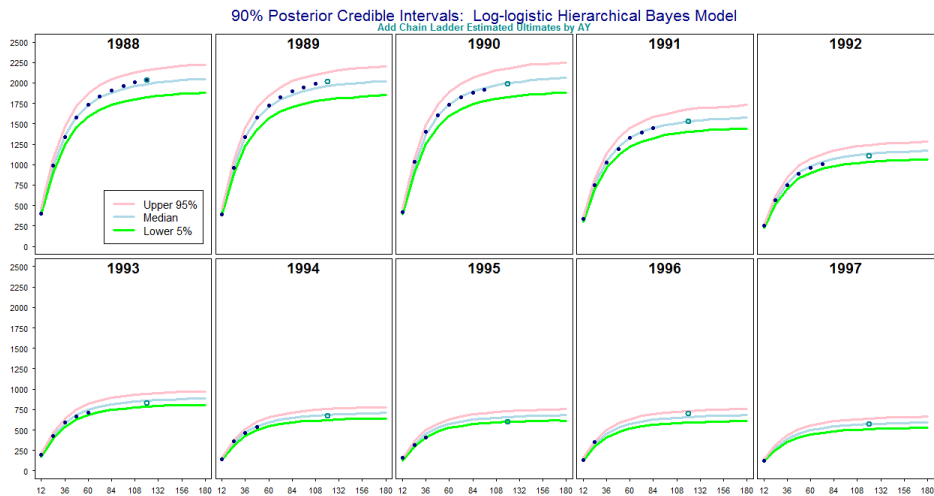
Bayesian Credible Intervals

- Now refit the model on all of the data and re-calculate the posterior credible intervals.



Comparison with the Chain Ladder

- For comparison, superimpose the “at 120 months” chain ladder estimates on the posterior credible intervals.



Posterior Distribution of Aggregate Outstanding Losses

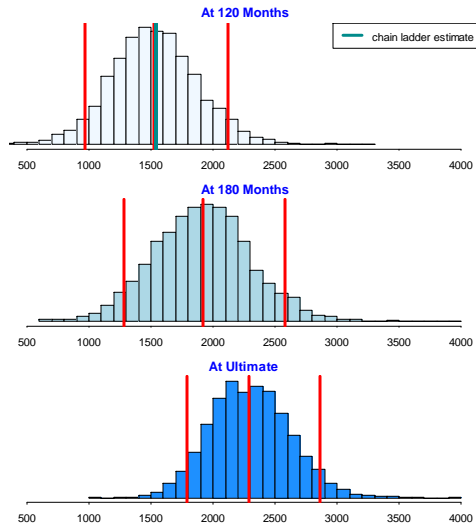
- In the top two images, we sum up the projected losses for all estimated AY's evaluated at 120 (180) months; then subtract losses to date (LTD).

- For the 120 month estimate, the posterior median (1519) comes very close to the chain ladder estimate (1543)

- In the bottom image, we multiply the estimated ultimate loss ratio parameters by premium and subtract LTD.

- Deciding which of these options is most appropriate is akin to selecting a tail factor.

Outstanding Loss Estimates at Different Evaluation Points
Estimated Ultimate Losses Minus Losses to Date



99

Testing the Predictive Distribution

Background

- Risk based capital proposals, e.g. EU Solvency II and USA SMI rely on stochastic models.
 - VaR@99.5% and TVaR@99%
- There are many stochastic loss reserve models that claim to predict the distribution of ultimate losses.

- ***How good are these models?***

- We now discuss tests of the predictions of currently popular stochastic loss reserve models on real data from 50 insurers in each of four lines of insurances.

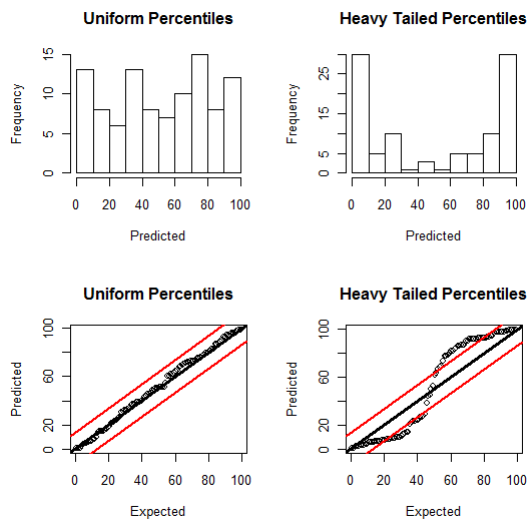
101

Criteria for a “Good” Stochastic Loss Reserve Model

- Using the upper triangle “training” data, predict the distribution of the outcomes in the lower triangle
- Can be observations from individual (AY, Lag) cells or sums of observations in different (AY,Lag) cells.
- Using the predictive distributions, find the percentiles of the outcome data.
- The percentiles should be uniformly distributed.
 - Histograms
 - Test with PP Plots/Kolmogorov-Smirnov (KS) tests
 - Plot Expected vs Predicted Percentiles
 - KS 95% critical values = 19.2 for $n = 50$ and 9.6 for $n = 200$

102

Illustrative Tests of Uniformity



103

The CAS Loss Reserve Database Created by Meyers and Shi With Permission of American NAIC

Schedule P (Data from Parts 1-4) for several US Insurers

- Private Passenger Auto
- Commercial Auto
- Workers' Compensation
- General Liability
- Product Liability
- Medical Malpractice (Claims Made)

Available on CAS Website

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

104

Notation

w = Accident Year $w = 1, \dots, 10$

d = Development Year $d = 1, \dots, 10$

$C_{w,d}$ = Cumulative (either incurred or paid) loss

$I_{w,d}$ = Incremental paid loss = $C_{w,d} - C_{w-1,d}$

105

Illustrative Insurer – Incurred Losses

Premium	AV/Lag	Cumulative Incurred Losses										Source
		1	2	3	4	5	6	7	8	9	10	
5812	1988	1722	3830	3603	3835	3873	3895	3918	3918	3917	3917	1997
4908	1989	1581	2192	2528	2533	2528	2530	2534	2541	2538	2532	1998
5454	1990	1834	3009	3488	4000	4105	4087	4112	4170	4271	4279	1999
5165	1991	2305	3473	3713	4018	4295	4334	4343	4340	4342	4341	2000
5214	1992	1832	2625	3086	3493	3521	3563	3542	3541	3541	3587	2001
5230	1993	2289	3160	3154	3204	3190	3206	3351	3289	3267	3268	2002
4992	1994	2881	4254	4841	5176	5551	5689	5683	5688	5684	5684	2003
5466	1995	2489	2956	3382	3755	4148	4123	4126	4127	4128	4128	2004
5226	1996	2541	3307	3789	3973	4031	4157	4143	4142	4144	4144	2005
4962	1997	2203	2934	3608	3977	4040	4121	4147	4155	4183	4181	2006

106

Illustrative Insurer – Paid Losses

Premium	AY/Lag	Cumulative Paid Losses										Source
		1	2	3	4	5	6	7	8	9	10	
5812	1988	952	1529	2813	3647	3724	3832	3899	3907	3911	3912	1997
4908	1989	849	1564	2202	2432	2468	2487	2513	2526	2531	2527	1998
5454	1990	983	2211	2830	3832	4039	4065	4102	4155	4268	4274	1999
5165	1991	1657	2685	3169	3600	3900	4320	4332	4338	4341	4341	2000
5214	1992	932	1940	2626	3332	3368	3491	3531	3540	3540	3583	2001
5230	1993	1162	2402	2799	2996	3034	3042	3230	3238	3241	3268	2002
4992	1994	1478	2980	3945	4714	5462	5680	5682	5683	5684	5684	2003
5466	1995	1240	2080	2607	3080	3678	4116	4117	4125	4128	4128	2004
5226	1996	1326	2412	3367	3843	3965	4127	4133	4141	4142	4144	2005
4962	1997	1413	2683	3173	3674	3805	4005	4020	4095	4132	4139	2006

107

Data Used in the Study

- Insurers listed in Meyers – Summer 2012 e-Forum
 - Also in files “CCL_IG10K.csv” (etc.) in “MCMC Workshop” directory
- 50 Insurers from four lines of business
 - Commercial Auto
 - Personal Auto
 - Workers’ Compensation
 - Other Liability
- Both paid and incurred losses
- In RStudio - open and run “Look at Triangle.R”

108

Exercise – Run the Mack Model

In RStudio – Open “Mack Model.R”

Key Steps in the Code

- Read data from CAS Loss Reserve Database
- Use R “ChainLadder” package to fit Mack Model
- Calculate 1st two moments of predicted outcomes
- Fit a lognormal distribution using moments
- Calculate percentile of actual outcome

Examine Output

109

Exercise – Run the Bootstrap ODP Model

In RStudio – Open “ODP Model.R”

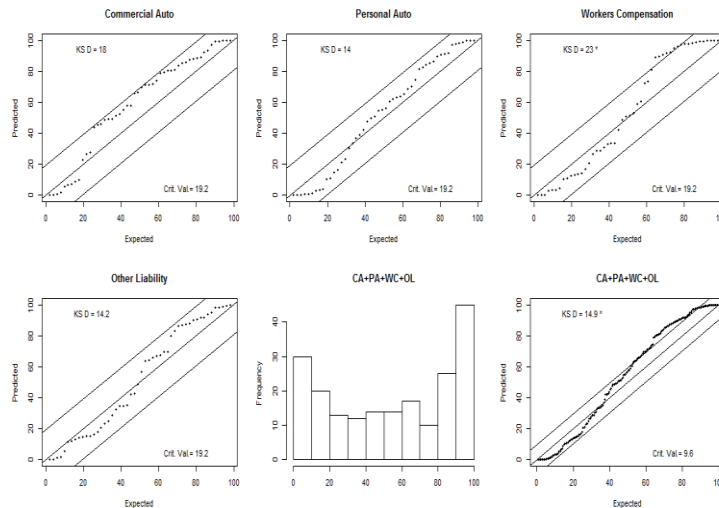
Key Steps in the Code

- Read data from CAS Loss Reserve Database
- Use R “ChainLadder” package to fit ODP Model
- Generate 10,000 outcomes
- Calculate percentile of actual outcome

Examine Output

110

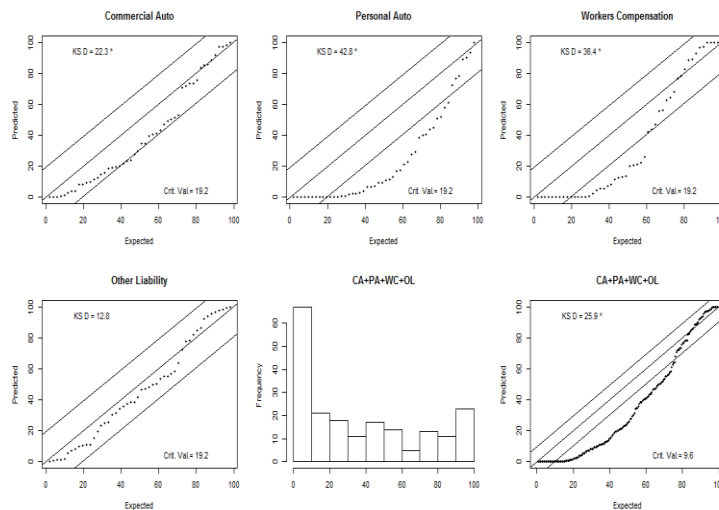
Test of Mack Model on Incurred Data



Conclusion – The Mack model predicts tails that are too light.

111

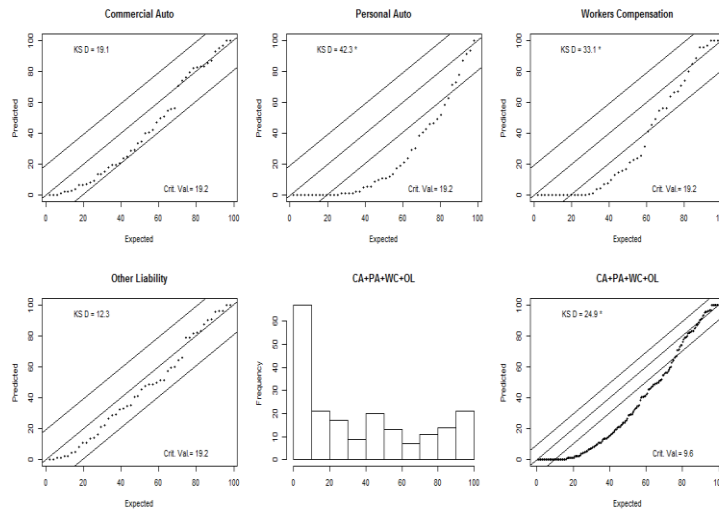
Test of Mack Model on Paid Data



Conclusion – The Mack model is biased upward.

112

Test of Bootstrap ODP on Paid Data



Conclusion – The Bootstrap ODP model is biased upward.

113

Response to Model
Failures

Possible Responses to the model failures

- The “Black Swans” got us again!
- We do the best we can in building our models, but the real world keeps throwing curve balls at us.
- Every few years, the world gives us a unique “black swan” event.
- Build a better model.
 - Use a model, or data, that sees the “black swans.”
 - MCMC is a good tool to use for stochastic loss reserve model building.

115

Bayesian MCMC Models

- Use R and JAGS packages
- Get a sample of 10,000 parameter sets from the posterior distribution of the model
- Use the parameter sets to get 10,000 simulated outcomes
- Calculate summary statistics of the simulated outcomes
 - Mean
 - Standard deviation
 - Percentile of the actual outcome

116

Discussion

Model Features with Incurred Data

- What do we know about the performance of the Mack model?
- Did not observe bias on our data.
- Predicted variance of the outcomes is too low.
- How do we increase the predicted variance?

117

How Can We Increase the Predicted Variance of Outcomes?

Model – $\log(C_{wd}) \sim \text{lognormal}(\mu_{wd}, \sigma_{wd})$

$$\mu_{wd} = \alpha_w + \beta_d$$

Mack assumes accident years are independent.

How can we introduce correlation between accident years?

$$\mu_{wd} = \alpha_w + \beta_d + \rho \cdot (\log(C_{w-1,d}) - \mu_{w-1,d})$$

118

How Can We Increase the Predicted Variance of Outcomes?

Model – $\log(C_{wd}) \sim \text{lognormal}(\mu_{wd}, \sigma_{wd})$

Note – Coefficient of variation is a function of σ .

$$\sigma_{wd} = \sigma_d$$

Do we know anything else about σ_d ?

$$\sigma_1 > \sigma_2 > \dots > \sigma_{10}$$

119

The Correlated Chain Ladder (CCL) Model

$$\mu_{1,d} = \alpha_1 + \beta_d$$

$$C_{1,d} \sim \text{lognormal}(\mu_{1,d}, \sigma_d)$$

$$\mu_{w,d} = \alpha_w + \beta_d + \rho (\log(C_{w-1,d}) - \mu_{w-1,d}) \text{ for } w = 2, \dots, 10$$

$$C_{w,d} \sim \text{lognormal}(\mu_{w,d}, \sigma_d)$$

$$\rho \sim U(-1, 1)$$

α_w and β_d are widely distributed with, $\beta_{10} = 0$.

$$\sigma_d = \sum_{i=d}^{10} a_i \quad a_i \sim U(0, 1) \text{ Forces } \sigma_d \text{ to decrease as } d \text{ increases}$$

Estimate distribution of $\sum_{w=1}^{10} C_{w,10}$

120

Exercise – Run the CCL Model

In RStudio – Open “CCL Model.R”

Key steps in the script

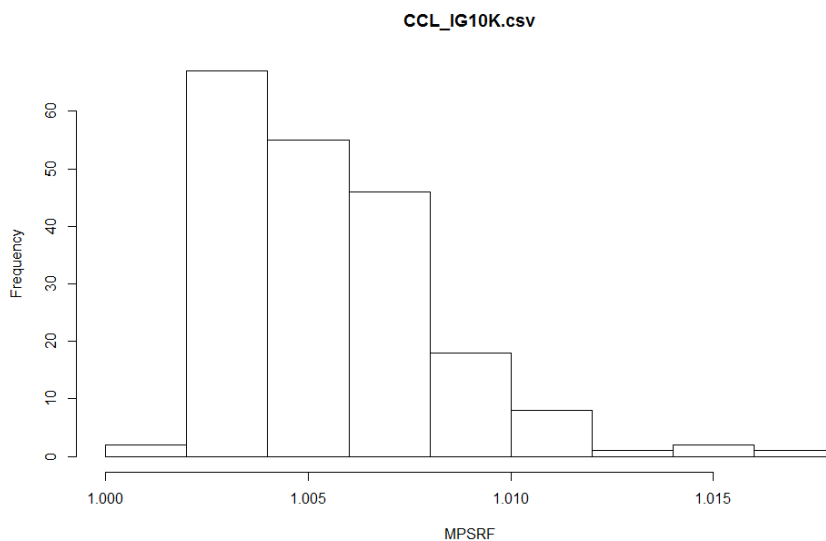
- Read data from CAS Loss Reserve Database
- Run JAGS to produce 10,000 parameter sets
- Generate convergence diagnostics
- Generate 10,000 outcomes by simulating loss from each parameter set.
- Calculate summary statistics
- Calculate percentile of actual outcome

Examine Output

- Look at convergence diagnostics
- Repeat exercise with “CCL Model Old.R”
- Look at convergence diagnostics

121

MPSRF Statistics on CCL Model for the 200 Triangles



122

The Correlated Chain Ladder Model Predicts Distributions with Thicker Tails

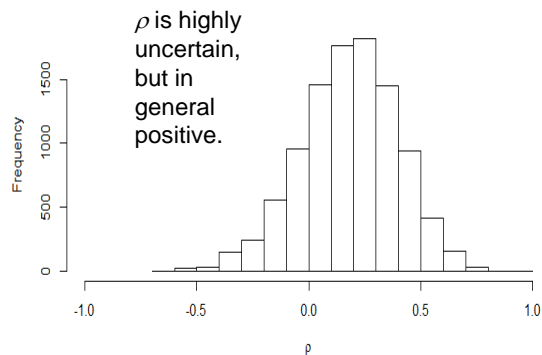
- Chain ladder applies factors to last **fixed** observation
- CCL uses **uncertain** “level” parameters for each accident year.

$$\text{Var}[C_{w,d}] = E_{\alpha_w}[\text{Var}[C_{w,d} | \alpha_w]] + \text{Var}_{\alpha_w}[E[C_{w,d} | \alpha_w]]$$

- Mack uses point estimations of parameters
- CCL uses Bayesian estimation to get a posterior distribution of parameters
- Mack assumes independence between accident years
- CCL allows for correlation between accident years
 - $\text{Corr}[\log(C_{w-1,d}), \log(C_{w,d})] = \rho$

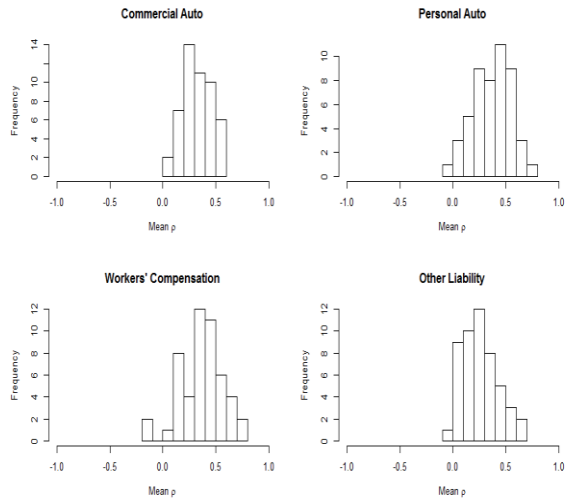
123

Posterior Distribution of ρ for Illustrative Insurer



124

Generally Positive Posterior Means of ρ



125

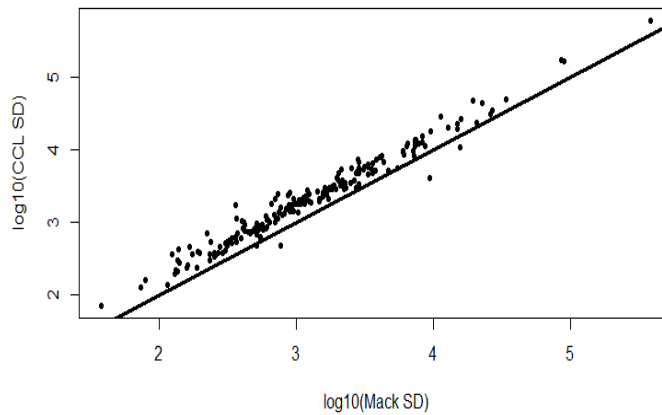
Results for the Illustrative Incurred Data with burn-in of 500,000 on old CCL

w	CCL			Mack			Outcome C_{w,10}
	C_{w,10}	SD	CV	C_{w,10}	SD	CV	
1	3,917	0	0.0000	3,917	0	0.0000	3,917
2	2,546	62	0.0244	2,538	0	0.0000	2,532
3	4,111	119	0.0289	4,167	3	0.0007	4,279
4	4,316	136	0.0315	4,367	37	0.0085	4,341
5	3,552	126	0.0355	3,597	34	0.0095	3,587
6	3,321	150	0.0452	3,236	40	0.0124	3,268
7	5,285	295	0.0558	5,358	146	0.0272	5,684
8	3,805	335	0.0880	3,765	225	0.0598	4,128
9	4,180	615	0.1471	4,013	412	0.1027	4,144
10	4,141	1,371	0.3311	3,955	878	0.2220	4,181
Total	39,174	1,869	0.0477	38,914	1,057	0.0272	40,061
Percentile		73.40			86.03		

Note the increase in the standard error of CCL over Mack.

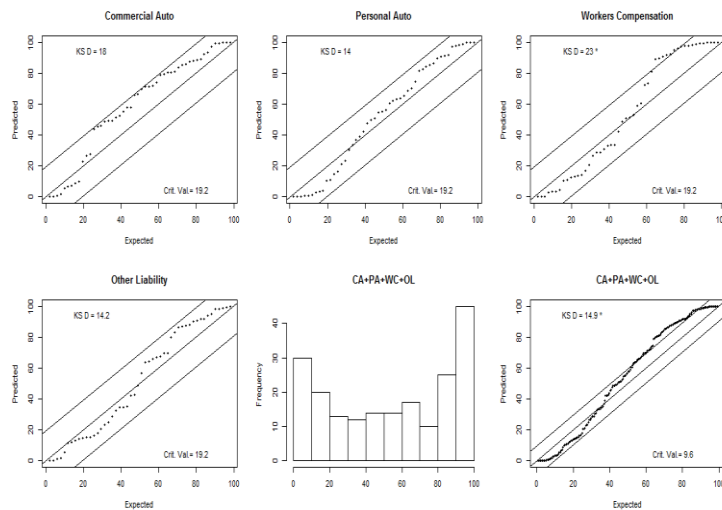
126

Compare SDs for All 200 Triangles



127

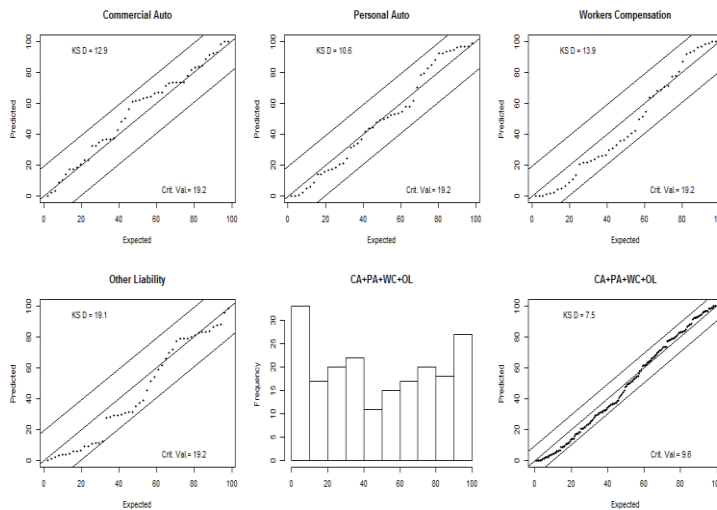
Test of Mack Model on Incurred Data



Conclusion – The Mack model predicts tails that are too light.

128

Test of CCL on Incurred Data



Conclusion – CCL model percentiles lie within KS statistical bounds.

129

Improvement with Incurred Data

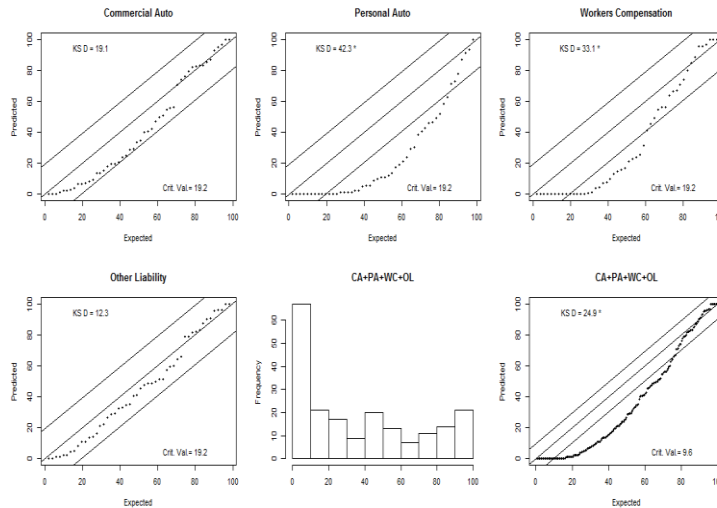
Accomplished by “pumping up” the variance of Mack model.

What About Paid Data?

Start by looking at CCL model on cumulative paid data.

130

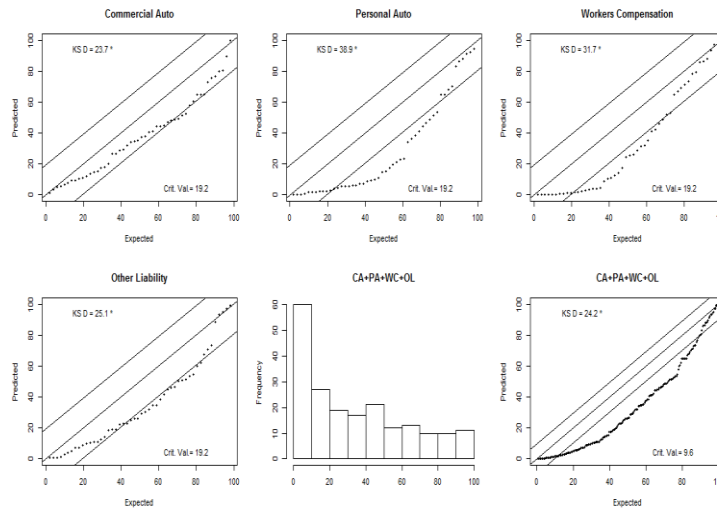
Test of Bootstrap ODP on Paid Data



Conclusion – The Bootstrap ODP model is biased upward.

131

Test of CCL on Paid Data



Conclusion – Roughly the same performance a bootstrapping and Mack

132

How Do We Correct the Bias?

Look at models with payment year trend.

- Ben Zehnwirth has been championing these for years.

Payment year trend does not make sense with cumulative data!

- Settled claims are unaffected by trend.

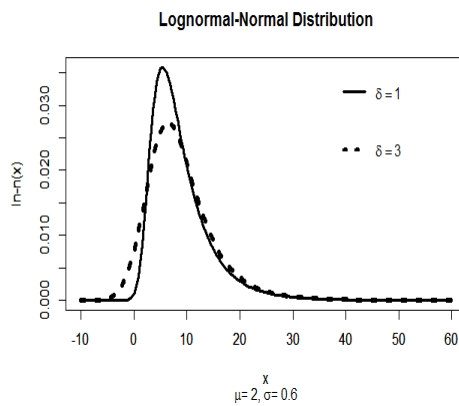
Recurring problem with incremental data – Negatives!

- We need a skewed distribution that has support over the entire real line.

133

The Lognormal-Normal (In-n) Mixture

$$X \sim \text{Normal}(Z, \delta), \quad Z \sim \text{Lognormal}(\mu, \sigma)$$



134

The Correlated Incremental Trend (CIT) Model

$$\mu_{w,d} = \alpha_w + \beta_d + \tau \cdot (w + d - 1)$$

$$Z_{w,d} \sim \text{lognormal}(\mu_{w,d}, \sigma_d) \text{ subject to } \sigma_1 < \sigma_2 < \dots < \sigma_{10}$$

$$I_{1,d} \sim \text{normal}(Z_{1,d}, \delta)$$

$$I_{w,d} \sim \text{normal}(Z_{w,d} + \rho \cdot (I_{w-1,d} - Z_{w-1,d}) \cdot e^\tau, \delta)$$

Estimate the distribution of $\sum_{w=1}^{10} C_{w,10}$

“Sensible” priors on α_w, σ_d , and τ . $\beta_1 = 0$

- Needed to control σ_d
- Interaction between τ , α_w and β_d .

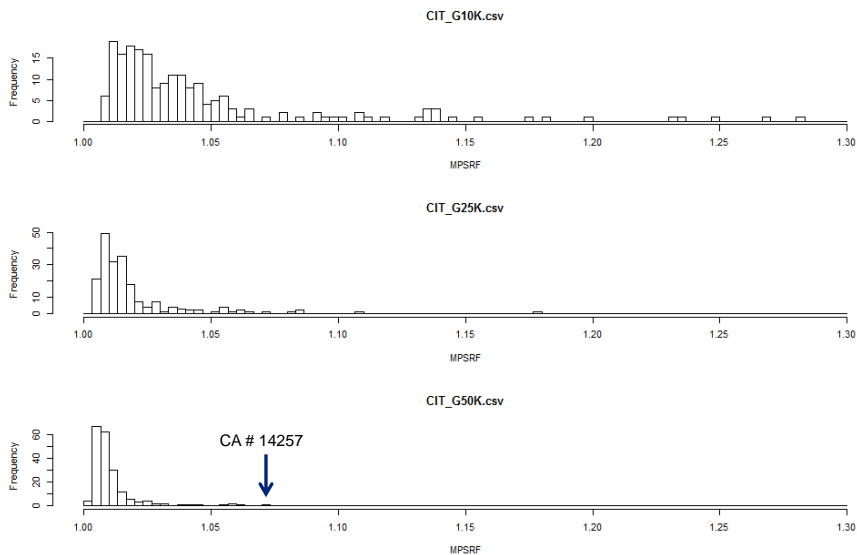
135

CIT Model for Illustrative Insurer with a burn-in of 500,000 iterations

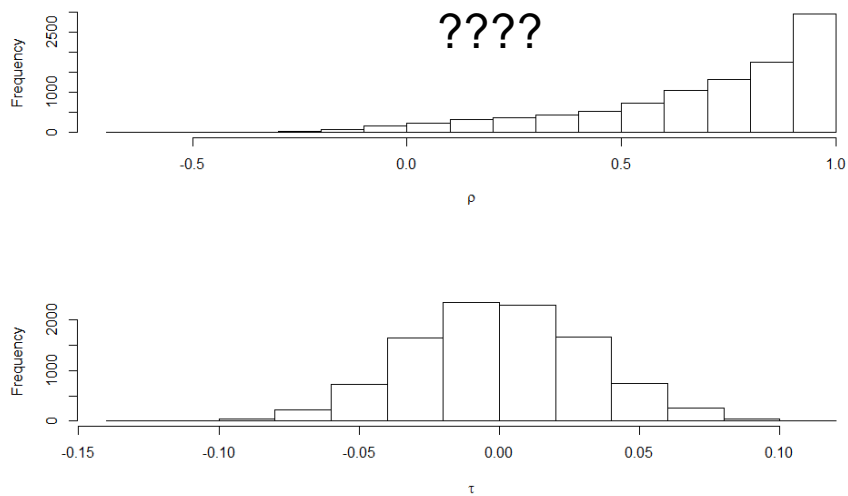
w	CIT			CCL			Outcome
	$C_{w,10}$	SD	CV	$C_{w,10}$	SD	CV	
1	3912	0	0	3912	0	0.0000	3912
2	2536	5	0.002	2563	110	0.0429	2527
3	4175	11	0.0026	4153	189	0.0455	4274
4	4378	29	0.0066	4320	224	0.0519	4341
5	3539	35	0.0099	3570	207	0.0580	3583
6	3043	105	0.0345	3403	255	0.0749	3268
7	5037	114	0.0226	5207	465	0.0893	5684
8	3501	556	0.1588	3649	467	0.1280	4128
9	3980	710	0.1784	4409	895	0.2030	4144
10	4661	1484	0.3184	5014	2435	0.4856	4139
Total	38763	1803	0.0465	40200	3070	0.0764	40000
Percentile		81.87			51.24		

136

MPSRF Statistics on CIT Model for the 200 Triangles



In-Depth Look at a Slow Mixing - CA # 14257 MPSRF = 1.072 with $n_{\text{adapt}} = n_{\text{burn}} = n_{\text{sample}} = 50,000$



In-Depth Look at a Slow Mixing - CA # 14257

MPSRF = 1.072 with $n_{\text{adapt}} = n_{\text{burn}} = n_{\text{sample}} = 50,000$

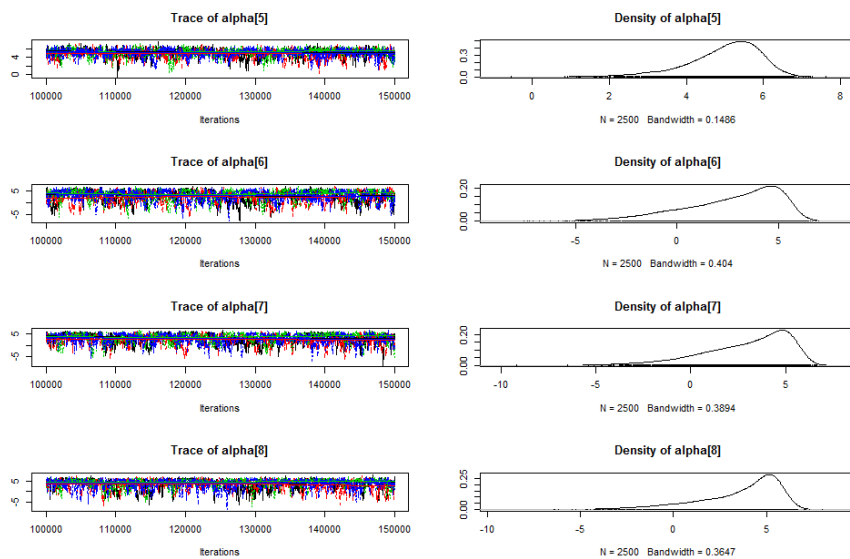
The Incremental Paid Triangle

w\d	1	2	3	4	5	6	7	8	9	10
1	216	168	112	65	23	0	0	0	0	0
2	245	280	104	96	52	5	0	0	0	
3	306	225	111	17	-3	0	-2	0		
4	400	162	181	165	1	0	0			
5	231	153	10	516	-361	0				
6	183	195	34	0	6					
7	306	150	-2	0						
8	333	128	62							
9	296	228								
10	309									

139

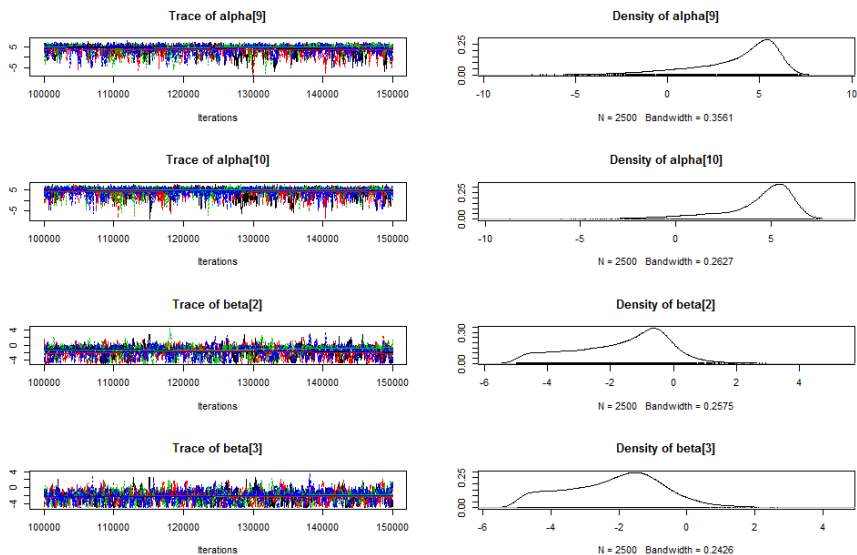
In-Depth Look at a Slow Mixing - CA # 14257

MPSRF = 1.072 with $n_{\text{adapt}} = n_{\text{burn}} = n_{\text{sample}} = 50,000$



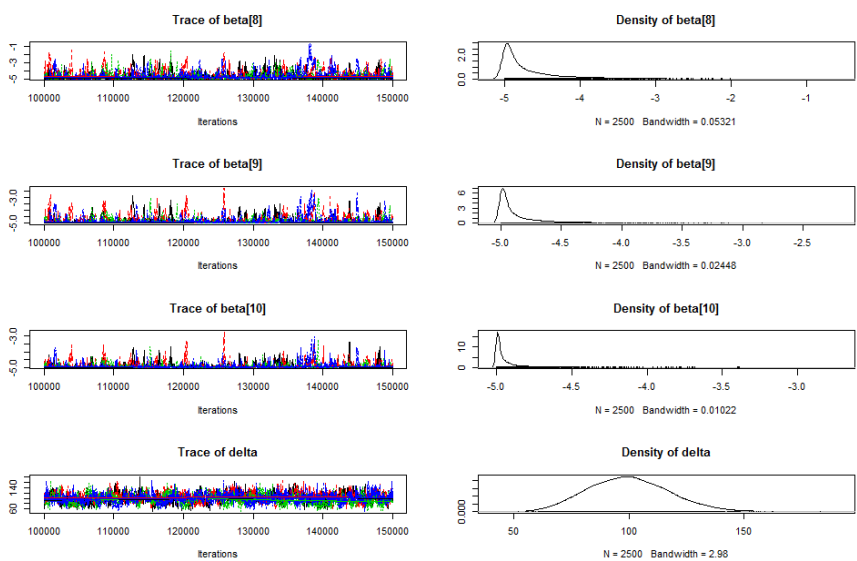
In-Depth Look at a Slow Mixing - CA # 14257

MPSRF = 1.072 with $n_{\text{adapt}} = n_{\text{burn}} = n_{\text{sample}} = 50,000$



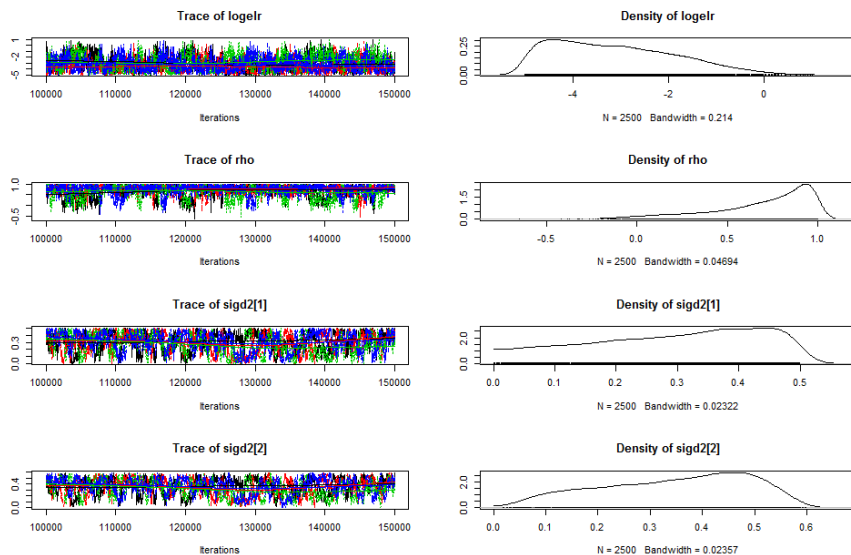
In-Depth Look at a Slow Mixing - CA # 14257

MPSRF = 1.072 with $n_{\text{adapt}} = n_{\text{burn}} = n_{\text{sample}} = 50,000$



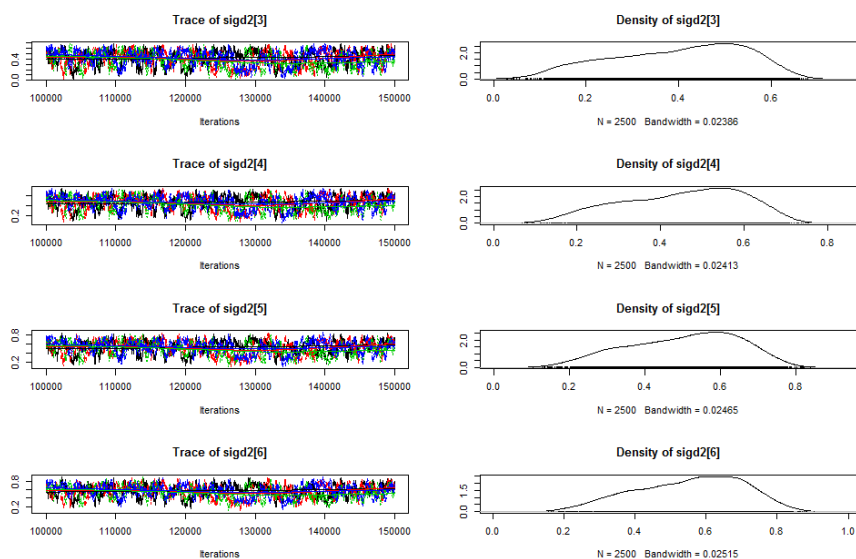
In-Depth Look at a Slow Mixing - CA # 14257

MPSRF = 1.072 with $n_{\text{adapt}} = n_{\text{burn}} = n_{\text{sample}} = 50,000$



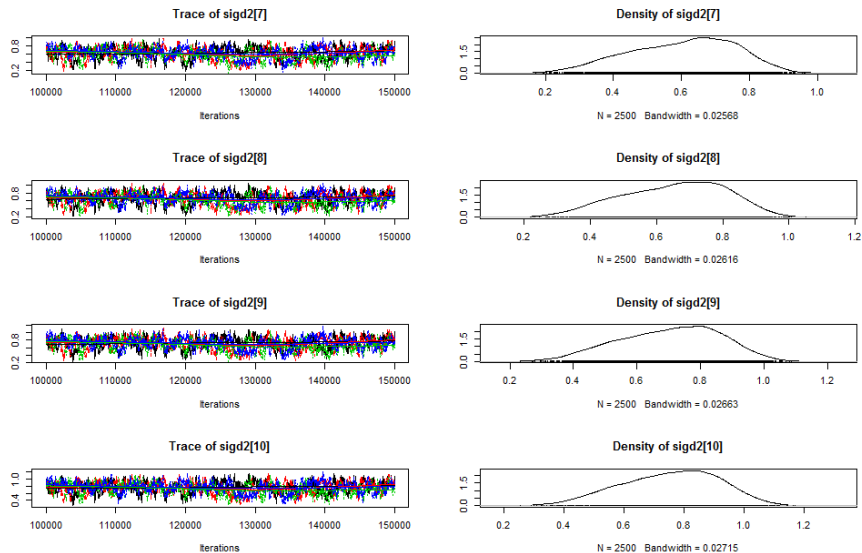
In-Depth Look at a Slow Mixing Model - CA # 14257

MPSRF = 1.072 with $n_{\text{adapt}} = n_{\text{burn}} = n_{\text{sample}} = 50,000$



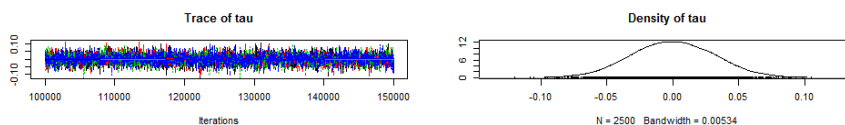
In-Depth Look at a Slow Mixing - CA # 14257

MPSRF = 1.072 with $n_{\text{adapt}} = n_{\text{burn}} = n_{\text{sample}} = 50,000$



In-Depth Look at a Slow Mixing - CA # 14257

MPSRF = 1.072 with $n_{\text{adapt}} = n_{\text{burn}} = n_{\text{sample}} = 50,000$



Is this a reasonable prediction?

w	Premium	CIT.Estimate	CIT.SE	CIT.CV	Outcome	CIT.Pct
1	1041	584	0	0	584	
2	1112	782	103	0.1317	782	
3	1077	656	165	0.2515	654	
4	713	916	206	0.2249	909	
5	819	557	242	0.4345	548	
6	1042	427	271	0.6347	419	
7	1165	680	328	0.4824	607	
8	1317	889	567	0.6378	607	
9	1463	901	929	1.0311	780	
10	1675	950	1390	1.4632	984	
Total	11424	7344	2347	0.3195	6874	46.13

My Current Practice on Convergence Testing

- Chapter 6 in Brooks, Gelman, Jones and Meng. Chapter authors are Andrew Gelman and Kenneth Shirley
1. Run model with four chains.
 2. $n_{\text{adapt}} = n_{\text{burn}} = n_{\text{sample}}$, with n_{thin} selected to get 10,000 parameter sets
 3. Select
 4. Run the “gelman.diag” function
 5. If $\text{MPSRF} < 1.05$, don't worry (too much) about nonconvergence. Gelman-Shirley suggest 1.1 (and use results from multiple chains).
 6. If worried, or if you have time and are not worried, look at the trace plots.

My prior practice – Brute force with burn in of 500,000+

147

147

Exercise – Run the CIT Model

In RStudio – Open “CIT Model.R”

Key Steps in the Code

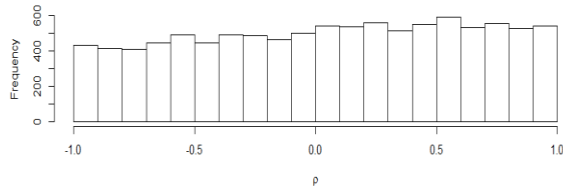
- Read data from CAS Loss Reserve Database
- Run JAGS to produce 10,000 parameter sets
- Generate 10,000 outcomes by simulating loss from each parameter set.
- Generate convergence diagnostics
- Calculate summary statistics
- Calculate percentile of actual outcome

Examine Output

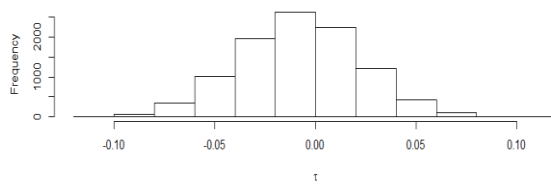
148

Posterior Distribution of μ and τ for Illustrative Insurer

Should we allow ρ in the model?

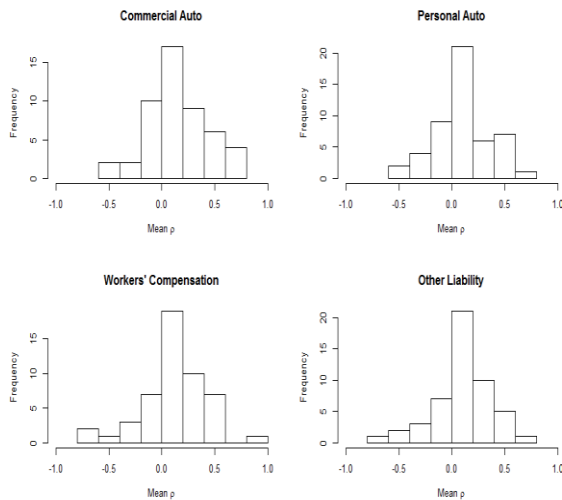


Predominantly negative trends



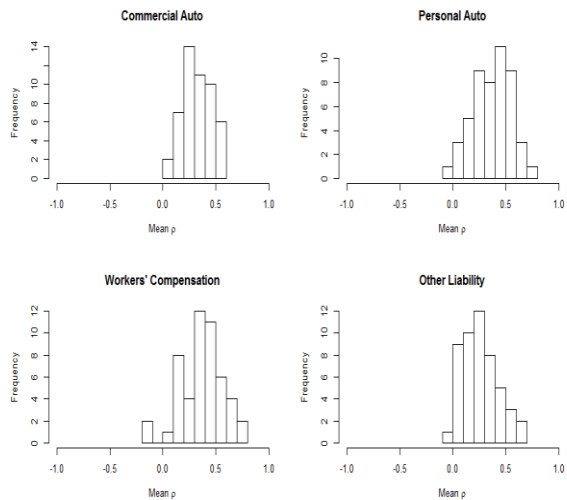
149

Posterior Mean ρ for All Insurers On Paid Data



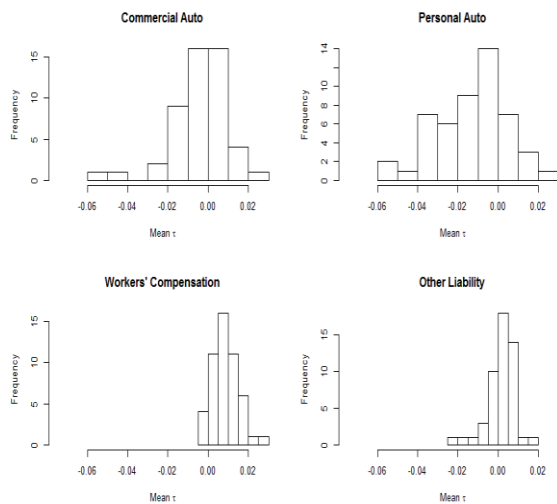
150

Posterior Mean ρ for All Insurers On Incurred Data



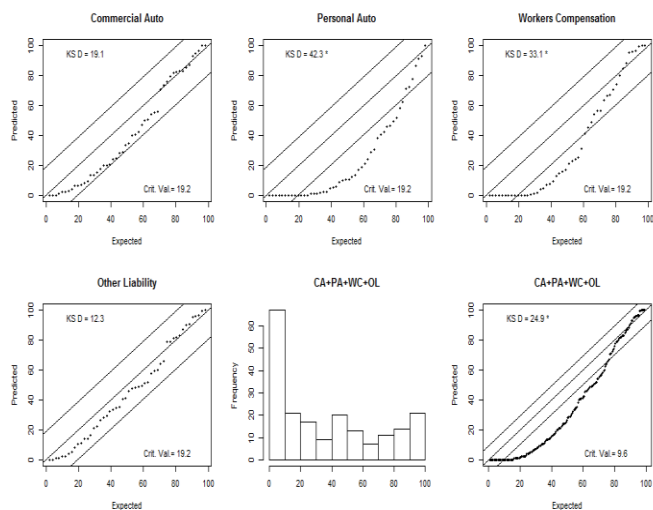
151

Posterior Mean τ for All Insurers



152

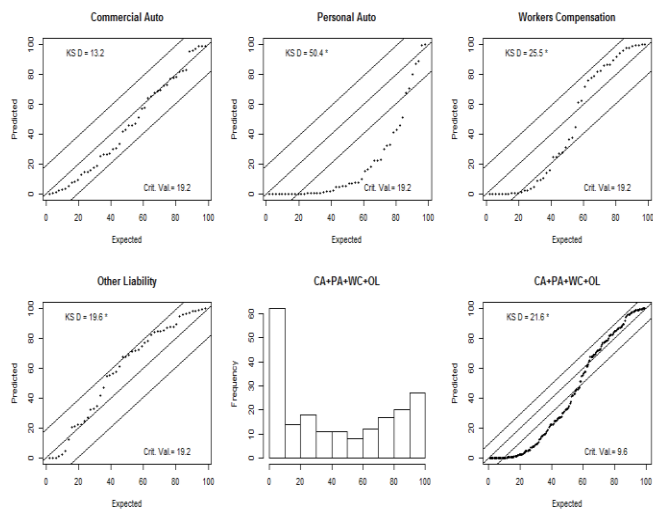
Test of Bootstrap ODP on Paid Data



Conclusion – The Bootstrap ODP model is biased upward.

153

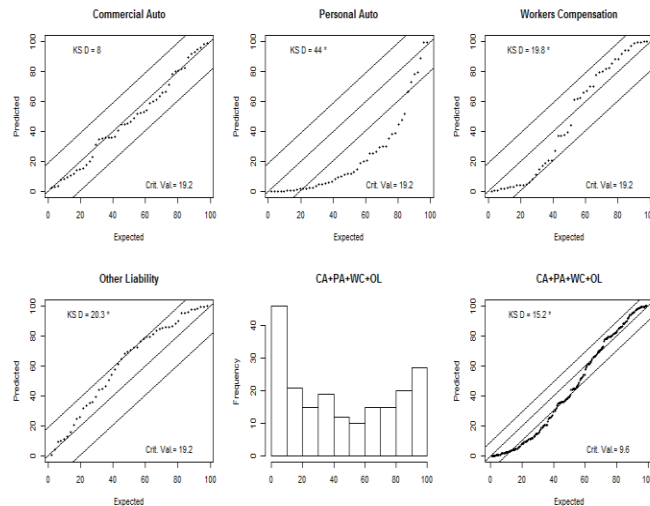
Test of CIT with $\rho = 0$ on Paid Data



Conclusion – Overall improvement but look at Personal Auto

154

Test of CIT on Paid Data



Conclusion – CIT model percentiles are an improvement but do not lie within the KS bounds.

155

Summary

Mack underpredicts the variability of outcomes with incurred data.

Both Mack and Bootstrap ODP are biased high with paid data.

Bayesian MCMC models

- Easily modified to produce new models.
- Easily implemented to produce predictive distributions of outcomes.

CCL model improves significantly on predictions with incurred data.

- Important feature – Correlation between accident years

CIT models improves somewhat on predictions with paid data.

- Important features – Payment year trend and correlation between accident years

Shortcoming – Study needs to be repeated on different time periods.

Goals of workshop

- Enable users to run Bayesian MCMC models for loss reserving
- Provide in depth understanding of CCL and CIT models so that users can explore improvements to those models.

156

References

1. Simon Jackman – *Bayesian Analysis for the Social Science*, Wiley 2009
 - A good introduction to the underlying theory and practice of Bayesian MCMC.
2. Steve Brooks, Andrew Gelman, Galin L. Jones and Xiao-Li Meng Editors, *Handbook of Markov Chain Monte Carlo*, Chapman and Hall 2011
 - Consists of chapters written by several authors on selected topics in MCMC
 - In depth view of the current state of the art that the editors think “may stand the test of time.”