Building a Stochastic Reserve Model with the BRMS Package

Presentation by Michael R Larsen
Agenda

• Goals
• Introduce brms
• Brms Examples
• Conclusion
Polling Questions

Actuarial Background
• Credit for Exam 5
• Credit for Exam 4/c
• Credit for current MAS II
• Credit for Exam 7
• You have squared a triangle at some point in your career on the job.
• You have done paid link ratio bootstrapping on the job

Modeling Background
• You have used R
• You have used Rstudio
• You have read a textbook on Bayesian MCMC modeling
• You need an idea of how to start building Bayesian MCMC models for reserving
• You have done Bayesian MCMC modeling for reserving and want to see what brms offers
Goals

• Make Bayesian MCMC analysis accessible to more actuaries
• Introduce the brms package plus tidyverse and shinystan
• Provide simple examples of stochastic reserve model construction
• Identify additional resources
Introduction to Packages in Presentation
What is BRMS

• Macro writer for STAN
• STAN is a Bayesian MCMC package
• BRMS let’s user describe model in linear model form with additions for Bayesian MCMC work
  • Lme4 framework (think of lm in R package as starting point)
  • Add in prior distributions
  • Correlation instructions
  • Group variables (random effects)
• Created by Paul Christian Buerkner in 2017
How brms helps

What it can do
• Reduces coding effort required to run in STAN
• Works with other packages
  • ShinyStan
  • TidyBayes
• Standard Bayesian MCMC Analysis options
  • Leave one out
  • WAIC

What it doesn’t do
• Eliminate need to understand Bayesian MCMC:
  • Concepts
  • Vocabulary
• Eliminate need to learn new software
  • Brms package
  • Rstudio IDE
  • Tidyverse assortment of R packages
Packages Supporting brms

**TidyVerse**
- Set of R packages to ease programming burden
- Tidybayes
  - Transforms Bayesian MCMC simulation to tidy data sets
- Tidyverse packages examples:
  - Ggplot2 & Cowplot presentation quality graphics
  - Dplyr data manipulation tool

**ShinyStan**
- Standardized reports on Bayesian MCMC object
- Reports on performance of STAN
  - Trace plots
  - Tree Depth
  - Divergence
  - Number of Effective Samples
- Reports on parameter estimates
RStudio Integrated Development Environment (IDE)

Create Programs for Modeling
- Pull in R packages
  - CRAN
  - Github
- Create source code using base R and packages
- View program results
  - Error Codes
  - Diagnostics as code executes

View Modeling Results
- Display Graphs
- Inspect data sets created
- Inspect objects created
Modeling Environment

- RStudio IDE
  - brms
  - STAN
  - MCMC OBJECT

- ggplot2
  - Graph Object
  - ShinyStan
  - Shiny Object
Working Example of RStudio Environment
Brms Structure

Brms structure

- Start with Regression Type formula:
  - response | aterms ~ pterms + (gterms | group)

- Add Other Modeling features:
  - Prior Distributions
  - Correlation structures
  - Variance modeling & other terms

Brms components

- Regression formula terms:
  - Response: dependent variable
  - Aterms: adjustments to dependent variable (exposure or censoring)
  - Pterms: GLM type betas for population
  - Group: variables to apply least squares credibility to in regression

- Other modeling features
  - Prior distribution: source to credibility weight against data set estimates
  - Correlation & Variance: options to model complex covariance structures
STAN Coding

Code Blocks
• Functions
• Data
• Transformed data
• Parameters
• Model
• Generated Quantities

Excerpt STAN Code 1st Model
• data {
  • int<lower=1> N; // number of observations
  • vector[N] Y; // response variable
  • int<lower=1> K; // number of population-level effects
  • matrix[N, K] X; // population-level design matrix
  • int<lower=1> K_sigma; // number of population-level effects
  • matrix[N, K_sigma] X_sigma; // population-level design matrix
  • // data for group-level effects of ID 1
  • int<lower=1> N_1; // number of grouping levels
  • int<lower=1> M_1; // number of coefficients per level
  • int<lower=1> J_1[N]; // grouping indicator per observation
  • // group-level predictor values
  • vector[N] Z_1_1;
  • int prior_only; // should the likelihood be ignored?
  • }

13
Why Bayesian MCMC?

Technical Issues
- Variance modeling matters
- Effect of correlation modeling
- Credibility weighting combined with regression
  - Prior knowledge to guide results
  - Least Squares credibility weighting options

Reserve Estimate Audience
- Variability estimates directly tied to reserve distribution
- Option to explicitly weight models based on predictive power
- Facilitates reserve distribution presentation to clarify reserve position choice
Stochastic Reserve Examples
Example Outline

• Simulated incremental paid loss triangle
• Start with graphs to identify likely structures using tidyverse
  • Data organized using dplyr
  • Graphs created with ggplot2
  • Use counts at 12 months as common exposure base
• Write brms package instructions to fit optional structures
  • Create three different models
  • Annotate first model instructions to give brms use example
• Analysis of results
  • Demonstrate ShinyStan diagnostics
  • Demonstrate tidybayes linked to ggplot2 diagnostics
Standard Deviation Log Incremental Loss Payment Per Claim

![Graph showing standard deviation of log incremental loss payment per claim over years.]

Y-axis: Std_Dev_Log_Inc_Paid_Per_Cnt
X-axis: Dev_Yr
QQ Plot for Normalized Log of Incremental Payments by Payment Lag
Observations on Graphs

- Mean incremental paid by development year shows quadratic, decreasing pattern
- Standard deviation is not a function of the mean and varies by development age
- Calendar year trend is evident
- Lognormal distribution fits by development age
Reserve Model #1 Instructions

• `rsv_model_1 <- brm(bf(Inc_Paid_Per_Cnt ~ Dev_Yr + Dev_Yr_Sqrd + Cal_Yr_Time + (1 | Acc_Yr),
  sigma ~ Dev_Yr + Dev_Yr_4_Cap + Dev_Yr_10_Cap ),
data = Reserve_Data_History,
prior = prior(normal(0,1)),
family = lognormal() )`
Reserve Model #1 Instruction Comments

• Example of modeling the distribution: mean and variance
• Mean: Inc_Paid_Per_Cnt ~ Dev_Yr + Dev_Yr_Sqrd + Cal_Yr_Time + (1|Acc_Yr)
• Variance: sigma ~ Dev_Yr + Dev_Yr_4_Cap + Dev_Yr_10_Cap
• Credibility with prior beliefs for betas: prior = prior(normal(0,1)),
• Least squares type credibility weighting: (1|Acc_Yr)
• Distribution: family = lognormal()
  • Note that sigma will be modeled with log transform to ensure positive result
• Data set: data = Reserve_Data_History
  • Compact code compared to STAN
Diagnostics for Reserve Models

ShinyStan
• Slides 26 – 29 are from ShinyStan for Reserve Model #1
• ShinyStan results are standard package results from applying ShinyStan to Reserve Model #1
• Some results are diagnostics on Hamiltonian algorithm
• Results on parameter distribution shown as well

Tidyverse Results
• Slides 30 – 32 are from tidyverse package applications
• Tidybayes transforms simulation result data set to tidy data
• Ggplot2 combined with tidybayes created plots on slides 31 & 32
<table>
<thead>
<tr>
<th>Parameter</th>
<th>n_eff</th>
<th>Rhat</th>
<th>mean</th>
<th>mcse</th>
<th>sd</th>
<th>2.5%</th>
<th>25%</th>
<th>50%</th>
<th>75%</th>
<th>97.5%</th>
</tr>
</thead>
<tbody>
<tr>
<td>b_Intercept</td>
<td>580</td>
<td>1.002</td>
<td>5.023</td>
<td>0.001</td>
<td>0.022</td>
<td>4.978</td>
<td>5.009</td>
<td>5.023</td>
<td>5.037</td>
<td>5.065</td>
</tr>
<tr>
<td>b_sigma_Interceptor</td>
<td>1,488</td>
<td>0.999</td>
<td>-4.194</td>
<td>0.004</td>
<td>0.162</td>
<td>-4.491</td>
<td>-4.309</td>
<td>-4.199</td>
<td>-4.087</td>
<td>-3.866</td>
</tr>
<tr>
<td>b_Yr</td>
<td>1,577</td>
<td>1</td>
<td>-0.096</td>
<td>0</td>
<td>0.003</td>
<td>-0.102</td>
<td>-0.098</td>
<td>-0.096</td>
<td>-0.093</td>
<td>-0.089</td>
</tr>
<tr>
<td>b_Yr_Sqrd</td>
<td>4,000</td>
<td>1.001</td>
<td>-0.011</td>
<td>0</td>
<td>0</td>
<td>-0.011</td>
<td>-0.011</td>
<td>-0.011</td>
<td>-0.01</td>
<td>-0.01</td>
</tr>
<tr>
<td>b_Cal_Yr_Time</td>
<td>579</td>
<td>1.001</td>
<td>0.041</td>
<td>0</td>
<td>0.002</td>
<td>0.038</td>
<td>0.04</td>
<td>0.041</td>
<td>0.042</td>
<td>0.045</td>
</tr>
<tr>
<td>b_sigma_Yr</td>
<td>1,133</td>
<td>1.003</td>
<td>0.111</td>
<td>0.001</td>
<td>0.031</td>
<td>0.051</td>
<td>0.09</td>
<td>0.111</td>
<td>0.131</td>
<td>0.172</td>
</tr>
<tr>
<td>b_sigma_Yr_4_Cap</td>
<td>1,559</td>
<td>1.001</td>
<td>0.215</td>
<td>0.002</td>
<td>0.076</td>
<td>0.068</td>
<td>0.164</td>
<td>0.215</td>
<td>0.267</td>
<td>0.365</td>
</tr>
<tr>
<td>b_sigma_Yr_10_Cap</td>
<td>996</td>
<td>1.004</td>
<td>0.075</td>
<td>0.002</td>
<td>0.054</td>
<td>-0.03</td>
<td>0.038</td>
<td>0.074</td>
<td>0.11</td>
<td>0.183</td>
</tr>
<tr>
<td>sd_Acc_Yr__Intercept</td>
<td>604</td>
<td>1.002</td>
<td>0.045</td>
<td>0</td>
<td>0.009</td>
<td>0.031</td>
<td>0.039</td>
<td>0.044</td>
<td>0.049</td>
<td>0.066</td>
</tr>
<tr>
<td>r_Acc_Yr[2000,Intercept]</td>
<td>672</td>
<td>1.002</td>
<td>0.014</td>
<td>0.001</td>
<td>0.022</td>
<td>-0.03</td>
<td>-0.001</td>
<td>0.013</td>
<td>0.028</td>
<td>0.059</td>
</tr>
</tbody>
</table>

Showing 1 to 10 of 31 entries
### Summary of sampler parameters

#### Warmup

<table>
<thead>
<tr>
<th></th>
<th>Omit</th>
<th>Include</th>
</tr>
</thead>
<tbody>
<tr>
<td>Statistic</td>
<td>Mean</td>
<td>SD</td>
</tr>
<tr>
<td>accept_stat</td>
<td>0.9358</td>
<td>0.0110</td>
</tr>
<tr>
<td>chain1</td>
<td>0.9233</td>
<td>0.0118</td>
</tr>
<tr>
<td>chain2</td>
<td>0.9541</td>
<td>0.0098</td>
</tr>
<tr>
<td>chain3</td>
<td>0.9370</td>
<td>0.0108</td>
</tr>
<tr>
<td>chain4</td>
<td>0.9289</td>
<td>0.0116</td>
</tr>
</tbody>
</table>
## Rsv_Model_1 Population Parameter Estimates

<table>
<thead>
<tr>
<th>Variable</th>
<th>Estimate</th>
<th>Lower</th>
<th>Upper</th>
<th>Interval</th>
<th>Estimate Type</th>
<th>Interval Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>b_Cal_Yr_Time</td>
<td>0.042</td>
<td>0.038</td>
<td>0.044</td>
<td>0.95</td>
<td>median</td>
<td>qi</td>
</tr>
<tr>
<td>b_Dev_Yr</td>
<td>-0.096</td>
<td>-0.102</td>
<td>-0.089</td>
<td>0.95</td>
<td>median</td>
<td>qi</td>
</tr>
<tr>
<td>b_Dev_Yr_Sqrd</td>
<td>-0.011</td>
<td>-0.011</td>
<td>-0.010</td>
<td>0.95</td>
<td>median</td>
<td>qi</td>
</tr>
<tr>
<td>b_Intercept</td>
<td>5.020</td>
<td>4.984</td>
<td>5.062</td>
<td>0.95</td>
<td>median</td>
<td>qi</td>
</tr>
<tr>
<td>b_sigma_Dev_Yr</td>
<td>0.109</td>
<td>0.050</td>
<td>0.169</td>
<td>0.95</td>
<td>median</td>
<td>qi</td>
</tr>
<tr>
<td>b_sigma_Dev_Yr_10_Cap</td>
<td>0.079</td>
<td>-0.026</td>
<td>0.184</td>
<td>0.95</td>
<td>median</td>
<td>qi</td>
</tr>
<tr>
<td>b_sigma_Dev_Yr_4_Cap</td>
<td>0.210</td>
<td>0.067</td>
<td>0.360</td>
<td>0.95</td>
<td>median</td>
<td>qi</td>
</tr>
<tr>
<td>b_sigma_Intercept</td>
<td>-4.200</td>
<td>-4.480</td>
<td>-3.874</td>
<td>0.95</td>
<td>median</td>
<td>qi</td>
</tr>
</tbody>
</table>

Note: Formatted using tidybayes. Beta estimates are on natural log scale. Beta estimates for sigma includes sigma in the variable name. Sigma estimates were developed assuming the natural log transform to ensure positive variance estimates for the lognormal distribution. Estimates were gathered from the Bayesian MCMC simulation results.
Reserve Modeling Options Selected

• Only change the variance estimate model formulas
  • Reserve Model #1: $\sigma \sim \text{Dev}_Yr + \text{Dev}_Yr\_4\_Cap + \text{Dev}_Yr\_10\_Cap$
  • Reserve Model #2: $\sigma \sim \text{Dev}_Yr\_10\_Cap + \text{Dev}_Yr\_10\_Spline$
  • Reserve Model #3: $\sigma \sim \text{Dev}_Yr\_8\_Cap + \text{Dev}_Yr\_8\_Spline$

• Compared models
  • Leave one out or WAIC
    • WAIC had technical issues and used leave one out

• Model comparison or weighting provides systematic way of bringing different views into the estimated distribution
Reserve Model Comparison Using Leave One Out Method

<table>
<thead>
<tr>
<th>Model</th>
<th>elpd_diff</th>
<th>se_diff</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reserve Model #1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Reserve Model #3</td>
<td>-0.8</td>
<td>2.4</td>
</tr>
<tr>
<td>Reserve Model #2</td>
<td>-4.5</td>
<td>3</td>
</tr>
</tbody>
</table>

Note: elpd = expected log pointwise predictive density for a new dataset
Results indicate that Reserve Model #1 performs best.
Conclusion

• Advances in software make Bayesian MCMC analysis more practical
  • Volume of coding is reduced
  • Open source help on web
  • STAN has more horsepower than earlier Bayesian MCMC tools

• Left with task of acquiring new vocabulary and skill set

• End result of analysis recognizes distribution of reserve estimates
  • See Slide #32
  • Sets up management discussion
    • What is the probability we are short at a given selected level?
    • What is the probability ultimate reserve estimates lie between two selected amounts?
Additional Resources

• Brms
  • Advanced Bayesian Multilevel Modeling with the R Package brms by Paul-Christian Bürkner, The R Journal Vol. 10/1, July 2018

• STAN
  • https://mc-stan.org/

• Rstudio
  • https://rstudio.com/

• Tidyverse
  • Rstudio help
  • R for Data Science: Import, Tidy, Transform, Visualize, and Model Data 1st Edition, Hadley Wickham, Garret Grolemund

• Bayesian MCMC textbooks