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

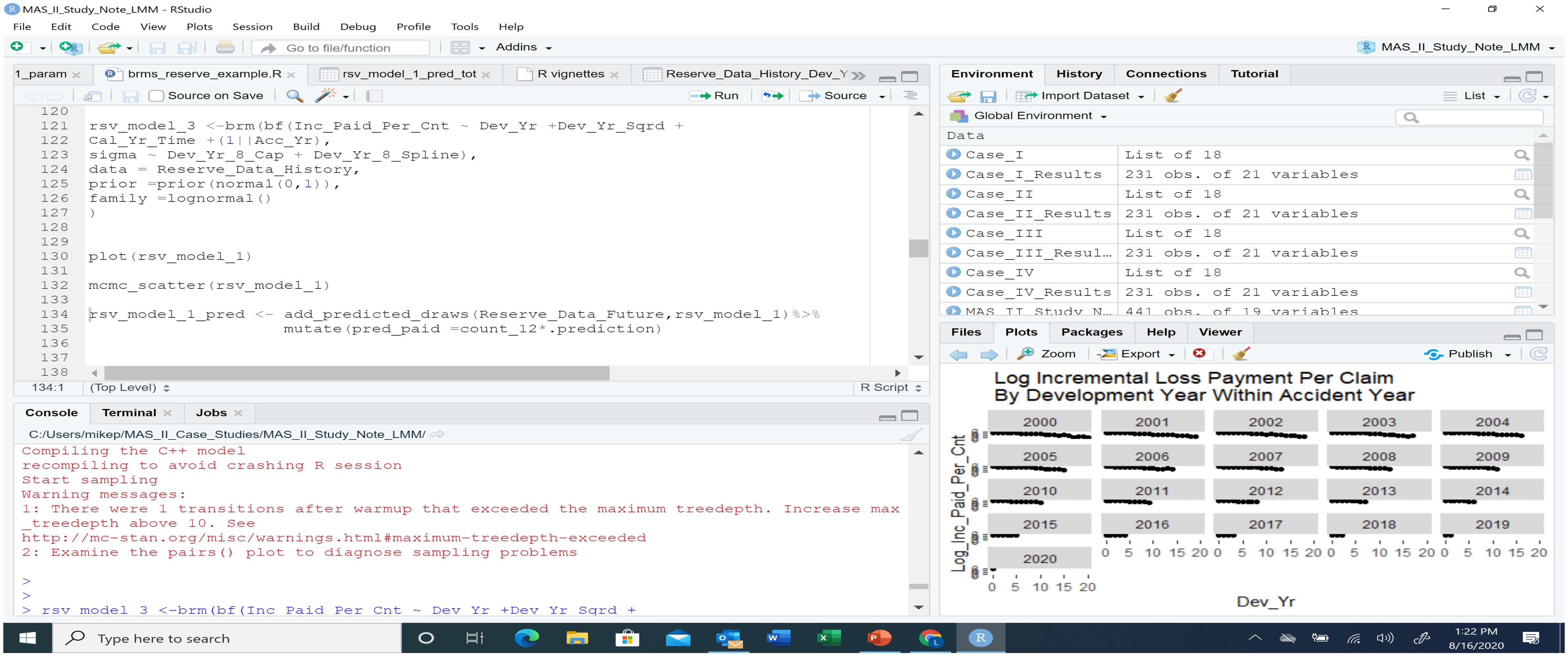
- Pull in R packages
  - CRAN
  - Github
- Create source code using base R and packages
- View program results
  - Error Codes
  - Diagnostics as code executes

View Modeling Results

- Display Graphs
- Inspect data sets created
- Inspect objects created

Modeling Environment

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
- Transformed 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?
- }

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

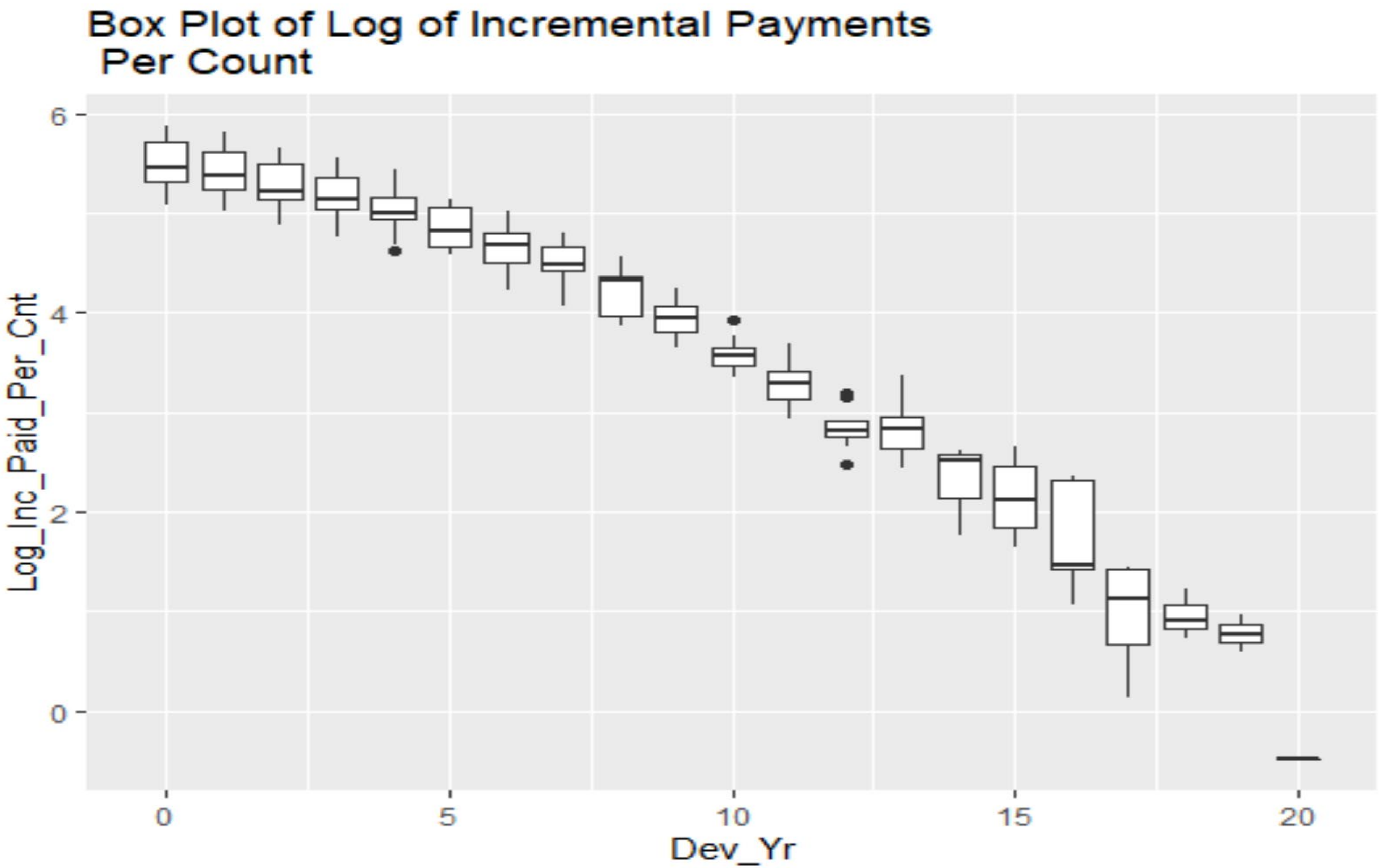
Reserve Estimate Audience

- 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

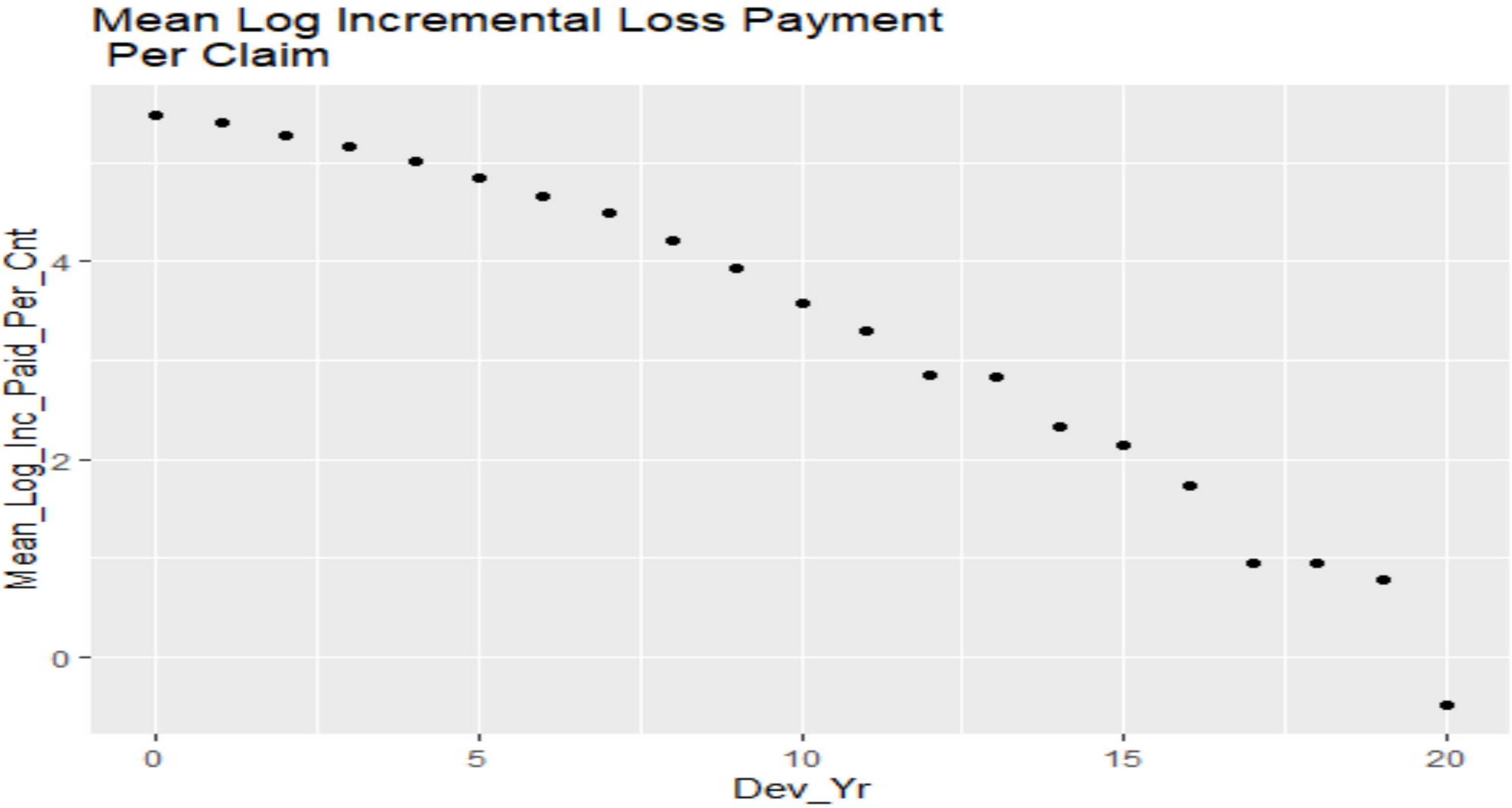
Stochastic Reserve Examples

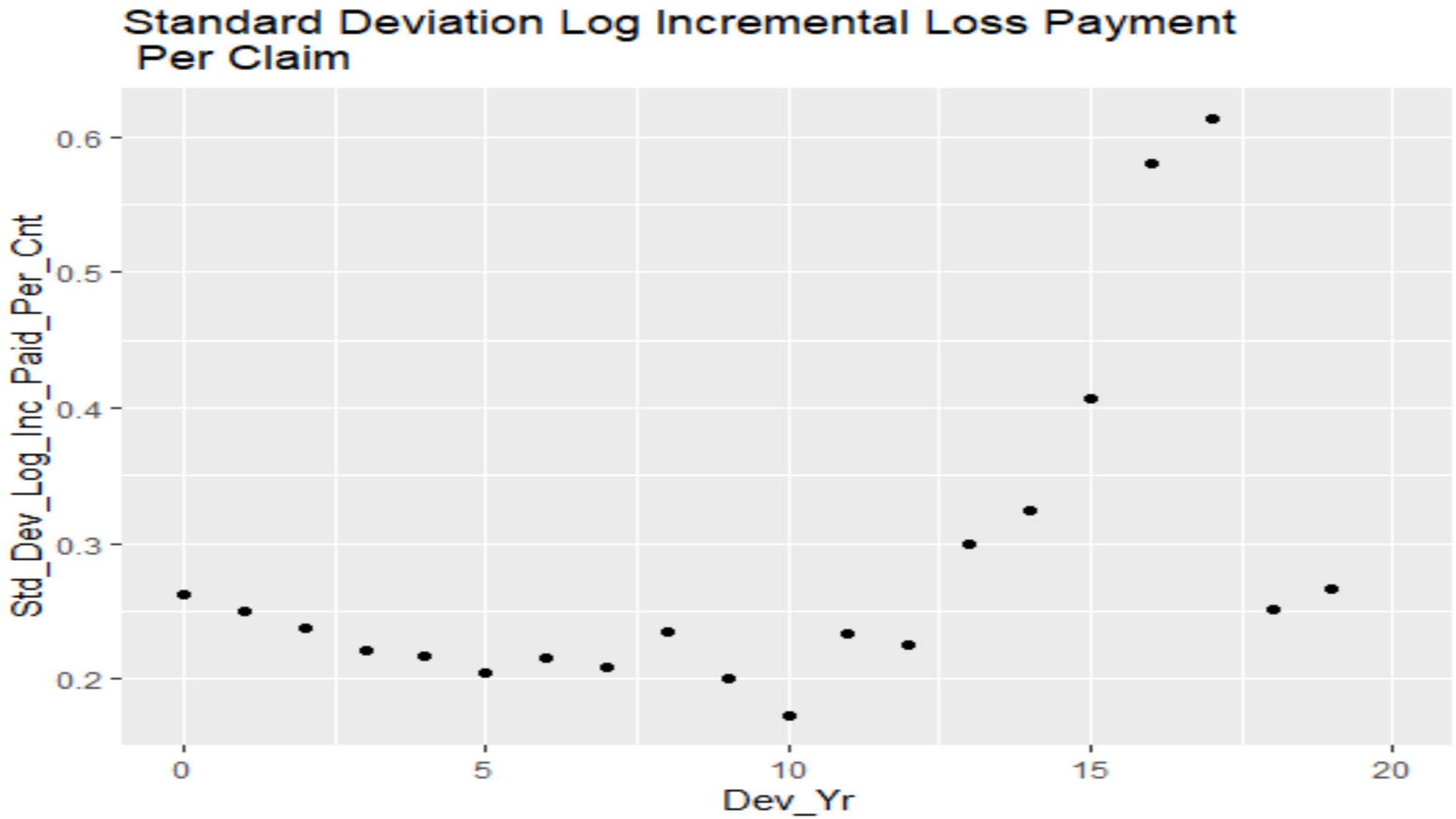
Example Outline

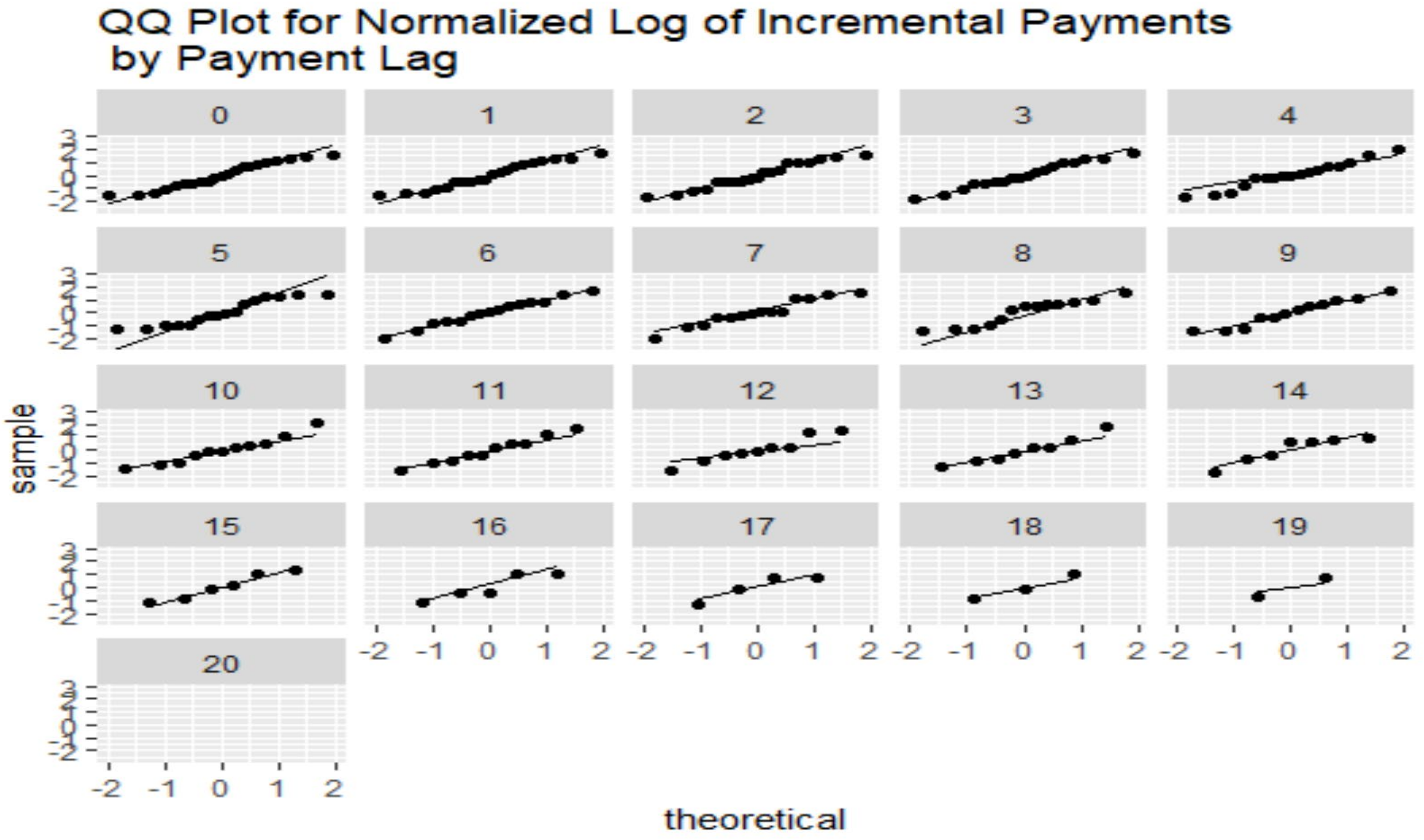
- Simulated incremental paid loss triangle
- Start with graphs to identify likely structures using tidyverse
  - Data organized using dplyr
  - Graphs created with ggplot2
  - Use counts at 12 months as common exposure base
- Write brms package instructions to fit optional structures
  - Create three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

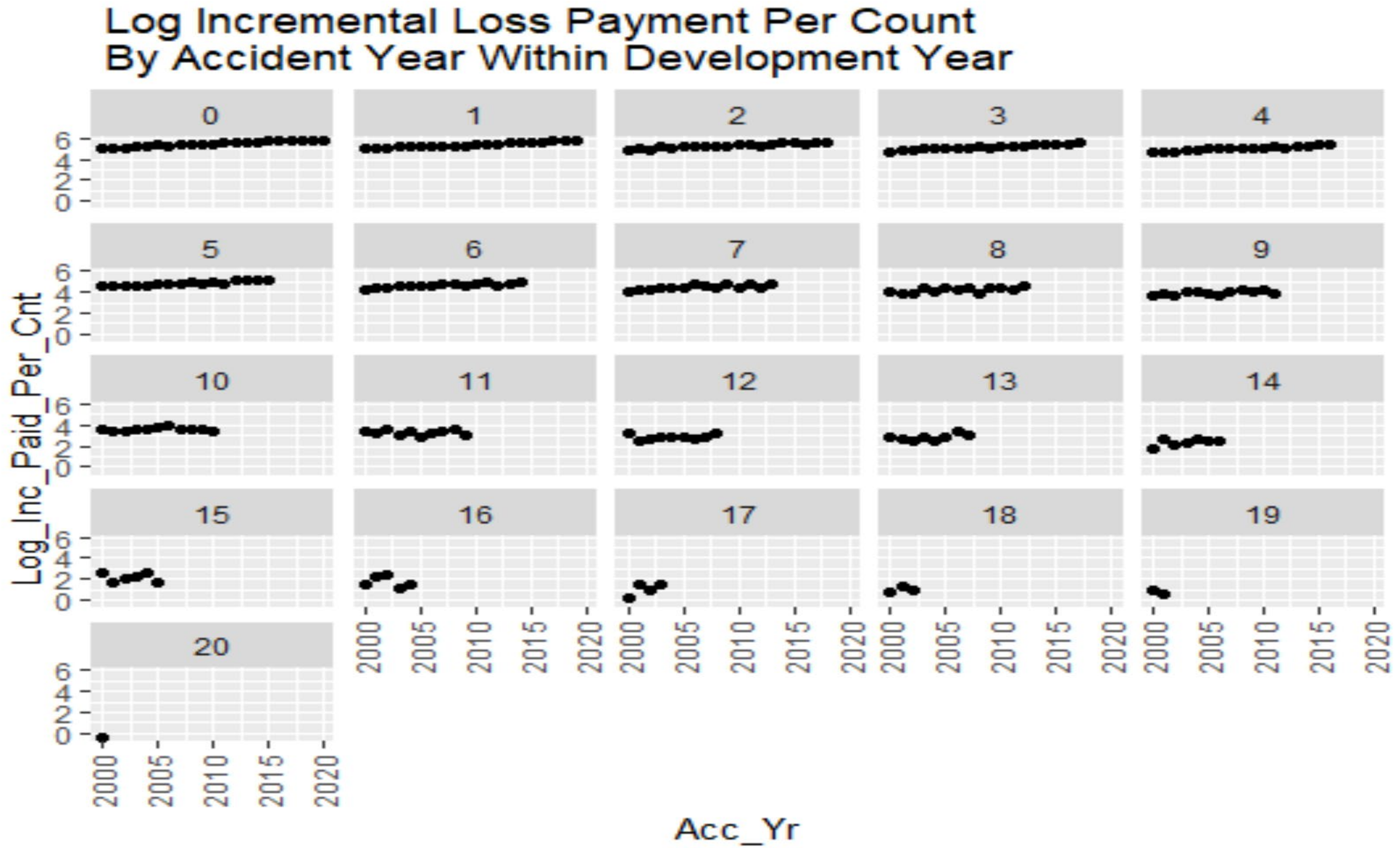


Box Plot of Log of Incremental Payments
Per Count
Log_Inc_Paid_Per_Cnt
6
4
2
0
0
5
10
15
20
Dev_Yr


Mean Log Incremental Loss Payment
Per Claim
Mean_Log_Inc_Paid_Per_Cnt
4
2
0
0
5
10
15
20
Dev_Yr


Standard Deviation Log Incremental Loss Payment
Per Claim
Std_Dev_Log_Inc_Paid_Per_Cnt
0.6
0.5
0.4
0.3
0.2
0
5
10
15
20
Dev_Yr


QQ Plot for Normalized Log of Incremental Payments
by Payment Lag
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
sample
theoretical


Log Incremental Loss Payment Per Count
By Accident Year Within Development Year
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
0
2
4
6
2000
2005
2010
2015
2020
Log_Inc_Paid_Per_Cnt
Acc_Yr

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

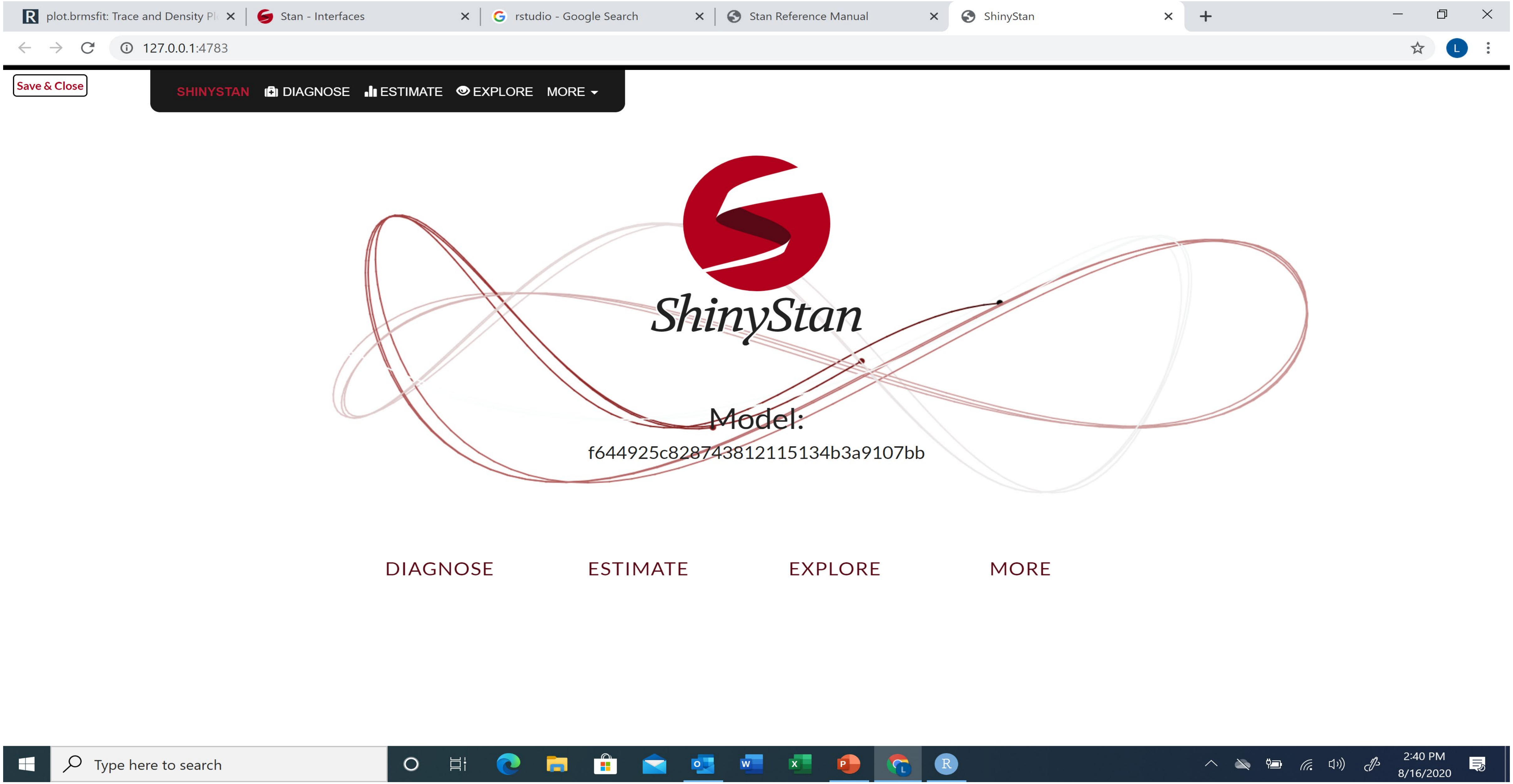
Diagnostics for Reserve Models

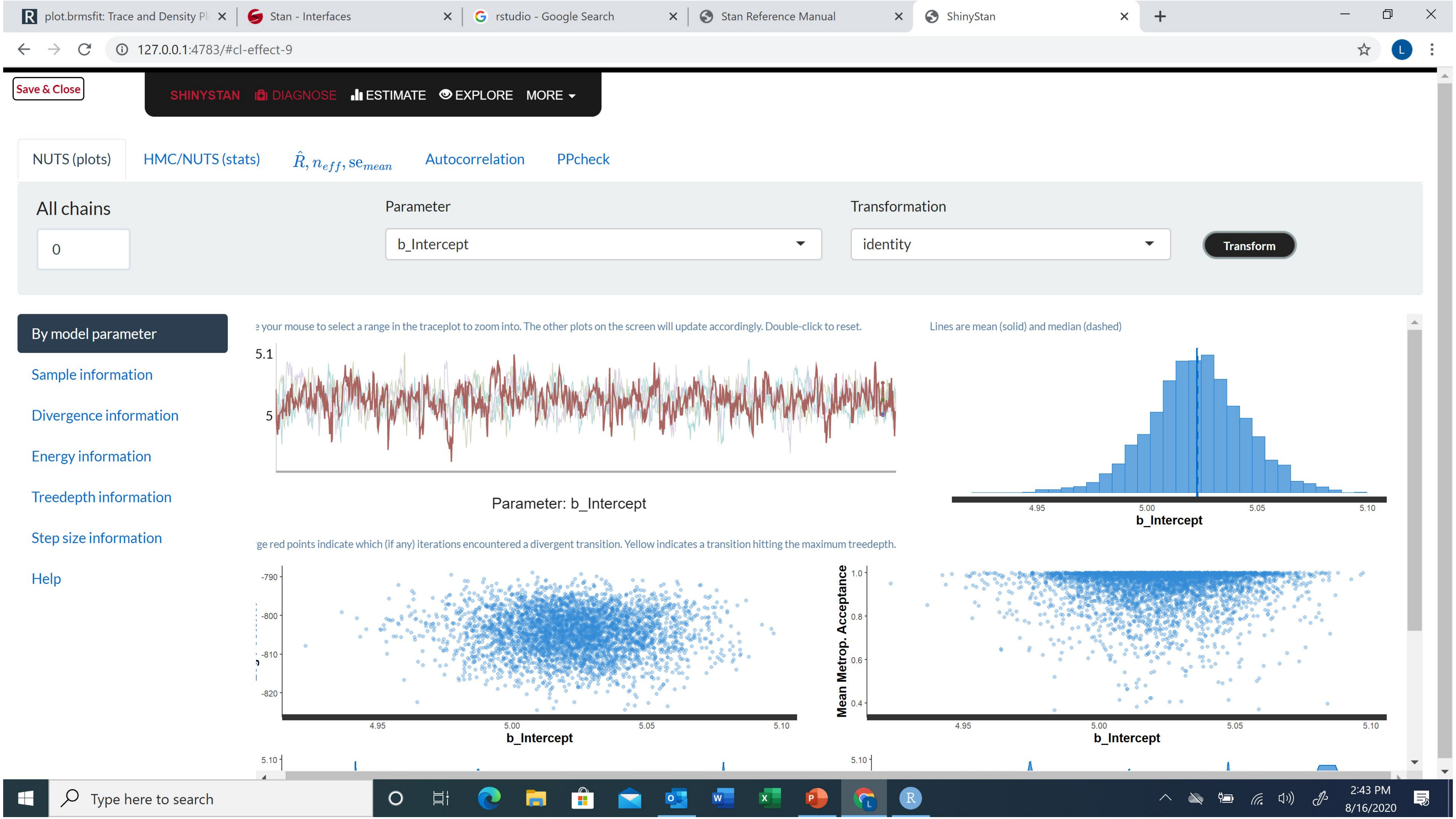
ShinyStan

- 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


plot.brmsfit: Trace and Density Pl
Stan - Interfaces
rstudio - Google Search
Stan Reference Manual
ShinyStan
127.0.0.1:4783
Save & Close
SHINYSTAN
DIAGNOSE
ESTIMATE
EXPLORE
MORE
ShinyStan
Model:
f644925c828743812115134b3a9107bb
DIAGNOSE
ESTIMATE
EXPLORE
MORE
Type here to search
2:40 PM
8/16/2020


plot.brmsfit: Trace and Density Pl
Stan - Interfaces
rstudio - Google Search
Stan Reference Manual
ShinyStan
127.0.0.1:4783/#cl-effect-9
Save & Close
SHINYSTAN
DIAGNOSE
ESTIMATE
EXPLORE
MORE
NUTS (plots)
HMC/NUTS (stats)
$\hat{R}, n_{eff}, se_{mean}$
Autocorrelation
PPcheck
All chains
0
Parameter
b_Intercept
Transformation
identity
Transform
By model parameter
Sample information
Divergence information
Energy information
Treedepth information
Step size information
Help
e your mouse to select a range in the traceplot to zoom into. The other plots on the screen will update accordingly. Double-click to reset.
Lines are mean (solid) and median (dashed)
Parameter: b_Intercept
b_Intercept
ge red points indicate which (if any) iterations encountered a divergent transition. Yellow indicates a transition hitting the maximum treedepth.
Mean Metrop. Acceptance
b_Intercept
b_Intercept
Type here to search
2:43 PM
8/16/2020

Save & Close

SHINYSTAN DIAGNOSE ESTIMATE EXPLORE MORE

Parameters plot | Posterior summary statistics | Generate LaTeX table

Digits 3

Glossary

Copy Print Download Column visibility Show 10 entries

Regex searching

	n_eff	Rhat	mean	mcse	sd	2.5%	25%	50%	75%	97.5%
b_Intercept	580	1.002	5.023	0.001	0.022	4.978	5.009	5.023	5.037	5.065
b_sigma_Intercept	1,488	0.999	-4.194	0.004	0.162	-4.491	-4.309	-4.199	-4.087	-3.866
b_Dev_Yr	1,577	1	-0.096	0	0.003	-0.102	-0.098	-0.096	-0.093	-0.089
b_Dev_Yr_Sqrd	4,000	1.001	-0.011	0	0	-0.011	-0.011	-0.011	-0.01	-0.01
b_Cal_Yr_Time	579	1.001	0.041	0	0.002	0.038	0.04	0.041	0.042	0.045
b_sigma_Dev_Yr	1,133	1.003	0.111	0.001	0.031	0.051	0.09	0.111	0.131	0.172
b_sigma_Dev_Yr_4_Cap	1,559	1.001	0.215	0.002	0.076	0.068	0.164	0.215	0.267	0.365
b_sigma_Dev_Yr_10_Cap	996	1.004	0.075	0.002	0.054	-0.03	0.038	0.074	0.11	0.183
sd_Acc_Yr__Intercept	604	1.002	0.045	0	0.009	0.031	0.039	0.044	0.049	0.066
r_Acc_Yr[2000,Intercept]	672	1.002	0.014	0.001	0.022	-0.03	-0.001	0.013	0.028	0.059

Showing 1 to 10 of 31 entries

First Previous Next Last

plot.brmsfit: Trace and Density Pl | Stan - Interfaces | rstudio - Google Search | Stan Reference Manual | ShinyStan

127.0.0.1:4783/#cl-effect-9

Save & Close

SHINYSTAN | DIAGNOSE | ESTIMATE | EXPLORE | MORE

NUTS (plots) | HMC/NUTS (stats) | $\hat{R}, n_{eff}, se_{mean}$ | Autocorrelation | PPcheck

Summary of sampler parameters

Glossary

Warmup: Omit / Include

Statistic: Mean / SD / Max / Min

Decimals: 4

	accept_stat	stepsize	treedepth	n_leapfrog	divergent	energy
All chains	0.9358	0.0110	7.9875	326.7920	0.0000	819.3621
chain1	0.9233	0.0118	7.7960	300.1200	0.0000	819.4552
chain2	0.9541	0.0098	8.1180	378.1360	0.0000	819.5688
chain3	0.9370	0.0108	8.0380	325.6560	0.0000	819.0719
chain4	0.9289	0.0116	7.9980	303.2560	0.0000	819.3523

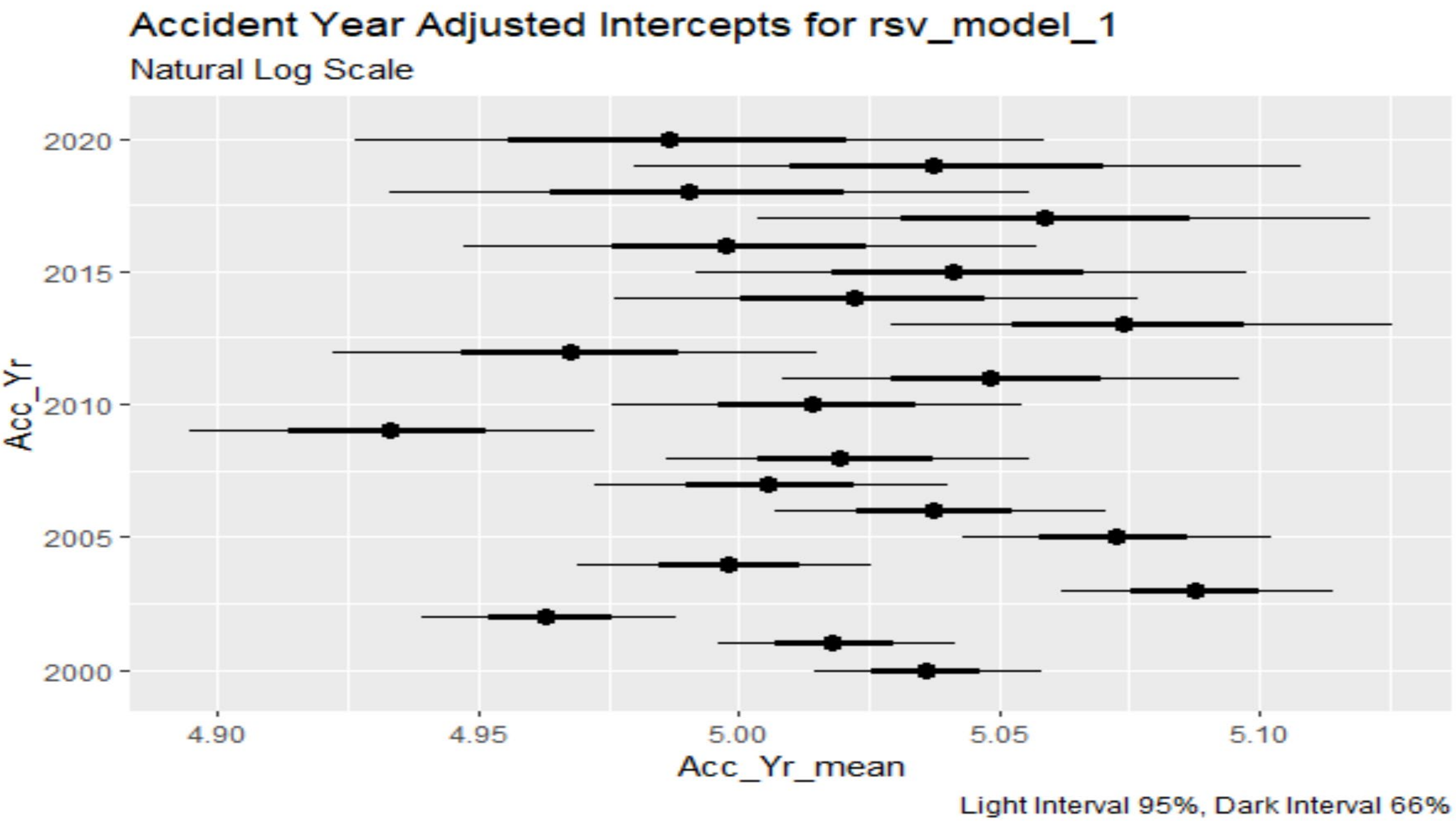
Type here to search

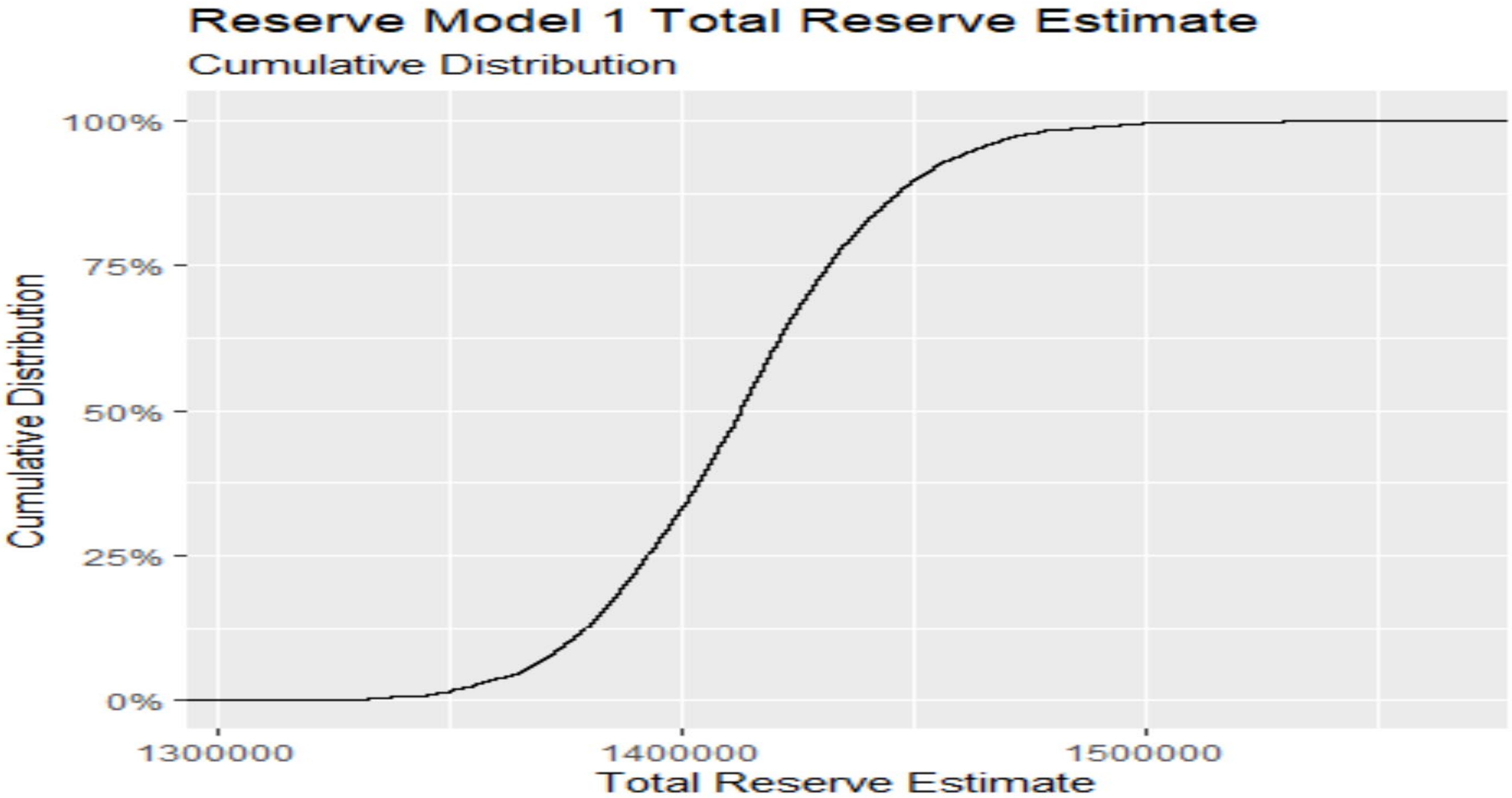
2:51 PM
8/16/2020

Rsv_Model_1 Population Parameter Estimates

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

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.


Accident Year Adjusted Intercepts for rsv_model_1
Natural Log Scale
2020
2015
2010
2005
2000
Acc_Yr
4.90
4.95
5.00
5.05
5.10
Acc_Yr_mean
Light Interval 95%, Dark Interval 66%


Reserve Model 1 Total Reserve Estimate
Cumulative Distribution
Cumulative Distribution
100%
75%
50%
25%
0%
1300000
1400000
1500000
Total Reserve Estimate

Reserve Modeling Options Selected

- Only change the variance estimate model formulas
  - Reserve Model #1: sigma ~ Dev_Yr + Dev_Yr_4_Cap +Dev_Yr_10_Cap
  - Reserve Model #2: sigma ~ Dev_Yr_10_Cap + Dev_Yr_10_Spline
  - Reserve Model #3: sigma ~ Dev_Yr_8_Cap + 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

	elpd_diff	se_diff
Reserve Model #1	0	0
Reserve Model #3	-0.8	2.4
Reserve Model #2	-4.5	3

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
  - brms: An R Package for Bayesian Multilevel Models using Stan, Paul-Christian Burkner, Journal of Statistical Software 2017
  - 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
  - Statistical Rethinking: A Bayesian Course with Examples in R and STAN (Chapman & Hall/CRC Texts in Statistical Science) 2nd Edition, by Richard McElreath
  - Bayesian Data Analysis (Chapman & Hall/CRC Texts in Statistical Science) 3rd Edition, Gelman et.al.