Paper 179-2013

Assessing Drug Safety with Bayesian Hierarchical Modeling Using PROC MCMC and JMP®

Richard C. Zink, Ph.D., JMP Life Sciences, SAS Institute, Inc.

ABSTRACT

Bayesian hierarchical models are advantageous for the analysis of adverse events in clinical trials. First, the models can borrow strength across related events within the MedDRA hierarchy. Second, the models can naturally temper findings likely due to chance. We describe the implementation of two Bayesian hierarchical models (Berry & Berry, 2004; Xia et al., 2010) used for the analysis of adverse events using PROC MCMC. Once models are fit, it is necessary to review convergence diagnostics to ensure that the posterior samples of parameters sufficiently approximate the target distribution. Numerous diagnostics are available within PROC MCMC, and we also present a freely available JMP® add-in for MCMC (Markov Chain Monte Carlo) dynamically interactive diagnostics, summary statistics and graphics.

INTRODUCTION

The analysis of adverse events is an important part of the safety assessment of any experimental drug or vaccine. Disease severity, trial duration and the number of patients under investigation contribute to the sheer number and variety of events that occur, all of which complicate the statistical analysis of comparing the safety profile of the new treatment to a suitable control. In a Frequentist paradigm, when confronted with numerous tests for adverse events, it is natural and appropriate to apply some form of multiplicity adjustment to reduce the likelihood of committing type I errors. In a superiority trial, type I errors would have us conclude that there is a non-zero difference in adverse event rates between the two treatments when in actuality, there is no difference. However, when it comes to the analysis of safety endpoints, committing type II errors due to low power is as important a consideration as committing type I errors (Berry & Berry, 2004; Crowe et al., 2009). Here, type II errors would have us conclude there is no difference in adverse event rates between the two treatments when a difference truly exists. When faced with a large number of comparisons, the False Discovery Rate (FDR) multiplicity adjustment of Benjamini and Hochberg (1995) and the Double FDR method (Mehrotra & Heyse, 2004; Mehrotra & Adewale, 2012) provide a more balanced approach between type I error and power, making them appropriate choices for the analysis of adverse events, and safety endpoints in general.

As an alternative to Frequentist approaches, Berry & Berry (2004) suggest a Bayesian three-level hierarchical mixture model for the analysis of adverse events as a way of coping with multiplicity. To determine whether treatment affects the incidence of a given event compared to control, their logistic model considers and incorporates how treatment affects all events being analyzed, particularly those from the same body system. Fitting such a model using ordinary logistic regression may not be possible since, as DuMouchel (2012) points out, the sparsity of many reported adverse events will likely cause estimation to fail. As a further benefit, the model naturally tempers extreme results that may occur due to the rarity of many events. Xia, Ma & Carlin (2010) study alternate specifications of this model, including a log-linear version that adjusts for the total subject-time at risk.

As Xia et al. (2010) point out, one reason the Berry & Berry model has not been more widely adopted is due to a lack of available software. To remedy this issue, they include code to fit these models using WinBugs software in the appendix of their manuscript. Gemperli (2010) shows how these models can be fit using PROC MCMC of SAS, although his code was written using an experimental version of the software included in SAS 9.2.

In this manuscript, we provide updated code for PROC MCMC to fit the Berry & Berry (2004) logistic regression model and the Xia et al. (2010) log-linear model using SAS 9.3 and SAS/STAT 12.1. Further, we introduce a freely available JMP® 10 add-in to assess MCMC diagnostics, generate forest plots of equal-tailed and highest posterior density (HPD) credible intervals, and calculate univariate and multivariate posterior probabilities (Zink, 2012).

We illustrate the use of these models using data from a vaccine trial described in Mehrotra & Heyse (2004). This clinical trial had 148 and 132 subjects in the treatment and control groups, respectively. There were 40 different adverse events reported across eight different body systems. Data are reproduced in the code in the Appendix.

BAYESIAN HIERARCHICAL MIXTURE MODELS

Suppose there are $s = 1, 2, \ldots S$ classifications of adverse events, which are often grouped by MedDRA system organ class (SOC). Within each SOC, there are numerous events $e = 1, 2, \ldots E_s$ often coded by MedDRA preferred terms. Let Y_{se} and X_{se} be the number of subjects with event e in system organ class s for the Treatment group of N_t subjects and the Control group of N_c subjects, respectively.

For the Berry & Berry (2004) hierarchical logistic regression model, assume $Y_{se} \sim \text{Bin}(N_t, t_{se})$ and $X_{se} \sim \text{Bin}(N_c, c_{se})$ and define $\text{logit}(c_{se}) = \gamma_{se}$ and $\text{logit}(t_{se}) = \gamma_{se} + \theta_{se}$. The parameter θ_{se} is the log-odds ratio for the treatment effect of event e in system organ class s.

Assume the following priors:

Stage 1 Priors:	$\gamma_{se} \sim N(\mu_{\gamma s}, \sigma^2_{\gamma s})$	$\theta_{se} \sim \pi_s \delta(0) + (1- \pi_s) N(\mu_{\theta s}, \sigma^2_{\theta s})$
Stage 2 Priors:	$\mu_{\gamma s} \sim N(\mu_{\gamma 0}, \tau^2_{\gamma 0})$	$\sigma^2_{\gamma s} \sim IG(\propto_\gamma, \beta_\gamma)$
	$\mu_{\theta s} \sim N(\mu_{\theta 0}, \tau^2_{\theta 0})$	$\sigma^2_{\theta s} \sim IG(\propto_\theta, \beta_\theta)$
Stage 3 Priors:	$\mu_{\gamma 0} \sim N(\mu_{\gamma 00}, \tau^2_{\gamma 00})$	$\tau^2_{\gamma 0} \sim IG(\propto_{\gamma 00}, \beta_{\gamma 00})$
	$\mu_{\theta 0} \sim N(\mu_{\theta 00}, \tau^2_{\theta 00})$	$\tau^2_{\theta 0} \sim IG(\propto_{\theta 00}, \beta_{\theta 00})$

Further assume that $\pi_s \sim \text{Beta}(\propto_\pi, \beta_\pi)$, $\propto_\pi \sim \text{Exp}(\lambda_\alpha)$ $I[\propto_\pi > 1]$ and $\beta_\pi \sim \text{Exp}(\lambda_\beta)$ $I[\beta_\pi > 1]$, and set $\mu_{\gamma 00} = \mu_{\theta 00} = 0$, $\tau^2_{\gamma 00} = \tau^2_{\theta 00} = 10$, $\propto_\gamma = \propto_\theta = \propto_{\gamma 00} = \propto_{\theta 00} = 3$, $\beta_\gamma = \beta_\theta = \beta_{\gamma 00} = \beta_{\theta 00} = 1$ and $\lambda_\alpha = \lambda_\beta = 0.1$. The value $\delta(0)$ is a distribution having unit point mass at 0, and $I[.]$ is an indicator function with value 1 if the condition inside the brackets is true, 0 otherwise.

For the Xia et al. (2010) model, let T_{se} and C_{se} be the total time at risk for event e in system organ class s for the treatment and control arms, respectively. For N_t treated subjects, define T_{se} as the summation of ξ_{sei} for $i = 1, 2 \ldots N_t$. Similarly, for N_c control subjects, define C_{se} as the summation of ζ_{sei} for $i = 1, 2 \ldots N_c$. The values ξ_{sei} and ζ_{sei} represent the time from first drug exposure until a subject experiences event e in system organ class s for the first time, or the total time on study if they did not experience the event. In lieu of the binomial assumptions above, assume $Y_{se} \sim \text{Pois}(t_{se}T_{se})$ and $X_{se} \sim \text{Pois}(c_{se}C_{se})$ and define $\log(c_{se}) = \gamma_{se}$ and $\log(t_{se}) = \gamma_{se} + \theta_{se}$. Here, the parameter θ_{se} is the log relative risk for the treatment effect of event e in system organ class s.

Rationale for assumed priors and constants are provided in Berry & Berry (2004) and Xia et al. (2010), though sensitivity analyses should examine the robustness of findings to alternate assumptions. Model fit can be assessed using the deviance information criterion (obtained through the DIC option in the PROC MCMC statement).

SPECIFICATION OF PROC MCMC

The PROC MCMC specification for the Berry & Berry (2004) hierarchical logistic regression model is contained in the Appendix. The code requests 20,000 samples from the posterior distribution after discarding a burn-in of 2,000 samples. Three separate Markov chains are requested to evaluate the convergence of the chains to the target distributions of the parameters. Unlike the data structures assumed in Gemperli (2010) and Xia et al. (2010), the PROC MCMC specification here assumes each treatment-event combination is a separate row, which is more in line with the vertical structure of many CDISC data standards (CDISC Submission Data Standards Team, 2012). When summarizing data from the AE domain, it is important that rows exist for treatment-event combinations that do not occur so that 0 events are explicitly specified in these cases.

Of particular note when comparing our specification to that of Gemperli (2010) is the use of RANDOM statements that simplify the specification of random-effects. While the RANDOM statement was available in earlier releases of PROC MCMC, this particular code requires SAS/STAT 12.1. Prior to SAS/STAT 12.1, random effects could not be specified as hyperparameters to other random effects. For example,

```
random mu_Ga ~ normal(mu_G0, prec = tau2_G0) subject = s monitor = (mu_Ga);
random G ~ normal(mu_Ga, prec = tau_Ga) subject = e monitor = (G);
```

would not have been previously permitted.

Time at risk was not available for the data described in Mehrotra & Heyse (2004). However, assuming time at risk is available in order to fit the Xia et al (2010) model, make the following modifications to incorporate T_{se} and C_{se}. We assume that the times at risk for each event, similar to the number of subjects experiencing the event, were previously summarized.

```
data ae;
       format term $28.;
       input e s term $ Y X T C;
       datalines;
       …
       …
run;

data ae(drop = Y X T C);
  set ae;
  count = Y; trt = 1; ntc = 148; risktime = T; output;
  count = X; trt = 0; ntc = 132; risktime = C; output;
run;
```

Further, in the PROC MCMC call change

```
ptc = logistic(lp);
model count ~ binomial(n = ntc, p = ptc);
```

to

```
ptc = log(lp);
model count ~ poisson(mean = ptc*risktime);
```

For either model, several instances of the following WARNING may appear in the log file after the PROC MCMC code compiles:

```
WARNING: There is still significant autocorrelation after 500 lags, and the effective sample
     size for the parameter XXXX might not be estimated accurately.
```

These warnings can typically be eliminated using the THIN=*k* option in the PROC MCMC statement to keep every *k*th posterior sample. However, this is particularly wasteful. High autocorrelation is an indicator of poor sampling efficiency, so unless storage of posterior samples is of concern, the recommendation is to keep all posterior samples.

As a note, parameters whose priors are defined using a PRIOR statement need to be included in one or more PARMS statements. From the PROC MCMC documentation: “Each PARMS statement defines a block of parameters, and the blocked Metropolis algorithm updates the parameters in each block simultaneously.” The code in the Appendix assumes one PARMS statement for each parameter, though these statements could have easily been specified differently. The user can examine the section “Blocking of Parameters” in the PROC MCMC documentation for details and recommendations.

As specified, the MCMC macro passes seeds 500, 203 and 140 to PROC MCMC to generate multiple chains. Users are encouraged to choose specific seeds in order to reproduce results. Alternatively, users can initialize parameters within the PARMS statement to evaluate whether parameters converge to the target distribution from extreme starting values. The example “Gelman-Rubin Diagnostics” in the PROC MCMC documentation provides further explanation.

To generate odds ratios or risk ratios within the PROC MCMC call, add the following lines:

```
array OR_[40] OR_1 - OR_40;
```

and

```
OR_[e] = exp(T_[e]);.
```

Similarly, indicators variables used to count whether treatment effects meet certain thresholds can be generated:

```
array IND_[40] IND_1 - IND_40;
```

and

```
IND_[e] = (OR_[e] > 1);.
```

Here, we have generated indicators based on whether there is an increase in risk for the treatment based on the odds or risk ratio. Remember to add these terms to the MONITOR option in the PROC MCMC statement.

Though PROC MCMC can calculate numerous summary statistics and diagnostics for the posterior samples, we did not include these specifications in our code in order to demonstrate the JMP10 MCMC Diagnostics add-in (Zink, 2012). Only posterior samples from the log odds ratios of the treatment effect (T_1 – T_40) are kept in the output data set for illustrative purposes, though in general, diagnostics for all parameters should be evaluated.

REVIEWING SUMMARY STATISTICS AND DIAGNOSTICS FOR POSTERIOR SAMPLES USING JMP®

Here we describe a freely available JMP 10 add-in for reviewing summary statistics and diagnostics for posterior samples from one or more Markov chains (Zink, 2012). The user must have a free SAS profile in order to download the MCMC Diagnostics add-in. The input data can be from any source (e.g., SAS, WinBugs, BRugs), but it is assumed that the data set of posterior samples is sorted by Markov chain number and iteration (whether provided in the dialog or not), and that each chain has the same number of posterior samples. An example data set is shown in Figure 1, which summarizes the treatment parameters from the MCMC model using data from Mehrotra & Heyse (2004). All credible intervals and diagnostic calculations are performed on samples from the Markov chain that appears first in the data set (here, number 1). Samples from any additional Markov chains are used only to compute Gelman-Rubin diagnostics. Most variables within the data set will represent parameters from the specified model. The dialog can accommodate many parameters at once for analysis, though sets of similar parameters should be submitted so that forest plots of credible intervals are most meaningful.

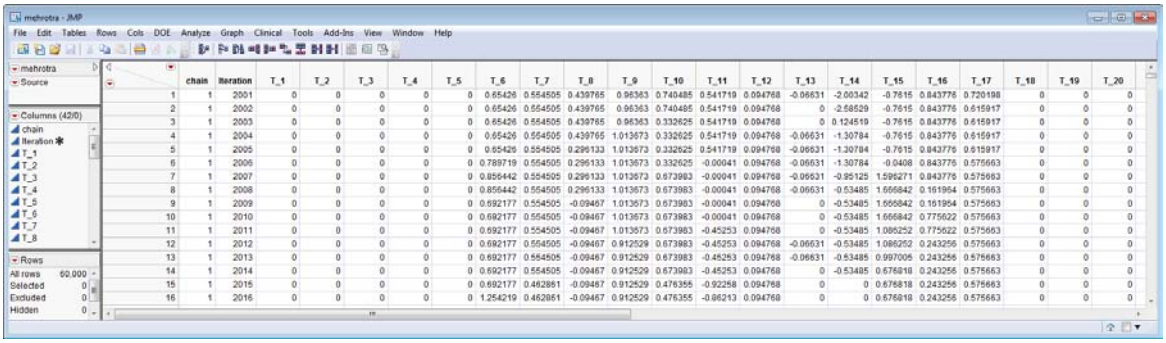

Figure 1. Posterior Samples from PROC MCMC Using Data from Mehrotra & Heyse (2004)

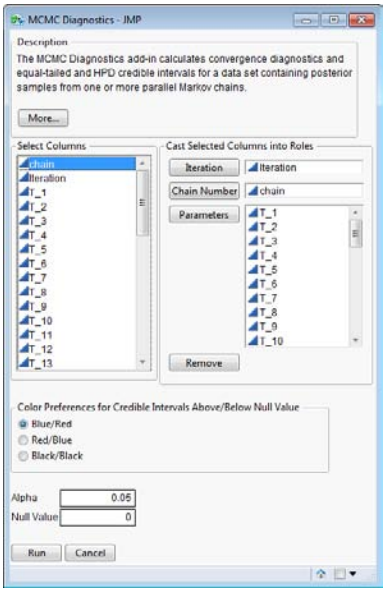

Figure 2. JMP10 MCMC Diagnostics Dialog

The MCMC Diagnostics dialog (Figure 2) displays all of the variables (COLUMNS) of the input data set. The only requirement to run the add-in is that at least one PARAMETER should be specified. In these instances, it is assumed that all samples are from a single Markov chain, and samples will be numbered in trace plots from 1 to the total number of rows in the data set. If ITERATION is provided, trace plots will reflect appropriate sample numbers (say, if burn-in samples were removed). CHAIN specifies a numeric value if multiple Markov chains are generated to assess parameter convergence to the target distribution. COLOR PREFERENCE specifies the color (default Blue/Red) of any credible intervals that exclude the NULL VALUE (default 0). Under the defaults, intervals entirely to the right or left of the null value will be blue or red, respectively. Color can be particularly useful for the Bayesian hierarchical models discussed in this paper. The mixture distribution including a point mass at zero could mean the endpoints of many credible intervals will be extremely close to 0 (meaning no treatment effect), making visual interpretation without color difficult. ALPHA (default 0.05) calculates $(1-\alpha)^*100\%$ credible intervals for the forest plots.

The add-in generates the MCMC Viewer Window in Figure 3. The Diagnostics Tab provides histograms, density function curves and summary statistics of the posterior samples from Chain 1 for all parameters. Trace plots summarize the behavior of the Chain 1 samples over the iterations and can be used to assess convergence of the chain to the target distribution. Histograms and summary statistics summarize the autocorrelation of Chain 1 posterior samples up to lag 25. If the analysis includes multiple Markov chains, trace plots summarize all chains simultaneously, and Gelman-Rubin Statistics are provided.

The interactivity of JMP is a key benefit of the add-in. The diagnostic output of all parameters except the first is initially collapsed. This output can be opened or closed by selecting the outline boxes in the Tab. By default, a non-parametric kernel density curve is fit to the posterior samples in the histograms. However, the user can add multiple reference lines from the red triangle menu of the histogram (Figure 4). If needed, a partial autocorrelation or variogram summary figure can be generated from the red triangle menu of any trace plot.

The Forest Plots of Credible Intervals Tab provides two figures of 95% credible intervals for the parameters using samples from Chain 1. Figure 5 summarizes equal-tailed credible intervals of the posterior samples. Here, the lower and upper endpoints for these intervals correspond to the 2.5^{th} and 97.5^{th} percentiles of the samples, respectively. Figure 6 summarizes the 95% highest posterior density (HPD) credible intervals, which are the narrowest intervals covering 95% of all samples. For both figures, the mean and median sample values are summarized using circles and diamonds, respectively. Only the intervals for T_17, which corresponds to the adverse event of irritability, exclude the assumed null value of 0. As in Berry & Berry (2004), we can conclude that treatment has an important effect on this adverse event. If needed, the underlying statistics for these figures are a button-click away (Figure 7). Note that PROC MCMC documentation refers to forest plots as caterpillar plots.

The Univariate Posterior Probability Calculator enables the user to define probability statements for the parameters, the results of which are summarized in a table (Figure 8). Ranges can be added manually, or the sliders can be used to select limits which are restricted to the minimum and maximum values of the samples for all parameters from Chain 1. The Multivariate Posterior Probability Calculator lets the user define probability statements that consider two or more parameters simultaneously. Figure 9 illustrates this calculator using only the treatment parameters from the first five adverse events. We calculate the posterior probability that treatment has an undesirable effect (essentially each parameter greater than 0) on astenia/fatigue, fever, infection-fungal, infection-viral and malaise simultaneously as 0.1756. However, since this particular model does not account for the association between different adverse events, this probability may be misleading. The multivariate calculator makes use of the JMP Data Filter to select data table rows meeting the criteria defined in the filter. Alternatively, the user can open the data table and select rows manually, or apply a function to the columns of interest. Once rows are selected, the user can push the Calculate Posterior Probability button.

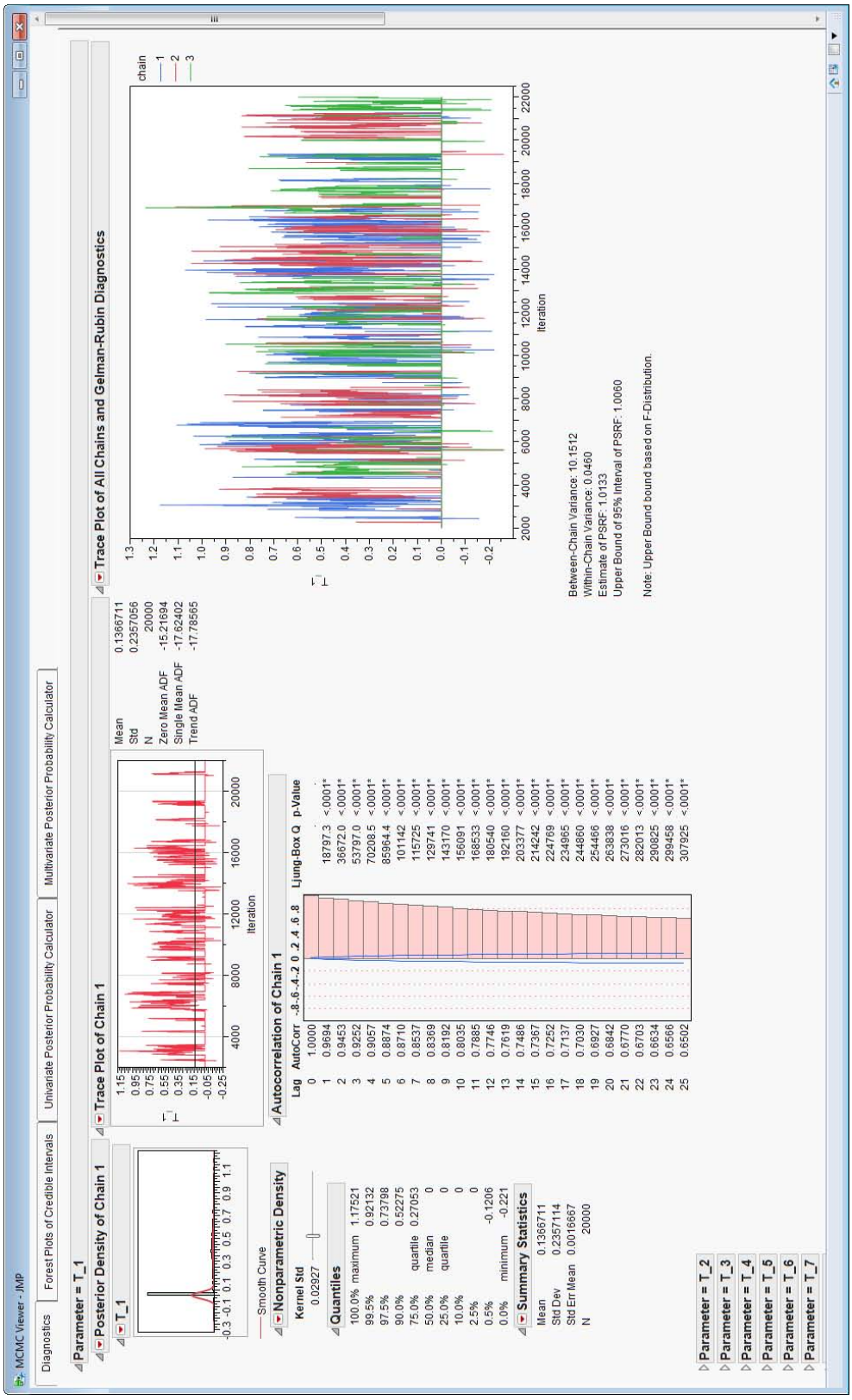

Figure 3. MCMC Diagnostics Including Histogram and Density Function of Posterior Samples, Trace Plots, Autocorrelation Assessment and Gelman-Rubin Statistics

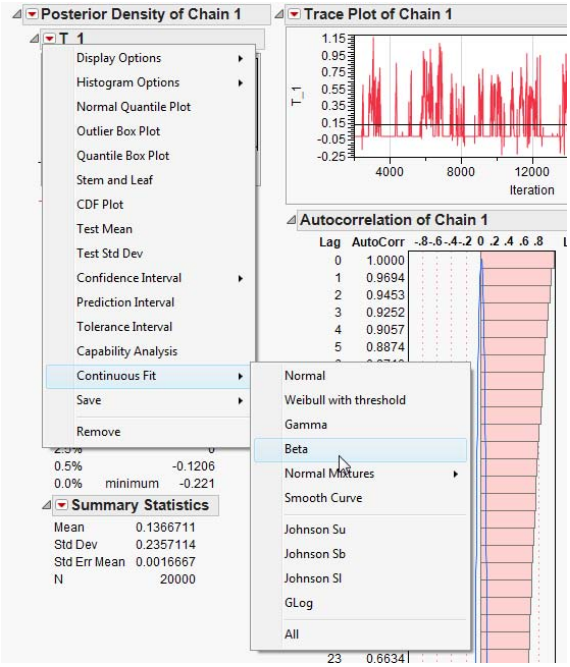

Figure 4. Adding Reference Curves Interactively

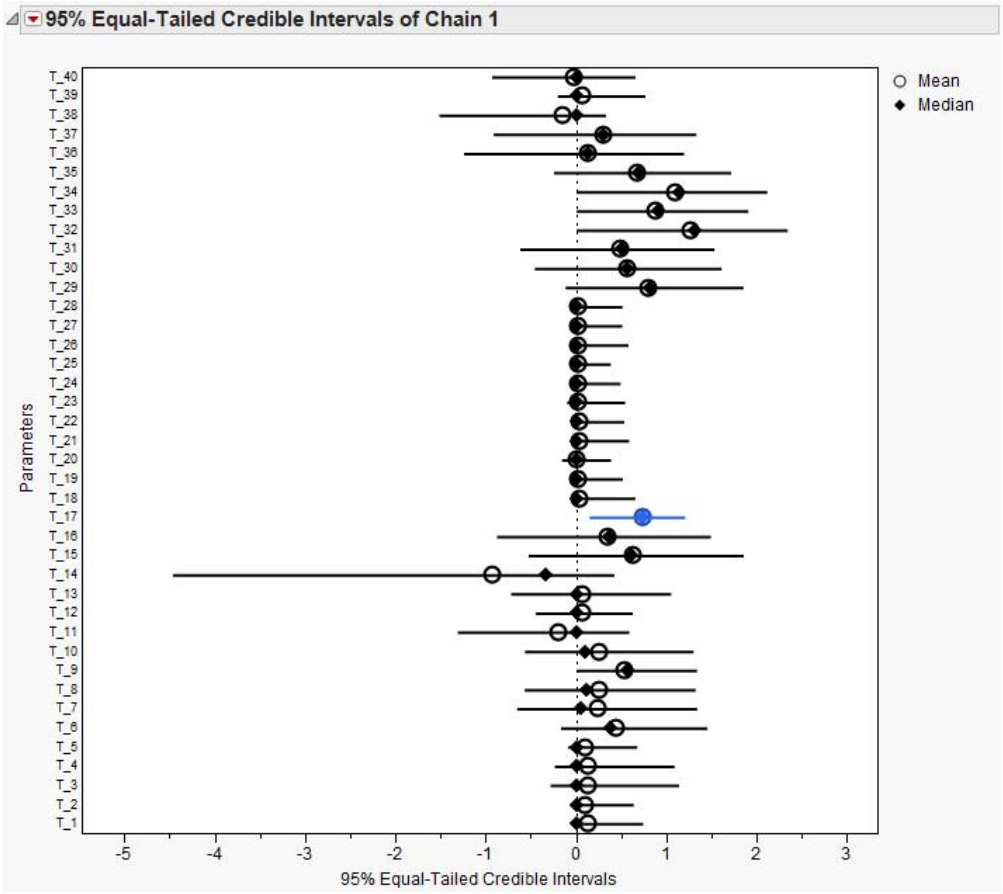

Figure 5. 95% Equal-Tailed Credible Intervals of Posterior Samples

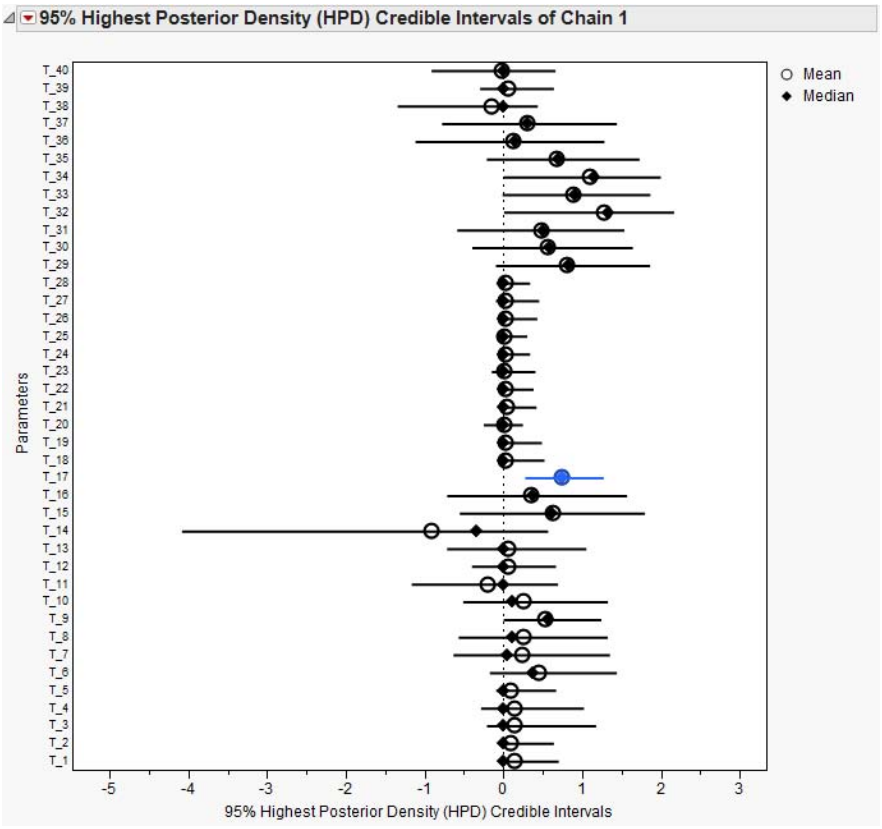

Figure 6. 95% Highest Posterior Density (HPD) Credible Intervals of Posterior Samples

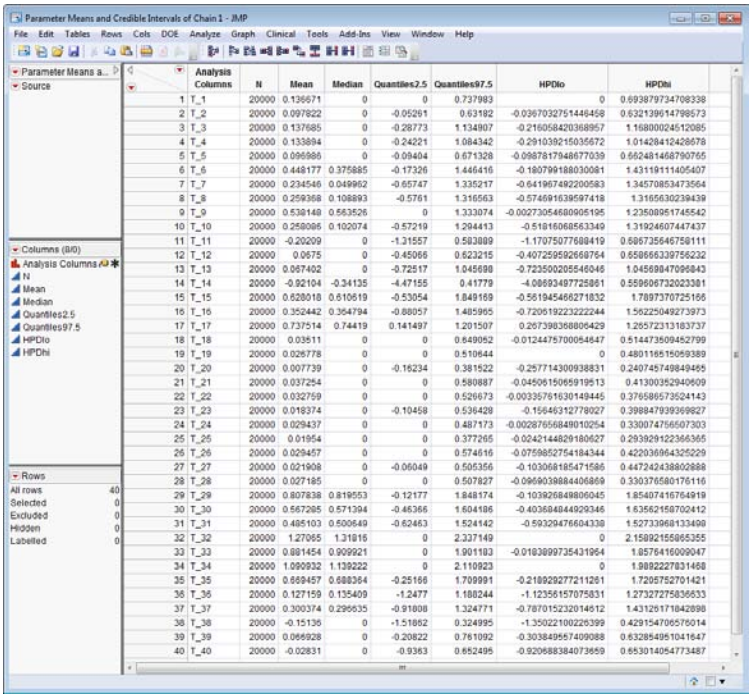

Figure 7. Data Table of Parameter Means and Credible Intervals of Chain 1

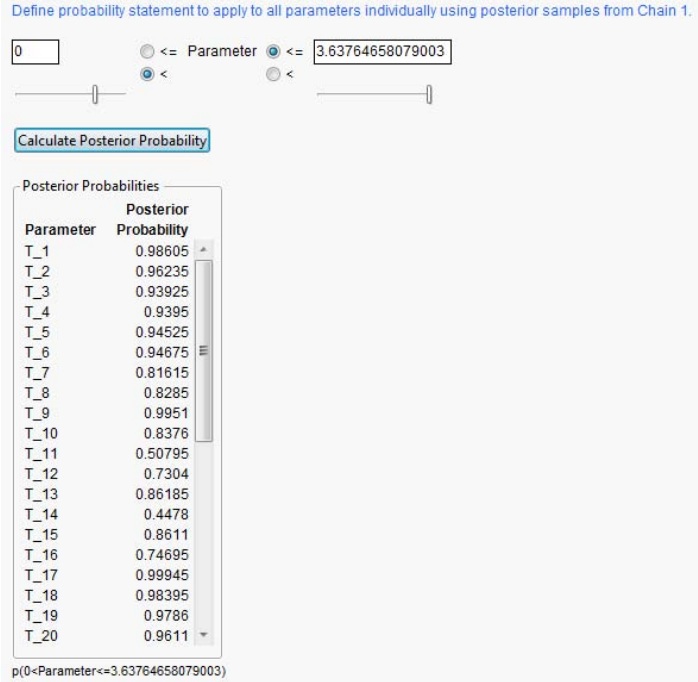

Figure 8. Univariate Probability Calculator

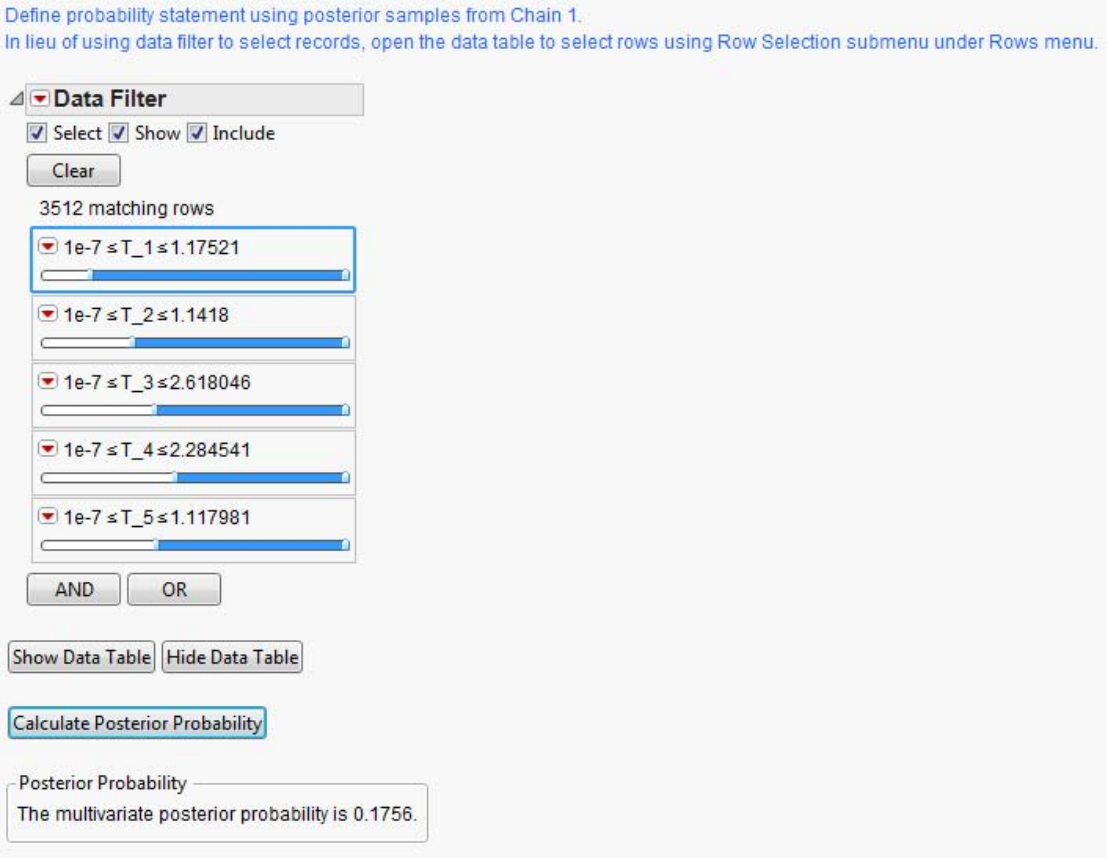

Figure 9. Multivariate Probability Calculator

CONCLUSIONS

In this paper, we summarized two Bayesian three-level hierarchical mixture models for the analysis of adverse events and provided PROC MCMC code using SAS 9.3 and SAS/STAT 12.1 software (Berry & Berry, 2004; Xia et al., 2010). Further, we demonstrated the flexibility of JMP for reviewing posterior samples from models fit using Markov Chain Monte Carlo (Zink, 2012). Though the models described here only consider a single study comprising two treatment arms, the PROC MCMC specification provided can be extended to include additional treatment arms or effects to account for multiple studies. These hierarchical models will be included in a future release of JMP Clinical.

The models as defined utilize summary statistics at the event level and do not explicitly model the association among adverse events. If interested, the reader can review the multivariate Bayesian logistic regression model of DuMouchel (2012) for an example of hierarchical models that take advantage of subject-level data. However, one benefit of relying on data summarized at the event level is that it is straightforward to model a mixture of ongoing trials and data from trials taken from the literature.

REFERENCES

Benjamini, Y & Hochberg, Y. (1995). “Controlling the False Discovery Rate: a Practical and Powerful Approach to Multiple Testing.” *Journal of the Royal Statistical Society B: Statistical Methodology* 57: 289–300.

Berry, SM & Berry, DA. (2004). “Accounting for Multiplicities in Assessing Drug Safety: A Three-Level Hierarchical Mixture Model.” *Biometrics* 60: 418-426.

CDISC Submission Data Standards Team. (2012). “Study Data Tabulation Model, Version 3.1.3” Round Rock, TX: Clinical Data Interchange Standards Consortium. Available for download: http://www.cdisc.org/sdtm

Crowe, BJ, Xia, HA, Berlin, JA, Watson, DJ, Shi, H, Lin, SL, Kuebler, J, Schriver, RC, Santanello, NC, Rochester, G, Porter, JB, Oster, M, Mehrotra, DV, Li, Z, King, EC, Harpur, ES & Hall, DB. (2009). “Recommendations for Safety Planning, Data Collection, Evaluation and Reporting During Drug, Biologic and Vaccine Development: a Report of the Safety Planning, Evaluation, and Reporting Team.” *Clinical Trials* 6: 430-440.

DuMouchel, W. (2012). “Multivariate Bayesian Logistic Regression for Analysis of Clinical Study Safety Issues.” *Statistical Science* 27: 319–339.

Gemperli, A. (2010). “Safety Analysis Using Bayesian Simulation Methods in SAS® 9.2.” *Pharmaceutical Programming* 3: 29-36.

Mehrotra, DV & Heyse, JF. (2004). “Use of the False Discovery Rate for Evaluating Clinical Safety Data.” *Statistical Methods in Medical Research* 13: 227-238.

Mehrotra, DV & Adewale, AJ. (2012). “Flagging Clinical Adverse Experiences: Reducing False Discoveries Without Materially Compromising Power for Detecting True Signals.” *Statistics in Medicine* 31: 1918–1930.

Xia, HA, Ma, H & Carlin, BP. (2011). “Bayesian Hierarchical Modeling for Detecting Safety Signals in Clinical Trials.” *Journal of Biopharmaceutical Statistics* 21: 1006-1029.

Zink, RC. (2012). MCMC Diagnostics Add-in for JMP 10. Available for download (must have free SAS profile): http://support.sas.com/demosdownloads/downarea_t4.jsp?productID=111697&jmpflag=N

ACKNOWLEDGMENTS

The author would like to thank the following individuals for their valuable feedback and suggestions: Fang K. Chen, Haijun Ma, Arati Mejdal, Kelci Miclaus and Amy Xia.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Name: Richard C. Zink, Ph.D.
Enterprise: JMP Life Sciences, SAS Institute Inc.
Address: SAS Campus Drive
City, State ZIP: Cary, North Carolina 27513
E-mail: richard.zink@jmp.com
Web: http://www.jmp.com/lifesciences-resources/
Blog: http://blogs.sas.com/content/jmp/author/rizink/
Twitter: @rczink

APPENDIX: PROC MCMC CODE

```
libname out "<<Your Directory Name Here>>";

data ae;
       format term $28.;
       input e s term $ N_T N_C ;
       datalines;
       1 1 Astenia/fatigue 57 40
       2 1 Fever 34 26
       3 1 Infection-fungal 2 0
       4 1 Infection-viral 3 1
       5 1 Malaise 27 20
       6 2 Anorexia 7 2
       7 2 Cendisiasis-oral 2 0
       8 2 Constipation 2 0
       9 2 Diarrhea 24 10
       10 2 Gastroenteritis 3 1
       11 2 Nausea 2 7
       12 2 Vomiting 19 19
       13 3 Lymphadenopathy 3 2
       14 4 Dehydration 0 2
       15 5 Crying 2 0
       16 5 Insomnia 2 2
       17 5 Irritability 75 43
       18 6 Bronchitis 4 1
       19 6 Congestion-nasal 4 2
       20 6 Congestion-respiratory 1 2
       21 6 Cough 13 8
       22 6 Infection-upper-respiratory 28 20
       23 6 Laryngotracheobronchitis 2 1
       24 6 Pharyngitis 13 8
       25 6 Rhinorrhea 15 14
       26 6 Sinusitis 3 1
       27 6 Tonsillitis 2 1
       28 6 Wheezing 3 1
       29 7 Bite/sting 4 0
       30 7 Eczema 2 0
       31 7 Pruritis 2 1
       32 7 Rash 13 3
       33 7 Rash-diaper 6 2
       34 7 Rash-measles/rubella-like 8 1
       35 7 Rash-varicella-like 4 2
       36 7 Urticaria 0 2
       37 7 Viral-exanthema 1 2
       38 8 Conjunctivitis 0 2
       39 8 Otitis-media 18 14
       40 8 Otorrhea 2 1
       ;
run;

data ae(drop = N_T N_C);
  set ae;
  count = N_T; trt = 1; ntc = 148; output;
  count = N_C; trt = 0; ntc = 132; output;
run;
```

```
%macro MCMC(suffix = , seed = );
  %let mu_G00 = 0;   %let mu_T00 = 0;
  %let tau2_G00 = 10;  %let tau2_T00 = 10;
  %let A_G = 3;     %let A_T = 3;
  %let A_G00 = 3;    %let A_T00 = 3;
  %let B_G = 1;     %let B_T = 1;
  %let B_G00 = 1;    %let B_T00 = 1;
  %let L_A = 0.1;    %let L_B = 0.1;

  proc mcmc data = ae seed = &seed nmc = 20000 dic maxtune = 500
            monitor = (_parms_ T_1-T_40) outpost = outpost&suffix nbi = 2000;
   *** Treatment Effects for the 40 Adverse Events ***;
   array T_[40] T_1 - T_40;
   parms mu_G0;
   parms tau2_G0;
   parms mu_T0;
   parms tau2_T0;
   parms A_P;
   parms B_P;

   *** Hyperpriors for Gammas ***;
   prior mu_G0 ~ normal(&mu_G00, prec = %sysevalf(1/&tau2_G00));
   prior tau2_G0 ~ gamma(&A_G00, iscale = &B_G00);
   *** Hyperpriors for Thetas ***;
   prior mu_T0 ~ normal(&mu_T00, prec = %sysevalf(1/&tau2_T00));
   prior tau2_T0 ~ gamma(&A_T00, iscale = &B_T00);
   *** Hyperpriors for Pis ***;
   prior A_P ~ expon(iscale = &L_A, lower = 1);
   prior B_P ~ expon(iscale = &L_B, lower = 1);

   random mu_Ga ~ normal(mu_G0, prec = tau2_G0) subject = s monitor = (mu_Ga);
   random tau_Ga ~ gamma(&A_G, iscale = &B_G) subject = s monitor = (tau_Ga);
   random G ~ normal(mu_Ga, prec = tau_Ga) subject = e monitor = (G);

   random mu_Ta ~ normal(mu_T0, prec = tau2_T0) subject = s monitor = (mu_Ta);
   random tau_Ta ~ gamma(&A_T, iscale = &B_T) subject = s monitor = (tau_Ta);
   random T1 ~ normal(mu_Ta, prec = tau_Ta) subject = e monitor = (T1);

   random P ~ beta(A_P, B_P) subject = s monitor = (P);
   random Bi ~ binary(P) subject = s monitor =(Bi);

   T_[e] = (1 - Bi) * T1;

   lp = G + T_[e] * trt;
   ptc = logistic(lp);
   model count ~ binomial(n = ntc, p = ptc);
  run;
%mend MCMC;

*** Generate 3 Independent Chains ***;
%MCMC(suffix = 1, seed = 500);
%MCMC(suffix = 2, seed = 203);
%MCMC(suffix = 3, seed = 140);

data out.events(keep = chain iteration T_1-T_40);
  set outpost1(in = ina) outpost2(in = inb) outpost3(in = inc);
  if ina then chain = 1;
  else if inb then chain = 2;
  else if inc then chain = 3;
run;
```