Paper SA-10-2012

Testing the Bayesian Suite of SAS® Procedures using Ecological Data and Comparing Simulations with WinBUGS

Matthew Russell, Dept. of Forest Resources, University of Minnesota, St. Paul, MN

ABSTRACT

There has been an increase in use in recent years of Bayesian modeling procedures in the fields of ecology and forestry. This paper uses a dataset of snags (standing dead trees) compiled across a range of forest types to test the PHREG, GENMOD, and MCMC procedures in SAS® which perform Bayesian analyses. Snags are an important component of ecological processes as they provide wildlife habitat, contribute to biodiversity, and provide a mix of forest structure. Strengths of the Bayesian procedures in SAS, such as the ability to simulate large datasets quickly, are highlighted using some example snag data. Comparisons of model output and processing time using the MCMC procedure are made with the popular WinBUGS program run using R statistical software.

INTRODUCTION

SAS® has recently introduced methodologies for doing Bayesian analysis. Some of these methodologies have been developed for procedures that have long existed (e.g. PHREG, GENMOD, and LIFEREG), while other procedures have been more recently developed (e.g. MCMC). In the fields of forestry and ecology, most researchers have traditionally used the WinBUGS program for doing Bayesian analyses, and there has been a breadth of literature devoted to the software (see McCarthy 2007 and Kéry 2010). An assessment of how the Bayesian methodologies in SAS perform with complex ecological data has yet to be tested.

Snags (defined as standing dead trees) are an integral component to maintaining diverse forest ecosystems and provide habitat for multiple wildlife species. Snag survival, defined as the period of time that snags remain standing in a forest, is needed to quantify the length of time in which snags provide a suitable habitat for wildlife. Similarly, the probability of a snag experiencing reduced height through time might provide information regarding which decomposing fungi species may occur on snags. Snag survival and snag height loss might be related to tree species, the diameter of the tree, and how long the tree has been standing dead.

The goal of this analysis was to use the snag data as a modeling dataset to test the multitude of procedures available for doing Bayesian methodologies in SAS. Specific objectives were to (1) employ the PHREG, GENMOD, and MCMC procedures to showcase their inherent differences, and (2) to compare the general effectiveness and model output of the SAS® procedures with those provided by the popular Bayesian software WinBUGS.

DATA

Snag measurements were collected on the Penobscot Experimental Forest in Bradley, Maine. Snags occurred in areas under a variety of differently managed forests. Measurements on snags were: tree number (`TreeNum`; a unique number for each snag measurement), snag diameter at breast height in centimeters, measured at 4.5 feet above the ground (`dbh`), snag height in meters from ground to top of snag (`ht`), number of years since tree death (`ysd`), a censoring variable for snag survival (`censored` = 1 if snag fell; `censored` = 0 if snag was standing), and an indicator for whether or not a snag experienced a loss in height (`snagloss` = 1 if snag lost height; `snagloss` = 0 if snag did not lose height).

Data were collected on 24 eastern white pine trees (Figure 1).

Obs	TreeNum	dbh	ht	ysd	censored	snagloss
1	1	6.4	2.9	6.5	0	1
2	2	15.7	8.5	8.5	0	1
3	3	7.9	3.6	4.5	1	.
4	4	17.8	17.3	12.5	0	1
5	5	5.6	7.2	2.5	0	0
.						
.						
.						

Figure 1. First five lines of snag data.

PROC PHREG

Proportional hazards (PH) regression models are a class of survival models that differ from others in that they describe how a unit increase in a covariate changes survival with respect to a baseline hazard rate (Cox 1972). The PH model is considered semiparametric because no assumption is placed on the shape of a baseline hazard (nonparametric component), but it assumes a linear effect of the covariates on the hazard (parametric component). A hazard function $h(t|X)$ assesses the risk of failure at some time t, conditioned on the probability of survival to time t:

$$h(t \mid X) = h(t)\exp\left(X_1\alpha_1 + ... + X_p\alpha_p\right)$$

where $h(t)$ is the baseline hazard, X_i's are the covariates, and α_i's are the estimated parameters. There are several advantages for using the PH model in ecology, especially for estimating snag survival. First, the PH model allows for independent variables that can change through time. This is an advantage when considering ecological data as biotic and abiotic conditions can change both temporally and spatially. Second, proportional hazards models ensure that probabilities and subsequent estimates of those probabilities are constrained between 0 and 1 (O'Quigley 2008). Lastly, since the effect of covariates is the same at all times t, hazard ratios can be developed which compare the changes in probability for each unit increase in a covariate.

Snag survival was modeled by censoring snag observations that fell. Given the usefulness of `dbh` in estimating snag survival (Garber et al. 2005), `dbh` was used as a predictor in the PH model. Mean number of years since death (`ysd`) was used as the censoring variable. Hence, the PH model estimating snag survival was:

$$h(\mathrm{ysd}|\,\mathrm{dbh}) = h(\mathrm{ysd})\,\exp\left(\mathrm{a}_1\mathrm{dbh}\right)$$

where h(ysd|dbh) is the hazard function predicting `ysd` conditioned on `dbh`.

SAS has recently incorporated the `bayes` statement into the GENMOD, LIFEREG, and PHREG procedures. As an example fitting the snag model in the PHREG procedure, common functions in the `bayes` statement would specify a random seed for the Monte Carlo simulation (`seed=`) and specify an output dataset which saves the posterior samples (`outpost=`). A prior displaying a normal distribution with mean of 0 and large variance (a noninformative prior) can be specified on the regression coefficients with the `coeffprior=` statement. By specifying plot=, users are able to view the appropriate trace plots of the simulations and posterior density plots for regression coefficients. The number of Monte Carlo simulations, number of simulations to use as burn-in, and the thinning parameter which aims to reduce autocorrelation between successive Monte Carlo runs can be set with the `nmc=`, `nbi=`, and `thin=`, statements, respectively. In the snag example, a 100,000 Monte Carlo run was following a 10,000-run burn-in.

Snag survival was modeled by censoring observations that fell. Here, `dbh` was used as the predictor variable. The `baseline` statement was specified, where two observations (a snag of dbh 10 and 20 cm) were contained in the dataset `pred`. Survival plots were also supplied by output, as displayed with the Bayesian highest probability density (`hpd`) intervals. The following code fit the proportional hazards model using PROC PHREG:

```
proc phreg data=snag plots(cl=hpd overlay)=survival;
baseline covariates=pred out=predout;
model ysd*censored(0)= dbh;
bayes seed=1  outpost=coeffout plots=trace plots=density coeffprior=normal(var=1e6)
      nmc=100000 nbi=10000 thin=3;
run; quit;
```

Trace plots indicated good mixing of the Markov chains. Posterior density plots indicated a smoothed curved with a negative value associated with the `dbh` parameter (Figure 2).

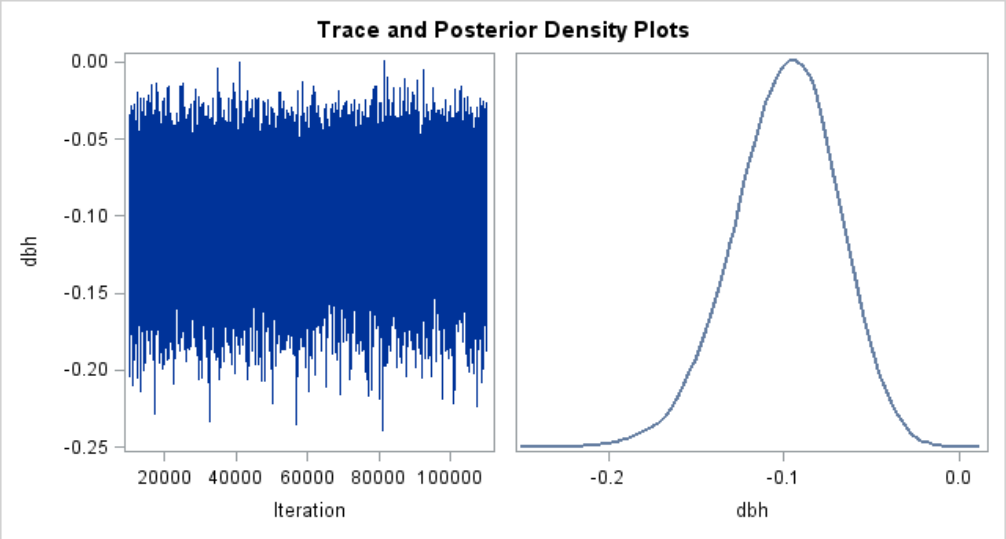

Figure 2. Trace and posterior densities, two diagnostic plots to assess model convergence.

Survival for the 10 and 20 cm snags indicated differences in probability of survival for the two differently-sized trees. These curves were non-overlapping when comparing their 95% highest probability density curves (Figure 3). Results indicate larger-sized snags to display higher probabilities of survival.

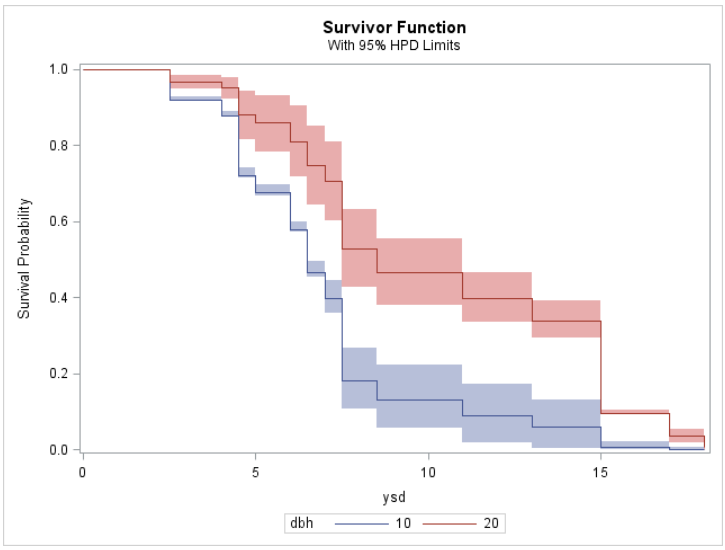

Figure 3. Estimated survival curves for two trees, with 95% highest probability density intervals.

We can fit the same statements in PROC PHREG, but this time using the `hazardratio` statement. Since the effect of covariates is the same at all times *t* with the PHREG model, hazard ratios compare the changes in probability of snag survival for each unit increase in a covariate. Here, hazard ratios can be specified for each unit increase in 5, 10, and 20 cm:

```
proc phreg data=snag plots(cl=hpd overlay)=survival;
baseline covariates=pred out=predout;
model ysd*censored(0)= dbh;
bayes seed=1  plots=trace plots=density coeffprior=normal(var=1e6)
       nmc=100000 nbi=10000 thin=3;

hazardratio 'HR DBH=5'  dbh /unit=5;
hazardratio 'HR DBH=10' dbh /unit=10;
hazardratio 'HR DBH=20' dbh /unit=20;
run; quit;
```

For the continuous variable `dbh`, the hazard ratio compares the hazard for each change in 5, 10, and 20 cm (Figure 4).

HR DBH=5: Hazard Ratios for dbh

Description	N	Mean	Standard Deviation	Quantiles			95% Equal-Tail Interval		95% HPD Interval	
				25%	50%	75%				
dbh Unit=5	33334	0.6158	0.0902	0.5530	0.6133	0.6753	0.4465	0.7993	0.4454	0.7969

HR DBH=10: Hazard Ratios for dbh

Description	N	Mean	Standard Deviation	Quantiles			95% Equal-Tail Interval		95% HPD Interval	
				25%	50%	75%				
dbh Unit=10	33334	0.3873	0.1129	0.3058	0.3761	0.4561	0.1994	0.6389	0.1906	0.6229

HR DBH=20: Hazard Ratios for dbh

Description	N	Mean	Standard Deviation	Quantiles			95% Equal-Tail Interval		95% HPD Interval	
				25%	50%	75%				
dbh Unit=20	33334	0.1628	0.0966	0.0935	0.1415	0.2080	0.0398	0.4082	0.0187	0.3515

Figure 4. Changes in hazards for an increase of 5, 10, and 20 cm on survival of the white pine snags.

PROC GENMOD

PROC GENMOD is a useful SAS procedure that can be used to fit a wide variety of general linear models, including logistic regressions. A logistic regression model predicting the probability of a snag experiencing no height loss (π_{noloss}) was fit using snag dbh as a covariate:

$$\pi_{noloss} = \frac{1}{1+\exp[b_0 + b_1 dbh]}$$

In SAS, this equation was fit in a Bayesian framework using a binomial distribution with a logit link function:

```
proc genmod data = snag;
model snagloss = dbh / dist=bin link=logit;
bayes seed=231 cprior=normal(var=1e6) plots=trace plots=density
      nmc=100000 nbi=10000 thin=3 ;
run;
```

Output indicated that dbh may not be a suitable covariate for predicting the probability of π_{noloss} for these white pine trees. Posterior summaries indicated large standard deviations in comparison to the mean of the posterior distribution, and the value 0 was contained in the 95% highest probability density interval (Figure 5).

The SAS System

The GENMOD Procedure

Bayesian Analysis

Posterior Summaries

Parameter	N	Mean	Standard Deviation	Percentiles		
				25%	50%	75%
Intercept	33334	-0.1302	0.7381	-0.6240	-0.1282	0.3665
dbh	33334	-0.00444	0.0367	-0.0285	-0.00419	0.0197

Posterior Intervals

Parameter	Alpha	Equal-Tail Interval		HPD Interval	
Intercept	0.050	-1.5882	1.3171	-1.5729	1.3298
dbh	0.050	-0.0774	0.0681	-0.0788	0.0666

Posterior Correlation Matrix

Parameter	Intercept	dbh
Intercept	1.000	-0.796
dbh	-0.796	1.000

Figure 5. Posterior summaries from the GENMOD procedure specifying the `bayes` statement.

PROC MCMC

The MCMC procedure is a general purpose Markov Chain Monte Carlo procedure designed to fit Bayesian models. PROC MCMC differs substantially from other SAS procedures in that inference is solely Bayesian (SAS Institute Inc. 2010). Coding in the MCMC procedure appears remarkably similar to what we've seen using the `bayes` statement in the PHREG and GENMOD procedures. Key differences are that parameters for coefficients b0 and b1 in addition to the variance parameter sigma are specified in the `parms` statement, and prior distributions with the `prior` statement. Here, noninformative priors are specified as normal for b0 and b1, and inverse gamma for sigma. The `model` statement brings everything together in the procedure to begin the Monte Carlo simulation.

To test PROC MCMC, we might be interested in conducting a simple linear regression to estimate snag `ht` using `dbh` as the predictor variable:

```
proc mcmc data=snag outpost=snagpost seed=1 nmc=100000 nbi=10000 thin=3 dic;
    parms b0 0 b1 0;
    parms sigma 1;
         prior b0 b1 ~ normal(mean = 0, var = 1e6);
         prior sigma ~ igamma(shape = 0.01, scale = 0.01);
              mu= b0+b1*dbh;
                   model  ht~ normal(mu,var=sigma);
run; quit;
```

A common indicator of model performance used, the deviance information criterion (`dic`) was 156.663 for this model. To expand on the model, we might be interested in fitting a multiple regression that includes both `dbh` and `ysd`:

```
proc mcmc data=snag_wp outpost=snagpost seed=1 nmc=100000 nbi=10000 thin=3 dic;
    parms c0 0 c1 0 c2;
    parms sigma 1;
         prior c0 c1 c2~ normal(mean = 0, var = 1e6);
         prior sigma ~ igamma(shape = 0.01, scale = 0.01);
              mu= c0+c1*dbh+c2*ysd;
                   model  ht~ normal(mu,var=sigma);
run; quit;
```

In this model, the deviance information criterion was 156.599. This indicates that there is little difference between the simple linear regression over the model including both `dbh` and `ysd` as predictors. For more on interpreting DIC values in a similar fashion to AIC values, see Burnham and Anderson 2002.

COMPARISONS WITH WINBUGS

WinBUGS is a software program initially developed in the UK that does Bayesian analysis using Markov Chain Monte Carlo methods (The BUGS Project 2012). Today, development of the BUGS software is accomplished in OpenBUGS, the open source version of BUGS (OpenBUGS 2012).

A typical WinBUGS session includes at least the following three windows to initiate an MCMC run: (1) a code file that specifies the statements to run (typically saved as a .bug file), (2) the Specification Tool that loads and compiles the data, and (3) the Sample Monitor Tool which allows the user to specify which output is seen. After output is specified and the MCMC run is complete, output is provided in the WinBUGS workspace (Figure 6).

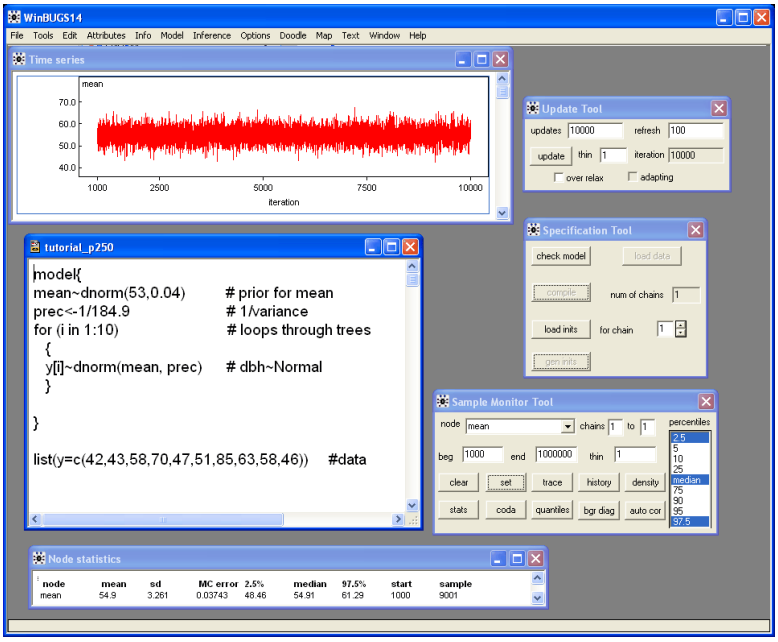

Figure 6. A typical WinBUGS session.

Several packages in the R statistical programming language have been developed which can execute BUGS script. Examples of these are the 'R2WinBUGS' (Sturtz et al. 2005) and 'BRugs' packages (Thomas et al. 2006). Figure 7 contains the code to perform an identical simple linear regression as shown in the PROC MCMC code above. The code makes use of the developed R packages, calling the BUGS model written in a .bug file:

whitepine.r

```
library("R2WinBUGS")
library("BRugs")

setwd("C:\\Users\\matt\\Documents"

#A simple linear regression
wp<-read.csv('C:\\Users\\matt\\Documents\\mwsug2012.csv')
wp.data = list(dbh=wp$dbh,ht=wp$ht)
wp.data$numTrees = 24

inits<- function() (list(b=c(0.1,0.1),prec=100))
params<-c('b[]')
bug.file='whitepine.bug'
modelCheck(bug.file)

SLR.sim=bugs(wp.data,inits,bug.file,par=params,
  n.iter=110000,n.burnin=10000,n.thin=3,
  bugs.directory='c:/Program Files/OpenBUGS/',program='openbugs')

samplesStats('*')
samplesHistory('*')
samplesDensity('*')
print(SLR.sim)
```

whitepine.bug

```
model{
   for(i in 1:2) {b[i]~dnorm(0,1.0E-6)}
   prec~dgamma(0.001,0.001)

    for(i in 1:numTrees) {
      preds[i]<-(b[1]+(b[2]*dbh[i]))

        ht[i]~dnorm(preds[i],prec)

   }
}
```

Figure 7. BUGS code for fitting a simple linear regression (right) with script employing 'R2WinBUGS' and 'BRugs' packages in R (left) to perform Markov chain Monte Carlo simulations.

Model output comparing PROC MCMC and WinBUGS results are shown in Table 1. Generally, both programs produced similar parameter estimates for the simple and multiple linear regressions under investigation, where the c2 parameter in the multiple linear regression showed the most difference between PROC MCMC and WinBUGS. A significant advantage of the PROC MCMC was that it completed the simulations at a much quicker speed than WinBUGS (e.g. 250% quicker for the simple linear regression).

Table 1. Model output from PROC MCMC and WinBUGS after fitting simple and multiple linear regressions simulating 100,000 runs that followed a 10,000-run burn-in (*n* = 24 observations).

	Simple linear regression			
	b0	b1	DIC	Runtime[3]
PROC MCMC[1]	2.036 [-2.042, 6.301]	0.423 [0.227, 0.618]	156.7	3.2 sec
WinBUGS[2]	2.048 [-2.122, 6.228]	0.423 [0.227, 0.617]	156.9	11.2 sec

	Multiple linear regression				
	c0	c1	c2	DIC	Runtime
PROC MCMC[1]	3.969 [-0.957, 8.934]	0.558 [0.278, 0.832]	-0.475 [-1.192, 0.191]	156.6	3.4 sec
WinBUGS[2]	3.933 [-1.050, 8.911]	0.560 [0.284, 0.835]	-0.473 [-1.165, 0.222]	157.0	15.9 sec

[1]Using SAS/STAT® 9.3

[2]Using OpenBUGS 3.2.2 release; run through R with the R2WinBUGS (Sturtz et al. 2005) and BRugs (Thomas et al. 2006) packages.

[3]Simulations run on a Dell Latitude E6420 x64-based PC with an Intel® Core™ i5-2540M CPU @ 2.60 GHz

CONCLUSIONS

Procedures for doing Bayesian analyses in SAS/STAT 9.3 were found to be straightforward with flexibility to account for complex datasets. By incorporating the `bayes` statement into existing procedures, users need not learn entire new procedures for specifying common modeling techniques. The generalized MCMC procedure allows users the opportunity to specify any variety of Bayesian model they wish.

From an ecological perspective, there may be several advantages of using WinBUGS over that of SAS/STAT® 9.3. First, although the SAS® procedures are excellent for modeling ecological data, little is available analyzing datasets using traditional experimental design techniques. This remains an important tool in ecological and agricultural studies. To showcase its importance, Kéry (2010) devotes four chapters to analysis of variance and covariance procedures under a Bayesian framework. Also, updating simulations after initially specifying a simulation can be carried out easily in WinBUGS, whereas a user would need to run an entire set of simulations again by increasing the number of samples to draw in SAS. However, given its speed in running simulations and the familiar SAS language, these SAS procedures should be in the toolbox of anyone employing Bayesian modeling techniques in the ecological disciplines.

REFERENCES AND RECOMMENDED READING

The BUGS Project. 2012. The BUGS project. Available online at http://www.mrc-bsu.cam.ac.uk/bugs/winbugs/contents.shtml; last accessed 16 August 2012.

Burnham, K.P., and Anderson, D.R. 2002. Model selection and multi-model inference: a practical information theoretic approach. New York, Springer-Verlag. 496 p.

Cox, D.R., 1972. Regression models and life tables. *Journal of the Royal Statistical Society, Series B*. 34, 187-220.

Ellison A.M. 2004. Bayesian inference in ecology. *Ecology Letters* 7: 509–520.

Garber, S.M., Brown, J.P., Wilson, D.S., Maguire, D.A., Heath, L.S., 2005. Snag longevity under alternative silvicultural regimes in mixed-species forests of central Maine. *Canadian Journal of Forest Research*. 35, 787-796.

Gardiner, J., 2012. Modeling heavy-tailed distributions in healthcare utilization by parametric and Bayesian methods. Proceedings of SAS® Global Forum 2012 Conference. April 22-25, 2012. 6 p.

Kéry, M., 2010. Introduction to WinBUGS for ecologists. Academic Press. 302 p.

McCarthy, M.A., 2007. Bayesian methods for ecology. Cambridge University Press. 296 p.

O'Quigley, J.O. 2008. Proportional hazards regression, Springer. 542 p.

OpenBUGS. 2012. OpenBUGS. Available online at http://www.openbugs.info/w/; last accessed 16 August 2012.

SAS Institute Inc. 2011. SAS/STAT(R) 9.3 user's guide. SAS Institute, Inc., Cary, NC.

Sturtz, S., Ligges, U., and Gelman, A. 2005. R2WinBUGS: a package for running WinBUGS from R. *Journal of Statistical Software* **12**(3): 1-16.

Thomas, A., O'Hara, B., Ligges, U., and Sturtz, S. 2006. Making BUGS open. *R News* **6**(1): 12-17.

ACKNOWLEDGEMENTS

Much was learned by the author from a session he attended by Fang Chen and Maura Stokes at the Northeast SAS® Users Group meeting in 2011: “Introduction to Bayesian Analysis Using SAS® Software”.

CONTACT INFORMATION

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

Matthew Russell, PhD
Dept. of Forest Resources, University of Minnesota
1530 Cleveland Ave. N.
St. Paul, MN 55108
Email: russellm@umn.edu