The Canadian Journal of Statistics
Vol. 29, No. 2, 2001, Pages 333–340
La revue canadienne de statistique

On cross-validation of Bayesian models

Fatemah ALQALLAF and Paul GUSTAFSON

Key words and phrases: Bayesian analysis; cross-validation; Markov chain Monte Carlo; predictive distribution.

MSC 2000: Primary 62F15; secondary 62M20.

Abstract: The authors examine several aspects of cross-validation for Bayesian models. In particular, they propose a computational scheme which does not require a separate posterior sample for each training sample.

De la validation croisée des modèles bayésiens

Résumé : Les auteurs abordent différents aspects de la validation croisée dans les modèles bayésiens. Ils proposent notamment une stratégie de calcul qui ne requiert pas l'obtention d'un échantillon de la loi a posteriori pour chaque échantillon d'apprentissage.

1. INTRODUCTION

Say that a set of data consists of n cases, each containing measurements of a response variable and some explanatory variables. The basic idea of cross-validation is to divide the cases at random into a *training sample* and a *validation sample*. Various models fit to the training sample can be used to predict the validation sample responses given the corresponding explanatory variables. Models and fitting techniques which yield better predictions in this scheme are then preferred.

An obvious problem with the basic cross-validation scheme is that the results may be sensitive to which particular partition of the data into training and validation cases is utilised. For this reason, various authors have suggested the use of *k-fold cross-validation*, which proceeds by randomly splitting the n cases into k segments which are as close to equal sized as possible. Consequently, there are k different ways to form a training sample comprised of $k-1$ segments and a validation sample comprised of the remaining segment. Cross-validation is applied to each of these k partitions, and the average predictive performance across the partitions is considered. The special case of n-fold cross-validation corresponds to the common "leave-one-out" scheme used in the jackknife and other statistical procedures.

In some circumstances, predictive assessments may even be sensitive to how the data are segmented for k-fold cross-validation. An alternative, referred to as *repeated data splitting* by Thall, Russell & Simon (1997), is to apply cross-validation to a very large number of randomly generated partitions of the data. In the context of a specific model selection algorithm, for instance, Thall, Russell and Simon demonstrated that repeated data splitting was preferred over k-fold cross-validation. Repeated data splitting has been investigated by several authors, including Burman (1989) and Shao (1993), amongst others.

While a pure Bayesian would undertake model assessment strictly on the basis of the posterior probabilities of competing Bayesian models, a pragmatic Bayesian might be interested in a cross-validatory comparison of these models. Moreover, cross-validation provides a level playing field for the comparison of Bayesian and non-Bayesian analyses of a particular data set. But cross-validation of Bayesian models using repeated data splitting can be very expensive computationally. In particular, fitting a model to just one training sample may require a considerable computational effort if a Markov Chain Monte Carlo (MCMC) algorithm is required for posterior sampling. Given this, the present article examines how cross-validation should be conducted in such contexts.

To be specific, Section 2 details the framework for cross-validation of Bayesian models based on the predictive distribution of the validation data given the training data. In particular, a reference distribution is introduced, so that cross-validatory summaries can be regarded as expectations with respect to this distribution. Given this foundation, Section 3 then describes the obvious estimators of such expectations using MCMC output. In implementing such an estimator, one must choose between fewer data partitions with a larger posterior sample for each training sample or more data partitions with a smaller posterior sample for each training sample. This tradeoff is examined in Section 4. Then Section 5 describes a different estimator which obviates the need for a separate posterior sample for each data partition. The estimators are compared in the examples of Sections 6 and 7, while some concluding thoughts appear in Section 8.

2. THE REFERENCE DISTRIBUTION

Say the observed data consist of responses $y^{\text{obs}} = (y_1^{\text{obs}}, \ldots, y_n^{\text{obs}})$, along with corresponding covariate vectors $x_1, \ldots, x_n$. Our cross-validatory assessments are based on expectations under a *reference distribution* for $(s, \theta, y^{\text{rep}})$, where the *split* s is a $0-1$ vector which divides the n cases into a training sample and a validation sample, the parameter vector θ describes the statistical model, and the *replicated response* y^{rep} is a hypothetical realization of the response vector. The *reference distribution* is most easily defined via the factorization

$$p(s, \theta, y^{\text{rep}}) \quad = \quad p(s)p(\theta \mid s)p(y^{\text{rep}} \mid \theta, s), \tag{1}$$

where each term on the right-hand side of (1) is described in turn.

We wish to average cross-validatory results across many data partitions. Specifically, this averaging is with respect to $p(s)$, the marginal distribution of s in (1). It seems sensible to fix the sizes of the training and validation samples at say n_T and n_V respectively, where of course $n_T + n_V = n$. In particular, $p(s)$ is taken to be the uniform distribution over such splits. That is, if the ones and zeros in s indicate training and validation cases respectively, then

$$p(s) \quad = \quad \begin{cases} \binom{n}{n_T}^{-1} & \text{if } \sum_{k=1}^{n} s_k = n_T, \\ 0 & \text{otherwise.} \end{cases}$$

Given the split s, then $p(\theta \mid s)$ in (1) is defined as the posterior distribution of θ given the training data only. We adopt the notation $T[s]$ and $V[s]$ to denote explicitly the training cases and the validation cases, respectively, and use $f(y \mid x, \theta)$ and $f(\theta)$ to denote the statistical model and prior. Thus a formal definition of $p(\theta \mid s)$ using the Bayes theorem is

$$p(\theta \mid s) \quad = \quad \{c(s)\}^{-1} \left\{ \prod_{i \in T[s]} f(y_i^{\text{obs}} \mid x_i, \theta) \right\} f(\theta), \tag{2}$$

where

$$c(s) \quad = \quad \int \left\{ \prod_{i \in T[s]} f(y_i^{\text{obs}} \mid x_i, \theta) \right\} f(\theta)\, d\theta$$

is the normalizing constant. Note that (1) does not explicitly acknowledge the dependence of the reference distribution on y^{obs} and x made apparent by (2).

Finally, cross-validation in the Bayesian context requires the predictive distribution of the replicated response vector y^{rep}. Typically, one is only interested in predicting the validation responses, but it is notationally simpler to define y^{rep} as a vector of length n to match y^{obs}. Given θ, then y^{rep} is simply distributed according to the model distribution. That is, $p(y^{\text{rep}} \mid \theta, s) = p(y^{\text{rep}} \mid \theta) = \prod_{i=1}^{n} f(y_i^{\text{rep}} \mid x_i, \theta)$. Thus under the reference distribution, $y_{V[s]}^{\text{rep}} \mid s$ is

distributed as the predictive distribution of the validation responses given the training data, for a particular data split.

Interest focusses on computing expectations with respect to the reference distribution, denoted generically as

$$\psi = \mathrm{E}\left\{r(s,\theta,y^{\mathrm{rep}})\right\}. \tag{3}$$

As a concrete example, if

$$r(s,\theta,y^{\mathrm{rep}}) = \left\|y^{\mathrm{rep}}_{V[s]} - y^{\mathrm{obs}}_{V[s]}\right\|^2 \tag{4}$$

then ψ is the average sum of squared prediction errors on the validation sample, where the average is over both the predictive distribution on the validation responses given the training data, and over different data splits. As a second example, if

$$r(s,\theta,y^{\mathrm{rep}}) = I\left\{T(y^{\mathrm{rep}}_{V[s]}) > T(y^{\mathrm{obs}}_{V[s]})\right\}$$

for some *checking function* T, then ψ is a cross-validated version of the Bayesian P-value introduced by Gelman, Meng & Stern (1996). While it is not pursued further here, a cross-validated version of the Bayesian P-value might be of considerable interest, since it obviates the criticism of using the same data twice. Draper (1996) considers such a cross-validation, but only for a single data split.

3. THE GOLD AND SILVER ESTIMATORS

The task at hand is to estimate ψ as given by (3). With very simple models and priors, $p(\theta \mid s)$ may have a closed form which permits analytic evaluation of $\mathrm{E}\left\{r(s,\theta,y^{\mathrm{rep}}) \mid s\right\}$. An obvious estimator of (3) in such instances is what we term the *gold estimator*,

$$\hat{\psi}_G = \frac{1}{I}\sum_{i=1}^{I} \mathrm{E}\left\{r(s,\theta,y^{\mathrm{rep}}) \mid s = s_i\right\},$$

where the split vectors $s_1,\ldots,s_I$ are simulated independently from $p(s)$.

When $p(\theta \mid s)$ does not have a closed form, MCMC methods may be used to simulate a dependent sample from $p(\theta \mid s)$. In this circumstance, the most straightforward way to estimate ψ is as follows. First, simulate independent split vectors $s_1,\ldots,s_I$ from $p(s)$. Then, for each s_i, use a MCMC run to draw a dependent sample $\theta_{i1},\ldots,\theta_{iJ}$ from $p(\theta \mid s = s_i)$. Finally, for each (i,j), simulate y^{rep}_{ij} from $p(y^{\mathrm{rep}} \mid \theta = \theta_{ij})$. Then each $(s_i,\theta_{ij},y^{\mathrm{rep}}_{ij})$ is distributed as the reference distribution (1), and so the *silver estimator*,

$$\hat{\psi}_S = \frac{1}{I}\sum_{i=1}^{I}\frac{1}{J}\sum_{j=1}^{J} r(s_i,\theta_{ij},y^{\mathrm{rep}}_{ij}), \tag{5}$$

is an unbiased estimator of ψ. Moreover, since (5) is an average of I independent and identically distributed terms, a simulation standard error for $\hat{\psi}_S$ is readily computed.

Two computational issues arise in the estimation of ψ by $\hat{\psi}_S$. First, the number of splits I and the size of the posterior sample for each split J must be specified. Given an overall constraint on computing resources, it is not clear *a priori* which is preferable: a smaller number of splits each with a large posterior sample based on the resultant training sample, or a larger number of splits each with a small posterior sample. Second, $\hat{\psi}_S$ is inherently expensive to compute, since each sampled split s requires a fresh MCMC run to sample from $p(\theta \mid s)$. It may then be valuable to develop an alternate estimate of ψ that is cheaper to compute. The next two sections focus on these two issues.

4. A FEW LARGE SAMPLES OR MANY SMALL SAMPLES?

To investigate the choice of I and J for the silver estimator (5), note that often the dependence in MCMC output roughly mimics an AR(1) process. Thus it might be reasonable to assume that

$$\operatorname{corr}\left\{r(s_i,\theta_{ij},y_{ij}^{\text{rep}}),r(s_i,\theta_{ik},y_{ik}^{\text{rep}})\mid s_i\right\} \;=\; \rho^{|j-k|},$$

for some correlation coefficient $\rho\in(0,1)$. Then the standard AR(1) approximation gives

$$\operatorname{var}\left\{\frac{1}{J}\sum_{j=1}^{J}r(s_i,\theta_{ij},y_{ij}^{\text{rep}})\mid s_i\right\} \;\approx\; \frac{1}{J}\operatorname{var}\left\{r(s,\theta,y^{\text{rep}})\mid s=s_i\right\}\left(\frac{1+\rho}{1-\rho}\right),$$

which in turn yields

$$\begin{aligned}\operatorname{var}(\hat{\psi}_S) &= \operatorname{var}\operatorname{E}(\hat{\psi}_S\mid s_1,\ldots,s_I)+\operatorname{E}\operatorname{var}(\hat{\psi}_S|s_1,\ldots,s_I) \qquad (6)\\ &\approx \frac{1}{I}\operatorname{var}\operatorname{E}\left\{r(s,\theta,y^{\text{rep}})\mid s\right\}+\frac{1}{IJ}\operatorname{E}\operatorname{var}\left\{r(s,\theta,y^{\text{rep}})\mid s\right\}\left(\frac{1+\rho}{1-\rho}\right).\end{aligned}$$

Ideally, we would choose the number of splits I and the posterior sample size J to minimize (6) subject to an appropriate constraint on the computation cost. Since MCMC algorithms require a burn-in period, the cost of Monte Carlo sampling behaves as

$$\text{COST} \;=\; I(J+B), \qquad (7)$$

where B is the number of burn-in iterations required. Temporarily forgetting that I and J are integers, straightforward calculus shows that the minimum of (6) subject to a fixed cost (7) is attained when

$$J \;=\; B^{1/2}\left(\frac{1+\rho}{1-\rho}\right)^{1/2}\left[\frac{\operatorname{E}\operatorname{var}\left\{r(s,\theta,y^{\text{rep}})\mid s\right\}}{\operatorname{var}\operatorname{E}\left\{r(s,\theta,y^{\text{rep}})\mid s\right\}}\right]^{1/2}. \qquad (8)$$

Note, in particular, that the optimal J scales as the square root of the number of burn-in iterations required. Often in practice B is on the order of 10^2 or 10^3, so (8) suggests that many splits with relatively small posterior samples is preferable. To use (8) more rigorously, one could carry out a small pilot simulation to obtain rough estimates of ρ, $\operatorname{E}\operatorname{var}\left\{r(s,\theta,y^{\text{rep}})\mid s\right\}$ and $\operatorname{var}\operatorname{E}\left\{r(s,\theta,y^{\text{rep}})\mid s\right\}$ before choosing I and J for the main simulation.

5. THE BRONZE ESTIMATOR

The silver estimator (5) requires as many MCMC runs as data splits, and each run requires B burn-in iterations before usable output is obtained. With a view to reducing the computational burden, we introduce the *bronze estimator* which requires only a handful of MCMC runs. If a simulation standard error is not required, then in fact this estimator can be based on a single MCMC run.

The posterior density for a given training sample can be expressed as $p(\theta\mid s)=\{c(s)\}^{-1}p^*(\theta\mid s)$, where

$$p^*(\theta\mid s) \;=\; \left\{\prod_{i\in T[s]}f(y_i^{\text{obs}}\mid x_i,\theta)\right\}f(\theta) \qquad (9)$$

is an unnormalized version of the posterior. We approximate (9) by a distribution which is based heuristically on the same amount of data but which does not depend on the specific split s. In particular, define

$$q^*(\theta)\propto\left\{\prod_{i=1}^{n}f(y_i^{\text{obs}}\mid x_i,\theta)\right\}^{\alpha}f(\theta), \qquad (10)$$

where $\alpha = n_T/n$. Also, let $q(\theta) = q^*(\theta)/\int q^*(\theta')\,d\theta'$ be the normalized version of (10). Raising the whole-data likelihood to the power α has the effect of flattening the posterior, to a degree commensurate with conditioning only on a fraction α of the data. Thus instead of drawing a MCMC sample from $p(\theta \mid s)$, which must be done afresh for every s, we draw a MCMC sample from $q(\theta)$, which is then reweighted using importance sampling to represent $p(\theta \mid s)$. This way, the same sample serves for every split s considered, offering a computational savings.

To be specific, say that $\theta_{h1}, \ldots, \theta_{hJ}$ is the hth of H independent MCMC samples simulated from $q(\theta)$. We expect H to be quite small, say $H = 5$. As well, we draw y_{hj}^{rep} from $p(y^{\text{rep}} \mid \theta = \theta_{hj})$, for $h = 1, \ldots, H$ and $j = 1, \ldots, J$. Now each of these H samples yields an importance-sampling estimate of $\mathrm{E}\left\{r(s, \theta, y^{\text{rep}}) \mid s = s_i\right\}$, namely

$$\hat{\psi}_{hi} = \frac{\sum_{j=1}^{J} r(s_i, \theta_{hj}, y_{hj}^{\text{rep}})\left\{p^*(\theta_{hj} \mid s_i)/q^*(\theta_{hj})\right\}}{\sum_{j=1}^{J}\left\{p^*(\theta_{hj} \mid s_i)/q^*(\theta_{hj})\right\}}, \tag{11}$$

where each weighting term $\left\{p^*(\theta_{hj} \mid s_i)/q^*(\theta_{hj})\right\}$ has the simple form

$$\log\left\{p^*(\theta_{hj} \mid s_i)/q^*(\theta_{hj})\right\} = \sum_{k=1}^{n}(s_{ik} - \alpha)\log f(y_k \mid x_k, \theta_{hj}).$$

The bronze estimator is defined as the average of $\hat{\psi}_{hi}$ across the I independent splits and the H independent samples from $q(\theta)$,

$$\hat{\psi}_B = \frac{1}{H}\sum_{h=1}^{H}\frac{1}{I}\sum_{i=1}^{I}\hat{\psi}_{hi}.$$

To determine a standard error for $\hat{\psi}_B$, view the terms $\hat{\psi}_{hi}$ as elements of an H by I matrix, and note that each element has the same distribution. Let c_0 denote the variance of this distribution, let c_1 be the common covariance of any pair of distinct elements from the same row, and let c_2 be the common covariance of any pair of distinct elements from the same column. Finally, note that any two elements from different rows and different columns are uncorrelated. Therefore,

$$\operatorname{var}(\hat{\psi}_B) = \frac{1}{H^2 I^2}\left\{(H)(I)c_0 + 2H\binom{I}{2}c_1 + 2I\binom{H}{2}c_2\right\},$$

which can be estimated as

$$\left\{\mathrm{SE}[\hat{\psi}_B]\right\}^2 = \frac{1}{H^2 I^2}\left\{\sum_{h=1}^{H}\sum_{i=1}^{I}(\hat{\psi}_{hi} - \hat{\psi}_B)^2 + 2\sum_{h=1}^{H}\sum_{i=1}^{I}\sum_{j=1}^{i-1}(\hat{\psi}_{hi} - \hat{\psi}_B)(\hat{\psi}_{hj} - \hat{\psi}_B) + 2\sum_{i=1}^{I}\sum_{h=1}^{H}\sum_{j=1}^{h-1}(\hat{\psi}_{hi} - \hat{\psi}_B)(\hat{\psi}_{ji} - \hat{\psi}_B)\right\}.$$

6. EXAMPLE: NORMAL LINEAR MODEL

We consider a data set from Weisberg (1995) comprised of body and brain weights for $n = 62$ mammals. Letting X and Y be the logarithms of the body weight and brain weight respectively, we entertain the linear model $Y \mid X \sim \mathsf{N}(\beta_0 + \beta_1 X, \sigma^2)$, in tandem with the noninformative prior $f(\beta_0, \beta_1, \sigma^2) \propto \sigma^{-2}$. We choose $n_T = n_V = 31$, so that the cross-validation is based on fitting the model to half of the cases and predicting the responses for the other half. Though it is possible to implement exact posterior sampling for this model, the Gibbs sampler is used for the sake of illustration. In particular, the Gibbs sampler iterates back and forth between updating σ

and updating (β_0, β_1). While interest focusses on the sum of squared prediction errors (4) as a summary of predictive performance, we can reduce Monte Carlo variability without changing ψ by using

$$r(s, \theta, y^{\text{rep}}) \;=\; \mathrm{E}\left\{ \left\| y^{\text{rep}}_{V[s]} - y^{\text{obs}}_{V[s]} \right\|^2 \mid s, \theta \right\} \;=\; n_V \sigma^2 + \left\| \beta_0 + \beta_1 x_{V[s]} - y^{\text{obs}}_{V[s]} \right\|^2. \quad (12)$$

We randomly sample $I = 200$ splits from $p(s)$ and then compute the gold, silver and bronze estimates of ψ with (12) as the function of interest. The silver estimate is computed using posterior samples of size $J = 25$ after $B = 100$ burn-in iterations for each split. The bronze estimate is computed with $H = 5$ samples of size $J = 150$ obtained after $B = 100$ burn-in iterations. With these choices, the total number of MCMC iterations required for the bronze estimate is only 5% of the number needed for the silver estimate.

We obtain gold, silver and bronze estimates (with simulation standard errors) of 33.10 (0.08), 33.21 (0.13), and 33.06 (0.10), respectively. Given that the bronze estimate requires less computation, it is surprising that it has a smaller simulation SE than the silver estimate. In fact, we redid the analysis nine further times, with newly sampled splits and MCMC samples each time. In each re-analysis the bronze SE is larger than the silver SE. Across the ten replicates, the average bronze SE is about 2.4 times larger than the average silver SE, which in turn is about 1.5 times larger than the average gold SE. However, the decreased precision of the bronze estimator is more than offset by the faster execution time. As implemented in *S-Plus*, the computing time for the bronze estimate is about one-tenth of that for the silver estimate.

7. EXAMPLE: COMPARING WEIBULL AND GAMMA REGRESSION MODELS FOR FAILURE-TIME DATA

As a second example, we consider Weibull and Gamma regression models for some failure-time data from Lawless (1982). The data consist of breakdown times for $n = 76$ specimens of an electrical insulating fluid. The specimens were split into seven groups, with each group receiving a different voltage stress. The model discussed by Lawless (1982) describes the lifetimes as Weibull distributed, with the scale parameter proportional to a power of the voltage. For some purposes, it is advantageous to describe this model in terms of the log failure-time. If Y and X denote the log failure-time and the log voltage, respectively, then

$$Y \;=\; \beta_0 + \beta_1 X + Z, \quad (13)$$

where Z has an extreme value distribution with location parameter 0 and scale parameter δ^{-1}. Thus $\theta = (\beta_0, \beta_1, \delta)$ is the vector of unknown parameters.

We consider a Gamma model as an alternative to the Weibull model. The Gamma model also follows (13), but now $\exp(Z)$ has a Gamma distribution with shape δ and scale δ^{-1}. In both models then, δ is the shape parameter describing the failure-time distribution. We assign the prior $f(\beta_0, \beta_1, \delta) = f(\beta_0, \beta_1) f(\delta)$, with (β_0, β_1) having a locally uniform prior, while $\log \delta \sim \mathsf{N}(0, 1)$. Note that the latter prior favours the exponential sub-model corresponding to $\delta = 1$ in both the Weibull and Gamma cases.

Again the data are split into equal sized samples for training and for validation, thus $n_T = n_V = 38$. Posterior sampling from both $p(\theta \mid s)$ and $q(\theta)$ is implemented via the Metropolis–Hastings algorithm with random walk proposals (Tierney 1994, Chib & Greenberg 1995). In particular, each component of θ is updated in turn by generating a candidate value, and then setting the new value to be either the candidate value or the current value depending on the appropriately computed acceptance probability. The candidates are generated according to

$$\beta_0^{(\text{cand})} \sim \mathsf{N}(\beta_0^{(\text{curr})}, 0.5^2), \quad \beta_1^{(\text{cand})} \sim \mathsf{N}(\beta_1^{(\text{curr})}, 4.0^2),$$

$$\log \delta^{(\text{cand})} \sim \mathsf{N}(\log \delta^{(\text{curr})}, 0.3^2),$$

which lead to reasonable acceptance rates in the vicinity of 40% to 60% for each parameter. Based on some preliminary runs, it seems safe to use $B = 250$ burn-in iterations.

Again, both silver and bronze estimates are based on averaging over the same $I = 200$ splits. The silver estimate uses posterior samples of size $J = 50$ after burn-in for each split, while the bronze estimate uses $H = 5$ samples of size $J = 350$ after burn-in. Thus the total number of MCMC iterations for the bronze estimate is again 5% of that for the silver estimate. The estimates and simulation standard errors appear in Table 1. It is clear that the Gamma model has better predictive performance than the Weibull model. The relative performance of the silver and bronze estimates is mixed. The silver estimate has a smaller simulation SE (than the bronze estimate) for the Weibull model but a larger simulation SE for the Gamma model. As implemented in *S-Plus*, the execution time for the bronze estimate is again better than that for the silver estimate, but only by about a factor of three in this example.

TABLE 1: Estimates of ψ in the failure-time example. Simulation standard errors appear in parentheses. Note that the standard error for the difference is based on pairing, as the same splits are considered for both the Weibull and the Gamma models.

	Weibull	Gamma	Difference
Silver	263.2 (2.5)	204.1 (1.3)	59.1 (2.7)
Bronze	258.6 (4.9)	200.4 (0.8)	58.2 (5.5)

8. DISCUSSION

This article has given a framework for Bayesian cross-validation through the introduction of the reference distribution. Moreover, we have provided two specific insights into the implementation of cross-validation via MCMC sampling. First, for the silver estimator, we have illustrated that many data splits with small posterior samples is typically preferred over fewer splits with larger posterior samples. Second, we have developed the bronze estimator which reduces the amount of Monte Carlo simulation required. While our examples are based on relatively small models, we hope that these insights will be useful to practitioners who desire cross-validatory assessments of complex Bayesian models fit using MCMC techniques.

The reference distribution formulation as per (1) suggests an alternative computational strategy. Given that we want to estimate expectations under this distribution, an attractive idea is to apply MCMC to (1) directly. In particular, this would involve a MCMC update to the split s itself, perhaps based on a proposal to switch a few randomly chosen observations from one sample (training or validation) to the other. Unfortunately it is not clear how to implement such an update, since $c(s)$, the normalization constant for $p(\theta \mid s)$ which appears in (1), is typically not available in closed form.

ACKNOWLEDGEMENTS

This work was supported by a research grant from the Natural Sciences and Engineering Research Council of Canada.

REFERENCES

P. Burman (1989). A comparative study of ordinary cross-validation, ν-fold cross-validation and the repeated learning-testing methods. *Biometrika*, 76, 503–514.

S. Chib & E. Greenberg (1995). Understanding the Metropolis–Hastings algorithm. *The American Statistician*, 49, 327–335.

D. Draper (1996). Comment on 'Posterior predictive assessment of model fitness via realized discrepancies'. *Statistica Sinica*, 6, 760–767.

A. Gelman, X. L. Meng & H. S. Stern (1996). Posterior predictive assessment of model fitness via realized discrepancies (with discussion). *Statistica Sinica*, 6, 733–807.

J. F. Lawless (1982). *Statistical Models and Methods for Lifetime Data.* Wiley, New York.

J. Shao (1993). Linear model selection by cross-validation. *Journal of the American Statistical Assocation*, 88, 486–494.

P. F. Thall, K. E. Russell & R. M. Simon (1997). Variable selection in regression via repeated data splitting. *Journal of Computational and Graphical Statistics*, 6, 416–434.

L. J. Tierney (1994). Markov chains for exploring posterior distributions. *The Annals of Statistics*, 22, 1701–1728.

S. Weisberg (1985). *Applied Linear Regression.* Second edition. Wiley, New York.

Received 13 July 2000
Accepted 19 February 2001

Fatemah ALQALLAF: fatemah@math.ubc.ca
Dept. of Mathematics, The University of British Columbia
Vancouver, British Columbia, Canada V6T 1Z2

Paul GUSTAFSON: gustaf@stat.ubc.ca
Dept. of Statistics, The University of British Columbia
Vancouver, British Columbia, Canada V6T 1Z2