Computational Aspects of Nonparametric Bayesian Analysis with Applications to the Modeling of Multiple Binary Sequences

Fernando A. Quintana* Michael A. Newton†

December, 1998

Abstract

We consider Markov mixture models for multiple longitudinal binary sequences. Prior uncertainty in the mixing distribution is characterized by a Dirichlet process centered on a matrix beta measure. We use this setting to evaluate and compare the performance of three competing algorithms which arise more generally in Dirichlet process mixture calculations: sequential imputations, Gibbs sampling, and a predictive recursion, for which an extension of the sequential calculations is introduced. This facilitates the estimation of quantities related to clustering structure which is not available in the original formulation. A numerical comparison is carried out in three examples. Our findings suggest that the sequential imputations method is most useful for relatively small problems, and that the predictive recursion can be an efficient

*Departamento de Probabilidad y Estadística, Facultad de Matemáticas, Pontificia Universidad Católica de Chile, Av. Vicuña Mackenna 4860, Casilla 306, Correo 22, Santiago, CHILE. Internet: quintana@riemann.mat.puc.cl. This manuscript is Facultad de Matemáticas Technical Report # PUC/PM-98/24.

†Department of Statistics, University of Wisconsin-Madison, 1210 West Dayton, Madison, WI 53706, USA. Internet: newton@stat.wisc.edu. This manuscript is UW Statistics Department Technical Report # 999.
preliminary tool for more reliable, but computationally intensive, Gibbs sampling implementations.

**Keywords:** Dirichlet Process, Gibbs sampling, Partial Exchangeability, Predictive Recursion, Sequential Imputations.

1 Introduction

Nonparametric hierarchical Bayesian analysis using the Dirichlet process has been the subject of much recent work. A critical advantage of this methodology compared to parametric Bayesian analysis is the ability to incorporate uncertainty at the level of distribution functions, thus allowing for infinite dimensional alternatives. However, this flexibility increases the computational complexity of the analysis. Gibbs sampling (Bush and MacEachern 1996), sequential imputations (MacEachern, Clyde and Liu in press) and predictive recursions (Newton and Zhang 1999, Newton, Quintana and Zhang 1998), are three algorithms for fitting nonparametric hierarchical Bayesian models. We pursue a comparative analysis of these methods.

To implement the comparison we focus on models for multiple binary sequence data. Essentially, we extend the work of Liu (1996) beyond exchangeable binary sequences to sequences with richer dependence structures. Three data sets facilitate a numerical study. We analyze at-bat sequences for 127 baseball players in the 1990 National League Baseball season, longitudinal data from a social science survey (a subsample of size 2,390 out of 12,000 young Americans), and incidence of pathogen infection in 278 dairy cows monitored regularly throughout a year.

Dirichlet-process posterior calculations involve a one-parameter distribution over partitions of experimental units, the Polya sequence, that may be described by a simple analogy. A restaurant has many large tables, positioned at random locations from a probability distribution $\alpha$. One by one, $n$ people enter the restaurant to be seated. The first person sits at the first table. Subsequently, for some weight parameter $c > 0$, the $i^{th}$ person either sits at the $i^{th}$ table, with probability $c/(c + i - 1)$, or
randomly selects a previously seated person and joins that dinner party; see also Arratia, Barbour and Tavaré (1992). The random sequence $\zeta_1, \zeta_2, \ldots, \zeta_n$ recording the – possibly vector valued – table locations of the $n$ seated people is a dependent, exchangeable sequence called a Polya sequence. The number of nonempty tables $N(z)$, where $z = (\zeta_1, \ldots, \zeta_n)$, equals the number of unique values among $\zeta_1, \ldots, \zeta_n$. In addition, the set of unique values $z^* = (\zeta^*_1, \ldots, \zeta^*_N(z))$ – the cluster locations – is a random sample from $\alpha$. Clearly, a random partition is induced by considering in the same element of the partition, i.e. the same cluster, those individuals seated at the same table. The parameter $c$ affects this distribution; for example $N(z)/\log(n) \xrightarrow{a.s.} c$ as $n \to \infty$ (Korwar and Hollander 1973). Thus small values of $c$ favor large clusters.

The connection to nonparametric hierarchical Bayesian analysis is rather interesting. One models data $y_1, y_2, \ldots, y_n$ from $n$ experimental units as conditionally independent given unobserved parameters $\zeta_1, \zeta_2, \ldots, \zeta_n$. These parameters are viewed as independent and identically distributed from an unknown mixing distribution $F$. Prior uncertainty in $F$ is characterized by a Dirichlet process centered on $\alpha$. Consequently, $\zeta_1, \ldots, \zeta_n$ is a Polya sequence, marginally over uncertainty on $F$ (Blackwell and MacQueen 1973), and the posterior distribution over $F$ is a mixture of Dirichlet processes. This model formulation has been considered by many authors in different contexts; in some cases a prior for $c$ and the parameters of $\alpha$ is also incorporated. We refer the reader to Mukhopadhyay and Gelfand (1997) and references therein, MacEachern and Müller (1998), Quintana (1998), and to the collection of papers presented in Dey, Müller and Sinha (1998). The partition structure induced on $z$ is naturally reflected in the observed data, in the sense that experimental units sharing a common parameter value in $z^*$ have a common distribution and thus reside in the same cluster.

Our interest concentrates on models for multiple correlated binary sequences. Thus, we consider each data point $y_i$ as a sequence of binary outcomes, a sequence forming a Markov chain of some known and fixed order (usually 0, 1 or 2) given transition parameters in $\zeta_i$, and thus marginally a partially exchangeable binary
sequence (Quintana and Newton 1998; Zaman 1984). As a consequence of the non-parametric Bayesian formulation, data points in the same cluster have a common transition matrix. Section 2 presents the model and develops the properties required for the subsequent calculations. These properties extend the results in Liu (1996) to partially exchangeable sequences.

We compare three algorithms for model fitting. Section 3 considers sequential imputations (SI). A Gibbs sampler (GS) implementation is discussed in Section 4. We develop an extended predictive recursion (PR) in Section 5. Using each algorithm we approximate three basic inference summaries in each numerical example: the posterior distribution of the number of clusters, the prior predictive probability as a function of c, and the posterior predictive distribution of a new transition matrix. Three data sets enable this comparison in Section 6, and Section 7 summarizes and discusses our main findings.

2 The model and some of its properties

2.1 The sampling model

Each of n experimental units provides a sequence of binary outcomes denoted by $y_i = (y_{i,1}, \ldots, y_{i,n_i})$, for $i = 1, 2, \ldots, n$. The complete observed data are $y = (y_1, \ldots, y_n)$. To model within-sequence correlation, we assume partial exchangeability. The joint distribution of a partially exchangeable sequence is invariant under permutations that leave unchanged the initial state and all $j \rightarrow k$ transition counts, for $j, k \in \{0, 1\}$. See, for instance, de Finetti (1938), Diaconis (1988), Diaconis and Freedman (1980a, 1980b, 1980c), and Zaman (1984). Due to the representation theorems by Freedman (1962) and Diaconis and Freedman (1980b), a recurrent, infinite, and partially exchangeable sequence corresponds to a mixture of Markov chains for a unique mixing measure $\mu$ defined on the Borel sets of $[0, 1]^2$. Thus, the partially exchangeable sequence $y_i$ can be obtained by choosing random effects $\zeta_i = (\zeta_{i,0}, \zeta_{i,1}) \sim \mu$, and then running a Markov chain with unobserved transition
matrix
\[
\begin{pmatrix}
\zeta_{0,0} & 1 - \zeta_{0,0} \\
\zeta_{1,1} & 1 - \zeta_{1,1}
\end{pmatrix}
\].

Transition counts will be denoted \( t^i = (t^i_{0,0}, t^i_{0,1}, t^i_{1,0}, t^i_{1,1}) \), where, for example, \( t^i_{0,1} \) denotes the number of times state 0 is followed by state 1 in sequence \( y_i \).

### 2.2 The prior

Following the development in Section 1, a Dirichlet process prior on the distribution \( F \) of random effects \( z = (\zeta_1, \ldots, \zeta_n) \) implies that this sequence of transition probabilities forms a Polyga sequence, with joint distribution

\[
P(z) = \prod_{i=1}^{n} \left\{ \frac{c \alpha(\zeta_i) + \sum_{j=1}^{i-1} \delta_{\zeta_j}(\zeta_i)}{c + i - 1} \right\}.
\]  

(1)

The centering probability measure \( \alpha \) is defined on \( \mathcal{B}([0,1]^2) \). In subsequent calculations, we chose \( \alpha \) to be the joint distribution of two independent random variables with marginal distributions \( \text{Beta}(a_0, b_0) \) and \( \text{Beta}(a_1, b_1) \), respectively, which we denote \( \text{Beta}_2(\lambda) \), where \( \lambda = (a_0, b_0, a_1, b_1) \in \mathbb{R}^4 \).

Some notation governing clustering is necessary. Let \( \mathcal{P} \) denote any partition of \( \{1, \ldots, n\} \) into \( |\mathcal{P}| \geq 1 \) nonempty exhaustive subsets \( E_1, \ldots, E_{|\mathcal{P}|} \), each of these having size \( e_i = |E_i| \) for \( i = 1, \ldots, |\mathcal{P}| \), and where we assume the elements in \( E_i = \{l_{i,1}, \ldots, l_{i,e_i}\} \) are sorted in ascending order. Note from the earlier discussion that each \( z \) can be represented as \((z^*, \mathcal{P})\). Moreover, this is a one-to-one correspondence, which can be expressed as \((z^*, s)\), where \( s = (s_1, \ldots, s_n) \) and \( s_i \) is the cluster where \( \zeta_i \) belongs. In particular, \( \zeta_i = \zeta^*_{s_i} \).

The following result due to Lo (1984) turns out to be fundamental: for \( \alpha \)-integrable real-valued functions \( g_1, \ldots, g_n \) on \([0,1]^2\) we have

\[
\int_{\mathbb{R}^2} \prod_{i=1}^{n} g_i(\zeta_i) \prod_{i=1}^{n} \left( c \alpha + \sum_{j=1}^{i-1} \delta_{\zeta_j} \right)(\zeta_i) = \sum_{\mathcal{P}} \phi(\mathcal{P}),
\]  

(2)
where the summation is over all partitions \( \mathcal{P} \), and
\[
\phi(\mathcal{P}) = c^{||\mathcal{P}||} \prod_{i=1}^{||\mathcal{P}||} \left\{ (e_i - 1)! \int_{\mathbb{R}^2} \prod_{t \in E_i} g_t(\zeta) \, d\alpha(\zeta) \right\}.
\] (3)

Next section develops the main properties of our model and thus extends results of Liu (1996) to the partially exchangeable case (order 1).

2.3 Properties of the model

The form of the centering probability measure \( \alpha \), together with the multiplicative structure of the right-hand side of (3) allow a straightforward evaluation of \( \phi(\mathcal{P}) \). For a given partition \( \mathcal{P} = \{ E_1, \ldots, E_{||\mathcal{P}||} \} \), denote the transition counts for the \( i^{th} \) cluster by \( t_{j_1,j_2}^{(i)} = \sum_{t \in E_i} t_{j_1,j_2}^{(i)} \), where \( j_1, j_2 \in \{0, 1\} \), and let \( t^{(i)} = (t^{(i)}_{0,0}, t^{(i)}_{0,1}, t^{(i)}_{1,0}, t^{(i)}_{1,1}) \). In addition, for \( u, v \in \mathbb{R}^2_+ \) we define
\[
B_{u,v} = \frac{B(u_1 + v_1, u_2 + v_2)}{B(u_1, u_2)} \times \frac{B(u_3 + v_3, u_4 + v_4)}{B(u_3, u_4)},
\]
where, for \( a, b > 0 \)
\[
B(a, b) = \int_0^1 x^{a-1}(1-x)^{b-1} \, dx = \frac{\Gamma(a)\Gamma(b)}{\Gamma(a+b)}.
\]

**Theorem 1** If the prior centering measure is \( c \text{Beta}_2(\lambda) \), then the predictive density of the data, assuming the first observation for each experimental unit to be fixed, is given by:
\[
P(y) = \frac{\sum_{m=1}^{n} c^m \mathcal{L}_m(y)}{\prod_{i=1}^{n} (c + i - 1)},
\] (4)

where
\[
\mathcal{L}_m(y) = \sum_{\mathcal{P} : ||\mathcal{P}|| = m} \left\{ \prod_{i=1}^{m} (e_i - 1)! B(\lambda(t^{(i)})) \right\}.
\] (5)
Proof: We have

\[ P(y|z) = \prod_{i=1}^{n} \frac{r_{i,0}^{y_i} (1 - \zeta_{i,0})^{y_i,1} \zeta_{i,1}^{y_i} (1 - \zeta_{i,1})^{y_i,1}}{s_{i,0}^{z_i} (1 - \zeta_{i,0})^{z_i,1} \zeta_{i,1}^{z_i} (1 - \zeta_{i,1})^{z_i,1}}. \]

Besides, \( P(y, z) = P(y|z) \times P(z) \), and the result then follows by applying (2) to the functions \( g_i(\zeta_i) = \frac{r_{i,0}^{y_i} (1 - \zeta_{i,0})^{y_i,1} \zeta_{i,1}^{y_i} (1 - \zeta_{i,1})^{y_i,1}}{s_{i,0}^{z_i} (1 - \zeta_{i,0})^{z_i,1} \zeta_{i,1}^{z_i} (1 - \zeta_{i,1})^{z_i,1}} \), since \( \mathcal{L}_m(y) = \sum_{P: |P| = m} \phi(P) \).

The next result is the key to establishing the clustering structure.

**Theorem 2** Under the assumptions of Theorem 1, the posterior distribution of \( N(z) \) is given by

\[ P\{N(z) = m|y\} = \frac{c^m \mathcal{L}_m(y)}{\sum_{j=1}^{n} c^j \mathcal{L}_j(y)} \quad (6) \]

Proof: It follows from (1) that

\[ P(y, z) = \prod_{i=1}^{n} \left\{ \frac{\zeta_{i,0}^{y_i} (1 - \zeta_{i,0})^{y_i,1} \zeta_{i,1}^{y_i} (1 - \zeta_{i,1})^{y_i,1}}{\zeta_{i,0}^{z_i} (1 - \zeta_{i,0})^{z_i,1} \zeta_{i,1}^{z_i} (1 - \zeta_{i,1})^{z_i,1}} \right\} \times \prod_{i=1}^{n} \left\{ \frac{c\alpha(\zeta_i) + \sum_{j=1}^{i-1} \delta(\zeta_j, \zeta_i)}{c + i - 1} \right\} \quad (7) \]

so that we can use (2) to integrate out \( z \) with the constraint that \( N(z) = m \), to get, after using Theorem 1, \( P(y, N(z) = m) \propto c^m \mathcal{L}_m(y) \), and the result follows.

The next result is useful for later developments, but it also provides a way of assessing how changes in the prior characteristic measure affect the posterior distribution.

**Theorem 3** If the continuous characteristic probability measure \( \alpha \) with weight \( c \) of the prior Dirichlet process is changed to \( \alpha_1 \), with weight \( c_1 \), then

\[ \frac{P_{c_1 \alpha_1}(y, z)}{P_{c \alpha}(y, z)} = \left\{ \prod_{i=0}^{n-1} \frac{c + i}{c_1 + i} \right\} \frac{c_1^N(z) \alpha_1(\zeta_1)^{N(z)} \cdots \alpha_1(\zeta_{N(z)})}{c^N(z) \alpha(\zeta_1)^{N(z)} \cdots \alpha(\zeta_{N(z)})} \quad (8) \]

In particular, if \( \alpha_1 = \alpha \) and \( q = \frac{c_1}{c} \) then (8) reduces to

\[ \frac{P_{c_1 \alpha_1}(y, z)}{P_{c \alpha}(y, z)} = q^{N(z)} \prod_{i=0}^{n-1} \left\{ \frac{c + i}{c_1 + i} \right\}. \]
\textbf{Proof:} The result follows directly from (7) and the earlier discussion. ■

Inference about the weight $c$ may be based on the marginal likelihood function

$$L(c) = P(y|c) = \frac{\sum_{m=1}^{n} c^{m} L_{m}(y)}{\prod_{i=1}^{n} (c + i - 1)}.$$  \hfill (9)

Liu (1996) shows that if $\hat{c}$ denotes the maximizer of (9), then the observed Fisher information of $d = \log(c)$ when evaluated at $\hat{d} = \log(\hat{c})$ is given by

$$I_{obs}(\hat{d}) = \text{Var}(N(z)) - \text{Var}(N(z)|y),$$ \hfill (10)

where, for this case,

$$\text{Var}(N(z)) = \sum_{i=1}^{n} \frac{\hat{c}(i-1)}{(\hat{c} + i - 1)^{2}}.$$

The same is valid for our partially exchangeable setting.

However, it is clear that $\hat{c}$ cannot be easily computed due to the presence of $L_{m}(y)$ in (9). Indeed, we would need to consider all partitions $P$ having exactly $m$ subsets, for every $m = 1, \ldots, n$, and then evaluate the corresponding expression for each of such partitions. This program is not feasible, unless $n$ is small. Likewise, computing the most likely number of clusters from its posterior distribution faces exactly the same problem. We discuss three alternative algorithms to approximate posterior summaries: sequential imputations, Gibbs sampling, and an extended predictive recursion.

3 Model Fitting via Sequential Imputations

The sequential imputations method is kind of importance sampling derived from a method due to Rubin (1987) and introduced by Kong, Liu and Wong (1994). The algorithm relies on a trick to construct an importance distribution on the space of random effects ($\zeta^*, s$). A limitation of sequential imputations, as implemented in Liu (1996), is that the importance weights can be highly skewed. This problem motivated
a second version of the algorithm that is based on collapsing the space within which
the algorithm operates to the set of possible cluster memberships, $S$, by eliminating
the locations via integration; see MacEachern, Clyde and Liu (in press). The main
advantage of this second version is a substantial reduction in variability of the im-
portance weights, thus yielding a more efficient algorithm. Quintana (1998) gives a
performance comparison of the two sequential imputation algorithms.

To carry out sequential imputations, we must specify all the necessary conditional
distributions involved in the algorithm. Following the reasoning in MacEachern,
Clyde and Liu (in press) and in Quintana (1998), let $x_{<i}$ denote the first $i - 1$
coordinates of an $n$-coordinate object $x$, for $i = 2, \ldots, n$. Now, observe that we
always have $s_1 = 1$. Let $\kappa_{<i} = \max\{s_1, \ldots, s_{i-1}\}$, that is, the number of clusters
up to experimental unit $i - 1$, where clusters are implicitly assumed to be assigned
consecutive numbers. Given $s_{<i}$, $s_i$ can assume any of the values $1, \ldots, \kappa_{<i} + 1$,
meaning that $\zeta_i$ can either join an already existing cluster, or start a new one. Let $m_{<i}^j$ represent the current size of cluster $j$, for $j = 1, \ldots, \kappa_{<i}$, and where for notational
convenience, we define $m_{<i}^{\kappa_{<i} + 1} = c$, the “size” of a future cluster. We note that up
to subject $i - 1$, the conditional distribution of the location for the current cluster $j$
given $s_{<i}$ and $y_{<i}$ is Beta$_2(\lambda_{<i}^j)$, where

$$\lambda_{<i}^j = \left\{ \begin{array}{ll}
\lambda + \sum_{i' < i | \kappa_{i'} = j} t_{i'} & \text{if } j = 1, \ldots, \kappa_{<i}, \\
\lambda & \text{if } j = \kappa_{<i} + 1 \end{array} \right. \quad (11)$$

This simple observation, a direct consequence of the Polya urn representation (1), is
crucial for stating the next result.

**Theorem 4** Under the assumptions of Theorem 1, the following statements are true:

(i) The conditional distribution of the membership for experimental unit $i$ given
all the data up to this point and previous memberships is given by

$$P(s_i = j | s_{<i}, y_{<i}, y_i) \propto \frac{B_{\lambda_{<i}^j}(t_i)}{c + i - 1} \times m_{<i}^j \quad \text{for } j = 1, \ldots, \kappa_{<i} + 1 \quad (12)$$
(ii) The probability required to update the importance weights is given by:

\[ \mathbb{P}(y_i|s_{<i}, y_{<i}) \propto \sum_{j=1}^{s_{<i}} \frac{m_{j,i} B_{X_{s_{<i}}}(t_j)}{c + i - 1} + \frac{c B_{X}(t_i)}{c + i - 1}. \] (13)

The proof of Theorem 4 is obtained by just following the calculations in MacEachern, Clyde and Liu (in press) or Quintana (1998), and is therefore omitted. The sequential imputations algorithm is now stated as follows: set \( s_1 = 1, w_1 = B_{X}(t^1) \), and repeat (a) and (b) below for \( i = 2, \ldots, n \):

(a) Draw \( s_i \) from (12).

(b) Compute \( w_i = \mathbb{P}(y_i|s_{<i}, y_{<i}) \) as in (13).

Upon completion of these steps, compute \( w = \prod_{i=1}^{n} w_i \), and repeat the whole procedure \( M \) times.

After running this algorithm we have \( M \) vectors \( s(1), \ldots, s(M) \) of cluster memberships with corresponding weights \( w(1), \ldots, w(M) \), where \( s(j) = (s_1(j), \ldots, s_n(j)) \), for \( j = 1, \ldots, M \). Put \( W = \sum_{j=1}^{M} w(j) \). Various approximations can be obtained from the algorithm output, for instance,

\[ \mathbb{P}(N(z) = r|y) \approx \frac{1}{W} \sum_{j=1}^{M} w(j) I_r \{ N(s(j)) \}, \] (14)

where \( I_r(x) \) is the indicator function that \( r = x, r = 1, 2, \ldots, n \), and \( N(s) \) is the number of clusters implied by the cluster membership vector \( s \). The Monte Carlo asymptotic variance of approximation (14) can be obtained from the following Lemma.

**Lemma 1** Let \( \theta = E_P(g(X)) \) be the parameter of interest. Let \( X_1, \ldots, X_M \) be an importance sample of size \( M \), with the trial distribution \( Q \), and with weights \( w(i) = \frac{dP}{dQ}(X_i) \) and \( W_M = \sum w(i) \). Denote

\[ \hat{\theta} = \sum_{i=1}^{M} g(X_i) w(i)/W_M \]
the estimate of $\theta$ from the importance sample. Then $\hat{\theta}$ is strongly consistent, and asymptotically normal, $\sqrt{M}(\hat{\theta} - \theta) \overset{D}{\to} N(0, \sigma^2)$, with asymptotic variance given by $\sigma^2 = E_Q((g(X) - \theta)^2 w^2)$, assumed to be finite. This variance can be consistently estimated by

$$M \sum_{i=1}^{M} \left\{ g(X_i) - \hat{\theta} \right\}^2 w_i^2 \over W_M^2,$$

(15)

and the effective sample size (ESS) for the importance sample can be computed as

$$\text{ESS} \approx \frac{M}{1 + \left( \frac{M}{W} \right)^2 \left( \frac{1}{M-1} \right) \sum_{j=1}^{M} \left( w(j) - \frac{W}{M} \right)^2}.$$

(16)

**Proof:** By delta method. See also Geweke (1989) and Kong, Liu and Wong (1994).

By Lemma 1, we can set $g(x) = I_{k}(x)$ for different values of $k$ to get a consistent estimate for the asymptotic variance of the estimated posterior distribution of $N(z)$.

In order to construct an approximation to (9), notice that if we run the sequential imputations with a given characteristic probability measure $\alpha_0$ with weight $c_0$, and want to change this weight to $c_1$, then by Theorem 3

$$\frac{P_{c_{1}\alpha_0}(y)}{P_{c_{0}\alpha_0}(y)} = \int \frac{P_{c_{1}\alpha_0}(y, z)}{P_{c_{0}\alpha_0}(y, z)} dz = \int \frac{P_{c_{1}\alpha_0}(y, z)}{P_{c_{0}\alpha_0}(y, z)} P_{c_{0}\alpha_0}(z | y) dz$$

$$= \sum_{r=1}^{n} P_{c_{0}\alpha_0}(N(z) = r | y) \left\{ \frac{c_1}{c_0} \right\}^r \left( \prod_{i=0}^{n-1} \frac{c_0 + i}{c_1 + i} \right).$$

For the situation at hand, a simple way to proceed consists of choosing the measure $\alpha_0$ to be a Beta-type distribution with a preselected weight $c_0$, e.g., $c_0 = 1$. Hence, for any weight $c$, the ratio of predictive densities (that is, the likelihood ratio against a fixed alternative) is

$$\frac{L(c)}{L(c_0)} = \sum_{r=1}^{n} P(N(z) = r | y) \left\{ \frac{c}{c_0} \right\}^r \left( \prod_{i=0}^{n-1} \frac{c_0 + i}{c + i} \right).$$

(17)
The right hand side in (17) can be approximated by

$$\left( \prod_{i=0}^{n-1} \frac{c_0 + i}{c + i} \right) \times \frac{1}{W} \sum_{j=1}^{M} w(j) \left\{ \frac{c}{c_0} \right\}^{N(j)}$$

so that the maximum likelihood estimator of $c$ can be found by maximizing the approximated log-likelihood function (Geyer and Thompson 1992, Liu 1996)

$$\hat{c}_M(c) = \log \left( \frac{1}{W} \sum_{j=1}^{M} w(j) \left\{ \frac{c}{c_0} \right\}^{N(j)} \right) - \sum_{i=0}^{n-1} \log (c + i). \quad (18)$$

Finally, the posterior predictive distribution for a new random transition matrix, $P(\zeta_{n+1}|y)$, which coincides with the posterior mean $E(F|y)$, can be approximated as proposed in MacEachern, Clyde and Liu (in press):

$$P(\zeta_{n+1}|y) \approx \frac{1}{W(c+n)} \sum_{j=1}^{M} w(j) \left( \sum_{r=1}^{N(j)+1} m_{\xi_n}^{r} \text{Beta}_2(\lambda_{\xi_n}(j); \zeta_{n+1}) \right), \quad (19)$$

where we recall that $(j)$ denotes importance sampling replicates.

4 Gibbs Sampling Implementation

In this section we describe a computational strategy based on the Gibbs sampling algorithm (Gelfand and Smith 1990). This specific version of the Gibbs sampler was presented in Bush and MacEachern (1996) in the context of randomized block designs, and is based on ideas from MacEachern (1994). A similar algorithm can be found in West, Müller and Escobar (1994). Additional details for the exchangeable model can be found in MacEachern, Clyde and Liu (in press).

The Gibbs sampler is implemented as follows. Starting with a given vector of random effects $z(0)$, at iteration $k$ we generate $z(k)$ in two steps:

(a) For $i = 1, \ldots, n$ generate $\zeta_i(k)$ from $\zeta_i|y, z_{-i}(k)$, where

$$z_{-i}(k) = (\zeta_1(k), \ldots, \zeta_{i-1}(k), \zeta_{i+1}(k-1), \ldots, \zeta_n(k-1)).$$
(b) After generating \( z(k) \) a new clustering structure \( s(k) \) is obtained, containing \( N(z(k)) \) different locations. The cluster locations are then generated from \( z^*|y, s(k) \). Finally, we put \( \zeta_i(k) = \zeta^*_i(k) \).

As discussed in Bush and MacEachern (1996) and MacEachern (1994), the Gibbs sampler without step (b) tends to generate clusters whose locations do not change for many iterations. This can result in very slow mixing over the posterior and thus to poor estimates of posterior quantities. Adding a move which modifies cluster locations can alleviate this problem.

Using the fact that the components of \( z \) are exchangeable we find that if \( z_{-i} \) denotes \( z \) with the \( i \)-th coordinate removed, then the conditional distributions for step (a) are

\[
\zeta_i|y, z_{-i} \sim cB_\lambda(t^i)\text{Beta}_2(\lambda + t^i) + \sum_{j \neq i} c_{s_{j,0}}^j c_{s_{j,1}}^j (1 - \zeta_j) \zeta_j (1 - \zeta_j) \delta_{\zeta_j}(\zeta_i).
\]

Finally, the locations are drawn from

\[
\zeta_i^* \sim \text{Beta}_2(\lambda + t^{(i)}), \quad i = 1, \ldots, N(z(k)).
\]

The various approximations discussed in Section 3 have parallel versions based on the Gibbs sampling algorithm. In fact, approximations (14) and (18) are obtained through replacing weighted averages by simple averages. For instance, the log-likelihood of \( c \) can be approximated by

\[
\hat{i}_M(c) = \log \left( \frac{1}{M} \sum_{j=1}^M \left\{ \frac{c}{c_0} \right\}^{N(z(j))} \right) - \sum_{i=0}^{n-1} \log (c + i),
\]

where \( M \) is the number of Gibbs iterations after discarding the customary burn-in period. Also, the posterior predictive density of a new transition matrix can be approximated as in (19), setting \( w(j) = 1/M \). Rao-Blackwellization of these quantities may be considered also.
5 Predictive Recursions

The predictive recursion proposed in Newton and Zhang (1999) is designed to approximate the Bayes estimate of $F_1$, or equivalently the posterior predictive distribution $F_{n+1}(B) \overset{\text{def}}{=} P(\zeta_{n+1} \in B | y)$, for measurable sets $B$ in the support of the prior guess $\alpha$. The exact equality

$$F_2(B) = P(\zeta_2 \in B | y_1) = \frac{c}{c+1} \alpha(B) + \frac{1}{c+1} P_0(\zeta_1 \in B | y_1),$$

motivates the recursive approximation:

$$F_i(B) = (1 - w_i) F_{i-1}(B) + w_i P_{i-1}(\zeta_i \in B | y_i),$$

where $P_{i-1}(\zeta_i \in B | y_i)$ is computed under $F_{i-1}(B)$, and with nominal weights $w_i = 1/(c + i)$. In terms of predictive densities,

$$f_i(\zeta) = (1 - w_i) f_{i-1}(\zeta) + w_i \left( \frac{P(y_i | \zeta) f_{i-1}(\zeta)}{\int P(y_i | x) f_{i-1}(x) dx} \right),$$

where $P(y_i | \zeta)$ is the Beta2($\lambda; \zeta$) density. A simple computational approach is to keep track of (23) on a finite grid. The resulting approximation depends on the order by which the data points are processed, but this dependence may be weak and can be nullified by averaging over a sample of orders, for example. The major benefit of this predictive recursion is its computational simplicity. Calculations are $O(n)$ and the cluster information is not part of the calculation. On the other hand, the quality of the approximation does not improve without increasing $n$.

A drawback of the predictive recursion is that information is lost concerning cluster structure, and so we do not obtain a direct approximation to $P(N(z) = k | y)$, for example. The following argument extends the recursive approximation to such posteriors. In analogy with the notation of Section 3, let $x_{\leq i}$ denote the first $i$ coordinates of an $n$-coordinate object $x$. Put $p_{i,k} = P(y_{\leq i}, N(z_{\leq i}) = k)$, where $1 \leq k \leq i \leq n$. Clearly, $P(N(z) = k | y) \propto p_{n,k}$. By the underlying Polya urn structure, it is immedi-
ately seen that for $i = 1, \ldots, n$ we have

$$p_{i,i} = \frac{c}{\prod_{j=1}^{i}(c+j-1)} \times (i-1)! B_{\lambda}(t^{i} + \cdots + t^{i}),$$

and

$$p_{i,i} = \frac{c^{i}}{\prod_{j=1}^{i}(c+j-1)} \times \prod_{j=1}^{i} B_{\lambda}(t^{i}).$$

In general, for $n \geq i > k \geq 1$ a simple calculation reveals that

$$P_{i,k} = P(y_{\leq i}, N(z_{\leq i}) = k - 1, N(z_{\leq i}) = k) + P(y_{\leq i}, N(z_{\leq i}) = k, N(z_{\leq i}) = k)$$

$$= P(y_{i}, N(z_{\leq i}) = k| y_{<i}, N(z_{<i}) = k - 1) P(y_{<i}, N(z_{<i}) = k - 1) +$$

$$P(y_{i}, N(z_{\leq i}) = k| y_{<i}, N(z_{<i}) = k) P(y_{<i}, N(z_{<i}) = k)$$

$$= P(y_{i}, N(z_{\leq i}) = k| y_{<i}, N(z_{<i}) = k - 1)p_{i-1,k-1} +$$

$$P(y_{i}, N(z_{\leq i}) = k| y_{<i}, N(z_{<i}) = k)p_{i-1,k},$$

where it is not hard to see that

$$P(y_{i}, N(z_{\leq i}) = k| y_{<i}, N(z_{<i}) = k - 1) = \frac{c}{c + i - 1} \times B_{\lambda}(t^{i}).$$

Now, when using nominal weights $w_{i} = 1/(c + i)$ in (23), it can be seen that

$$f_{i}(\zeta) = \frac{c}{c + i} \times \text{Beta}_{2}(\lambda; \zeta) + \frac{i}{c + i} \times h_{i}(\zeta),$$

(24)

where

$$h_{i}(\zeta) = \frac{f_{i}(\zeta) - c \text{Beta}_{2}(\lambda; \zeta)/(c + i)}{i/(c + i)}.$$

The importance of the decomposition in (24) is that the first term to the right-hand side can be seen as the density for a newly generated random effect $\zeta_{i+1}$, multiplied by the probability of forming a new cluster. Likewise, $h_{i}$ approximates the conditional density of $\zeta_{i+1}$, given that this will join one of the already existing clusters.

In general, for a given sequence of weights $\{w_{i}\}$, the earlier reasoning can be easily extended to get

$$f_{i}(\zeta) = \left(\prod_{j=1}^{i}(1 - w_{j})\right) \times \text{Beta}_{2}(\lambda; \zeta) + \left(1 - \prod_{j=1}^{i}(1 - w_{j})\right) \times h_{i}(\zeta),$$

(25)
so that \( h_i \) is now computed as

\[
h_i(\zeta) = \frac{f_i(\zeta) - \prod_{j=1}^{i} (1 - w_j) \times \text{Beta}(\lambda; \zeta)}{1 - \prod_{j=1}^{i} (1 - w_j)}.
\]

Thus, the proposed recursive approximation starts with \( p_{1,1} = B(\lambda, t^1) \), and for \( 2 \leq i \leq n \):

\[
p_{i,k} = \frac{c}{c + i - 1} \times B(\lambda, t^i) \times p_{i-1,k-1} + \frac{i - 1}{c + i - 1} \times \int P(y_i|\zeta)h_{i-1}(\zeta)d\zeta \times p_{i-1,k}.
\]

(26)

This scheme runs for \( k = 1, \ldots, k_{\text{max}} \), where \( k_{\text{max}} \) is a reasonable upper limit, e.g., \( k_{\text{max}} = 20 \), and where we set \( p_{i,0} = 0 \) and \( p_{i,k} = 0 \) for \( k > i \). In practice, (26) can be computed over a suitable grid on \([0,1] \times [0,1]\), which also facilitates evaluating the required integrals. At the end of a run, we can use the approximation

\[
\hat{P}(N(z) = k|y) \approx \frac{p_{n,k}}{\sum_{j=1}^{k_{\text{max}}} p_{n,j}}, \quad k = 1, \ldots, k_{\text{max}}.
\]

(27)

The estimates (27) are later averaged over a number \( M \) of random orderings of the experimental units. In addition, the form of the log-likelihood (17) suggests computing

\[
\hat{l}(c) = \log \left( \sum_{k=1}^{k_{\text{max}}} \hat{P}(N(z) = k|y) \left\{ \frac{c}{c_0} \right\}^k \right) - \sum_{j=0}^{n-1} \log(c + j)
\]

(28)

as an approximation to (17).

6 Computational Experience

We now apply the earlier theory and algorithms to three concrete examples, with special emphasis given to algorithm performance and computational aspects. The datasets are:
**D1:** sequences of hits and outs for 127 players in both American and National Baseball leagues in 1990 (Albright 1993). The binary response is defined as 1 if a hit, walk or sacrifice occurred at bat and 0 if an out occurred at bat.

**D2:** sequences of pathogen infection indicators for each of 278 cows, at several examination times. The binary response is 1 if some portion of the udder was infected at a given time, and 0 otherwise.

**D3:** sequences of employment status for 2,390 young people, a subset of the National Longitudinal Survey of Youth (NLSY; see Borus 1984), measured over a number of consecutive years. Each binary response is 1 if at the given year the person was employed and 0 if unemployed. The original subsample contained 2,500 sequences, of which 110 were entirely missing and hence removed from this application. We note here that 87 of the remaining sequences are of length 1.

For a more detailed description of each dataset, we refer the reader to Quintana and Newton (1998). We also consider a subset of D2, referred to as D2*, which consists of only the first three data points.

From a modeling viewpoint, in all three cases the sequence for each subject is assumed to be a Markov chain, conditional on a subject–specific transition matrix. In turn, these transition matrices are a random sample from $F$, and a priori, $F$ is a Dirichlet process, with characteristic measure $c \text{Beta}_2(1, 1, 1, 1)$, and weight $c > 0$. For comparison purposes, we also consider order 0 models.

The inference (and computational effort) will focus on the estimation of (i) posterior distribution of the number of clusters defined in (6); (ii) log-likelihood of the weight parameter $c$, derived from (9); and (iii) posterior predictive distribution of a new transition matrix, $P(\zeta_{n+1} | y)$.

For the sake of estimating $l(c)$, we chose $c_0 = 1$ when running all the algorithms. The value of $M$ was set to 20,000 for sequential imputations (SI) and also for Gibbs sampling (GS), with the exception of D3, for which, due to the large sample size, we used only $M = 10,000$, and $D2^*$, for which $M = 1,000$ suffices. The customary
burn-in period for GS was set to be of length 250. The number of random orderings 
considered for predictive recursions (PR) is \( M = 100 \), using the sequence of weights 
\( w_i = 1/\sqrt{(c+1)(c+i)} \). In addition, we used equally spaced \( 100 \times 100 \) grid on 
\([0,1] \times [0,1]\) in all cases – including SI and GS – for the purpose of evaluating the 
posterior predictive density.

6.1 Some General Considerations

In general terms, the three algorithms are rather straightforward to implement. In 
terms of execution time, the three algorithms take similar CPU times to produce all 
the estimates of interest, using identical value of \( M \), and with SI being the fastest 
and GS the slowest (data not shown). This makes PR an attractive choice, because it 
seems to require moderate values of \( M \) for good approximations. One other interesting 
point is that each run of PR requires a sequence of grid evaluations, from which all 
subsequent calculations are derived, independent of what we want to compute. In 
contrast, if the interest is not on the predictive distribution, both SI and GS are more 
efficient, in the sense of not requiring additional calculations, and in this case, SI is 
about 20% faster than GS.

6.2 Estimating \( P(N(z) = k|y) \) and \( l(c) \)

Each algorithm was run on each of the three datasets for both order 0 and 1 models. 
The ESS (16) for each run of SI are presented in Table 1. We can see a rapid decaying 
pattern. In particular, the estimation for dataset D3 is highly unreliable. By contrast, 
allocation to D2* gives results that are almost as good as sampling from the true 
posterior. This is consistent with earlier experience; see Quintana (1998). Thus, SI 
may be convenient choice for small to moderate sample sizes, but the importance 
weights can be very skewed for large values of \( n \).

Table 2 displays all the estimated posterior means and variances of the number of clusters. The estimates from SI and GS are very similar, but PR tends to under-
estimate the mean and overestimate the variance. The same trend is observed for
other sequences of weights, including $w_i = 1/i$, not reported here. This situation
is represented in Figures 1, 2, and 3, which show the estimated posterior distributions
for $N(z)$. For D1, SI and GS give very similar results, but for D2 and particularly
D3, discrepancies are evident, even though the estimates generally concentrate on
the same region. The fact that SI becomes less reliable as the sample size increases is
clearly shown in the standard errors on Figures 2 and 3. We point out that while the
standard error bars for GS and SI in these plots represent the Monte Carlo variability,
the situation is different for PR. Indeed, the bars reflect the inherent permutation
variability, but this might not bound the actual difference between estimates and
target quantities.

Now, excluding the case of D1, order 0, the PR estimates closely match the shape
found with GS, but have heavier tails, which explains the underestimated means
and overestimated variances mentioned earlier. Thus, PR seems to oversmooth the
clustering structure.

Comparing order 0 and 1 models, we see similar distributions for D1 and D3, but
not for D2. This suggests that the order of dependence is 0 for D1 and D3 and 1 for
D2, which is consistent with the results obtained in Quintana and Newton (1998),
Table 1, using goodness-of-fit tests.

Figures 4, 5, and 6 show the approximated log-likelihood functions of the weight
parameter $c$ for all cases considered. In general, the form of these functions is about
the same, but the estimated modes vary; see Table 3 which presents the maximizers
of these estimated log-likelihood, and the variances of $\hat{d}$, i.e., $I_{obs}(\hat{d})^{-1}$ from (10).
Comparing the values of $\hat{d}$ we see some discrepancies, particularly for D1 (order 0) and
D3 (order 1), mostly due to the differences among the estimated posterior distribution
of number of clusters. This is also why we see some apparently abnormally large values
of $I_{obs}(\hat{d})^{-1}$ from PR in D2 (order 0) and D3 (both orders).

Comparing the values of $l(\hat{c})$ from Table 3, we find that the values for the order 1
models do not always exceed those obtained for order 0. The reason of this apparent
contradiction is that while order 1 models treat the first observation from each chain as fixed – allowing proper computation of order 1 transition counts – order 0 models do not assume this and hence in a strict sense, they are not nested.

6.3 Estimating $P(\zeta_{n+1}|y)$

Newton, Quintana and Zhang (1998) applied PR and GS to D3 (order 1), finding a very close agreement among estimated posterior predictive distributions. The GS used in that work generated such estimates by merely computing the frequencies of $\zeta_i$'s falling into each cell of a $100 \times 100$ grid, considering all scans produced. Here we implement a better approximation, which consists of just using (19) for SI, and its GS counterpart. Figures 7 and 8 show the contour plots obtained for D1 and D2. For D1, GS and PR produce nearly the same posterior estimates, although for the latter algorithm we see smoother contour lines. When $c = 1.0$ and $c = 5.0$, SI yields shapes that are very similar to GS and PR. However, for $c = 0.1$ SI yields an estimated predictive distribution that is slightly less dispersed than the other two methods. For D2 we note the almost identical patterns obtained with SI and GS. PR yields a posterior distribution that is concentrated on nearly the same region, but it misses what the other two algorithms clearly show as a mode. The hint of a second mode is seen in the case of SI, but as already pointed out, this is unreliable due to the low ESS.

Figure 9 shows the results obtained for D2*, this time including the contours derived from the exact distribution. In this case, all the plots are almost identical, although PR misses a bit of the curvature near the boundaries of $[0, 1] \times [0, 1]$.

7 Discussion

We have studied three algorithms for fitting a nonparametric Bayesian model to multiple binary sequences that exhibit serial correlation. Exact posterior calculations are not feasible given the dimensions of the three datasets analyzed here.
Each algorithm has strengths and drawbacks, but some general guidelines can be suggested. First, SI seems to be most useful for problems with relatively moderate sample size such as in D1. This is corroborated by the exact results in Quintana (1998), in a rather different setting, and also in the analysis of D2*. As the sample size increases, skewed importance weights inflate Monte Carlo standard error to the point that the estimation is highly unreliable; e.g., D3 having more than 2000 data points.

GS is a general purpose algorithm that has been used successfully in nonparametric Bayesian problems, and may represent the gold standard. Our experience suggests that GS is a reliable method, but depending on the target inference, it can be significantly more CPU-consuming than other choices, especially for large problems like D3. It is important to note that in general, the other algorithms give very similar results but with much less CPU time.

Taking advantage of the form of (25) we have proposed a recursive approximation that takes into account the clustering structure, thus widening the scope of the predictive recursion method. The method is very simple to implement, and although the extended version does increase the CPU usage, the resulting extended PR is still a very cheap alternative. PR finds reasonable accurate approximations to the quantities of interest at much lower cost than SI and GS. However, our experience suggests that PR tends to oversmooth, resulting in that some features in the model might be missed. As a consequence, we obtain posterior distributions of number of clusters that have heavier left tails than what is found with the other methods. PR may be most useful as a preliminary tool for unveiling major features of posterior distributions.

References


<table>
<thead>
<tr>
<th>Data set</th>
<th>Model (order)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0</td>
</tr>
<tr>
<td>D1</td>
<td>250 (1.25%)</td>
</tr>
<tr>
<td>D2</td>
<td>22 (0.11%)</td>
</tr>
<tr>
<td>D3</td>
<td>4 (0.04%)</td>
</tr>
<tr>
<td>D2*</td>
<td>996 (99.6%)</td>
</tr>
</tbody>
</table>

Table 1: Effective sample sizes for all runs of sequential imputations. The numbers, expressed as percentages of $M$, are indicated between parentheses.

<table>
<thead>
<tr>
<th>Data set</th>
<th>Order 0 Model</th>
<th>Order 1 Model</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Mean</td>
<td>Variance</td>
</tr>
<tr>
<td>D1</td>
<td>SI</td>
<td>3.72 (0.060)</td>
</tr>
<tr>
<td></td>
<td>GS</td>
<td>3.69 (0.028)</td>
</tr>
<tr>
<td></td>
<td>PR</td>
<td>3.00 (0.130)</td>
</tr>
<tr>
<td>D2</td>
<td>SI</td>
<td>8.34 (0.115)</td>
</tr>
<tr>
<td></td>
<td>GS</td>
<td>8.49 (0.061)</td>
</tr>
<tr>
<td></td>
<td>PR</td>
<td>7.52 (0.198)</td>
</tr>
<tr>
<td>D3</td>
<td>SI</td>
<td>9.99 (0.256)</td>
</tr>
<tr>
<td></td>
<td>GS</td>
<td>9.65 (0.130)</td>
</tr>
<tr>
<td></td>
<td>PR</td>
<td>8.95 (0.292)</td>
</tr>
</tbody>
</table>

Table 2: Posterior mean and variance of $N(z)$ for each dataset, estimated from Sequential Imputations (SI), Gibbs Sampling (GS), and Predictive Recursion (PR). Monte Carlo standard errors are indicated between parentheses.
<table>
<thead>
<tr>
<th>Data</th>
<th>Method</th>
<th>Order 0 Model</th>
<th></th>
<th></th>
<th>Order 1 Model</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>$\hat{c}$</td>
<td>$l(\hat{c})$</td>
<td>$\hat{d}$</td>
<td>$I_{obs}(\hat{d})$</td>
<td>$\hat{c}$</td>
<td>$l(\hat{c})$</td>
</tr>
<tr>
<td>D1</td>
<td>SI</td>
<td>0.45</td>
<td>-490.95</td>
<td>-0.80</td>
<td>0.34</td>
<td>0.21</td>
<td>-489.65</td>
</tr>
<tr>
<td></td>
<td>GS</td>
<td>0.45</td>
<td>-490.94</td>
<td>-0.80</td>
<td>0.33</td>
<td>0.22</td>
<td>-489.75</td>
</tr>
<tr>
<td></td>
<td>PR</td>
<td>0.25</td>
<td>-490.18</td>
<td>-1.38</td>
<td>0.35</td>
<td>0.20</td>
<td>-489.54</td>
</tr>
<tr>
<td>D2</td>
<td>SI</td>
<td>2.59</td>
<td>-1288.99</td>
<td>0.95</td>
<td>0.78</td>
<td>0.29</td>
<td>-1289.09</td>
</tr>
<tr>
<td></td>
<td>GS</td>
<td>3.01</td>
<td>-1288.41</td>
<td>1.10</td>
<td>1.25</td>
<td>0.26</td>
<td>-1288.87</td>
</tr>
<tr>
<td></td>
<td>PR</td>
<td>3.43</td>
<td>-1288.95</td>
<td>1.23</td>
<td>7.80</td>
<td>0.24</td>
<td>-1288.81</td>
</tr>
<tr>
<td>D3</td>
<td>SI</td>
<td>1.60</td>
<td>-16206.28</td>
<td>0.47</td>
<td>0.49</td>
<td>1.15</td>
<td>-16206.71</td>
</tr>
<tr>
<td></td>
<td>GS</td>
<td>1.90</td>
<td>-16206.19</td>
<td>0.64</td>
<td>0.52</td>
<td>1.00</td>
<td>-16206.73</td>
</tr>
<tr>
<td></td>
<td>PR</td>
<td>1.96</td>
<td>-16206.44</td>
<td>0.67</td>
<td>5.27</td>
<td>0.59</td>
<td>-16206.66</td>
</tr>
</tbody>
</table>

Table 3: Maximum likelihood estimate of $c$ for each dataset, using Sequential Imputations (SI), Gibbs Sampling (GS), and Predictive Recursion (PR).
Figure 1: Estimated posterior distribution of number of clusters after $M = 20,000$ iterations of SI, $M = 20,000$ GS scans and $M = 100$ runs of PR for D1.
Figure 2: Estimated posterior distribution of number of clusters after $M = 20,000$ iterations of SI, $M = 20,000$ GS scans and $M = 100$ runs of PR for D2.
Figure 3: Estimated posterior distribution of number of clusters after $M = 10,000$ iterations of SI, $M = 10,000$ GS scans and $M = 100$ runs of PR for D3.
Figure 4: Estimated log-likelihood of $c$ after $M = 20,000$ iterations of SI, $M = 20,000$ GS scans and $M = 100$ runs of PR for D1. Plotted values correspond to $i_M(c) + 489.5$ versus $c$. 
Figure 5: Estimated log-likelihood of $c$ after $M = 20,000$ iterations of SI, $M = 20,000$ GS scans and $M = 100$ runs of PR for D2. Plotted values correspond to $\hat{i}_M(c)+1,288$ versus $c$. 
Figure 6: Estimated log-likelihood of $c$ after $M = 10,000$ iterations of SI, $M = 10,000$ GS scans and $M = 100$ runs of PR for D3. Plotted values correspond to $\hat{i}_M(c) + 16,206$ versus $c$. 
Figure 7: Contour plot of estimated posterior predictive distribution for D1 using all three methods. Contours represent probability content at the levels (0.5, 0.75, 0.85, 0.9, 0.925, 0.95, 0.975, 0.99).
Figure 8: Contour plot of estimated posterior predictive distribution for D2 using all three methods. Contours represent probability content at the levels (0.5, 0.75, 0.85, 0.9, 0.925, 0.95, 0.975, 0.99).
Figure 9: Contour plot of estimated posterior predictive distribution for D2\textsuperscript{+}, and including the exact values of P(z_4 | y_{\leq 3}). Contours represent probability content at the levels (0.75, 0.85, 0.9, 0.925, 0.95, 0.975, 0.99).