Journal of Statistical Software

October 2011, Volume 44, Issue 2. *http://www.jstatsoft.org/*

Markov Chain Monte Carlo Random Effects Modeling in Magnetic Resonance Image Processing Using the BRugs Interface to WinBUGS

Martin D. King
University College London

Fernando Calamante
Florey Neuroscience Institutes

Chris A. Clark
University College London

David G. Gadian
University College London

Abstract

A common feature of many magnetic resonance image (MRI) data processing methods is the voxel-by-voxel (a voxel is a volume element) manner in which the processing is performed. In general, however, MRI data are expected to exhibit some level of spatial correlation, rendering an independent-voxels treatment inefficient in its use of the data. Bayesian random effect models are expected to be more efficient owing to their information-borrowing behaviour.

To illustrate the Bayesian random effects approach, this paper outlines a Markov chain Monte Carlo (MCMC) analysis of a perfusion MRI dataset, implemented in R using the **BRugs** package. **BRugs** provides an interface to **WinBUGS** and its **GeoBUGS** add-on. **WinBUGS** is a widely used programme for performing MCMC analyses, with a focus on Bayesian random effect models. A simultaneous modeling of both voxels (restricted to a region of interest) and multiple subjects is demonstrated. Despite the low signal-to-noise ratio in the magnetic resonance signal intensity data, useful model signal intensity profiles are obtained. The merits of random effects modeling are discussed in comparison with the alternative approaches based on region-of-interest averaging and repeated independent voxels analysis.

This paper focuses on perfusion MRI for the purpose of illustration, the main proposition being that random effects modeling is expected to be beneficial in many other MRI applications in which the signal-to-noise ratio is a limiting factor.

Keywords: R, random effects, Markov chain Monte Carlo, **BRugs**, **WinBUGS**, **GeoBUGS**, magnetic resonance imaging.

1. Introduction

A common feature of many approaches adopted in the analysis of magnetic resonance (MR) image data is an independent voxels (a voxel is a volume element) treatment in which a given algorithm is applied in a voxel-by-voxel manner. In a recent publication we outlined an alternative approach based on Bayesian random effects modeling (King *et al.* 2009). The distinguishing feature of the latter approach is simultaneous, multiple-voxel processing based on some suitable spatial or non-spatial statistical distribution. Thus, each of the voxel-specific random effect parameters is specified as belonging to a population with an underlying statistical distribution. There are a number of consequencies, including the constraining effect of the model distribution that disallows extremely improbable parameter values.

Random effect models appear to have received relatively little attention in the MR literature, with the notable exception of functional magnetic resonance imaging (MRI) (Penny and Friston 2004), although Bayesian random effect models have occasionally been adopted in other MRI applications. The latter includes dynamic contrast enhanced (DCE) MRI (Schmid *et al.* 2006, 2009) and diffusion MRI in tumour radiology (Walker-Samuel *et al.* 2011). Nevertheless, the approach has been largely ignored, despite the fact that MRI parameter estimation based on some form of random effects model (spatial or otherwise) is expected to make better use of the available data, assuming that the ensemble of voxels is informative in relation to the parameter estimates sought for an individual voxel. This assumption is reasonable, since the converse implies that the voxels that make up an MR image are statistically independent.

1.1. Random effects versus complete averaging and unpooled analyses

Some MRI data analysts may be dubious about the validity of the modeling approach outlined in this paper, or the need for an alternative to an independent-voxel (i.e., separate voxels) analysis or complete region-of-interest (ROI) averaging (i.e., completely-pooled data analysis). The point we wish to stress at the outset is that random effects modeling is well established and has a huge statistics literature. Jones (1993) and Hand and Crowder (1996) both provide a useful introduction to random effect models, focussing on frequentist statistical methods, while Gelman *et al.* (2004) devote a chapter to Bayesian random effect models under the title *hierarchical models*. The latter includes numerous references to the Bayesian statistics literature in the form of a bibliographic note. Readers wishing to delve into the random effects literature might note that the terms *hierarchical model* and *multilevel model* are both widely used in the random effects context. A hierarchical model includes more than one level of random variation, as illustrated by the example given in this paper, in which random voxels are nested within random subjects. These models provide an optimum combination of the two extreme types of estimate, i.e, the completely pooled estimate at one extreme, and the estimates obtained from a set of separate analyses at the other extreme. In any situation in which there is a true, non-negligible underlying variation between the units under consideration (voxels and subjects in the present example), combined with a non-negligible measurement error, neither completely-pooled estimation nor the estimates obtained through a set of separate analyses are uncompromised. Sections 5.4 and 5.5 in Gelman *et al.* (2004) discuss this issue and provide a simple dataset that is used to compare the results obtained via a random effects treatment with the corresponding completely pooled result and an independent units analysis. They make a convincing case for hierarchical/random effects modeling.

1.2. Random effects modeling software

Numerous tools/packages are provided within the R programming environment (R Development Core Team 2011) for performing Bayesian random effects analyses, with a focus on Markov chain Monte Carlo (MCMC). A list can be obtained by searching for *Bayesian inference* at the Comprehensive R Archive Network (see, for example, the Bayesian task view maintained by Park 2011). The focus of this paper is the **BRugs** interface to **WinBUGS** and its **GeoBUGS** add-on (Thomas *et al.* 2006).

Before proceeding, a brief explanation of the term *mixed model* is required. This term is commonly encountered in various R help documents on random effects modeling and it is relevant to understanding the distinction between standard regression models and the model outlined in this paper. Mixed models include two kinds of explanatory variable (parameter), namely fixed effect and random effect variables, the latter of which is the focus of this paper. Unfortunately, the distinction between them can depend on the context, leading to possible confusion. For the purpose of this paper a random effect parameter is assigned to a unit under investigation (person, for example) that has been drawn from an underlying population of related/similar units. In contrast, fixed effect parameters apply when each level of the factor under consideration is of primary interest in its own right, with no reference to an underlying population. An example will clarify the difference. In an analysis involving MRI data acquired using two specific scanners, a scanner variable might be included in the model as a fixed effect to allow for differences between the two specific scanners used in the study. If, on the other hand, a multi-centre study was undertaken involving a relatively large number of scanners, perhaps with no particular interest in the individual scanners, then the scanner effect might be included in the regression model as a random effect variable. Crowder (1992) discusses this issue in relation to inter-laboratory measurements and their comparison. It might be noted that BLUPs (best linear unbiased predictors) and the concept of inference space are both important in this context (McLean *et al.* 1991). Random effect models can be formulated within both the frequentist and Bayesian statistical frameworks. Pinheiro and Bates (2000) provide a comprehensive account of mixed-effects modeling using the `lme` and `nlme` functions for linear mixed-effect and non-linear mixed-effect models, respectively, both of which are provided by the **nlme** package (Pinheiro *et al.* 2011). Similarly, the `lmer` function, which is provided by the **lme4** pakckage (Bates *et al.* 2011), can be used to perform linear mixed model, generalized linear mixed model or nonlinear mixed model analyses (Crawley 2007). These functions focus on frequentist/likelihood based mixed-model statistical inference. The seminal paper on linear random effect models is Laird and Ware (1982). The Laird-Ware paper was, however, preceded by a number of papers on Bayesian random effect models, including the notable paper by Lindley and Smith (1972). But it was not until the mid 1990s that applied statisticians started to take much interest in Bayesian random effect models, and the literature on this subject started to escalate. This was mainly attributable to the increase in computer power that had taken place in the preceding decade. Modern Bayesian analytical methods tend to be computationally demanding. Specifically, MCMC is a posterior simulation method that now dominates the applied Bayesian modeling field, and it is only in the last two decades that MCMC has become computationally feasible as a routine analytical tool.

In Section 5 we shall return briefly to some of the problems associated with frequentist statistical inference in mixed-model regression analysis. The important point is that a Bayesian analysis, in combination with MCMC, offers a number of advantages. As stated in the previous paragraph, prior to the development of contemporary MCMC algorithms and modern computing power, Bayesian analysis was not an option for the majority of practising statisticians. The problem was that a fundamental component of any Bayesian analysis is the computation of various posterior probability distributions. This invariably involves complicated high-dimensional integrals that have no analytical solution. Although numerical integration was adopted as a solution, this proved problematic in practice. MCMC provides a solution in which integration is performed by simulation, thus circumventing the analytical and/or numerical intractability of many applied Bayesian statistical problems. These issues are discussed at length in the texts by Gelman *et al.* (2004) and Carlin and Louis (2000), which provide an introduction to Bayesian analysis and its implementation using MCMC. The monograph edited by Gilks *et al.* (1996) provides a collection of articles on various aspects of MCMC practice, as it stood in the mid 1990s. Despite its age, the contents of this book remain relevant today.

Prominent among the available programs for performing Bayesian analyses using MCMC is **WinBUGS** and its various add-ons, including **GeoBUGS** for Bayesian spatial modeling and **PKBugs** for pharmacokinetic analysis. Its predecessor **BUGS** (Bayesian inference using Gibbs sampling) was launched in the early 90s, followed by the Microsoft Windows version, **WinBUGS**, in the mid 1990s (Lunn *et al.* 2009). **WinBUGS** provides a flexible tool for specifying a given Bayesian model (in terms of the likelihood and priors) and for generating samples from specified posterior distributions using the Gibbs algorithm. The Gibbs sampler is among the most widely used MCMC methods. According to Lunn *et al.* (2009) **WinBUGS** has in excess of 30,000 registered users, justifying our focus on this particular software. Carlin and Louis (2000, Appendix C) provides a software guide that includes a brief description of alternative software for mixed/random effects analysis, both Bayesian and non-Bayesian. We should emphasize that our focus on the **BRugs** (Thomas *et al.* 2006) interface to **WinBUGS** reflects little more than the fact that this is the package that we used in the work outlined in this paper. **R2WinBUGS** (Sturtz *et al.* 2005) provides a similar R interface to **WinBUGS**, while the **rjags** package (Plummer 2011) functions as an interface to **JAGS** (just another Gibbs sampler). In addition, a variety of other tools are available for performing random effects analysis, both Bayesian and non-Bayesian.

1.3. Gibbs sampling and WinBUGS

A detailed description of the manner in which Gibbs sampling is implemented within **WinBUGS** is beyond the scope of this paper. Nevertheless, a brief summary is warranted. Additional information is provided by Lunn *et al.* (2000, 2009). As stated above, Gibbs sampling is an MCMC algorithm. It is a special case of the Metropolis-Hastings algorithm based on a sampling of the so-called full conditional posterior distribution. Gelman *et al.* (2004) and Carlin and Louis (2000) both provide descriptions of these two algorithms. Among the reasons for the popularity of **WinBUGS** is that it frees the analyst from the burden of deriving the full conditional distributions and setting up the required sampling functions. Armed with the measurement model in the form of a likelihood and the associated priors, the rest of the problem is handled automatically within **WinBUGS**. The likelihood and priors can be provided in the form of programming statements or, alternatively, the model can be specified in terms of a graphical model (Lunn *et al.* 2009). In fact graphical models, or to be specific, so-called *directed acyclic graphical models* (DAGs), lie at the heart of **WinBUGS**. The important point is that DAGs provide a mechanism for factorizing a joint posterior probability and thus obtaining the set of full-conditional distributions required for Gibbs sampling. **WinBUGS** determines which of these full conditional distributions can be sampled directly, the remainder of which are sampled using an embedded Metropolis-Hastings or slice sampler.

Although the **WinBUGS** package is versatile in terms of the variety of models that it can handle, usually an analysis will involve a degree of post-simulation processing. At the very least, this will include some kind of convergence assessment, as required to guard against misleading inferences caused by poor sampling of the posterior parameter space (see Section 2). There is an obvious advantage to adopting an approach in which the sequence of steps involved in the analysis, namely, pre-simulation data processing (e.g., stripping data from MR image files and rearrangement to obtain the format required for input to **WinBUGS**), model compilation, MCMC simulation and post-processing are all contained within a single programme. The **WinBUGS** Gibbs sampler is provided within a GUI (graphical user interface) and, although menu-driven manual operations can be replaced by using **WinBUGS** batch-mode script commands, this does not have the flexibility required for performing a complete analysis with ease. Instead, this can be achieved within R by using one of the R-**WinBUGS** interfacing packages, as exemplified in this paper using **BRugs**. We provide a simple demonstration of the manner in which **BRugs** can be used as an R interface to **WinBUGS** using a dataset consisting of dynamic susceptibility contrast (DSC) signal intensity data acquired in a number of individuals. A Bayesian spatiotemporal random effects model is adopted in the analysis of the time-dependent DSC data. Specifically, we illustrate the MCMC random effects treatment by showing results obtained by simultaneously modeling the signal intensity data obtained from a 16-voxel (4-by-4) region of interest in five subjects, using the **BRugs** interface to **WinBUGS**.

To summarize, the purpose of this paper is two-fold. Firstly, we wish to alert the MRI research community to the advantages of Bayesian random effects modeling and to promote its more widespread use. We demonstrate the method using DSC signal intensity data, but maintain that the general approach is expected to be beneficial in many MRI applications. Secondly, the paper shows how a Bayesian random effects analysis can be performed in R, using the **BRugs** interface to **WinBUGS**.

1.4. Dynamic susceptibility contrast MRI

In this paper we use a DSC-MRI dataset to demonstrate a random effects analysis performed using **BRugs**. DSC-MRI is a perfusion estimation technique that involves the measurement of signal intensity following the rapid injection of an MR contrast agent (Calamante *et al.* 1999). These signal intensity data are used to calculate the time-dependent concentration of contrast agent in the tissue of interest. Perfusion parameter estimates are subsequently calculated by solving a system of equations derived from these tissue concentration data (Østergaard *et al.* 1996). DSC parameter estimation is challenging for several reasons. In particular, the technique tends to suffer from low signal-to-noise ratios, compounded by the ill-conditioned inverse calculation that arises in DSC data processing (Østergaard *et al.* 1996). Summary parameter estimation is an alternative empirical approach to DSC data quantification, one that avoids the 'inversion problem' (Calamante *et al.* 1999). Low signal-to-noise ratio remains a fundamental limitation, however, particularly in areas of low perfusion, irrespective of the type of perfusion parameter estimates that are used. DSC data processing is typical of the kind of MRI application in which random effects modeling might be used to advantage. For example, a random effects spatiotemporal model of the type outlined in this paper might be used in the first of a two-stage data processing procedure, involving a first-stage random effects smoothing across voxels within an ROI and, where applicable, across subjects. This would be followed by parameter estimation at the second stage, performed using inversion or some other method.

2. Materials and methods

2.1. Subjects and MRI data acquisition

The DSC-MRI data used in this paper were acquired from five children attending the Great Ormond Street Hospital for Children for a variety of clinical investigations. These children belonged to a mixed-gender cohort, aged between 6 and 23 months, all with no sign of a brain abnormality, as judged by visual assessment of their MRI scans. The signal intensity data were acquired on a 1.5T Siemens (Erlangen, Germany) Magnetom Vision system, using a spin-echo echo-planar imaging sequence (repetition time 1.25 or 1.5s; echo time 100ms, matrix size 128×128, field-of-view 250×250mm, 5mm slice thickness). After approximately 15 baseline acquisitions (1.5s sampling interval), gadolinium diethylenetriamine pentaacetic acid (Gd-DTPA) was administered by intravenous injection (0.15mmol/kg), followed by a saline flush. MRI data acquisition was continued, without interruption, before, during and after Gd-DTPA administration. A 16-voxel (4-by-4) region was selected from within the thalamus of each of the five subjects, positioned to achieve a high level of uniformity in the first baseline T2-weighted image, as shown in Figure 1 for one of the subjects.

2.2. Random effects model

A commonly used approach to DSC-MRI data processing is to use the so-called gamma-

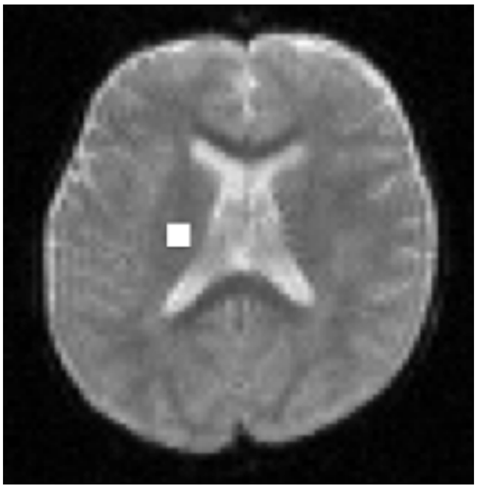


Figure 1: Baseline T2-weighted image (first time point) acquired in one of five subjects, showing the position of the 4-by-4 ROI within the thalamus. Signal intensity data obtained from selected voxels within this ROI are shown in Figure 3.

variate function (Calamante *et al.* 1999) as a model for contrast-agent concentration. This can be inaccurate in some situations (Calamante 2005), however, and we have adopted an alternative approach based on modeling the observed signal intensity and using a changepoint treatment (Carlin *et al.* 1992) of contrast-agent arrival, combined with an exponentially-damped polynomial (EDP; Crowder and Tredger 1981) signal response function. An EDP takes the general form $f(t) = \alpha + t^q \beta(t) e^{-\lambda t}$, $\beta(t) = \beta_0 + \beta_1 t + \ldots + \beta_p t^p$, $t \geq 0$, where q and p are integers. It can be considered a generalization of the gamma-variate function, the latter of which is given by the EDP with $\alpha = 0$ and $p = 0$.

The full random coefficients, EDP spatiotemporal changepoint model used in the present analysis took the form

$$\begin{aligned} y_{ij}(t_k) &\sim N(\mu_{ij}(t_k), \tau_\varepsilon) && (1)\\ \mu_{ij}(t_k) &= \alpha_{ij} + \beta_{0_{ij}},\ t_k < \kappa_{ij} && (2)\\ \mu_{ij}(t_k) &= \alpha_{ij} + (\beta_{0_{ij}} + \beta_{4_{ij}} t_{ijk}^{'4}) \exp\left(-\lambda_{ij} t_{ijk}^{'}\right),\ t_k \geq \kappa_{ij} && (3)\\ t_{ijk}^{'} &= t_k - \kappa_{ij} && (4) \end{aligned}$$

where $\sim$ indicates *distributed as*, $N(\mu, \tau)$ is the normal distribution with mean μ and precision τ (precision = 1/variance), $y_{ij}(t_k)$ is the observed DSC signal at the kth measurement occasion within the jth voxel within the ith subject, t_k is time at the kth measurement occasion, and κ_{ij} is the changepoint (contrast-agent arrival time) in the jth voxel of the ith subject. This model for signal intensity was incorporated into a Bayesian random effects structure in which each EDP parameter was treated as a sum of subject-specific ($\theta_i^{(\mathrm{s})}$) and voxel-specific ($\theta_{ij}^{(\mathrm{v})}$) terms. Thus $\theta_{ij} = \theta_i^{(\mathrm{s})} + \theta_{ij}^{(\mathrm{v})}$, where θ_{ij} is a given EDP model parameter. Subjects were treated as exchangeable (i.e., invariant to permutations of the subject labels), while voxels were modelled as spatially correlated. With the exception of $\beta_{0i}^{(\mathrm{s})}$, each of the subject-level random effect terms was assigned a hierarchical distribution

$$\begin{aligned} \theta_i^{(\mathrm{s})} &\sim N(\bar{\theta}_i^{(\mathrm{s})}, \tau_\theta^{(\mathrm{s})}) && (5)\\ \bar{\theta}_i^{(\mathrm{s})} &\sim N(0, 10^{-4}) && (6)\\ \tau_\theta^{(\mathrm{s})} &= 1/\sigma_\theta^2 && (7)\\ \sigma_\theta &\sim U(0, L) && (8) \end{aligned}$$

where L is some suitable large number. $\beta_{0i}^{(\mathrm{s})}$ was assigned a truncated normal distribution to constrain the asymptotic signal intensity to be no greater than the mean baseline level, noting that T1-induced signal enhancement is not expected in the absence of blood-brain barrier breakdown given the acquisition parameters that were used in this study (Calamante *et al.* 2007). The voxel-level random effect terms were assigned a spatial CAR (conditional autoregressive) prior (Wakefield *et al.* 2000). This takes the form

$$\begin{aligned} \eta_{ij} &\sim N(\bar{\eta}_{ij}, \sigma_\eta^2 / r_{ij}) && (9)\\ \bar{\eta}_{ij} &= \sum_{m \in \mathcal{N}_{ij}} \eta_m / r_{ij} && (10) \end{aligned}$$

where, to simplify the notation, η_{ij} is used to represent a given spatial random coefficient (i.e., $\eta_{ij} \equiv \theta_{ij}^{(\mathrm{v})}$), and σ_η^2 is an overall scale parameter, specified in terms of its reciprocal, i.e., precision, the latter of which was assigned a Gamma(0.5, 0.0005) prior. The notation $m \in \mathcal{N}_{ij}$ denotes the set of voxel labels belonging to those voxels in the immediate neighbourhood of the jth voxel in the ith subject. Voxel m belongs to that set, and r_{ij} is the number of voxels in the set. In the present analysis this set of neighbours included all voxels with at least one corner in contact with the voxel under consideration, i.e., all eight surrounding voxels, except at the boundary of the ROI, where the number of adjacent voxels is three and five for the corner and edge voxels, respectively. Thus, $\bar{\eta}_{ij}$ is the mean of η_m in voxels neighbouring voxel j in subject i. The measurement precision, τ_ε, was assigned an uninformative gamma prior. The specific form of the model used in this analysis was based on results obtained from an examination of various competing models, but it should not be regarded as definitive. For example, with some datasets the model may require refinement to capture properly the signal intensity behaviour at the changepoint. We emphasize that a more comprehensive model assessment study may yield an improved expression for the signal intensity, although it is expected that some form of changepoint treatment will prove indispensable. Similarly, the precise form of the various priors are not definitive. In part, these are expected to depend on the application and data under consideration. Some recommendations and other information regarding the specification of non-informative priors in the random effects context are given in Section 9.2 of the *Classic* ***BUGS*** manual (Spiegelhalter *et al.* 1995), while a more comprehensive discussion is provided by Gelman (2006).

Gibbs sampling was performed using the **BRugs** interface to **WinBUGS** in conjunction with the **GeoBUGS** `car.normal` distribution and associated spatial modeling tools. Three parallel chains were generated, each consisting of 5000 samples (after thinning at run time from 50,000 samples). This was followed by a post-simulation thinning to 1000 samples per chain.

2.3. Convergence assessment

MCMC convergence was assessed for a number of key parameters, focusing on various derived parameters of particular interest. Overlaid-chain trace plots were generated and inspected for visual signs of convergence failure. All were entirely satisfactory. This was followed by a semi-formal analysis performed using three convergence test procedures, namely the Gelman-Rubin shrink factor diagnostic and associated shrink factor plots, the Geweke Z-score diagnostic and Z-score plots, and the Raftery-Lewis diagnostic procedure (Cowles and Carlin 1996). These analyses were performed using the **coda** (convergence diagnosis and output analysis) package (Plummer *et al.* 2006). All test results were satisfactory, confirming the impression given by the overlaid trace plots. The Raftery-Lewis calculations indicated that after combining the three parallel chains, the 0.025 quantile estimates obtained for various statistics of interest had an accuracy of at least ± 0.0057 with probability 0.95, giving nominal 95% credible intervals with a true coverage of between 93.9% and 96.1%, with probability 0.95.

3. Example DSC-MRI WinBUGS analysis

The spatiotemporal statistical model was specified in the form of **WinBUGS** programming statements, which were saved in an ASCII text file (filename `DscMriModel.txt`). The programme, which is based on the spatiotemporal modeling code given in Lawson *et al.* (2003), was as follows:

```
model {
  for (i in 1:S) {
    alphaS[i] ~ dnorm(alpha.c, alpha.tau)
    beta4S[i] ~ dnorm(beta4.c, beta4.tau)
    beta0S[i] ~ dnorm(beta0.c, beta0.tau) I(0,)
    deltaS[i] ~ dnorm(delta.c, delta.tau)
    t_shiftS[i] ~ dnorm(t_shift.c, t_shift.tau)
    for (j in 1:M) {
      for (k in 1:T) {
        time[i,j,k] <- k
        tprime[i,j,k] <- time[i,j,k] - (t_shiftV[i,j] + t_shiftS[i])
        y[i,j,k] ~ dnorm(mu[i,j,k],tau.c)
        mu[i,j,k] <- step(-tprime[i,j,k]) * (alphaV[i,j] + alphaS[i] +
        beta0V[i,j] + beta0S[i]) + step(tprime[i,j,k]) * (alphaV[i,j] +
        alphaS[i] + (beta0V[i,j] + beta0S[i] + (beta4V[i,j] +
        beta4S[i]) * pow(tprime[i,j,k],4)) * exp(- (deltaV[i,j] +
        deltaS[i]) * tprime[i,j,k]))
      }
    }
  }

  for (i in 1:S) {
    alphaV[i,1:M] ~ car.normal(adj[], weights[], num[], tau.alpha[i])
    beta4V[i,1:M] ~ car.normal(adj[], weights[], num[], tau.beta4[i])
    beta0V[i,1:M] ~ car.normal(adj[], weights[], num[], tau.beta0[i])
    deltaV[i,1:M] ~ car.normal(adj[], weights[], num[], tau.delta[i])
    t_shiftV[i,1:M] ~ car.normal(adj[], weights[], num[], tau.t_shift[i])
  }

  for (k in 1:sumNumNeigh) {
    weights[k] <- 1
  }

  t_shift.c ~ dunif(21.0, 24.0)
  alpha.c ~ dflat()
  beta0.c ~ dnorm(0, 1.0e-04) I(0,)
  beta4.c ~ dflat()
  delta.c ~ dflat()

  alpha.sigma ~ dunif(0, 100)
  alpha.tau <- 1 / (alpha.sigma * alpha.sigma)

  beta0.sigma ~ dunif(0, 100)
  beta0.tau <- 1 / (beta0.sigma * beta0.sigma)
  beta4.sigma ~ dunif(0, 100)
  beta4.tau <- 1 / (beta4.sigma * beta4.sigma)

  delta.sigma ~ dunif(0, 100)
  delta.tau <- 1 / (delta.sigma * delta.sigma)

  t_shift.sigma ~ dunif(0, 100)
  t_shift.tau <- 1 / (t_shift.sigma * t_shift.sigma)

  tau.c ~ dgamma(0.001, 0.001)
  for (i in 1:S) {
    tau.alpha[i] ~ dgamma(0.5, 0.0005)
    tau.beta0[i] ~ dgamma(0.5, 0.0005)
    tau.beta4[i] ~ dgamma(0.5, 0.0005)
    tau.delta[i] ~ dgamma(0.5, 0.0005)
    tau.t_shift[i] ~ dgamma(0.5, 0.0005)
  }
}
```

The following four points might be noted. Firstly, in **BUGS/WinBUGS** syntax the tilde symbol `~` is used to define stochastic nodes, while the assignment symbol pair `<-` is used to define deterministic nodes (Spiegelhalter *et al.* 2003). Secondly, the **WinBUGS** language is somewhat limited in terms of its conditional constructs (it has no `if` statement, for example), and the `step` function is used as a substitute. Thirdly, the `I()` construction is used in the preceding code as a mechanism for obtaining a truncated distribution. This is common practice, and is satisfactory in the present application if the aim is spatiotemporal smoothing with a focus on voxel-specific Gd-DTPA response profiles, as opposed to statistical inference. It should be noted, however, that the proper purpose of the `I()` construct is for modeling censored data and that it can lead to invalid inferences (Lunn *et al.* 2009) if the `I()` construct is used in the specification of truncated prior distributions involving unknown parameters, and the focus is on mean behaviour. Fourthly, the exchangeable random effect precision parameter priors are specified in terms of their root variances, each of which has been assigned a uniform distribution. The upper limit of these distributions might be modified, based on the precision parameter posterior distributions obtained in preliminary analyses.

The Gibbs sampler requires starting values for each chain, and these are provided in an `inits` file (the file `DscMriInits.txt` used in the example R session contained the final state of the sampler in a preceding simulation). The initialization values are supplied as a list of the form:

```
list(alpha.c = 400, beta4.c = -300, beta0.c = 11, delta.c = 2.5, ...)
```

The list does not need to include values for every parameter in the model, since any remaining model parameters can be initialized using the `modelGenInits` function (see the information on `gen inits` in the *Model Menu* section of the **WinBUGS** 1.4 manual (Spiegelhalter *et al.* 2003)). This is especially useful in random effects modeling applications involving a large number of random effect parameters. Finally, a data file (`DscMriData.txt`) containing the observed signal intensities, and other information, is required. The **WinBUGS** documentation (Spiegelhalter *et al.* 2003) provides details regarding the data formats accepted by **WinBUGS**.

The preceding **WinBUGS** code is reproduced in the supplemental material to this article, together with an `inits` file and the data file, the latter of which includes the signal intensity data, which were acquired with a sampling interval of 1.5 seconds.

The following very basic R session provides a rudimentary sequence of function calls that might be used to perform an analysis. It serves to illustrate the most basic use of some key **BRugs** functions. Although largely self-explanatory, a brief description of the sequence of statements is given below.

```
R> library("BRugs")
R> modelCheck("DscMriModel.txt")
R> modelData("DscMriData.txt")
R> modelCompile(numChains = 1)
R> modelInits("DscMriInits.txt")
R> modelUpdate(5000)
R> samplesSetThin(10)
R> samplesSet(c("alpha.c", "alphaS", "alphaV", "beta0.c", "beta0S",
+    "beta0V", "beta4.c", "beta4S", "beta4V", "delta.c", "deltaS", "deltaV",
+    "t_shift.c", "t_shiftS", "t_shiftV"))
R> modelUpdate(50000)
R> alpha.c.sim <- samplesSample("alpha.c")
R> parms <- samplesMonitors("alphaS")
R> alphaS.sim <- sapply(parms, samplesSample)
...
R> samplesHistory("*", mfrow = c(2, 2))
R> samplesDensity("*")
```

This R session performs the basic steps required to implement an analysis. It checks the model for syntax errors, loads the data and performs the compilation, noting that the model is compiled in the context of the data that have been loaded. Thus, any incompatibility between the model statements and data will be detected at this stage. Although this is unconventional, it is standard in **BUGS**/**WinBUGS**. Assuming the model/data compile successfully, the initial values are loaded, and/or generated, as required. In the example, 5000 burn-in iterations are performed, which starts with an initial adaptive phase as required to tune those samplers (Metropolis and slice-samplers) that are used when direct sampling is not possible. After setting the thinning level the `samplesSet` function is used to specify which of the model parameter chains should be stored. Thinning is standard practice given a high degree of autocorrelation in the parameter chains, thus saving storage and reducing the CPU demands of any subsequent processing work. In the example the thinning parameter is set to 10, with the result that 1 in 10 of the MCMC samples is returned to R. Similarly, only 1 in 10 samples is used in subsequent statistical summary calculations performed via **BRugs** functions. Finally, a second call to `modelUpdate` restarts the simulation, which generates the specified number of samples. After the simulation has completed, control is passed back to R. At this stage, the MCMC output is not directly available to R. Three distinct approaches can be adopted to using the MCMC output, namely, (1) by making calls to various **BRugs** functions, thus generating various summary statistics, for example, (2) by creating a special list (with class `mcmc.list`) which can be used as input to the **coda** package for subsequent analysis and (3) by using the `samplesSample` function to create R arrays from the stored values. These arrays are directly available for post-MCMC processing in R. In the first of these approaches the MCMC output remains invisible to R. The preceding example R session illustrates the third approach.

3.1. Method 1: Working with BRugs plotting and summary functions

The function samplesSet is used to set one or more nodes for recording during subsequent chain updates. A number of functions are provided for generating statistics and plots from the resulting stored values. For example, the call to samplesStats in the sequence of statements

```
R> samplesSet(c("alpha.c", "alphaS", "alphaV", "beta0.c", "beta0S", "beta0V",
+    "beta4.c", "beta4S", "beta4V", "delta.c", "deltaS", "deltaV",
+    "t_shift.c", "t_shiftS", "t_shiftV"))
R> modelUpdate(50000)
R> samplesStats(c("alpha.c", "beta0.c", "beta4.c", "delta.c", "t_shift.c"))
```

lists the mean, median and other quantiles for each specified node (Figure 2). Trace plots (plots of sample value versus iteration number) and smoothed kernel density plots can be generated using the plotHistory and plotDensity functions, respectively, as illustrated by the following statements:

```
R> plotHistory("beta0.c")
R> plotDensity("beta0S[1]")
```

The pair of functions summarySet and summaryStats provide an alternative mechanism for generating statistics. Thus, for example,

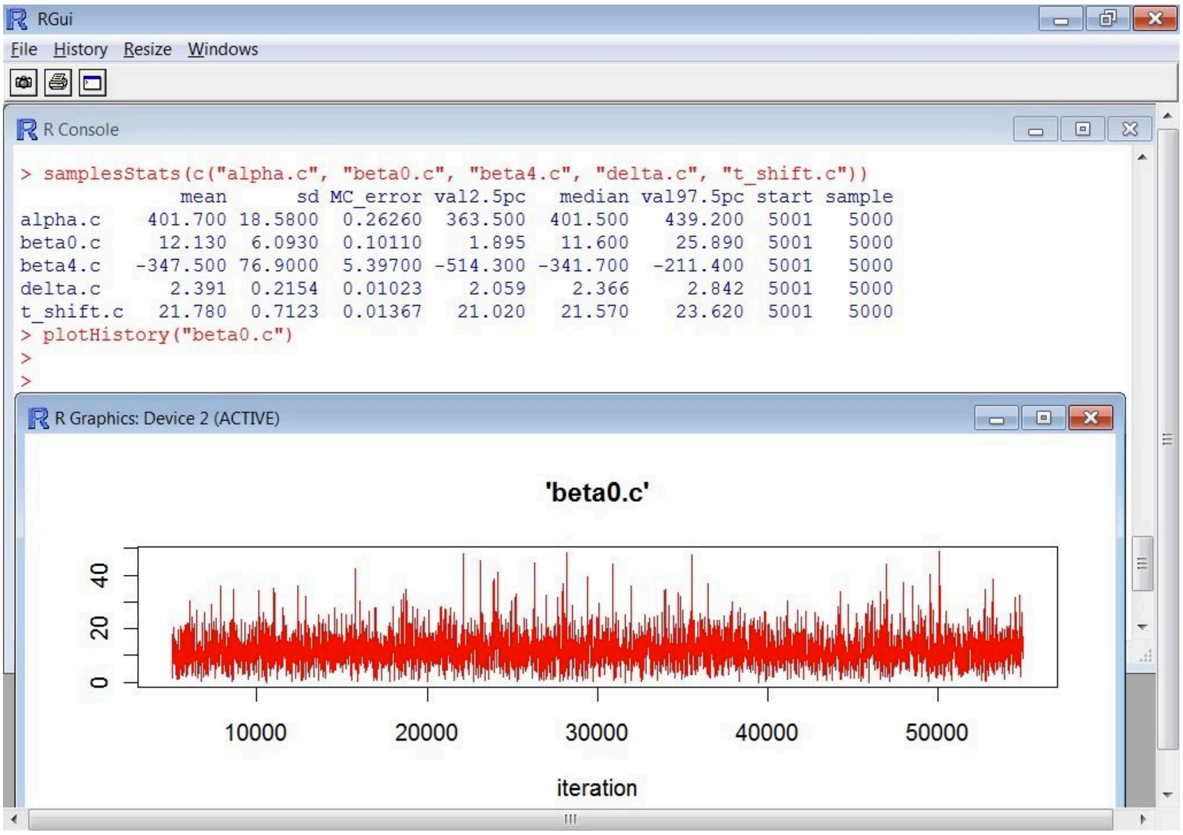
```
R> summarySet(c("alpha.c", "alphaS", "alphaV", "beta0.c", "beta0S", "beta0V",
+    "beta4.c", "beta4S", "beta4V", "delta.c", "deltaS", "deltaV",
+    "t_shift.c", "t_shiftS", "t_shiftV"))
R> modelUpdate(50000)
R> summaryStats(c("alpha.c", "beta0.c", "beta4.c", "delta.c", "t_shift.c"))
```

might be used in place of the preceding samplesSet-samplesStats pair of calls. The **BRugs** help document states that the resulting quantiles are approximate and should be used with caution, but offer the advantage of requiring less storage than those obtained with sampleSet.

3.2. Method 2: Working with mcmc.list objects

The function buildMCMC generates an R object with class mcmc.list, which can be accessed within R using standard commands, as illustrated in the following code, which uses double-bracket indexing notation (returning either the entire beta0.c chain and its first sample, respectively):

```
R> samplesSet(c("alpha.c", "alphaS", "alphaV", "beta0.c", "beta0S", "beta0V",
+    "beta4.c", "beta4S", "beta4V", "delta.c", "deltaS", "deltaV",
+    "t_shift.c", "t_shiftS", "t_shiftV"))
R> modelUpdate(50000)
```



```
> samplesStats(c("alpha.c", "beta0.c", "beta4.c", "delta.c", "t_shift.c"))
              mean      sd MC_error val2.5pc   median val97.5pc start sample
alpha.c    401.700 18.5800  0.26260  363.500  401.500   439.200  5001   5000
beta0.c     12.130  6.0930  0.10110    1.895   11.600    25.890  5001   5000
beta4.c   -347.500 76.9000  5.39700 -514.300 -341.700  -211.400  5001   5000
delta.c      2.391  0.2154  0.01023    2.059    2.366     2.842  5001   5000
t_shift.c   21.780  0.7123  0.01367   21.020   21.570    23.620  5001   5000
> plotHistory("beta0.c")
>
>
```

Figure 2: The R GUI window captured after using the `samplesStats` function to generate summary statistics for a set of specified variables, together with the `plotHistory` function, which produces a trace plot for the specified variable(s). Trace plots are indispensable as a tool for detecting convergence failure, as discussed in numerous publications, including Gelman and Rubin (1992), Carlin and Louis (2000, Chapter 5), Gelman (1996) and Kass *et al.* (1998). Trace plots can be usefully displayed individually, as shown here, and as overlaid plots. Well-behaved MCMC output is characterized by parallel-chain trace plots showing that each sampler in a set of parallel simulations (started at overdispersed points in parameter space) has moved freely around a common parameter space, with no sign of being stuck within a localized region, and with no systematic trends.

```
R> codaobject.beta0.c <- buildMCMC("beta0.c")
R> codaobject.beta0.c[[1]]
R> codaobject.beta0.c[[1]][1]
```

The main purpose of `mcmc.list` objects is, however, to provide input to the **coda** package, which includes functions for generating various convergence diagnostic plots and for performing a number of convergence test calculations. Visual convergence diagnostic checks and their associated calculations are important because validity of any inferences depends on convergence to a stationary distribution. Some of these checks can be performed using the `summary` function which, in conjunction with `mcmc.list` objects, provides various summary statistics including the so-called time-series standard error and specified quantiles. For example, summary statistics for the node `beta0.c`, together with Geweke convergence test results and diagnostic plot are obtained with the code:

```
R> summary(codaobject.beta0.c, quantiles = c(0.025, 0.5, 0.975))
R> geweke.diag(codaobject.beta0.c, frac1 = 0.1, frac2 = 0.5)
R> geweke.plot(codaobject.beta0.c, frac1 = 0.1, frac2 = 0.5, nbins = 20,
+    pvalue = 0.05, auto.layout = TRUE, ask = TRUE)
```

3.3. Method 3: Using `samplesSample` to access the sampled data

The **BRugs** function `samplesSample`, used in conjunction with `samplesSet`, returns an array of selected values, as illustrated in the preceding R session. It should be noted that the `samplesSample` argument must be a single scalar node, and that some additional code is required to deal with vector, matrix or array nodes, as illustrated in the example R session, which uses the **BRugs** `samplesMonitors` function in conjunction with the R `sapply` function to return the specified matrix nodes. In that example, each call to `samplesMonitors` returns the name of each element of the specified matrix node, and the accompanying call to `sapply` uses the **BRugs** function `samplesSample` to return the MCMC output for each of these elements.

The various code fragments and example R session given in this Section are rudimentary, the objective being to provide an immediate and simple indication of the manner in which various **BRugs** tools might be used to retrieve and process **WinBUGS** MCMC output. Obviously these function calls can be wrapped in additional code to achieve any level of programming sophistication. This is demonstrated, for example, in the open source code available from `http://code.google.com/p/alzheimers-disease-progression-model-adascog/source/browse/trunk/cfbmodel/code`, provided by William Gillespie.

4. Results

Bayesian random effects modeling was performed using the DSC-MRI signal intensity data obtained from a 16-voxel (4-by-4) ROI placed within the thalamus of each of five subjects. The main feature of the analytical approach outlined in this paper is that it permits a simultaneous modeling of all five subjects and all 16 voxels within each subject. Figure 3 shows the signal intensity data in six selected voxels taken from one of the five subjects, with the modelled data superimposed. In some voxels the signal intensity minimum is reasonably well defined, while in others the Gd-DTPA response is partially obscured by noise. Statistical analysis (results not shown) indicate a lack of evidence for an equivalent response in selected voxel pairs, providing justification for the spatial random effects treatment, as opposed to ROI averaging.

Figure 4 shows Gd-DTPA response data taken from two other subjects, with the modelled data superimposed. In the first of these subjects (Subject A) the response profile is well defined in all 16 voxels, hence the good agreement between the observed and modelled data. In the second case (Subject B), the response in some voxels within the ROI is swamped by noise, with a minority of voxels providing evidence of a well-defined minimum in signal intensity. Subject B provides an example in which an independent-voxels analysis is expected to be problematic.

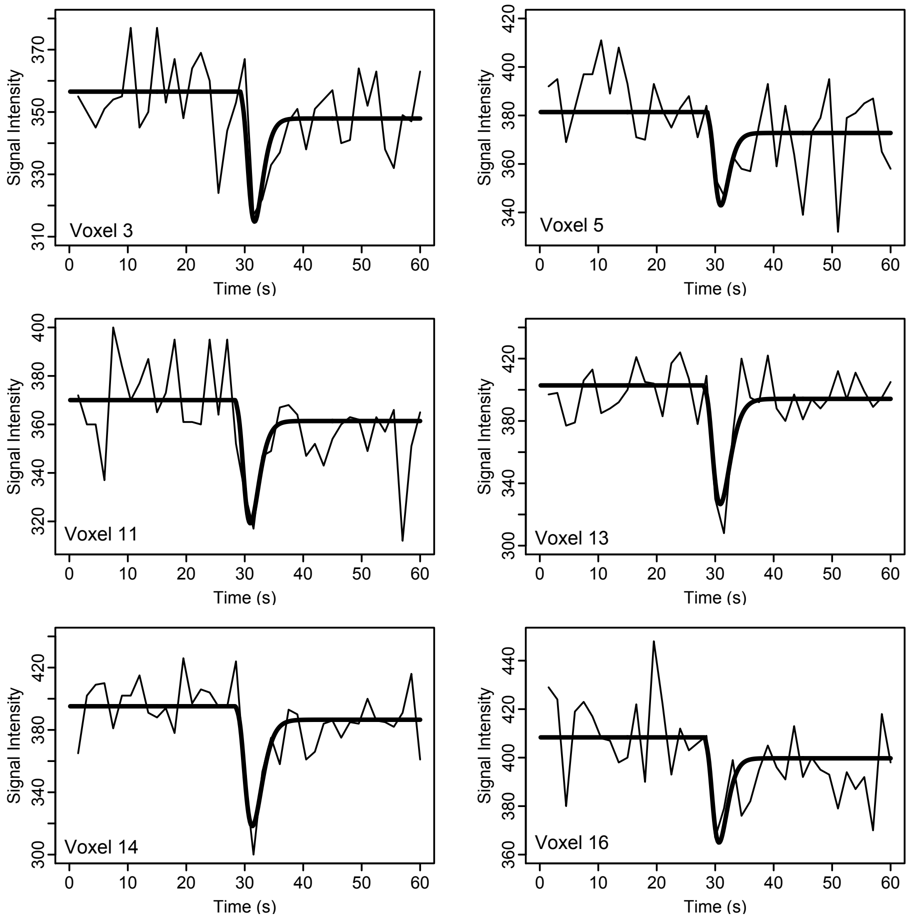


Figure 3: Signal intensity data acquired from six selected voxels in one of the five subjects (thin line), with the model responses superimposed (broad line). Voxels 11 and 14 show a reasonably well-defined minimum in the observed signal intensity, close to the expected peak position, while in voxels 3 and 16 the position of the minimum is partly obscured by noise.

5. Discussion

The main purpose of this paper is to alert MRI researchers to Bayesian random effects modeling as a general spatial or spatiotemporal data processing method, and to suggest that it might be applied with advantage to a wide variety of MR image processing problems. We have extended the spatiotemporal random effects treatment by incorporating it into an hierarchical model, as required to achieve a formal analysis of between-subject variation in response. We demonstrate the manner in which this can be achieved by using the **BRugs** package, which provides an R interface to **WinBUGS** and its **GeoBUGS** add-on. A DSC-MRI dataset has been used for the purpose of illustration, demonstrating that a random effects analysis is capable of providing useful voxel-specific contrast-agent response profiles, despite the low signal-to-noise ratio of the DSC data. In previous publications we have outlined the application of Bayesian random effect models to diffusion MR image processing and the analysis of multivariate longitudinal MR data (King *et al.* 2003, 2005, 2009). These provide other examples of the advantages of the Bayesian random effects modeling approach, although these

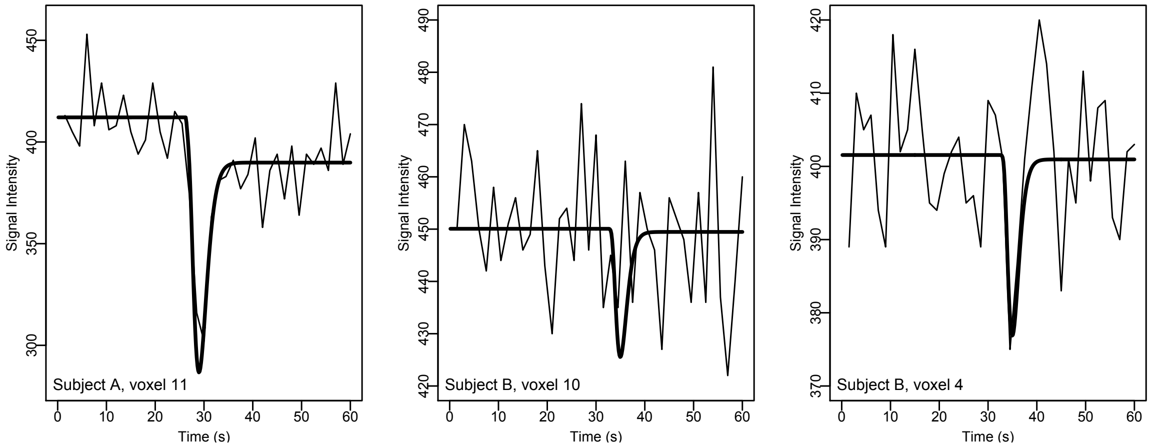


Figure 4: Single-voxel signal intensity data obtained in two of the five subjects not featured in Figure 3 (thin line). Subject A has a well-defined Gd-DTPA response in each of the 16 selected voxels, one of which is shown (Sugject A, voxel 11). In contrast, the response in Subject B is less well defined in the majority of voxels, an extreme example of which is shown (Subject B, voxel 10), together with the data obtained from one voxel with a better defined minimum (Subject B, voxel 4). In each case the Gd-DTPA response profile generated by the random effects model is shown superimposed (broad line).

analyses were not undertaken using R.

Among the attractive properties of random effect models in general, and Bayesian random effect models in particular, is a formal pooling of information across the entire dataset. In the Introduction we refer to the concept of a weighted combination of the estimates that might be obtained from a set of separate, independent analyses and those provided by a completely pooled (e.g., ROI averaged) data analysis. In the present application information pooling occurs across subjects, and across voxels within subjects. Thus the data that are provided by each voxel within each subject supplies information with a capacity to improve the parameter estimates obtained for every other voxel in the dataset. We emphasize that there is no *direct* correlation between voxels in different subjects under the random effects model outlined in this paper. This coupling occurs only indirectly via the combined influence of the exchangeable-subjects random effect and the spatial priors. For the purpose of general illustration we have shown an analysis based on combining information across a set of subjects, recognising that some MRI data analysts might ague for a separate and independent analysis of each subject when working in a clinical setting. The distinguishing feature of most clinical investigative work is subject-specific diagnosis which, in the regression context, implies an emphasis on individual-specific parameter estimation as distinct from group mean parameter estimation. Nevertheless, we wish to emphasize that pooling information across a group of similar subjects can be beneficial when dealing with sparse or noisy data, even if interest is restricted to parameter estimation in individual subjects. This strategy is well documented in the statistics literature. For example, a commonly encountered difficulty in clinical pharmacokinetics/pharmacodynamics is the impracticality of acquiring sufficient data on an individual under investigation to obtain the required kinetic parameter estimates with the necessary precision. In fact, it is not uncommon to face the situation in which the data obtained for some subjects are so sparse that the resulting analytical problem is either underdetermined or too ill conditioned to allow any useful estimates to be obtained (Gelman *et al.* 1996). The Bayesian random effects approach can turn an ill-conditioned problem into one that is well conditioned. This is achieved through a combination of two features of the Bayesian formalism. Firstly, the explicit incorporation of prior information on one or more of the model parameters is a mechanism through which constraints can be imposed, thus producing an improvement in conditioning (Gelman *et al.* 1996; Wakefield 1996). Relevant prior information is often available as published physiological data. A second mechanism through which a Bayesian random effects model provides an alleviation of the effects of ill-conditioning is through the formal combination of information afforded by similar subjects, taken as a whole in the form of distributional information. Thus, although the data obtained for some individuals might be insufficient to provide useable parameter estimates if used in isolation, the distributional information provided by other subjects has a constraining effect that can yield useful estimates for otherwise poorly identified individual-specific parameters. This pooling of information is referred to as information borrowing and it is achieved via various distributional constraints that are incorporated into the model.

The Introduction to this paper includes a section on the merits of random effects modeling, as compared with the alternatives, namely a single analysis based on completely pooled data and a set of separate analyses. Of course, averaging over subjects is not an option in a clinical diagnostic setting, so random effects modeling is critical when pooling data across subjects but with individual diagnoses as the objective. That said, averaging over voxels within an ROI remains an option. ROI averaging can, however, be criticized on theoretical grounds, since it can cause a marked change in response profile shape. It is an inescapable fact that, given an underlying nonlinear model, the mean curve (i.e., the curve defined by the parameter means) is not, in general, identical to the curve obtained by fitting the mean observations (Hand and Crowder 1996, p. 122). Data averaging can cause a substantial corruption of the underlying functional dependencies when working with non-linear models. The perfusion data used in this paper exhibit a demonstrable lack of equivalence in the temporal response among voxels in at least one subject (results not shown), suggesting that ROI averaging might be inappropriate. It could be argued that marked differences between the voxel-specific parameter estimates are an indication of poor ROI selection, despite using regions selected from a part of the thalamus that appeared uniform on baseline T2-weighted images. We suggest, however, that restricting an analysis to a sample of voxels exhibiting marked homogeneity may not be realistic in a clinical setting involving pathology. In applications in which a lack of homogeneity is a concern, the random effects model can be extended with the distinct purpose of dealing with heterogeneity. For example, in a previous paper (King *et al.* 2009) we examined the Besag-York-Mollié model (Besag *et al.* 1991) which uses a combination of an exchangeable prior and a spatial prior as a mechanism for dealing with boundaries. (The concept of exchangeability is well known in random effect models analysis. In this context the term *exchangeable* indicates that the model is invariant to permutation of the unit labels, where units are voxels and subjects in the present analysis. Thus an exchangeable voxels model takes no account of the spatial relationship between voxels, but makes a distribution assumption for each voxel-specific parameter.) The Besag-York-Mollié model, together with mixture models and other approaches to dealing with heterogeneity, are well documented in the statistics literature. For example, a number of researchers engaged in disease mapping have developed spatial models with a capacity to deal with localized heterogeneity; see Richardson *et al.* (2004) and references therein. The analytical challenge is to deal with marked dissimilarity among the units under investigation, for example, the presence of abrupt spatial differences or boundaries.

The analytical challenges encountered in DSC-MRI data processing are not atypical of those that are met by MRI data analysts in general. Among the defining features of a typical clinical DSC-MRI data set is the longitudinal nature of the observations, coupled with the need to adopt some form of nonlinear model. Furthermore, the example dataset used in this paper is characteristic of many clinical datasets which fall into the category of *small sample size* problems. *Large sample approximation*, based on so-called asymptotic theory, is central to the traditional frequentist statistical calculations that are typically carried out subsequent to parameter estimation as performed using some minimization (optimization) algorithm. This applies to both standard non-linear regression modeling problems (Rawlings 1988, Section 14.4) and mixed model regression analysis; see, for example, Chapters 5 and 6 in Crowder and Hand (1990). The Wald statistic is well known in this context (Hand and Crowder 1996). By definition, asymptotic results are not applicable when the sample size is small. Standard errors and confidence intervals based on asymptotic theory are approximate, and the approximation is expected to be particularly poor in small-sample problems. Although some of the problems that arise in frequentist mixed model regression analysis are avoided by adopting a Bayesian random effects approach, related computational problems often remain in practice due to the need to evaluate complex, multidimensional integrals, often of high order. Asymptotic approximation (also referred to as normal approximation) is among the various computational methods that were adopted as a mechanism for avoiding intractable integrals (Carlin and Louis 2000; Gelman *et al.* 2004). Although this provided a solution, the need for some kind of approximation together with the computational demands of the calculations remained a major impediment for many years. Fortunately, modern MCMC algorithms combined with cheap but powerful computers, as required for their implementation, has removed these obstacles. MCMC effectively allows a complete characterization of the posterior distribution, and traditional methods of posterior evaluation no longer play a prominent role. Although simulation error remains a source of inaccuracy, in principle simulation error can be reduced to any specified level by collecting a sufficient number of samples. To this end the Raftery-Lewis calculation, which is implemented in the **coda** package, provides an estimate of the number of iterations required to obtain quantile estimates with a specified accuracy (see Materials and Methods section). In summary, Bayesian random effects modeling is regarded by many analysts as the method of choice for multilevel data modeling.

This paper has focussed on fitting DSC time course data using a Bayesian spatiotemporal random effects model, implemented using MCMC. This model provides a direct estimate of contrast-agent arrival time (i.e., the change point) and, with some additional computation, other summary variables, time to maximum response, for example, could be estimated as part of the MCMC analysis. But there is a CPU time limit to the amount of computation that can be realistically incorporated into the MCMC simulation. This practical problem might be circumvented by adopting an approach in which the random effects analysis is used in the first of a two-stage data processing procedure, involving a first-stage smoothing across voxels within an ROI and, where applicable, across subjects. This would be followed by parameter estimation at the second stage, performed using inversion, for example (Østergaard *et al.* 1996). The latter approach is admittedly informal and lacks statistical rigour. Unfortunately a formal treatment in which inversion and parameter estimation is performed within the MCMC simulation is not computationally feasible at present.

To conclude, the main purpose of this paper is to advertise Bayesian random effects modeling as a general MRI data analytical tool, and to demonstrate an implementation in R. As stated previously, we do not claim to provide a definitive model for DSC-MRI data analysis as opposed to using the DSC-MRI data for illustrative purposes, noting that the analytical challenges resemble those discussed in relation to the sparse data problem that arises in pharmacokinetics/pharmacodynamics. Specifically, the DSC-MRI data are sparse in the sense that the sampling interval is too long to capture the MR response to the contrast agent in detail, as the agent passes through the tissue under investigation. Furthermore, the signal-to-noise ratio is poor. With so few temporal observations, and given the lack of precision in the signal intensity data, it is not possible to obtain direct and reliable voxel-level perfusion parameter estimates based on an independent-voxels analysis. The analytical challenge that arises from the sparse and noisy nature of the data is compounded by the need to adopt some form of non-linear model. Despite these difficulties, perfusion parameter estimation is an essential component of the investigations that are undertaken. The results presented in this paper show that sensible contrast-agent response profiles can be obtained from these data. For example, in one subject the Gd-DTPA response is particularly ill-defined at the voxel level (Figure 4), providing an example where random effects modeling is especially advantageous. Thus, in those voxels in which the response is more poorly determined, the parameter estimates are pulled relatively strongly towards the average of the neighbouring voxels. This example serves to illustrate the power of the random effects modeling approach to dealing with sparse, messy observations obtained with a method that suffers from poor signal-to-noise ratio. As an aside, we wish to alert readers to the changepoint (breakpoint) treatment (Bacon and Watts 1971; Carlin *et al.* 1992) used in this analysis. This kind of piecewise model is ideal for tackling problems involving abrupt change, including disease/pathology onset and related applications that arise frequently in the MR literature. It might be argued that changepoint models are underused by MRI data analysts as a mechanism for modeling abrupt onset. That said, our main point is that there are many MR applications in which noise limits the spatial and/or temporal resolution that can be achieved. These include diffusion imaging, spectroscopic imaging, time-resolved-MRI and the structural imaging of some pathologies. This paper, taken together with our previous papers on random effects modeling and other work cited in the Introduction, makes the case for adopting some form of random effects analysis in some of these applications.

Acknowledgments

The work presented in this paper was funded by the EPSRC (EP/G025452/1). Fernando Calamante is grateful to the National Health and Medical Research Council (NHMRC) of Australia for financial support.

References

Bacon DW, Watts DG (1971). “Estimating the Transition Between Two Intersecting Straight Lines.” *Biometrika*, **58**, 525–534.

Bates D, Maechler M, Bolker B (2011). ***lme4**: Linear Mixed-Effects Models Using S4 Classes.* R package version 0.999375-38, URL http://CRAN.R-project.org/package=lme4.

Besag J, York J, Mollié A (1991). "Bayesian Image Restoration, with Two Applications in Spatial Statistics." *Annals of the Institute of Statistical Mathematics*, **43**, 1–59.

Calamante F (2005). "Artifacts and Pitfalls in Perfusion MR Imaging." In JH Gillard, AD Waldman, PB Barker (eds.), *Clinical MR Neuroimaging: Diffusion, Perfusion and Spectroscopy*, pp. 141–160. Cambridge University Press, Cambridge.

Calamante F, Thomas DL, Pell GS, Wiersma J, Turner R (1999). "Measuring Cerebral Blood Flow Using Magnetic Resonance Imaging Techniques." *Journal of Cerebral Blood Flow and Metabolism*, **19**, 701–735.

Calamante F, Vonken EPA, van Osch MJP (2007). "Contrast Agent Concentration Measurements Affecting Quantification of Bolus-Tracking Perfusion MRI." *Magnetic Resonance in Medicine*, **58**, 544–553.

Carlin BP, Gelfand AE, Smith AFM (1992). "Hierarchical Bayesian Analysis of Changepoint Problems." *Applied Statistics*, **41**, 389–405.

Carlin BP, Louis TA (2000). *Bayes and Empirical Bayes Methods for Data Analysis.* 2nd edition. Chapman & Hall/CRC, Boca Raton.

Cowles MK, Carlin BP (1996). "Markov Chain Monte Carlo Convergence Diagnostics: A Comparative Review." *Journal of the American Statistical Association*, **91**, 883–904.

Crawley MJ (2007). *The R Book.* John Wiley & Sons, Chichester.

Crowder M (1992). "Interlaboratory Comparisons: Round Robins with Random Effects." *Journal of the Royal Statistical Society C*, **42**, 409–425.

Crowder MJ, Hand DJ (1990). *Analysis of Repeated Measures.* Chapman & Hall, London.

Crowder MJ, Tredger JA (1981). "The Use of Exponentially Damped Polynomials for Biological Recovery Data." *Applied Statistics*, **30**, 147–152.

Gelman A (1996). "Inference and Monitoring Convergence." In WR Gilks, S Richardson, DJ Spiegelhalter (eds.), *Markov Chain Monte Carlo in Practice*, pp. 131–143. Chapman & Hall, London.

Gelman A (2006). "Prior Distributions for Variance Parameters in Hierarchical Models (Comment on Article by Browne and Draper)." *Bayesian Analysis*, **1**, 515–534.

Gelman A, Bois F, Jiang J (1996). "Physiological Pharmacokinetic Analysis using Population Modeling and Informative Prior Distributions." *Journal of the American Statistical Association*, **91**, 1400–1412.

Gelman A, Carlin JB, Stern HS, Rubin DB (2004). *Bayesian Data Analysis.* 2nd edition. Chapman & Hall/CRC, Boca Raton.

Gelman A, Rubin DB (1992). "A Single Series from the Gibbs Sampler Provides a False Sense of Security." In JM Bernardo, JO Berger, AP Dawid, AFM Smith (eds.), *Bayesian Statistics*, volume 4, pp. 625–631. Oxford University Press, Oxford.

Gilks WR, Richardson S, Spiegelhalter DJ (eds.) (1996). *Markov Chain Monte Carlo in Practice*. Chapman & Hall, London.

Hand D, Crowder M (1996). *Practical Longitudinal Data Analysis*. Chapman & Hall, London.

Jones RH (1993). *Longitudinal Data with Serial Correlation: A State-Space Approach*. Chapman & Hall, London.

Kass RE, Carlin BP, Gelman A, Neal RM (1998). "Markov Chain Monte Carlo in Practice: A Roundtable Discussion." *The American Statistician*, **52**, 93–100.

King MD, Crowder MJ, Hand DJ, Harris NG, Williams SR, Obrenovitch TP, Gadian DG (2003). "Temporal Relation Between the ADC and DC Potential Responses to Transient Focal Ischemia in the Rat: A Markov Chain Monte Carlo Simulation Analysis." *Journal of Cerebral Blood Flow and Metabolism*, **23**, 677–688.

King MD, Crowder MJ, Hand DJ, Harris NG, Williams SR, Obrenovitch TP, Gadian DG (2005). "Is Anoxic Depolarisation Associated with an ADC Threshold? A Markov Chain Monte Carlo Analysis." *NMR in Biomedicine*, **18**, 587–594.

King MD, Gadian DG, Clark CA (2009). "A Random Effects Modelling Approach to the Crossing-Fibre Problem in Tractography." *NeuroImage*, **44**, 753–768.

Laird NM, Ware JH (1982). "Random-Effects Models for Longitudinal Data." *Biometrics*, **38**, 963–974.

Lawson AB, Browne WJ, Vidal Rodeiro CL (2003). *Disease Mapping with **WinBUGS** and **MLwiN***. John Wiley & Sons, Chichester.

Lindley DV, Smith AFM (1972). "Bayes Estimates for the Linear Model." *Journal of the Royal Statistical Society B*, **34**, 1–41.

Lunn D, Spiegelhalter D, Thomas A, Best N (2009). "The **BUGS** Project: Evolution, Critique and Future Directions." *Statistics in Medicine*, **28**, 3049–3067.

Lunn DJ, Thomas A, Best N, Spiegelhalter D (2000). "**WinBUGS** – A Bayesian Modelling Framework: Concepts, Structure, and Extensibility." *Statistics and Computing*, **10**, 325–337.

McLean RA, Sanders WL, Stroup WW (1991). "A Unified Approach to Mixed Linear Models." *The American Statistician*, **45**, 54–64.

Østergaard L, Weisskoff RM, Chesler DA, Gyldensted C, Rosen BR (1996). "High Resolution Measurement of Cerebral Blood Flow Using Intravascular Tracer Bolus Passages. Part 1: Mathematical Approach and Statistical Analysis." *Magnetic Resonance in Medicine*, **36**, 715–725.

Park JH (2011). "CRAN Task View: Bayesian Inference." Version 2011-10-07, URL `http://CRAN.R-project.org/view=Bayesian`.

Penny WD, Friston KJ (2004). "Hierarchical Models." In RSJ Frackowiak, KJ Friston, CD Frith, RJ Dolan, CJ Price, S Zeki, JT Ashburner, WD Penny (eds.), *Human Brain Function*, pp. 851–863. Academic Press, San Diego.

Pinheiro J, Bates D, DebRoy S, Sarkar D, R Development Core Team (2011). ***nlme**: Linear and Nonlinear Mixed Effects Models.* R package version 3.1-98, URL http://CRAN.R-project.org/package=nlme.

Pinheiro JC, Bates DM (2000). *Mixed-Effects Models in S and S-PLUS*. Springer Verlag, New York.

Plummer M (2011). ***rjags**: Bayesian Graphical Models using MCMC.* R package version 2.1.0-6, URL http://CRAN.R-project.org/package=rjags.

Plummer M, Best N, Cowles K, Vines K (2006). "**coda**: Convergence Diagnosis and Output Analysis for MCMC." *R News*, **6**(1), 7–11. URL http://CRAN.R-project.org/doc/Rnews/.

Rawlings JO (1988). *Applied Regression Analysis: A Research Tool*. Wadsworth and Brooks, Belmont.

R Development Core Team (2011). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0, URL http://www.R-project.org/.

Richardson S, Thomson A, Best N, Elliott P (2004). "Interpreting Posterior Relative Risk Estimates in Disease-Mapping Studies." *Environmental Health Perspectives*, **112**, 1016–1025.

Schmid VJ, Whitcher B, Padhani AR, Taylor NJ, Yang GZ (2006). "Bayesian Methods for Pharmacokinetic Models in Dynamic Contrast-Enhanced Magnetic Resonance Imaging." *IEEE Transactions on Medical Imaging*, **25**, 1627–1636.

Schmid VJ, Whitcher B, Padhani AR, Taylor NJ, Yang GZ (2009). "A Bayesian Hierarchical Model for the Analysis of a Longitudinal Dynamic Contrast-Enhanced MRI Oncology Study." *Magnetic Resonance in Medicine*, **61**, 163–174.

Spiegelhalter D, Thomas A, Best N, Gilks W (1995). "**BUGS** 0.5. Bayesian Inference Using Gibbs Sampling Manual." MRC Biostatistics Unit, Institute of Public Health, Cambridge, URL http://www.mrc-bsu.cam.ac.uk/bugs/documentation/bugs05/manual05.html.

Spiegelhalter D, Thomas A, Best N, Lunn D (2003). "**WinBUGS** User Manual." MRC Biostatistics Unit, Institute of Public Health, Cambridge, URL http://www.mrc-bsu.cam.ac.uk/bugs/winbugs/manual14.pdf.

Sturtz S, Ligges U, Gelman A (2005). "**R2WinBUGS**: A Package for Running **WinBUGS** from R." *Journal of Statistical Software*, **12**(3), 1–16. URL http://www.jstatsoft.org/v12/i03/.

Thomas A, O'Hara B, Ligges U, Sturtz S (2006). "Making **BUGS** Open." *R News*, **6**(1), 12–17. URL http://CRAN.R-project.org/doc/Rnews/.

Wakefield J (1996). "The Bayesian Analysis of Population Pharmacokinetic Models." *Journal of the American Statistical Association*, **91**, 62–75.

Wakefield JC, Best NG, Waller L (2000). "Bayesian Approaches to Disease Mapping." In P Elliott, JC Wakefield, NG Best, DJ Briggs (eds.), *Spatial Epidemiology: Methods and Applications*, pp. 104–127. Oxford University Press, Oxford.

Walker-Samuel S, Orton M, Boult JKR, Robinson SP (2011). "Improving Apparent Diffusion Coefficient Estimates and Elucidating Tumor Heterogeneity using Bayesian Adaptive Smoothing." *Magnetic Resonance in Medicine*, **65**, 438–447.

Affiliation:

Martin D. King, Chris A. Clark and David G Gadian
Imaging and Biophysics Unit
UCL Institute of Child Health
30 Guilford Street
London WC1N 1EH, United Kingdom
E-mail: M.King@ich.ucl.ac.uk

Fernando Calamante
Brain Research Institute
Florey Neuroscience Institutes
Melbourne, Australia
and
Department of Medicine
University of Melbourne
Melbourne, Australia

Journal of Statistical Software http://www.jstatsoft.org/
published by the American Statistical Association http://www.amstat.org/
Volume 44, Issue 2 *Submitted:* 2010-10-01
October 2011 *Accepted:* 2011-06-10