



SynMRI: An R Package for Synthetic Magnetic Resonance Imaging

Martin Silerio-Vazquez.
Iowa State University

Abstract

Synthetic magnetic resonance (MR) imaging refers to the procedure used to predict magnetic resonance images for any given design parameter settings using at least three observed MRI scans. **SynMRI** is an R package we developed for this purpose. We implemented the method proposed in Maitra and Riddles (2010), which used a model based on the Bloch equation, an empirical expression describing the nuclear magnetic resonance phenomena, to get voxel-wise estimates that are used to predict the intensity values for a given design parameters settings. The noise on the MR signal is modeled using the Rice distribution. All the parameters involved are estimated using the EM algorithm. **SynMRI** includes functions that compute the estimates, and visualize and evaluate the results. The EM algorithm estimation stage is performed in C since it is the part most compute intensive.

Keywords: Bloch equation, EM algorithm, Rice distribution, L-BFGS-B, penalized log likelihood, spin-echo sequence.

1. Introduction

Magnetic resonance imaging (MRI) is a tool used in radiology to visualize tissue structure and study cerebral function. Tissues can be characterized with their longitudinal or spin-lattice relaxation time (T_1), the transverse or spin-spin relaxation time (T_2) and the proton density (ρ). These physical quantities are typically unobservable, but their influence on the imaging process can be modulated with user-controlled scanner parameters. For the spin-echo sequence imaging (the type of imaging we focus on) the parameters repetition time (TR), echo time (TE) and flip angle (α) are used. Also for the spin-echo sequence imaging the magnitude of the true MR signal (denoted by ν) can be expressed in terms of a solution to the Bloch equation. The Bloch equation is an empirical expression describing the nuclear magnetic

resonance phenomena. Therefore, for the i th voxel corresponding to the MRI obtained with the design parameters (TE_j, TR_j) the true (noiseless) MR signal has a magnitude ν_{ij} given by

$$\nu_{ij} = \rho_i \exp\left(-\frac{TE_j}{T_{2i}}\right) \left\{1 - \exp\left(-\frac{TR_j}{T_{1i}}\right)\right\}. \quad (1)$$

It is important to note that (1) is a simplified model, as explained on [Maitra and Riddles \(2010\)](#). Different pairs of (TE, TR) are used to highlight contrasts between different types of tissue, providing a method for their identification. However the optimal design parameters to visualize a specific type of tissues vary from patient to patient and between types of tissue. This problem led to the development of *synthetic magnetic resonance imaging*. This method consist in acquiring a set of *training images* at a few settings (TE, TR) and use them to estimate the underlying physical quantities (T_1, T_2, ρ) at each voxel. These estimates are then used on the Bloch transform in combination with the desired design parameters values to generate the corresponding images. One problem, and the reason why this approach has not been developed and exploited further, is the ill-posed nature of the Bloch transform. This characteristic of the Bloch transform leads to unstable estimates of (T_1, T_2, ρ) when using classical estimation approaches like least-squares (LS). The authors in [Maitra and Riddles \(2010\)](#) propose two remedies to deal with the ill-posed nature of the Bloch transform 1) modelling the noise in the acquisition process using a Rice distribution $R(\sigma; \nu)$, 2) adding a regularization term to the estimation process that penalizes when estimates for (T_1, T_2, ρ) at a voxel i are quite different than the estimates for its immediate neighbors. For the later a Matrix normal MRF distribution is used. Consider $r_{i,j}$, the i th voxel intensity of the image j (MRI acquired at the design parameters (TE_j, TR_j)). [Maitra and Riddles \(2010\)](#) assumed that $r_{i,j}$ follows a Rice distribution with density

$$\varrho(r_{ij}; \nu_{ij}, \sigma_j) = \frac{r_{ij}}{\sigma_j^2} \exp\left(-\frac{r_{ij}^2 + \nu_{ij}^2}{2\sigma_j^2}\right) \mathbf{I}_0\left(\frac{r_{ij}\nu_{ij}}{\sigma_j^2}\right), \quad (2)$$

where $r_{i,j} > 0$ is the observed intensity, $\nu_{i,j}$ is the true (noiseless) intensity and σ_j is the common standard distribution of the real and imaginary parts of the signal that arise after the Fourier reconstruction of the original MR signal. $\mathbf{I}_0(\cdot)$ is the modified Bessel function of the first kind of order zero.

2. Theory and Methods

Let's assume that we are working with a training set consisting of m MRI, each one acquired at design parameters (TE_j, TR_j) , $j = 1, 2, \dots, m$. We consider each image to be a set of independent observations from $R(\sigma_j; \nu_{ij})$, $i = 1, 2, \dots, n$, $j = 1, 2, \dots, m$, where n is the total number of voxels per image, m the total number of parameter settings and ν_{ij} is related with (T_{1i}, T_{2i}, ρ_i) and (TE_j, TR_j) as stated on (1). Since we (as in [Maitra and Riddles \(2010\)](#)) worked with 3D images, $n = n_x n_y n_z$, where n_x, n_y, n_z are the dimensions of the 3D images. The observed loglikelihood of the unknown parameters (T_{1i}, T_{2i}, ρ_i) given the observations r_{ij} 's is

$$\ell(T_{1i}, T_{2i}, \rho_i; r_{ij}) = \prod_{j=1}^m \prod_{i=1}^n \varrho(r_{ij}; \nu_{ij}, \sigma_j). \quad (3)$$

We $m \geq 3$ for identifiability of (T_{1i}, T_{2i}, ρ_i) . The noise parameter σ_j was computed using the methodology presented on [Maitra \(2013\)](#) with code provided by the author.

2.1. EM Algorithm

[Maitra and Riddles \(2010\)](#) presents in detail the EM algorithm derivation. The resulting formula to update voxel-wise the values of (T_{1i}, T_{2i}, ρ_i) (and in consequence the values of $\nu_{i,j}$) at each iteration is

$$Q\left(\nu_{ij} \mid r_{ij}, \sigma_j, \nu_{ij}^{(t)}\right) = \sum_{i=1}^n \sum_{j=1}^m \left(-\frac{\nu_{ij}^2}{2\sigma_j^2} + \frac{r_{ij}\nu_{ij}\mathbf{I}_1\left(\frac{r_{ij}\nu_{ij}^{(t)}}{\sigma_j^2}\right)}{\sigma_j^2\mathbf{I}_0\left(\frac{r_{ij}\nu_{ij}^{(t)}}{\sigma_j^2}\right)} \right). \quad (4)$$

At each iteration the Q function is maximized over the range of values of (T_{1i}, T_{2i}, ρ_i) . Such parameters need to have their values restricted to values that make sense on the underlying physical phenomenon. We use a C implementation of the L-BFGS-B optimization algorithm ([GitHub \(2019\)](#)). The range of values we use are $\rho_i \in [1, 400]$, $T_{1i} \in [300, 12000]$ and $T_{2i} \in [5, 400]$. ρ_i is expressed in units of the image voxel intensity while T_{1i} and T_{2i} are expressed in milliseconds. The starting values for the EM algorithm are maybe the most determining factor in how the algorithm will perform. As in [Maitra and Riddles \(2010\)](#), we used the least squares estimates as starting values. The results obtained with this approach are presented later.

2.2. Penalized Loglikelihood Estimation

According to [Maitra and Riddles \(2010\)](#), (4) leads to unstable estimates. One remedy to deal with the problem is to change the range of the parameter values using some transformations. We use $W_{i1} = \rho_i$, $W_{i2} = \exp(-T_{1i}^{-1})$ and $W_{i3} = \exp(-T_{2i}^{-1})$. However, the most important change done was the addition of a regularization factor to the loglikelihood. The loglikelihood penalized by this regularization is given by

$$\ell(W_{i1}, W_{i2}, \rho_i) = Q\left(W_{i1}, W_{i2}, \rho_i \mid r_{ij}, \sigma_j, \nu_{ij}^{(t)}\right) - \log f(W, \psi, \Gamma), \quad (5)$$

where $f(W, \psi, \Gamma)$ is the density of a Matrix Normal distribution $N_{n,3}(0_{n,3}, \psi, \Gamma)$ with density

$$f(W; \Psi, \Gamma) = \frac{\exp\left(-\frac{1}{2}\text{tr}(\Psi^{-1}W'\Gamma^{-1}W)\right)}{(2\pi)^{\frac{3n}{2}} |\Psi|^{\frac{n}{2}} |\Gamma|^{\frac{3}{2}}}. \quad (6)$$

Again, we use the EM algorithm to find estimates, this time not just for $W = (W_{i1}, W_{i2}, W_{i3})$, but also for ψ and Γ . We split the maximization step in three, one for each of W , ψ and Γ . For W we proceed the same we did before, maximized $\ell(W_{i1}, W_{i2}, \rho_i)$ over the range of values of (W_{i1}, W_{i2}, W_{i3}) that makes sense, while keeping ψ and Γ fixed. For ψ we just compute the usual covariance matrix ML estimate, since the observations W_i are multivariate normal when W and Γ are fixed. Finally, we get the estimate for Γ following exactly what is described on [Maitra and Riddles \(2010\)](#). That is, assume that Γ has a special form dependent on a parameter $\beta = (\beta_x, \beta_y, \beta_z)$, whose values determine the loglikelihood value that is penalized

when values of the estimates (W_{i1}, W_{i2}, W_{i3}) deviate from the values of the estimates of their neighbors.

3. Experiments

We reproduced the main example presented in [Maitra and Riddles \(2010\)](#). We worked with a dataset consisting of a MRI taken from the brain of a healthy human male at $m = 12$ design parameters configuration (TE_j, TR_j) . The MRI's are 3D images of dimensions $n_x = 256$, $n_y = 256$ and $n_z = 20$. We used just three images as training set. These three images in the middle section across the z-axis are displayed below.

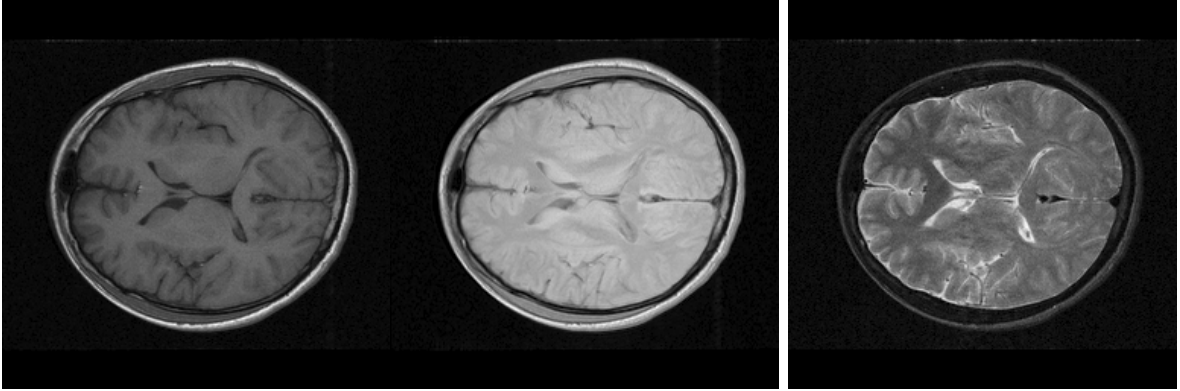


Figure 1: Image 1, Image 10 and Image 12

After fitting the parameters W , ψ and Γ , we used the resulting estimates to make predictions for the 12 design parameters settings, and we compared the predictions with the real images. We used the scaled Root Mean Square error to measure how good the prediction are. The results are in the table below.

Image	TE (ms)	TR (ms)	Maitra and R,	LSE	Penalized LL	Improvement?
1	10	600	0.08	0.05073	0.05537	No
2	15	600	0.14	0.10798	0.10745	Yes
3	20	600	0.16	0.12554	0.12519	Yes
4	10	1000	0.14	0.13476	0.13419	Yes
5	30	1000	0.2	0.16711	0.16676	Yes
6	40	1000	0.23	0.18649	0.18641	Yes
7	10	2000	–	0.27659	0.27563	Yes
8	40	2000	0.19	0.16563	0.16553	Yes
9	80	2000	0.22	0.18120	0.18106	Yes
10	10	3000	0.07	0.02197	0.02665	No
11	60	3000	0.21	0.16825	0.16830	No
12	100	3000	0.18	0.05473	0.05009	Yes

Table 1: Scaled RMSPE ν^* . Rows 1, 10 and 12 correspond to the training set.

Below we have a visual comparison between three of the original images and the predicted images from the test set. We selected the three images for that we got the worst prediction performance.

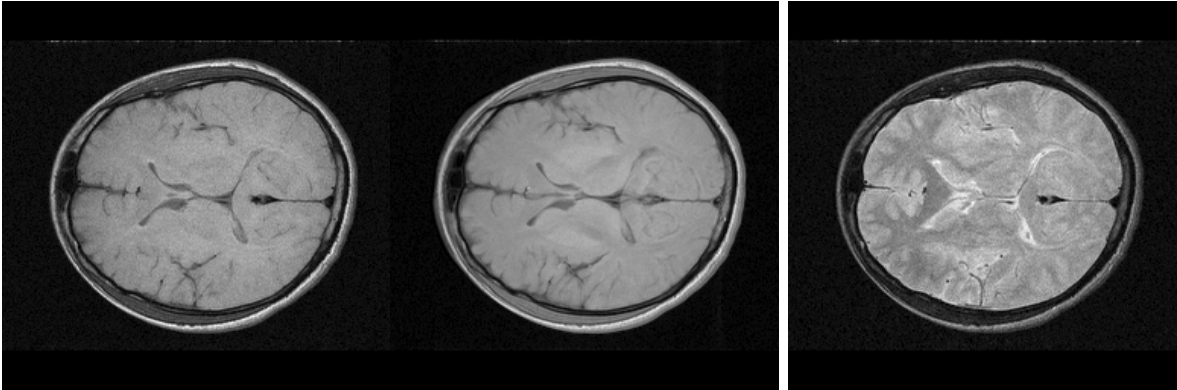


Figure 2: Image 6, Image 7 and Image 9

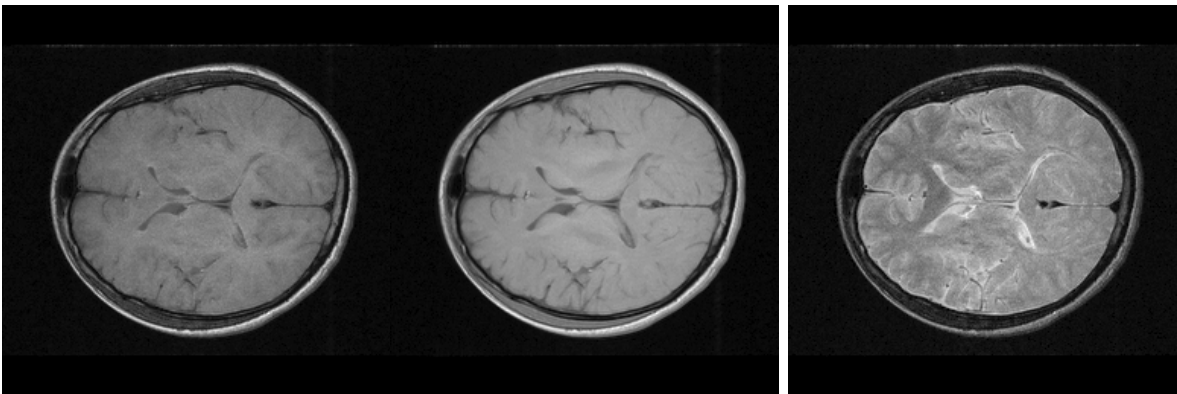


Figure 3: Predictions for Image 6, Image 7 and Image 9

4. Discussion

Our biggest concern after studying the results of our implementation is the discrepancy respect with the results obtained in [Maitra and Riddles \(2010\)](#). We need to research further to find out what is happening. But, since the results now obtained are numerically better than the ones reported in [Maitra and Riddles \(2010\)](#), and visually they look pretty close to the original images, the concern does not seem that bad after all. The package **SynMRI** and the code producing the results presented are available to download at [GitHub \(2020\)](#).

References

- GitHub I (2019). “L-BFGS-B-C.” <https://github.com/stephenbecker/L-BFGS-B-C>.
- GitHub I (2020). “SynMRI.” <https://github.com/martinsilerio/SynMRI>.
- Maitra R (2013). “On the Expectation-Maximization Algorithm for Rice-Rayleigh Mixtures With Application to Noise Parameter Estimation in Magnitude MR Datasets.” *Sankhya B*, **75**. doi:10.1007/s13571-012-0055-y.
- Maitra R, Riddles JJ (2010). “Synthetic magnetic resonance imaging revisited.” *IEEE transactions on medical imaging*, **29**(3), 895–902.

Affiliation:

Martin Silerio-Vazquez
Department of Statistics
Iowa State University
Ames, IA
E-mail: silerio@iastate.edu