Bayesian Analysis Users Guide
Release 4.00, Manual Version 1

G. Larry Bretthorst
Biomedical MR Laboratory
Washington University School Of Medicine,
Campus Box 8227
Room 2313, East Bldg.,
4525 Scott Ave.
St. Louis MO 63110
http://bayes.wustl.edu
Email: larry@bayes.wustl.edu

October 21, 2016

Contents

List of Figures

List of Tables

Appendix B

Markov chain Monte Carlo With Simulated Annealing

Most of the packages in the Bayesian Analysis software use Markov chain Monte Carlo simulations to approximate the Bayesian posterior probability. To understand how a Markov chain can be used to do this, suppose there is some quantity called M. This quantity could be a set of parameters or it could be a selection of models and the object of the Bayesian calculation is to estimate the parameters or to determine which model best characterizes the data. While these two problems sound very different, they are really one and the same problem. To see this suppose M is just the decay rate constant in a simple exponential decay, then discrete values of M, just specify a set of models $\{M_1, M_2, \ldots, M_n\}$ and in parameter estimation we compute the posterior probability for each model, just as we do in model selection. So model selection and parameter estimation are fundamentally the same problem. The major difference is that in parameter estimation problems, the functional form of the models is the same, while it can be different in model selection.

The estimation problems addressed by the various packages are all structurally similar, suppose M consists of a set of parameters or model indicators, $M \in \{M_1, M_2, \ldots, M_n\}$, and we wish to compute the posterior probability for an individual parameters $P(M_j|DI)$ where M_j is the hypothesis of interest, D represents all of the data and I is the prior information. Applying the rules of Bayesian probability theory, the joint posterior probability for all of the parameters is given by

$$P(M_1 \ldots M_n|DI) = \frac{P(M_1 \ldots M_n|I)P(D|M_1 \ldots M_n I)}{P(D|I)} \tag{B.1}$$

which is Bayes' theorem [1]. For those unfamiliar with the rules of Bayesian probability theory, see Chapter 4 for a tutorial on probability theory or consult [31, 3, 11, 33, 61, 32] for more detailed descriptions of probability theory when treated as extended logic.

If we normalize this posterior probability at the end of the calculation, then $P(D|I)$ can be dropped and one obtains:

$$P(M_1 \ldots M_n|DI) \propto P(M_1 \ldots M_n|I)P(D|M_1 \ldots M_n I). \tag{B.2}$$

Using logical independence and the product rule, the joint prior probability, $P(M_1 \ldots M_n|DI)$, can

be factored to obtain

$$P(M_1 \dots M_n|DI) \propto P(M_1|I) \cdots P(M_n|I)P(D|M_1 \dots M_n I) \tag{B.3}$$

and it is this joint posterior probability that most of the Markov chain Monte Carlo simulation approximate. To determine exactly how an individual package implements this calculation, see the Chapter describing that package.

If all a Markov chain Monte Carlo simulation did was to approximate this joint posterior probability, it would not be very useful because what is really needed in the Bayesian calculation is not $P(M_1 \dots M_n|DI)$, but $P(M_j|DI)$ where $P(M_j|DI)$ is computed using the sum rule of probability theory:

$$P(M_j|DI) = \int_{M_1} \cdots \int_{M_n} P(M_1|I) \cdots P(M_n|I)P(D|M_1 \dots M_n I)dM_1 \cdots dM_n \tag{B.4}$$

where the integrals are over all M_l except j. In the Markov chain Monte Carlo simulation this part of the calculation is done using Monte Carlo integration which consists of sampling the joint posterior probability, the integrand, and then using the samples for each parameter as samples from the marginal posterior probability for each parameter separately.

B.1 Metropolis-Hastings Algorithm

All of the packages that implement their calculations using Markov chain Monte Carlo use the Metropolis-Hastings algorithm and you can read more about the Metropolis-Hastings algorithm at the previous WikipediA link. You can consult the original 1953 paper [45] paper, Gilks et al. have written extensively on Markov chain Monte Carlo as used in Bayesian probability theory [24] and Radford Neal did his dissertation on that subject [46]. Here we will briefly summarize how the Metropolis-Hastings algorithm is used to approximate the Bayesian posterior probabilities.

At their heart all Markov chain simulations are random number generators in which you, the author, can choose the distribution of the random numbers. In the Bayesian calculations done in this software the chosen distribution is the joint posterior probability for all of the parameters and model indicators given the data and the prior information. By running the chain one can sample the joint posterior probability built into the Markov chain simulation. To run a Markov chain Monte Carlo simulation, one must be able to compute Eq. (B.3) for a given set of parameters and model indicators. Here is a very toy version of how one runs a Markov chain Monte Carlo simulation to sample the joint posterior probability:

1. One begins the process of generating a Markov chain by simply sampling the parameter from their valid range. We are going to call these parameters M_0 and the joint posterior probability computed using M_0 will be designated as P_0.

2. Next propose a new value for one or more of the parameter. Call this new proposed set of parameters M_1, and compute the joint posterior probability, P_1 using the proposed values.

3. Accept the proposal if P_1 is greater than P_0. Here accepting the proposal means that you replace M_0 and P_0 by M_1 and P_1 respectively and go back to Step 2.

4. If P_1 is less than or equal to P_0, then draw a random number, r, from a uniform $(0-1)$ random number generator, and if the ratio P_1/P_0 is greater than r, accept the proposed value of P_1 and M_1, i.e., replace M_0 and P_0 with M_1 and P_1 and go to Step 2.

5. Otherwise, reject the proposed values. Here rejecting the proposed values simply means going back to Step 2 without replacing M_0 or P_0.

This simple 5 step procedure is all it takes to generate a Markov chain Monte Carlo simulation. Unfortunately, implementing the calculation in practice, is more of an art than a science and, shortly, we will describe a few of the tricks used in ensuring the calculations work correctly.

There are several major problems with the Markov chain Monte Carlo simulation as described so far: First, it is possible for the simulation to become stuck in local maxima and one would never know it. So one needs a mechanism for detecting simulations that are trapped in local maxima. Second, even if the chain converges correctly its very difficult to tell if the Markov chain has reached a stationary point. One can run a single chain over multiple steps and then look at the path of the simulation, but this is a very unreliable method of testing whether or not a simulation has converged because simulations often deviate from the maximum. And third, with a single chain it is very difficult to adjust the acceptance rate, the number of times a change to a simulation is accepted divided by the total number of times one changed the simulation.

In the following sections we are going to describe how multiple simulations, simulated annealing, killing simulations and adjusting the rate of acceptance in the simulation can be used to generate Markov chains that are highly robust, and almost impossible to get stuck in a local maxima.

B.2 Multiple Simulations

We do not run a single Markov chain Monte Carlo simulation; rather we run an ensemble of simulations in parallel. Typically, the ensemble is on the order of a few 10's, for example the defaults number of simulations used in the interface is 50, 50 because experience with running multiple simulations indicates that most of the time 50 simulations is enough to explore most parameter spaces; while running fewer increases the risk of nonconvergence and running more usually make things run longer without improving convergence.

For reasons that will become apparent shortly, we initialize the simulations from the prior probability for the parameters. We then run the simulations through a fixed number of steps. Here running a simulation means that we vary the parameters in one simulation, and then either accept or reject the modified simulation based on the prescription given above, Section B.1. This procedure is repeated for each parameter in each simulation and we repeat this procedure at least 25 times for each parameter. So for example if there are 50 simulations, 20 parameters, and each parameter is varied 25 times, the operations count is about $50 \times 20 \times 25 = 25,000$ operations to bring the ensemble of simulations to equilibrium.

Between annealing steps various statistics are computed from the multiple simulations and these statistics are used to aid in judging convergence. Additionally, the expected value of the logarithm of the likelihood is used in thermodynamic integration, Section C, and the trajectories of each simulation are good visual aids in determining convergence.

B.3 Simulated Annealing

These simulations are run using simulated annealing. In simulated annealing one introduces an annealing parameter, which we call β, into the calculation of the joint posterior probability, Eq. (B.3). This annealing parameter is introduced by raising the direct probability for the data to the β power:

$$P(M_1 \dots M_n|\beta DI) = P(M_1|I) \cdots P(M_n|I)P(D|M_1 \dots M_n I)^{\beta} \tag{B.5}$$

where we have modified the notation to indicate that the joint posterior probability is a function of β. It is this modified joint probability density function that is used in the Markov chain Monte Carlo simulation. When $\beta = 0$, the likelihood is raised to the power of zero and the data completely disappears from the problem, one is sampling the prior. Consequently, when the Markov chain Monte Carlo simulations are initialized they are initialized from the prior probability for the parameters. Also note, that when $\beta = 1$ we are sampling the full joint posterior probability for the parameters and model indicators. The annealing parameter, β, is varied from zero to one according to some annealing schedule, discussed shortly.

Typically one starts the simulations with $\beta = 0$ and runs the simulations until they reach equilibrium. Running the simulations means changing the parameters in a simulation and then accepting or rejecting the change according to the simple perception given earlier, Chapter B.1. For a given value of β, when running the simulations, the posterior probability will increase to an equilibrium point. That is to say, the posterior probability will quit increasing and simply fluctuate about the peak in the posterior probability.

Ounce the simulations are in equilibrium, we increase the annealing parameter by small amount. This has the effect of knocking the ensemble of simulations out of equilibrium, so we again run the simulations until they reach equilibrium at this new value of β. When the annealing parameter is increased, the likelihood becomes more important and the simulations will begin to cluster around the high likelihood regions. However, because the annealing parameter is still small, the simulations will explore a much larger part of parameter space simply because the likelihood is not let strongly constraining them. As the annealing parameter is increased the likelihood becomes increasing important and the simulations begin to cluster abound increasingly high and higher likelihood regions.

B.4 The Annealing Schedule

The annealing schedule, the way the annealing parameter β is varied from zero to one, can be something as simple as varying the annealing parameter linearly to something much more elaborate. In earlier versions of the software, a linear annealing schedule was used. In a linear annealing schedule the annealing parameter was given by:

$$\beta = \frac{j}{n} \qquad (0 \leq j \leq n) \tag{B.6}$$

where n is the number of nonzero steps taken in the annealing. This worked well for many problems, but sometimes ran into difficulty when the logarithm of the likelihood is very rapidly changing, because the first tentative steps in simulated annealing can raise the likelihood many hundreds of orders of magnitude and consequently the simulations can fail to local the global maximum of the posterior probability.

In the current version of the software the annealing parameter is adjusted dynamically as follows. The annealing parameter starts at zero, and the simulations are run until they reaches equilibrium. Call this step n. For the next step, the $n+1$ step, the annealing parameter is given by

$$\beta_{n+1} = \text{Min}(1, \beta_n + d\beta_n) \tag{B.7}$$

where β_{n+1} is the value of β to be used in the next annealing step. If the minimum number of annealing steps is N, then $d\beta$ is given by:

$$d\beta = \text{Min}\left(\frac{1}{\sigma + N}, 1 - \beta\right) \tag{B.8}$$

where σ is the standard deviation of the logarithm of the likelihood computed form the ensemble of simulations. Note that if the standard deviation of the logarithm likelihood is small, then this method of computing β just reduces to Eq. (B.6), i.e., a linear annealing schedule. However, when the simulations are first initialized by sampling the prior probability for the parameters, the standard deviation of the logarithm of the likelihood is usually large, and consequently, the simulations initially move slowly, gaining speed as the simulations converge on the global maxima.

It is this initial slow annealing that allows the multiple Markov chain simulations to explore the parameter space and locate the global maximum of the posterior probability. However, slowing the annealing down at small values of β is not enough to ensure that the simulations reach the global maximum; it is still possible for simulations to become stuck in local maxima.

B.5 Killing Simulations

As noted, slowing down the annealing for small values of β works very well for giving the simulations time to find the global maxima. However, it is not enough, it is still possible for simulations to become stuck in a local maxima. These trapped simulations must be found and fixed as quickly as possible if the simulations are to reach a stationary point.

Up to now the Markov chain simulations have been described as having multiple chains running in parallel using simulated annealing with an annealing parameter set dynamically based on the standard deviation of the logarithm of the likelihood. Each step in the Markov chain proceeds roughly as follows, the value of the annealing parameter is computed and set for this step in the simulated annealing cycle. Prior to setting to the annealing parameter, simulations should be in a equilibrium. However, incriminating the annealing parameter throws the simulations out of equilibrium and because the simulations are out of equilibrium, we make a number of other modifications to the simulations, we adjust the rate of acceptance, and we kill off a number of simulations.

Between annealing steps, the algorithm doing the Markov chain simulation kills off low probability simulations. To do this the algorithm computes the logarithm of the posterior probability for each simulation. This table of logarithms is then indexed, sorted, and used to replace low probability simulations. In this step typically 10% of the simulations are replaced by higher probability simulations. The program simply takes the lowest probability simulation and then replaces the simulation by one of the simulations having higher probability. The higher probability replacement is chosen by drawing a random number from a Gaussian having a standard deviation that is roughly one third of the total simulations. So when a simulation is replaced, it is replaced by a higher probability simulation, but not necessarily the highest probability simulation.

B.6 the Proposal

When doing a Metropolis-Hastings Markov chain Monte Carlo simulation one must be careful in proposing new values of a parameter. If the current value of a parameter is M_0, one proposes a new parameter value M_1 as follows:

$$M_1 = M_0 + \delta M \tag{B.9}$$

where δM is the change that is being made to the parameter. The exact method one obtains this δM doesn't matter except for one propriety that must be enforced. If the probability of moving from M_0 to M_1 is given by $P(M_1|M_0I)$ then the probability of moving from M_1 to M_0 must be the same:

$$P(M_0|M_1I) = P(M_1|M_0I) \tag{B.10}$$

That is to say jumps in the proposal probability must be symmetric. There are countless modifications and addendum to this rule and you can look at the various references on Markov chain simulations on what these modifications are, but in the calculations implemented in this software package, Eq. (B.10) is the rule implemented using a simple Gaussian proposal. A Gaussian proposal has a number of advantages, for example it is symmetric in its argument and thus automatically satisfies Eq. (B.10). Additionally, a Gaussian has one additional parameter that is important, its standard deviation. By adjusting the size of the standard deviation of the proposal one can control how often the Markov chain transitions from one proposed value to another.

Now one might ask why this is important and the answer is simple, if the proposal it too small the simulation sill not explore the parameter space and if the proposal is too large, the change in the posterior probability will be so great that the probability of accepting the change is zero and again the simulations do now explore the parameter space. Consequently, it is important to monitor the size of the proposal and to adjust it between annealing steps to ensure it is neither too small or too large.

The way the programs that implement the Markov chain simulations control the proposal is by keeping track of the acceptance rate for a given parameter. The acceptance rate is simply the ratio of the number of times a proposed parameter was accepted divided by the total number of times one proposed a new parameter value. There is not hard and fast rule on how often one should accept a parameter, but too often or too little are both bad. Additionally, if one is to error, then err on the side of more exploration of the parameter space is probably a good thing. Consequently, the programs that implement these calculations try and keep the acceptance rate between 20 and 30%. If the acceptance rate falls below 20% the proposal is decreased and if the acceptance rate is above 30% the proposal is increased. If it is between 20 and 30% no change is made. There are many addendum that could be added to this description, but it captures what the program actually does. Indeed, there is an output report generated by most of the packages in the Bayesian Analysis Software called an accepted report and that report is available while a package is running and needles to say, its primary output is the current acceptance rate for the various parameters.

Bibliography

[1] Rev. Thomas Bayes (1763), "An Essay Toward Solving a Problem in the Doctrine of Chances," *Philos. Trans. R. Soc. London*, **53,** pp. 370-418; reprinted in *Biometrika*, **45**, pp. 293-315 (1958), and *Facsimiles of Two Papers by Bayes,* with commentary by W. Edwards Deming, New York, Hafner, 1963.

[2] G. Larry Bretthorst (1988), "Bayesian Spectrum Analysis and Parameter Estimation," in *Lecture Notes in Statistics,* **48**, J. Berger, S. Fienberg, J. Gani, K. Krickenberg, and B. Singer (eds), Springer-Verlag, New York, New York.

[3] G. Larry Bretthorst (1990), "An Introduction to Parameter Estimation Using Bayesian Probability Theory," in *Maximum Entropy and Bayesian Methods,* Dartmouth College 1989, P. Fougère *ed.,* pp. 53-79, Kluwer Academic Publishers, Dordrecht the Netherlands.

[4] G. Larry Bretthorst (1990), "Bayesian Analysis I. Parameter Estimation Using Quadrature NMR Models" *J. Magn. Reson.*, **88**, pp. 533-551.

[5] G. Larry Bretthorst (1990), "Bayesian Analysis II. Signal Detection And Model Selection" *J. Magn. Reson.*, **88**, pp. 552-570.

[6] G. Larry Bretthorst (1990), "Bayesian Analysis III. Examples Relevant to NMR" *J. Magn. Reson.*, **88**, pp. 571-595.

[7] G. Larry Bretthorst (1991), "Bayesian Analysis. IV. Noise and Computing Time Considerations," *J. Magn. Reson.*, **93**, pp. 369-394.

[8] G. Larry Bretthorst (1992), "Bayesian Analysis. V. Amplitude Estimation for Multiple Well-Separated Sinusoids," *J. Magn. Reson.*, **98**, pp. 501-523.

[9] G. Larry Bretthorst (1992), "Estimating The Ratio Of Two Amplitudes In Nuclear Magnetic Resonance Data," in *Maximum Entropy and Bayesian Methods,* C. R. Smith *et al.* (eds.), pp. 67-77, Kluwer Academic Publishers, the Netherlands.

[10] G. Larry Bretthorst (1993), "On The Difference In Means," in Physics & Probability Essays in honor of Edwin T. Jaynes, W. T. Grandy and P. W. Milonni (eds.), pp. 177-194, Cambridge University Press, England.

[11] G. Larry Bretthorst (1996), "An Introduction To Model Selection Using Bayesian Probability Theory," in *Maximum Entropy and Bayesian Methods*, G. R. Heidbreder, ed., pp. 1-42, Kluwer Academic Publishers, Printed in the Netherlands.

[12] G. Larry Bretthorst (1999), "The Near-Irrelevance of Sampling Frequency Distributions," in *Maximum Entropy and Bayesian Methods,* W. von der Linden *et al.* (eds.), pp. 21-46, Kluwer Academic Publishers, the Netherlands.

[13] G. Larry Bretthorst (2001), "Nonuniform Sampling: Bandwidth and Aliasing," in *Maximum Entropy and Bayesian Methods in Science and Engineering,* Joshua Rychert, Gary Erickson and C. Ray Smith *eds.*, pp. 1-28, American Institute of Physics, USA.

[14] G. Larry Bretthorst, Christopher D. Kroenke, and Jeffrey J. Neil (2004), "Characterizing Water Diffusion In Fixed Baboon Brain," in Bayesian Inference And Maximum Entropy Methods In Science And Engineering, Rainer Fischer, Roland Preuss and Udo von Toussaint *eds.*, AIP conference Proceedings, **735**, pp. 3-15.

[15] G. Larry Bretthorst, William C. Hutton, Joel R. Garbow, and Joseph J.H. Ackerman (2005), "Exponential parameter estimation (in NMR) using Bayesian probability theory," *Concepts in Magnetic Resonance*, 27A, Issue 2, pp. 55-63.

[16] G. Larry Bretthorst, William C. Hutton, Joel R. Garbow, and Joseph J. H. Ackerman (2005), "Exponential model selection (in NMR) using Bayesian probability theory," *Concepts in Magnetic Resonance*, 27A, Issue 2, pp. 64-72.

[17] G. Larry Bretthorst, William C. Hutton, Joel R. Garbow, and Joseph J.H. Ackerman (2005), "How accurately can parameters from exponential models be estimated? A Bayesian view," *Concepts in Magnetic Resonance*, 27A, Issue 2, pp. 73-83.

[18] G. Larry Bretthorst, W. C. Hutton, J. R. Garbow, and Joseph J. H. Ackerman (2008), "High Dynamic Range MRS Time-Domain Signal Analysis," *Magn. Reson. in Med.*, **62**, pp. 1026-1035.

[19] V. Chandramouli, K. Ekberg, W. C. Schumann, S. C. Kalhan, J. Wahren, and B. R. Landau (1997), "Quantifying gluconeogenesis during fasting," *American Journal of Physiology*, **273**, pp. H1209-H1215.

[20] R. T. Cox (1961), "The Algebra of Probable Inference," Johns Hopkins Univ. Press, Baltimore.

[21] André d'Avignon, G. Larry Bretthorst, Marlyn Emerson Holtzer, and Alfred Holtzer (1998), "Site-Specific Thermodynamics and Kinetics of a Coiled-Coil Transition by Spin Inversion Transfer NMR," *Biophysical Journal*, **74**, pp. 3190-3197.

[22] André d'Avignon, G. Larry Bretthorst, Marlyn Emerson Holtzer, and Alfred Holtzer (1999), "Thermodynamics and Kinetics of a Folded-Folded Transition at Valine-9 of a GCN4-Like Leucine Zipper," *Biophysical Journal*, **76**, pp. 2752-2759.

[23] David Freedman, and Persi Diaconis (1981), "On the histogram as a density estimator: L_2 theory," *Zeitschrift f¨r Wahrscheinlichkeitstheorie und verwandte Gebiete*, **57**, 4, pp. 453-476.

[24] W. R. Gilks, S. Richardson, and D. J. Spiegelhalter (1996), "Markov Chain Monte Carlo in Practice," Chapman & Hall, London.

[25] Paul M. Goggans, and Ying Chi (2004), "Using Thermodynamic Integration to Calculate the Posterior Probability in Bayesian Model Selection Problems," in *Bayesian Inference and Maximum Entropy Methods in Science and Engineering: 23rd International Workshop*, **707**, pp. 59-66.

[26] Marlyn Emerson Holtzer, G. Larry Bretthorst, D. André d'Avignon, Ruth Hogue Angelette, Lisa Mints, and Alfred Holtzer (2001), "Temperature Dependence of the Folding and Unfolding Kinetics of the GCN4 Leucine Lipper via 13C alpha-NMR," *Biophysical Journal*, **80**, pp. 939-951.

[27] E. T. Jaynes (1968), "Prior Probabilities," *IEEE Transactions on Systems Science and Cybernetics*, SSC-4, pp. 227-241; reprinted in [30].

[28] E. T. Jaynes (1978), "Where Do We Stand On Maximum Entropy?" in *The Maximum Entropy Formalism*, R. D. Levine and M. Tribus *Eds.*, pp. 15-118, Cambridge: MIT Press, Reprinted in [30].

[29] E. T. Jaynes (1980), "Marginalization and Prior Probabilities," in *Bayesian Analysis in Econometrics and Statistics*, A. Zellner *ed.,* North-Holland Publishing Company, Amsterdam; reprinted in [30].

[30] E. T. Jaynes (1983), "Papers on Probability, Statistics and Statistical Physics," a reprint collection, D. Reidel, Dordrecht the Netherlands; second edition Kluwer Academic Publishers, Dordrecht the Netherlands, 1989.

[31] E. T. Jaynes (1957), "How Does the Brain do Plausible Reasoning?" unpublished Stanford University Microwave Laboratory Report No. 421; reprinted in *Maximum-Entropy and Bayesian Methods in Science and Engineering* **1,** pp. 1-24, G. J. Erickson and C. R. Smith *Eds.,* 1988.

[32] E. T. Jaynes (2003), "Probability Theory—The Logic of Science," edited by G. Larry Bretthorst, Cambridge University Press, Cambridge UK.

[33] Sir Harold Jeffreys (1939), "Theory of Probability," Oxford Univ. Press, London; Later editions, 1948, 1961.

[34] John G. Jones, Michael A. Solomon, Suzanne M. Cole, A. Dean Sherry, and Craig R. Malloy (2001) "An integrated ^{2}H and ^{13}C NMR study of gluconeogenesis and TCA cycle flux in humans," *American Journal of Physiology, Endocrinology, and Metabolism,* **281,** pp. H848-H856.

[35] John Kotyk, N. G. Hoffman, W. C. Hutton, G. Larry Bretthorst, and J. J. H. Ackerman (1992), "Comparison of Fourier and Bayesian Analysis of NMR Signals. I. Well-Separated Resonances (The Single-Frequency Case)," *J. Magn. Reson.*, **98**, pp. 483–500.

[36] Pierre Simon Laplace (1814), "A Philosophical Essay on Probabilities," John Wiley & Sons, London, Chapman & Hall, Limited 1902. Translated from the 6th edition by F. W. Truscott and F. L. Emory.

[37] N. Lartillot, and H. Philippe (2006), "Computing Bayes Factors Using Thermodynamic Integration," *Systematic Biology*, **55** (2), pp. 195-207.

[38] D. Le Bihan, and E. Breton (1985), "Imagerie de diffusion in-vivo par rsonance," Comptes rendus de l'Acadmie des Sciences (Paris), **301** (15), pp. 1109-1112.

[39] N. R. Lomb (1976), "Least-Squares Frequency Analysis of Unevenly Spaced Data," *Astrophysical and Space Science*, **39**, pp. 447-462.

[40] T. J. Loredo (1990), "From Laplace To SN 1987A: Bayesian Inference In Astrophysics," in *Maximum Entropy and Bayesian Methods,* P. F. Fougere (ed), Kluwer Academic Publishers, Dordrecht, The Netherlands.

[41] Craig R. Malloy, A. Dean Sherry, and Mark Jeffrey (1988), "Evaluation of Carbon Flux and Substrate Selection through Alternate Pathways Involving the Citric Acid Cycle of the Heart by 13C NMR Spectroscopy," *Journal of Biological Chemistry*, **263** (15), pp. 6964-6971.

[42] Craig R. Malloy, Dean Sherry, and Mark Jeffrey (1990), "Analysis of tricarboxylic acid cycle of the heart using ^{13}C isotope isomers," *American Journal of Physiology*, **259**, pp. H987-H995.

[43] Lawrence R. Mead and Nikos Papanicolaou, "Maximum entropy in the problem of moments," *J. Math. Phys.* **25**, 2404–2417 (1984).

[44] K. Merboldt, Wolfgang Hanicke, and Jens Frahm (1969), "Self-diffusion NMR imaging using stimulated echoes," Journal of Magnetic Resonance, **64** (3), pp. 479-486.

[45] Nicholas Metropolis, Arianna W. Rosenbluth, Marshall N. Rosenbluth, Augusta H. Teller, and Edward Teller (1953), "Equation of State Calculations by Fast Computing Machines," Journal of Chemical Physics. The previous link is to the Americain Institute of Physics and if you do not have access to Science Sitations you many not be able to retrieve this paper.

[46] Radford M. Neal (1993), "Probabilistic Inference Using Markov Chain Monte Carlo Methods," technical report CRG-TR-93-1, Dept. of Computer Science, University of Toronto.

[47] Jeffrey J. Neil, and G. Larry Bretthorst (1993), "On the Use of Bayesian Probability Theory for Analysis of Exponential Decay Data: An Example Taken from Intravoxel Incoherent Motion Experiments," *Magn. Reson. in Med.*, **29**, pp. 642–647.

[48] H. Nyquist (1924), "Certain Factors Affecting Telegraph Speed," *Bell System Technical Journal*, **3**, pp. 324-346.

[49] H. Nyquist (1928), "Certain Topics in Telegraph Transmission Theory," *Transactions AIEE*, **3**, pp. 617-644.

[50] William H. Press, Saul A. Teukolsky, William T. Vetterling and Brian P. Flannery (1992), "Numerical Recipes The Art of Scientific Computing Second Edition," Cambridge University Press, Cambridge UK.

[51] Emanuel Parzen (1962), "On Estimation of a Probability Density Function and Mode," *Annals of Mathematical Statistics* **33**, 1065–1076

[52] Karl Pearson (1895), "Contributions to the Mathematical Theory of Evolution. II. Skew Variation in Homogeneous Material," *Phil. Trans. R. Soc. A* **186**, 343–326.

[53] Murray Rosenblatt, "Remarks on Some Nonparametric Estimates of a Density Function," *Annals of Mathematical Statistics* **27**, 832–837 (1956).

[54] Jeffery D. Scargle (1981), "Studies in Astronomical Time Series Analysis I. Random Process In The Time Domain," *Astrophysical Journal Supplement Series*, **45**, pp. 1-71.

[55] Jeffery D. Scargle (1982), "Studies in Astronomical Time Series Analysis II. Statistical Aspects of Spectral Analysis of Unevenly Sampled Data," *Astrophysical Journal*, **263**, pp. 835-853.

[56] Jeffery D. Scargle (1989), "Studies in Astronomical Time Series Analysis. III. Fourier Transforms, Autocorrelation Functions, and Cross-correlation Functions of Unevenly Spaced Data," *Astrophysical Journal*, **343**, pp. 874-887.

[57] Arthur Schuster (1905), "The Periodogram and its Optical Analogy," *Proceedings of the Royal Society of London*, **77**, p. 136-140.

[58] Claude E. Shannon (1948), "A Mathematical Theory of Communication," *Bell Syst. Tech. J.*, **27**, pp. 379-423.

[59] John E. Shore, and Rodney W. Johnson (1981), "Properties of cross-entropy minimization," *IEEE Trans. on Information Theory*, **IT-27,** No. 4, pp. 472-482.

[60] John E. Shore and Rodney W. Johnson (1980), "Axiomatic derivation of the principle of maximum entropy and the principle of minimum cross-entropy," *IEEE Trans. on Information Theory*, **IT-26** (1), pp. 26-37.

[61] Devinderjit Sivia, and John Skilling (2006), "Data Analysis: A Bayesian Tutorial," Oxford University Press, USA.

[62] Edward O. Stejskal and Tanner, J. E. (1965), "Spin Diffusion Measurements: Spin Echoes in the Presence of a Time-Dependent Field Gradient." Journal of Chemical Physics, **42** (1), pp. 288-292.

[63] D. G. Taylor and Bushell, M. C. (1985), "The spatial mapping of translational diffusion coefficients by the NMR imaging technique," Physics in Medicine and Biology, **30** (4), pp. 345-349.

[64] Myron Tribus (1969), "Rational Descriptions, Decisions and Designs," Pergamon Press, Oxford.

[65] P. M. Woodward (1953), "Probability and Information Theory, with Applications to Radar," McGraw-Hill, N. Y. Second edition (1987); R. E. Krieger Pub. Co., Malabar, Florida.

[66] Arnold Zellner (1971), "An Introduction to Bayesian Inference in Econometrics," John Wiley and Sons, New York.