UNIVERSIDAD CARLOS III DE MADRID

working papers

Working Paper 08-25
Statistics and Econometrics Series 07
May 2008

Departamento de Estadística
Universidad Carlos III de Madrid
Calle Madrid, 126
28903 Getafe (Spain)
Fax (34) 91 624-98-49

BAYESIAN NON-LINEAR MATCHING OF PAIRWISE MICROARRAY GENE EXPRESSIONS

J.M. Marin[1] and C. Nieto[2]

Abstract

In this paper, we present a Bayesian non-linear model to analyze matching pairs of microarray expression data. This model generalizes, in terms of neural networks, standard linear matching models. As a practical application, we analyze data of patients with Acute Lymphoblastic Leukemia and we find out the best neural net model that relates the expression levels of two types of cytogenetically different samples from them.

Keywords: MCMC computation, microarray data analysis, multidimensional scaling, neural network, spatial Bayesian methods.

1) Departamento de Estadística, Universidad Carlos III de Madrid,
e-mail: jmmarin @est-econ.uc3m.es

2) Departamento de Estadística e IO. III, Universidad Complutense de Madrid,
e-mail: tita@estad.ucm.es

Bayesian non-linear matching of pairwise microarray gene expressions

J.M. Marin[1] and C. Nieto[2]

1: Dep. de Estadística, U. Carlos III, Getafe

2: Dep. de Estadística e I.O. III, U. Complutense, Madrid

Abstract

In this paper, we present a Bayesian non-linear model to analyze matching pairs of microarray expression data. This model generalizes, in terms of neural networks, standard linear matching models. As a practical application, we analyze data of patients with Acute Lymphoblastic Leukemia and we find out the best neural net model that relates the expression levels of two types of cytogenetically different samples from them.

Keywords: MCMC computation, microarray data analysis, multidimensional scaling, neural network, spatial Bayesian methods.

1 Introduction

In Shape and Procrustes Analysis a typical problem is how to match two or more configurations of labelled points or landmarks (see Dryden and Mardia (1998)) by applying a geometrical transformation. In Bioinformatics, Green and Mardia (2006) studied the problem of matching two configurations under a linear Bayesian hierarchical model, and Marin and Nieto (2008) derived a generalization of this problem in terms of multiple configurations. In Chemoinformatics, Dryden et al. (2007) considered the problem of matching unlabelled point sets under a linear application when two configurations of points were compared, one being treated as a perturbation of the other one.

In this paper, we consider a non-linear model consisting of a linear term plus a neural network, between two configurations of points with applications in microarray data analysis. This is a very general approach that enables to consider complex relationships among different levels of expression data. We consider a Bayesian methodology in order to estimate the corresponding parameters of the model.

In Section 2 and Subsection 2.1 we present the problem of matching two configurations by using a non-linear model, under normality, and we derive the posterior distributions of the parameters therein.

In Section 3 we apply previous results in the case of comparing expression levels of a selected group of genes involved in patients with Acute Lymphoblastic Leukemia, based on data derived by Chiaretti et al. (2004).

Finally, in Section 4, we discuss future lines of researching and further developments.

2 Non-linear relations between configurations of points

Let consider two configurations of n matched points in $\mathbb{R}^d$, $\mathbf{x} = x_i$ $(i = 1, \ldots, n)$ and $\mathbf{y} = y_i$ $(i = 1, \ldots, n)$ where one of the configurations is considered fixed, e.g. $\mathbf{x}$, and the other is a random perturbed version of the previous one. If the relation between configurations cannot be expressed in terms of translations or rotations, we can consider a more general non-linear model consisting of a linear model plus a one-layer feed forward neural network,

$$\begin{aligned} y_{ij} &= \beta_{j0} + \lambda_j^T x_i + \sum_{k=1}^{M} \beta_{jk} \Psi(\gamma_{k0} + x_i^T \gamma_k) + \varepsilon_{ij} \\ i &= 1, \ldots, n \qquad j = 1, \ldots, d \end{aligned}$$

where, for all $i = 1, \ldots n$, $j = 1, \ldots, d$ and $k = 1, \ldots, M$, the parameters of the model are $\beta_{j0} \in \mathbb{R}$, $\lambda_j \in \mathbb{R}^d$, $\beta_{jk} \in \mathbb{R}$, $\gamma_{k0} \in \mathbb{R}$ and $\gamma_k \in \mathbb{R}^d$. Errors ε_{ij} are independently distributed with a given density f_i and $\Psi(\cdot)$ is the activation function that may be a logistic function (see for a revision Lee (2004)).

In this model each coordinate of point y_i is related with each point x_i $(i = 1, \ldots, n)$ by means a linear part plus a non linear one in terms of one-hidden layer neural network. Note that the number of hidden nodes, M, may be considered as an unknown parameter.

Alternatively, the model may be expressed in a matrix form as

$$y_i = \beta_{00} + \Lambda x_i^T + B\mathbf{\Psi}(\gamma_0 + x_i^T \gamma) + \varepsilon_i \qquad i = 1, \ldots, n$$

where

$$\underset{d\times 1}{\beta_{00}} = \begin{pmatrix} \beta_{10} \\ \vdots \\ \beta_{d0} \end{pmatrix}, \quad \underset{M\times 1}{\mathbf{\Psi}(\gamma_0 + x_i^T \gamma)} = \begin{pmatrix} \Psi(\gamma_{10} + x_i^T \gamma_1) \\ \vdots \\ \Psi(\gamma_{M0} + x_i^T \gamma_M) \end{pmatrix},$$

$$\underset{d\times M}{B} = \begin{pmatrix} \beta_{11} & \cdots & \beta_{1M} \\ \vdots & \ddots & \vdots \\ \beta_{d1} & \cdots & \beta_{dM} \end{pmatrix}, \quad \underset{d\times d}{\Lambda} = \begin{pmatrix} \lambda_{11} & \cdots & \lambda_{1d} \\ \vdots & \ddots & \vdots \\ \lambda_{d1} & \cdots & \lambda_{dd} \end{pmatrix}$$

2.1 Bayesian inference

In this Section, we shall assume that the number of hidden nodes is known.

If normal errors are supposed, the likelihood is

$$L(\beta_{00}, \Lambda, B, \gamma_0, \gamma, \sigma^2 | \mathbf{y}) \propto \frac{1}{(\sigma^2)^{nd/2}} \exp\left\{ -\frac{1}{2\sigma^2} \sum_{i=1}^{n} \sum_{j=1}^{d} (y_{ij} - \beta_{j0} - \lambda_j^T x_i - \right.$$
$$\left. - \sum_{k=1}^{M} \beta_{jk} \Psi(\gamma_{k0} + x_i^T \gamma_k))^2 \right\}.$$

In $\mathbb{R}^2$ the model is simplified as

$$L(\beta_{00}, \Lambda, B, \gamma_0, \gamma, \sigma^2|\mathbf{y}) \propto (\frac{1}{\sigma^2})^n \exp\left\{-\frac{1}{2\sigma^2}\left[\sum_{i=1}^{n}(y_{i1} - \beta_{10} - \lambda_1^T x_i - \right.\right.$$

$$\left.\left. - \sum_{k=1}^{M} \beta_{1k}\Psi(\gamma_{k0} + x_i^T\gamma_k))^2 + \sum_{i=1}^{n}(y_{i2} - \beta_{20} - \lambda_2^T x_i - \sum_{k=1}^{M}\beta_{2k}\Psi(\gamma_{k0} + x_i^T\gamma_k))^2\right]\right\}.$$

In order to carry out Bayesian inference, we must first introduce the prior distributions for the parameters of the model assuming a relatively diffuse prior distribution structure.

Let assume that the prior distribution of σ^2 is inverted-gamma $IG(\alpha, \beta)$, then the posterior distribution of σ^2 is

$$p(\sigma^2 \mid \mathbf{y}, \cdots) \propto (\sigma^2)^{-(\alpha+1)} \exp\left\{-\frac{\beta}{\sigma^2}\right\} \cdot (\frac{1}{\sigma^2})^n \exp\left\{-\frac{1}{2\sigma^2}\left[\sum_{i=1}^{n}(y_{i1} - \beta_{10} - \lambda_1^T x_i - \right.\right.$$

$$\left.\left. - \sum_{k=1}^{M}\beta_{1k}\Psi(\gamma_{k0} + x_i^T\gamma_k))^2 + \sum_{i=1}^{n}(y_{i2} - \beta_{20} - \lambda_2^T x_i - \sum_{k=1}^{M}\beta_{2k}\Psi(\gamma_{k0} + x_i^T\gamma_k))^2\right]\right\} \propto$$

$$(\sigma^2)^{-(\alpha+1+n)} \exp\left\{-\frac{1}{\sigma^2}(\beta + A)\right\},$$

where

$$\begin{aligned} A &= \frac{1}{2}\left[\sum_{i=1}^{n}(y_{i1} - \beta_{10} - \lambda_1^T x_i - \sum_{k=1}^{M}\beta_{1k}\Psi(\gamma_{k0} + x_i^T\gamma_k))^2 \right. \\ &\left. + \sum_{i=1}^{n}(y_{i2} - \beta_{20} - \lambda_2^T x_i - \sum_{k=1}^{M}\beta_{2k}\Psi(\gamma_{k0} + x_i^T\gamma_k))^2\right]. \end{aligned}$$

Hence, the posterior distribution is also inverted gamma, $\sigma^2 \mid \mathbf{y}, \cdots \sim IG(\alpha + n, \beta + A)$.

Let assume that prior distributions of $\beta_{j0} \sim N(\mu_{\beta_{j0}}, \sigma^2_{\beta_{j0}})$, with $j = 1, 2$. Then, the posterior distribution is

$$\beta_{j0} \mid \mathbf{y}, \cdots \sim N(\frac{D_{j0}}{2C_{j0}}, C_{j0}^{-1})$$

where

$$C_{j0} = \tfrac{1}{\sigma^2_{\beta_{j0}}} + \tfrac{n}{\sigma^2}; \qquad D_{j0} = \tfrac{2\mu_{\beta_{j0}}}{\sigma^2_{\beta_{j0}}} + \tfrac{2\sum_{i=1}^{n} R_{ij0}}{\sigma^2};$$

$$R_{ij0} = y_{ij} - \lambda_j^T x_i - \sum_{k=1}^{M}\beta_{jk}\Psi(\gamma_{k0} + x_i^T\gamma_k).$$

Let assume that the prior distribution of λ_{rs}, for $r = 1, 2$ and $s = 1, 2$ is $\lambda_{rs} \sim N(\mu_{\lambda_{rs}}, \sigma^2_{\lambda_{rs}})$; then, the posterior distribution is

$$\lambda_{rs} \mid \mathbf{y}, \cdots \sim N(\frac{D_{rs}}{C_{rs}}, C_{rs}^{-1})$$

where

$$C_{rs} = \frac{\sum\limits_{i=1}^{n} x_{is}^2}{\sigma^2} + \frac{1}{\sigma^2_{\lambda_{rs}}}; \qquad D_{rs} = \frac{\sum\limits_{i=1}^{n} M_{irs} x_{is}}{\sigma^2} + \frac{\mu_{\lambda_{rs}}}{\sigma^2_{\lambda_{rs}}};$$
$$M_{irs} = y_{ir} - \beta_{r0} - \lambda_{rs} x_{is} - \sum_{k=1}^{M} \beta_{rk} \Psi(\gamma_{k0} + x_i^T \gamma_k).$$

Let assume, for $j = 1, 2$ and $s = 1, \ldots, M$, that the prior distribution of β_{js}, is $\beta_{js} \sim N(\mu_{\beta_{js}}, \sigma^2_{\beta_{js}})$; then, the posterior distribution is

$$\beta_{js} \mid \mathbf{y}, \cdots \sim N(\frac{D_j}{C_j}, C_j^{-1})$$

where

$$\begin{aligned}
C_j &= \frac{\sum\limits_{i=1}^{n} \Psi^2(\gamma_{s0} + x_i^T \gamma_s)}{\sigma^2} + \frac{1}{\sigma^2_{\beta_{js}}}; \\
D_j &= \frac{\sum\limits_{i=1}^{n} R_{ij} \Psi(\gamma_{s0} + x_i^T \gamma_s)}{\sigma^2} + \frac{\mu_{\beta_{js}}}{\sigma^2_{\beta_{js}}}; \\
R_{ij} &= y_{ij} - \beta_{j0} - \lambda_j^T x_i - \sum_{\substack{k=1 \\ k \neq s}}^{M} \beta_{jk} \Psi(\gamma_{k0} + x_i^T \gamma_k).
\end{aligned}$$

Let assume that the prior distribution of γ_{rs}, for $r = 1, \ldots, M$ and $s = 0, 1, 2$ is $\gamma_{rs} \sim N(\mu_{\gamma_{rs}}, \sigma^2_{\gamma_{rs}})$; then, the posterior distribution is

$$p(\gamma_{rs} \mid \mathbf{y}, \cdots) \propto \exp\left\{ -\frac{1}{2\sigma^2} \left[\sum_{i=1}^{n} (y_{i1} - \beta_{10} - \lambda_1^T x_i - \sum_{\substack{k=1 \\ k \neq r}}^{M} \beta_{1k} \Psi(\gamma_{k0} + x_i^T \gamma_k) - \right.\right.$$
$$- \beta_{1r} \Psi(\gamma_{r0} + x_i^T \gamma_r))^2 + \sum_{i=1}^{n} (y_{i2} - \beta_{20} - \lambda_2^T x_i -$$
$$\left.\left. - \sum_{\substack{k=1 \\ k \neq r}}^{M} \beta_{2r} \Psi(\gamma_{k0} + x_i^T \gamma_k) - \beta_{2r} \Psi(\gamma_{r0} + x_i^T \gamma_r))^2 \right] \right\} \cdot \exp\left\{ -\frac{1}{2\sigma^2_{\gamma_{rs}}} (\gamma_{rs}^2 - 2\mu_{\gamma_{rs}} \gamma_{rs}) \right\}$$
$$\propto \exp\left\{ -\frac{1}{2\sigma^2_{\gamma_{rs}}} (\gamma_{rs}^2 - 2\mu_{\gamma_{rs}} \gamma_{rs}) \right\} \cdot \exp\left\{ -\frac{1}{2\sigma^2} \left[\sum_{i=1}^{n} (R_{i1} - \beta_{1r} \Psi(\gamma_{r0} + x_i^T \gamma_r))^2 + \right.\right.$$
$$\exp\left\{ -\frac{1}{2\sigma^2} \left[\sum_{i=1}^{n} (R_{i1} - \beta_{1r} \Psi(\gamma_{r0} + x_i^T \gamma_r))^2 + \sum_{i=1}^{n} (R_{i2} - \beta_{2r} \Psi(\gamma_{r0} + x_i^T \gamma_r))^2 \right] \right\},$$

where, for all $i = 1, \ldots, n$ and $j = 1, 2$,

$$R_{ij} = y_{ij} - \beta_{j0} - \lambda_j^T x_i - \sum_{\substack{k=1 \\ k \neq r}}^{M} \beta_{jk} \Psi(\gamma_{k0} + x_i^T \gamma_k).$$

Given these conditional distributions, we can apply an MCMC algorithm to simulate a sample from the joint posterior distribution. All conditional distributions are normal and inverted gamma, except that of parameter γ. Hence, we can introduce a Metropolis-Hasting step, inside a Gibbs sampling algorithm, with a normal as proposal distribution. As an alternative to standard Gibbs sampling, Wong and Liang (1997) proposed a *simulated tempering with dynamic weighting* (STWD) algorithm to deal with training of neural networks.

The size of the neural net, namely, the number of hidden nodes M can be considered as a parameter. In this case, it may be applied a reversible jump methodology to explore among spaces of different dimension (see e.g. Muller and Rios-Insua (1998) and Andrieu et al. (2001)). Nevertheless, in complex models like neural nets, problems of identifiability may appear and other methods may be considered (see Titterington (2004) for a revision). As a measure of complexity and fit of models, we have used a modified version of a DIC coefficient as pointed out by Richardson (2002), and Celeux et al. (2006) who labeled this version as DIC$_3$. This criterion is well adapted to the case of neural networks with different possible architecture. The expression of this coefficient is

$$\text{DIC}_3 = -4E_{\theta|\mathbf{y}}[\log f(\mathbf{y} \mid \theta)] + 2 \log \hat{f}(\mathbf{y}),$$

where $\hat{f}(\mathbf{y}) = \prod_{i=1}^{n} \hat{f}(y_i)$, and $\hat{f}(y_i) = E_{\theta|\mathbf{y}}[f(y_i \mid \theta)]$.

3 Application in microarray data expressions of Lymphoblastic Leukemia

We have chosen a microarray experiment, depicted in the database *ALL* from the BioConductor bundle (see Gentleman et al. (2004b)), originally rendered by Chiaretti et al. (2004). We have taken a subset of 79 samples representing patients with B-cell acute lymphoblastic leukemia, and we have considered, first, the samples with BCR/ABL fusion gene (37 patients) and then, the cytogenetically normal samples (42 patients).

We sought a group of genes with different expression levels in above samples. For that, we have used the genefilter package of the BioConductor bundle (Gentleman et al. (2004) and Gentleman et al. (2004b)) and we have selected 2391 probesets from the original 12625 ones for our analysis, deleting those with no expression or low variability across all samples. Then, by using the multtest package, we have considered the criterion provided by the false discovery rate (FDR), that is, the expected proportion of false positives among the genes that are significant. We have used the procedure of Benjamin and Hochberg (1995) to control the FDR at a level of 0.05. After this procedure, we have obtained 102 significantly different expressed genes.

As a previous step we have projected the selected genes in a two-dimensional map in order to determine which function relates them. For that, we have computed the Euclidean distances between pairs of genes in each group and we have built a map by using an *INDSCAL* Analysis (see for references therein e.g. Borg and Groenen (2005) and Marin and Nieto (2008)). In order to program the *INDSCAL* Analysis we have used software `SAS` v. 9.1 running in a Pentium IV, 3.2 Ghz. processor.

In figures 1 and 2 we show the positions of genes in the two groups of patients, after applying the *INDSCAL* Analysis. In these figures, numbers identify two specific genes in order to visualize their situation in the space. We will consider the cytogenetically normal group as the fixed configuration of points.

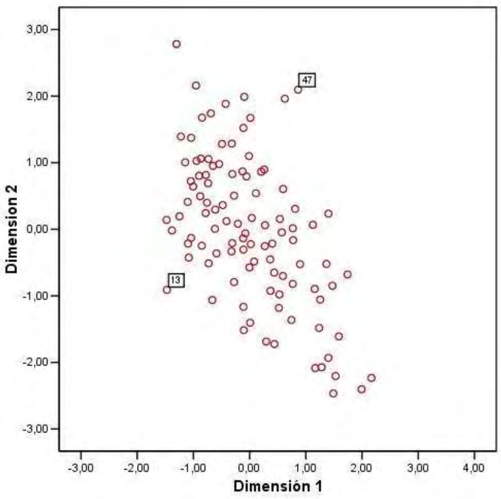

Figure 1: Group cytogenetically normal

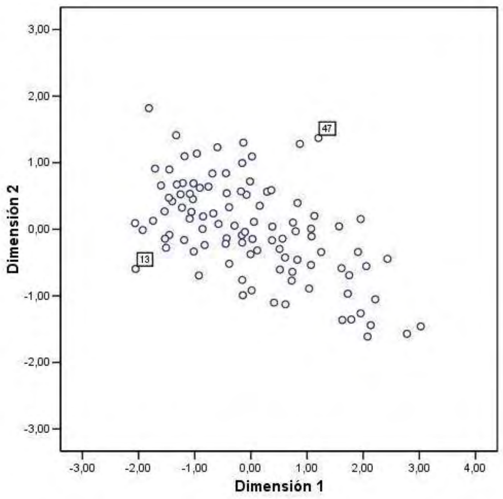

Figure 2: Group with the BCR/ABL fusion

In order to program model 2.1 we have used as software `WinBugs` (see Lunn et al. (2000)) running in a Pentium IV, 3.2 Ghz. processor. We have used, as prior distributions of $\beta_{00}, \Lambda, B, \gamma_0$ and γ, normal distributions with zero means and high variances. For parameter σ^2 we have used a prior inverted gamma distribution with high variance also. All programs are disposable by request to the corresponding author.

In table 1 we show DIC$_3$ values for different number of hidden nodes M, after a total of 100000 iterations with 50000 iterations for burn-in. Lowest value is obtained for $M = 1$.

M	**1**	**2**	**3**	**4**
DIC$_3$	-1494.565	-1493.17	-1492.177	-1488.328
M	**5**	**6**	**7**	**8**
DIC$_3$	-1486.23	-1486.049	-1484.409	-1484.714
M	**9**	**10**		
DIC$_3$	-1483.864	-1481.183		

Table 1: DIC$_3$ values vs number of hidden nodes M

We derive the posterior distributions of the parameters of the $M = 1$ model. We test the results by dividing the sample of 102 genes, in a random training set with 80% of observations and a test set with the resting ones. We considered a total of 600000 iterations, with 300000 iterations for burn-in, and three different chains with random initialization points. *Post hoc* analysis of chains showed that there were not a significant departure from convergence. Results of predictions and actual values of the validation set are in table 2, showing that predictions are accurate.

Actual values	(-0.8526 , 0.1925)	(0.2886 , 0.5641)
Estimated values	(-0.8524 , 0.1918)	(0.2885 , 0.5623)
Actual values	(1.5697 , 0.0429)	(1.0721 , 0.0091)
Estimated values	(1.5700 , 0.0417)	(1.0720 , 0.0077)
Actual values	(0.0537 , 0.1107)	(2.2148 , -1.0531)
Estimated values	(0.0538 , 0.1089)	(2.2150 , -1.0540)
Actual values	(0.5167 , -0.6055)	(1.9068 , -0.3414)
Estimated values	(0.5168 , -0.6073)	(1.9070 , -0.3424)
Actual values	(-0.2918 , 0.0534)	(2.4316 , -0.4448)
Estimated values	(-0.2913 , 0.0519)	(2.4320 , -0.4459)
Actual values	(-1.4535 , -0.0849)	(0.8263 , -0.4591)
Estimated values	(-1.4530 , -0.0858)	(0.8264 , -0.4605)
Actual values	(1.7875 , -1.3557)	(-1.7069 , 0.9099)
Estimated values	(1.7880 , -1.3570)	(-1.7070 , 0.9091)
Actual values	(-0.4324 , -0.1370)	(-1.4613 , 0.4721)
Estimated values	(-0.4321 , -0.1383)	(-1.4610 , 0.4714)
Actual values	(-1.2277 , 0.3238)	(-0.5963 , 1.2311)
Estimated values	(-1.2280 , 0.3230)	(-0.5964 , 1.2300)
Actual values	(1.6281 , -1.3630)	(-1.5291 , -0.1417)
Estimated values	(1.6280 , -1.3640)	(-1.5290 , -0.1424)

Table 2: Predictions and actual values of validation set

4 Discussion and future developments

The proposed model can be useful to relate different expression levels in two microarrays, corresponding to different conditions in a group of genes. Moreover, this model can be useful to find relations among expressions, by selecting some critical genes, or to assess diagnostic forecasting by comparing positions of genes in different times of a given illness.

One possible generalization may be to consider multiple comparisons among more than two microarrays. As a technical issue, in order to deal with non-linear matching problems, it may be interesting to introduce the class of Gaussian Processes that generalize neural nets models.

References

Andrieu, C., de Freitas, N. and Doucet, A. (2001). Robust Full Bayesian Learning for Radial Basis Networks. *Neural Computation* **13**(10), 2359–2407.

Benjamini, Y. and Hochberg, Y. (1995). Controlling the false discovery rate: a practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society, Series B*, **57**, 289–300.

Borg, I. and Groenen, P. J. F. (2005). *Modern Multidimensional Scaling.* Springer, New York.

Celeux, G., Forbes, F., Robert, C. P. and Titterington, D. M. (2006). Deviance Information Criteria for Missing Data Models. *Bayesian Analysis* **1**(4), 651–674.

Chiaretti, V., Li, X., Gentleman, R., Vitale, A., Vignetti, M., Mandelli, F., Ritz, J. and Foa, R. (2004). Gene expression profile of adult T-cell acute lymphocytic leukemia identifies distinct subsets of patients with different response to therapy and survival. *Blood*, **103**(7), 2771–2778.

Dryden, I. L., Hirst, J. D. and Melville,J.L. (2007). Statistical analysis of unlabelled point sets: comparing molecules in chemoinformatics. it Biometrics **63**(1), 237–251.

Dryden, I. L. and Mardia, K. V. (1998). *Statistical shape analysis.* Wiley, Chichester.

Gentleman, R., Carey, V., Huber, W., Irizarry, R. and Dudoit, S. (Eds.) (2004). *Bioinformatics and Computational Biology Solutions Using R and Bioconductor.* Springer, New York.

Gentleman, R., Carey, V., Bates, D., Bolstad, B., Dettling, M., Dudoit, S., Ellis, B., Gautier, L., Ge, Y., Gentry, J., Hornik, K., Hothorn, T., Huber, W., Iacus, S., Irizarryand, R., Leisch, F., Li, C., Maechler, M., Rossiniand, A., Sawitzki, G., Smith, C., Smyth, G., Tierneyand, L., Yang, J. and Zhang, J. (2004b). Bioconductor: Open software development for computational biology and bioinformatics. *Genome Biology* **5**, R80. url = http://www.bioconductor.org

Green, P. J. and Mardia, K. V. (2006). Bayesian alignment using hierarchical models with applications in protein Bioinformatics. *Biometrika* **93**(2), 235–254.

Lee, H. K. H. (2004). *Bayesian nonparametrics via neural networks.* Asa-Siam, Philadelphia.

Lunn, D. J., Thomas, A., Best, N. and Spiegelhalter, D. (2000). WinBUGS – a Bayesian modelling framework: concepts, structure, and extensibility. *Statistics and Computing* **10** 325–337.

Marin, J. M. and Nieto, C. (2008). Spatial Matching of Multiple Configuratios of Points with a Bioinformatics Application. *Comm. in Stat. T. and Meth.* **37**, 1977–1995.

Muller, P. and Rios-Insua, D. (1998). Issues in Bayesian Analysis of Neural Network Models. *Neural Computation* **10**(3), 749–770.

Richardson, S. (2002). Discussion of Spiegelhalter et al. *Journal of the Royal Statistical Society, Series B* 631.

Titterington, D. M. (2004). Bayesian Methods for Neural Networks and Related Models. *Statistical Science* **19**(1), 128–139.

Wong, W. H. and Liang, F. M. (1997). Dynamic weighting in Monte Carlo and optimization. *Proceedings of the National Academy of Sciences of USA* **94**(26), 14220–14224.