A NEURAL-BAYESIAN APPROACH FOR PREDICTING SMEAR NEGATIVE PULMONARY TUBERCULOSIS

Alcione M. dos Santos*, Aristófanes C. Silva†, Leonardo de O. Martins†

* *Universidade Federal do Maranhão - UFMA*
Departamento de Saúde Pública
Rua Barão de Itapary,155, Centro
CEP: 65020 - 070, São Luís-MA,Brasil

† *Universidade Federal do Maranhão - UFMA*
Departamento de Engenharia e Eletricidade
Av. dos Portugueses, SN, Campus do Bacanga, Bacanga
65085-580, São Luís, MA, Brasil

Emails: alcione.miranda@terra.com.br, ari@dee.ufma.br, leonardo@gia.deinf.ufma.br

Abstract— This work analyzes the application Bayesian neural network for predicting Smear Negative Pulmonary Tuberculosis (SNPT). The data used for developing the proposed model comprised of one hundred and thirty-six patients from Health Care Units. They were referred to the University Hospital in Rio de Janeiro, Brazil, from March 2001 to September 2002, with clinical-radiological suspicion of smear negative pulmonary tuberculosis. Only symptoms and physical signs were used for constructing the Bayesian neural network modelling, which was able to correctly classify 82% of patients from a test sample.

Keywords— Neural networks, Bayesian learning, Markov chain Monte Carlo, medical diagnosis.

Resumo— Este trabalho analisa a aplicação de rede neural Bayesiana para predizer o diagnóstico de tuberculose pulmonar paucibacilar. Os dados utilizados para desenvolvimento do modelo neural referem-se ao estudo de pacientes atendidos no Hospital Universitário Clementino Fraga Filho da Universidade Federal do Rio de Janeiro, que apresentaram suspeita de TB pulmonar e com resultado de baciloscopia negativa, no período de março de 2001 a setembro de 2002. Apenas sintomas e sinais físicos foram usados para construção da rede neural Bayesiana, a qual foi capaz de classificar corretamente 82% dos pacientes da amostra de teste.

Palavras-chave— Redes Neurais Artificiais, Aprendizado Bayesiano, Método Monte Carlo via Cadeia de Markov, Diagnóstico Médico.

1 Introduction

During the last years, the application of artificial neural networks (ANN) for prognostic and diagnostic classification in clinical medicine has attracted growing interest in the medical literature (Edwards et al., 1999; El-Solh et al., 1999; Tu, 1996).

(Santos, 2003) uses neural networks and classification trees to identify patients with clinical-radiological suspicion of smear negative pulmonary tuberculosis. (Kiyan and Yildirim, 2003) employed radial basis function, general regression neural network and probabilistic neural network in order to breast cancer diagnosis. This work also indicates that statistical neural networks can be effectively used for breast cancer diagnosis to help oncologists.

An attractive feature of ANN is their ability to model highly complex non-linear relationships in data (Bishop, 1996). Recently, Bayesian methods have been proposed for neural networks to solve regression and classification problems (Lampinen and Vehtari, 2001). These methods claim to overcome some difficulties encountered in the standard approach such as overfitting.

The Bayesian neural networks (BNN) can provide us not only the means of predictive weights but also their uncertainties. Another advantage of BNN is the appropriate choices of a number of hidden layers and their dimensions (Rios Insua and Müller, 1998).

In this work, we use Bayesian neural networks for developing and evaluating a prediction model for diagnosing SNPT, useful for patients attended in Health Care Units within areas of limited resources. In addition, we analyze the information that is extracted by the model and compare it to expert analysis for diagnosis. This analysis aims at helping doctors to understand the way the model works and to make them more confident in its practical application.

2 Bayesian Neural Networks

In this paper, we focus exclusively on feedforward networks with a single hidden layer with m nodes and k outputs, besides do not allow direct connections from the inputs and outputs. The particular form of the neural model we will work is

$$f_k(\mathbf{x}, \mathbf{w}) = \phi_k\{w_{k0} + \sum_{j=1}^{m} w_{kj}\phi_j(w_{j0} + \sum_{i=1}^{p} w_{ij}x_i)\} \quad (1)$$

where $\mathbf{x}$ is the input vector with p explanatory variables, $\mathbf{x_i}$ and ϕ represents the activation function and the set of all weights (parameters), represented by the vector $\mathbf{w}$, including input-hidden weights, biases and hidden-output weights. If the neural model is used for classification problem, the output $f_k(\mathbf{x}, \mathbf{w})$ is the final value used for classification process.

In the Bayesian approach to learning neural network (Buntine and Weigend, 1991; Mackay, 1991; Mackay, 1992), the objective is to find the weights posterior distribution mode. To obtain the posterior distribution of the weights, we need to specify the prior distribution, which is a probability distribution that represents the prior information associated with the weights of the network, and the data likelihood. Firstly, we will discuss how we choose the prior distribution of the weights.

2.1 Prior Distribution for Neural Networks

Many implementations of Bayesian neural networks use Gaussian distribution, with zero mean and some specified width, as the priors for all weights and biases in the network. To specify the prior distributions, the weights were divided into three separate groups: bias terms, input to hidden weights and hidden to output weights.

We consider a Gaussian prior with zero mean and unknown variance $1/\lambda_\alpha$ for the input to hidden weights, where λ_α is a *precision parameter*. Instead of fixing the λ_α value, we regard it as another parameter. We would then call it a hyperparameter to separate it from weights and biases. Now, we need to specify a *hyperprior* distribution for λ_α.

Although there are several ways to implement the required hyperprior, we choose a Gamma hyperprior, with mean and specified shape parameter (Berger, 1985). This process can be extended, where each input weight have different priors and hyperpriors. This process is called *Automatic Relevance Determination* (ARD) (Neal, 1996). Using this prior distribution, it is possible determine the relative importance of the different inputs. The relevance of each input is considered to be inversely proportional to the variance of this distribution.

In order, the prior distribution for hidden to output weights was also considered Gaussian with zero mean and unknown variance λ_β. We use Gamma hyperprior with mean and specified shape to hyperparameter λ_β.

Finally, all biases terms are then assumed to be distributed according to a gaussian prior with mean zero and variance λ_γ, where the Gamma hyperprior, with mean and specified shape, was again used to the hyperparameter η_γ. To facilitate the notation, let's denote a set of hyperparameters $\varphi = (\lambda_\alpha, \lambda_\beta, \lambda_\gamma)$.

Once we have chosen the prior distributions, we combine the evidence from the data to get the posterior distribution for the parameters and hyperparameters.

2.2 Likelihood Function and Posterior Distribution

We assume that we have a data set consisting of n input vectors $\mathbf{x_1}, \ldots, \mathbf{x_n}$ and the corresponding target y_i. For classification problems with two classes, it is known that under conditions the output $f_k(\mathbf{x}, \mathbf{w})$ can be interpreted as the probability that a target y_i belongs to a certain class. For the specific problem at hand, the BNN is used to find the probability of the patient is with active pulmonary tuberculosis (PT).

Assuming that $(\mathbf{x_1}, \mathbf{y_1}), \ldots, (\mathbf{x_n}, \mathbf{y_n})$ are independents and identically distributed according to Bernoulli distribution, we have the following likelihood function for the training data

$$P(\mathcal{D} \mid \mathbf{w}) = \prod_{\mathbf{i=1}}^{\mathbf{n}} \mathbf{f_k}(\mathbf{x_i}, \mathbf{w})^{\mathbf{y_i}} [\mathbf{1} - \mathbf{f_k}(\mathbf{x_i}, \mathbf{w})]^{\mathbf{1-y_i}} \quad (2)$$

After observing the data, using Bayes theorem and likelihood, prior distribution is updated to the posterior distribution

$$\begin{aligned} P(\mathbf{w}, \varphi \mid \mathcal{D}) &= \frac{P(\mathcal{D} \mid \mathbf{w}, \varphi)\mathbf{P}(\mathbf{w}, \varphi)}{P(\mathcal{D})} \\ &= \frac{P(\mathcal{D} \mid \mathbf{w}, \varphi)\mathbf{P}(\mathbf{w} \mid \varphi)\mathbf{P}(\varphi)}{\int\int P(\mathcal{D} \mid \mathbf{w}, \varphi)\mathbf{P}(\mathbf{w}, \varphi)\mathbf{dwd}\varphi} \end{aligned} \quad (3)$$

The denominator in the Equation 3 is sometimes called *normalizing constant* that ensures that the posterior distribution integral is equal to one. This constant can be ignored since it is irrelevant to the first level of inference. Therefore, the theorem may also be written as

$$P(\mathbf{w}, \varphi \mid \mathcal{D}) \propto P(\mathcal{D} \mid \mathbf{w}, \varphi)\mathbf{P}(\mathbf{w} \mid \varphi)\mathbf{P}(\varphi) \quad (4)$$

Given a training data, to find the weight vector $\mathbf{w}^*$, corresponding to the maximum of the posterior distribution, is equivalent to minimize the error-function $E(\mathbf{w})$, which is given by

$$E(\mathbf{w}) = -\ln \mathbf{P}(\mathcal{D} \mid \mathbf{w}) + \ln \mathbf{P}(\mathbf{w}, \varphi) \quad (5)$$

where $P(\mathcal{D} \mid \mathbf{w})$ is the likelihood function presented in Equation 2.

In the Bayesian learning, the posterior distribution is used to find the predictive distributions for the target values in the new test case given the inputs for that case as well as inputs and target for training cases (Ghosh et al., 2004). To predict the new output y_{n+1} for new input $\mathbf{x_{n+1}}$, predictive

distribution is obtained by integrating the predictions of the model with respect to the posterior distribution of the model parameters

$$P(y_{(n+1)} \mid \mathbf{x_{(n+1)}}, \mathcal{D}) = \int\int P(y_{(n+1)} \mid \mathbf{x_{(n+1)}}, \mathbf{w}, \varphi)\mathbf{P}(\mathbf{w}, \varphi \mid \mathcal{D})\mathbf{dwd}\varphi \quad (6)$$

The posterior distribution presented in Equation 4 is very complex and its evaluation requires high-dimensional numerical integration, then it is impossible to compute it exactly. In order, to make this integral analytically tractable, we need to introduce some simplifying approximations.

2.3 Markov Chain Monte Carlo Methods

There are different approaches to calculate the posterior distribution. In (Mackay, 1992) is used a Gaussian approximation for the posterior distribution, while in (Neal, 1996) is introduced a hybrid Monte Carlo method. Another approach to approximate the posterior distribution uses Markov Chain Monte Carlo method (MCMC) (Rios Insua and Müller, 1998). For a review of these methods see, for instance (Bishop, 1996).

The idea of MCMC is to draw a sample of values $\mathbf{w^{(t)}}$, $t = 1, \ldots, M$ from the posterior distribution of network parameters. In this work, we used Gibbs sampling (Geman and Geman, 1984) to generate samples to the posterior distribution.

Gibbs sampling is perhaps the simplest MCMC method and it is applicable when the joint distribution is not known explicitly, but the full conditional distribution of each parameter is known. In a single iteration, Gibbs sampling involves sampling one parameter from full conditional distribution given all other parameters.

Gibbs sampling requires that all conditional distributions of the target distribution can be sampled exactly. When the full conditional distribution was unknown, it was used the Metropolis-Hasting algorithm (Hastings, 1970) or adaptive sampling procedure (Gilks and Wild, 1992). For more details of this method, see (Gamerman, 1997).

We can observe that the integral of Equation 6 is the expectation of function $f_k(\mathbf{x_{(n+1)}}, \mathbf{w})$ with respect to the posterior distribution of the parameters. This expectation can be approximated by MCMC, using a sample of values $\mathbf{w^{(t)}}$ drawn from the posterior distribution of parameters. These values are then used to calculate

$$y_{(n+1)} \approx \frac{1}{M}\sum_{t=1}^{M} f_k(\mathbf{x_{(n+1)}}, \mathbf{w^{(t)}}) \quad (7)$$

3 Methodology

3.1 Data Set

The data set refers to one hundred and thirty-six patients, who agreed to participate in the study. They were referred to Hospital Clementino Fraga Filho, a University Hospital of Federal University of Rio de Janeiro, Brazil, from March, 2001 to September, 2002, with clinical-radiological suspicion of SNPT. The data consisted of information from anamnesis interview and included demographic and risk factors typically known for tuberculosis diagnosis.

These patients were under suspicion of active pulmonary tuberculosis, presenting negative smear. Forty three per cent of these patients actually showed PT in activity.

Twelve clinical variables were considered for model development. These included: age, cough, sputum, sweat, fever, weight loss, chest pain, shudder, alcoholism and others.

To obtain the training and testing sets, the original data were randomly divided into the training and testing sets. Given the statistical limitations of the available data, a division in the form of 80% of the patients for the training set and 20% for the testing set would be preferable. With this strategy, model development was made varying the number of hidden neurons. It was observed that some neural networks exhibited poor performance. This is due to poor statistical representation of training set with respect to patterns belonging to testing set (Santos et al., 2006).

To avoid the possibility of some regions not represented, training and testing sets were also obtained data clustering. Efficient alternative for data selection is *cluster analysis* (Morrison, 1990). This technique permits to detect the existence of clusters in given data. This grouping process can be seen as an unsupervised learning technique.

The clustering method under investigation pointed out three clusters in the data set. In this case, the training set was obtained by randomly selecting 75% of the patients in each cluster and 25% of patients was selected to form the testing set.

3.2 Implementation of BNN

We generated a BNN with 12 inputs, a single-layer feedforward with a fixed number m of hidden nodes and one output node. The nonlinear activation function used for the hidden units and output units was the logistic sigmoid, which produces an output between 0 and 1.

We used the Gaussian prior distribution as described in Section 2, with three separate weight-

groups. The prior over network parameters are

$$\begin{aligned} u_i \mid \lambda_\alpha &\sim N(0, \lambda_\alpha^{-1}), \quad i = 1, \ldots, I. \\ v_j \mid \lambda_\beta &\sim N(0, \lambda_\beta^{-1}), \quad j = 1, \ldots, H. \\ b_k \mid \lambda_\gamma &\sim N(0, \lambda_\gamma^{-1}), \quad k = 1, \ldots, S. \end{aligned} \tag{8}$$

where u_i represents the input-hidden weights, v_j the hidden-outputs weights and b_k the biases terms.

A convenient form for the hyperprior distributions is vague Gamma distribution. Here, we considered all hyperparameters distributed according Gamma distribution with the scale and shape parameter equal to 0.001. The priors for different parameters and hyperparameters are all independent.

The software WinBUGS (Spiegelhalter et al., 2003) was used to implement the Bayesian neural network. Through WinBUGS, we specified the model described in Section 2. Next, the software simulated the posterior distribution values for each parameter of interest, using the Metropolis-within-Gibbs procedure. We computed a single chain of a MCMC sampler in WinBUGS for each parameter of interest (weights and bias). We simulate 20000 iterations, and discarded the 10000 first in each sequence. The experiment was configured with 102 training samples and 34 samples for tests.

The posterior distribution samples for the model parameters were used to estimate the predictive distribution for the new test inputs. For each iteration t, the BNN has parameters $\mathbf{w^{(t)}}$ and produces an output $y^{(t)}$, for an input vector $\mathbf{x}$. Thus, for each test sample, we calculate the arithmetic mean of the M network outputs, according to Equation 7.

Several feedforward neural networks also were tested. The networks have log-sigmoid activation functions, one output neuron and one hidden layer, with the optimum number of hidden neurons found empirically. Alternative parameters were used (learning rate, momentum and number of iterations) of the backpropagation algorithm.

The performance of the BNN and ANN was evaluated through the classification for the testing set, which is referred to here as accuracy. Other descriptive statistics were also used to evaluate the performance of neural networks in study, they are: *sensitivity* and *specificity*, since those measured are of general use in the medicine. *Sensitivity* of the neural model will tell us how the model is classifying the patients with PT in activity, while the *specificity* will tell how the model is classifying patients without PT in activity.

4 Results

Bayesian neural networks with 2, 3, 4 and 5 hidden neurons were tested and we report just the best one, that was obtained with two hidden neurons.

The results of BNN are comparable to those of artificial neural network with four hidden neurons. This network was selected because it has the smallest classification errors in the test set.

According to Table 1, the BNN with presented the largest accuracy (82%), as well as the largest specificity (85%). However, the ANN possesses the largest sensitivity (100%), i.e., this network better classifies the individuals with active PT.

Tabela 1: Classification efficiencies

	ANN	BNN
Accuracy	76%	82%
Specificity	60%	85%
Sensitivity	100%	79%

5 Conclusion

The present study has applied Bayesian neural network to predict medical diagnosis. The Bayesian neural network model achieved good classification performance, exhibiting sensitivity from 79% and specificity from 85%.

It is known that parsimonious models with few hidden neurons are preferable, because they tend to show a better generalization ability, reducing the *overfitting* problem. Although, the ANN possesses the largest sensitivity, this network has more hidden neurons, and it is not good when the amount of available data is small.

Many issues remain to be explored. For example, we should try other prior distributions and to treat the number hidden neurons as an additional parameter.

Acknowledgements

The authors thank the staff and students of the Tuberculosis Research Unit, Faculty of Medicine, Federal University of Rio de Janeiro, for making available the data used in this work.

References

Berger, J. O. (1985). *Statistical Decision Theory and Bayesian Analysis*, 2^a edn, Springer-Verlag, New York.

Bishop, C. M. (1996). *Neural Networks for Pattern Recognition*, 2^a edn, Oxford University Press, New York.

Buntine, W. L. and Weigend, A. S. (1991). Bayesian backpropagation, *Complex Systems* **5**: 603–643.

Edwards, D. F., H., H., Zazulia, A. R. and Diringer, M. N. (1999). Artificial neural

networks improve the prediction of mortality in intracerebral hemorrhage, *Neurology* **53**: 351–356.

El-Solh, A. A., Hsiao, C.-B., Goodnough, S., Serghani, J. and Grant, B. J. B. (1999). Predicting active pulmonary tuberculosis using an artificial neural network, *Chest* **116**(4): 968–973.
URL: *1*

Gamerman, D. (1997). *Markov Chain Monte Carlo: Stochastic Simulation for Bayesian Inference*, Chapman & Hall, Londres.

Geman, S. and Geman, D. (1984). Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **6**: 721–741.

Ghosh, M., Maiti, T., Chakraborty, S. and Tewari, A. (2004). Hierarchical bayesian neural networks: An application to prostate cancer study, *Journal of the American Statistical Association* **99**(467): 601–608.

Gilks, W. and Wild, P. (1992). Adaptive rejection sampling for Gibbs sampling, *Journal of the Royal Statistical Society Series C (Applied Statistics)* (41): 337–348.

Hastings, W. (1970). Monte Carlo sampling methods using Markov chains and their applications, *Biometrika* **57**: 97–109.

Kiyan, T. and Yildirim, T. (2003). Breast cancer diagnosis using statistical neural networks, *International XII. Turkish Symposium on Artificial Intelligence and Neural Networks* .

Lampinen, J. and Vehtari, A. (2001). Bayesian approach for neural networksreview and case studies, *Neural Network* (14): 7–24.

Mackay, D. J. C. (1991). *Bayesian Mathods for Adaptive Models*, Tese de doutorado, California Institute of Technology.

Mackay, D. J. C. (1992). A practical Bayesian framework for backpropagation networks, *Neural Computation* **4**(3): 448–472.

Morrison, D. F. (1990). *Multivariate Statistical Methods*, 3^a edn, McGraw Hill, New York.

Neal, R. M. (1996). *Bayesian Learning for Neural Networks*, Springer-Verlag, New York.

Rios Insua, D. and Müller, P. (1998). Feedforward neural networks for nonparametric regression, *in* D. Dey, P. Müller and D. Sinha (eds), *Practical Nonparametric and Semiparametric Bayesian Statistics*, Springer-Verlag, New York, pp. 181–193.

Santos, A. M. (2003). *Redes neurais e árvores de classificação aplicadas ao diagnóstico da tuberculose pulmonar paucibacilar*, PhD thesis, COPPE/ UFRJ, Rio de Janeiro.

Santos, A. M., Pereira, B. B., Seixas, J. M., Mello, F. C. Q. and Kritski, A. L. (2006). Neural networks: An application for predicting smear negative pulmonary tuberculosis, *in* A. Jean-Louis, N. Balakrishnan, M. Mesbah and G. Molenberghs (eds), *Advances in Statistical Methods for the Health Sciences: Applications to Cancer and AIDS Studies, Genome Sequence Analysis, and Survival Analysis*, Birkhäuser Boston, New York, pp. 181–193.

Spiegelhalter, D. J., Thomas, A., Best, N. and Lunn, D. (2003). WinBUGS manual version 1.4, *Technical report*, MRC Biostatistics Unit, Institute of Public Health, Robinson Way, Cambridge.

Tu, J. V. (1996). Advantages and disadvantages of using artificial neural networks versus logistic regression for predicting medical outcomes, *Journal Clinical of Epidemiology* **49**(11): 1225–1231.