STATISTICS IN MEDICINE, VOL. 14, 73–82 (1995)

A NEURAL NETWORK MODEL FOR SURVIVAL DATA

DAVID FARAGGI AND RICHARD SIMON

Biometric Research Branch, National Cancer Institute, 6130 Executive Blvd., Room 739, Rockville, MD 20852, U.S.A.

SUMMARY

Neural networks have received considerable attention recently, mostly by non-statisticians. They are considered by many to be very promising tools for classification and prediction. In this paper we present an approach to modelling censored survival data using the input–output relationship associated with a simple feed-forward neural network as the basis for a non-linear proportional hazards model. This approach can be extended to other models used with censored survival data. The proportional hazards neural network parameters are estimated using the method of maximum likelihood. These maximum likelihood based models can be compared, using readily available techniques such as the likelihood ratio test and the Akaike criterion. The neural network models are illustrated using data on the survival of men with prostatic carcinoma. A method of interpreting the neural network predictions based on the factorial contrasts is presented.

1. INTRODUCTION

Neural networks[1] have received much attention recently by computer scientists, neurophysiologists, psychologists and engineers interested in biological nervous system organization and artificial intelligence. Statisticians have given little attention to them, although neural networks are increasingly applied to problems of classification and prediction. For example, neural networks have been used for speech recognition,[2] diagnostic image analysis,[3] clinical diagnosis,[4] macromolecule sequence analysis[5] and prediction of mechanism of action for new cancer drugs.[6] There has also been a concern that the capabilities of neural networks have been overestimated by individuals with limited knowledge of standard statistical procedures.

Evaluation of the potential of neural networks for classification and prediction should be based on empirical comparisons of neural network methods with other statistical methods on real data. Many problems of medical prediction involve the use of right-censored survival data. Although neural networks have been previously applied to survival or duration of stay data,[7–9] the outcome has been generally coded as an uncensored discrete variable with all censored observations omitted or included in the highest category. Often, survival is considered binary (for example, survival for duration of hospital stay or not). The one attempt we found to incorporate general right-censored observations was based on a complex procedure for predicting censoring indicators using multiple representations of each case and survival time as an input variable to the network.[10–11] In this paper, we follow a different route to provide an extension of neural network prediction methods to accommodate right-censored data. The models are based on the likelihood function and we use maximum likelihood estimates of the network parameters, so likelihood ratio tests and the Akaike criterion[12] can be used to determine the network topology and to select variables.

In Section 2 we review the basic single hidden layer feed forward neural network. In Section 3, we describe a model for utilizing neural network predictors with censored data. We discuss the use of this approach mainly in conjunction with Cox's proportional hazards model.[13] We show,

CCC 0277–6715/95/010073–10

Received May 1993
Revised January 1994

however, that the idea is easily extended to other models developed for censored data such as the accelerated failure time model[14] and the Buckley–James model.[15] In Section 4 we discuss optimization methods for fitting parameters. In Section 5 we illustrate the neural network models and the standard Cox proportional hazards additive model using data on the survival of patients with prostate cancer. We show how both the likelihood ratio test and the Akaike criterion are used to evaluate nested models. We then present a method for interpreting neural network predictions based on factorial contrasts. This method can also be used for non-censored data.

2. THE NEURAL NETWORK

Most neural networks used for prediction have one layer of input nodes, one layer of output nodes and one layer of 'hidden' nodes. In this paper we will restrict ourselves to the single output node network, an example of which is shown in Figure 1. Each input is connected directly to all but one node in the hidden layer. In the hidden nodes a non-linear transformation is performed on a weighted sum of the inputs. Each hidden node is connected directly to the output node. No feedback connections are permitted in these 'feed-forward' networks. Consider a vector of covariates $x_i = (x_{i0}, x_{i1}, \ldots, x_{iP})'$ for the ith case as an input to the network ($i = 1, \ldots, n$), where x_{i0} is always 1. The pth input node takes value x_{ip} for this input vector ($p = 0, \ldots, P$). A weight w_{ph} is associated with the connection between input p and hidden node h ($h = 1, \ldots, H$). In general, a neural network has a special hidden node (node 0) connected to the output nodes but not the input nodes. Its role is similar to the constant term in linear regression. However, in the proportional hazards setting it is not needed because its effect is incorporated into the baseline hazard. The vector of weights associated with the inputs to hidden node h will be denoted $w_h = (w_{0h}, w_{1h}, \ldots, w_{Ph})'$. The input to hidden node h for the ith case is a linear projection $w_h' x_i$. The output of hidden node h is $f(w_h' x_i)$ where f is a 'squashing function'. The most commonly used squashing function is the logistic function $f(z) = 1/(1 + \exp(-z))$. Other squashing functions include the linear function and the hyperbolic tangent function. When the identity squashing function $f(z) = z$ is used the output becomes a linear function of the inputs. In this paper we restrict ourselves to the logistic squashing function. We omit the use of a squashing function for the output layer for reasons discussed below. The weights w_{0h} associated with the unity input x_{i0} are called 'bias' terms. The output from the network is a weighted sum of the output of the hidden nodes with weights $\alpha_0, \ldots, \alpha_H$. Therefore the functional representation of the output from a single hidden layer neural network with H hidden nodes for a given input vector x_i is

$$g(x_i, \theta) = \alpha_0 + \sum_{h=1}^{H} \alpha_h f(w_h' x_i) = \alpha_0 + \sum_{h=1}^{H} \alpha_h / [1 + \exp(-w_h' x_i)] \tag{1}$$

where θ denotes the vector of unknown parameters, $(w_{01}, w_{11}, \cdots, w_{P1}, w_{02}, w_{12}, \ldots, w_{P2}, \ldots, w_{0H}, w_{1H}, \ldots, w_{PH}, \alpha_0, \alpha_1, \ldots, \alpha_H)'$. The number of parameters in (1) is $m = (H + 1) + (P + 1)H$. We define $g(x, \theta) = [g(x_1, \theta), \ldots, g(x_n, \theta)]'$ as the vector of outputs for all cases. From model (1), the relationship between neural network and projection pursuit regression[16] is seen. In projection pursuit regression both the projections θ and the function f in (1) are estimated from the data while in parametric neural networks the squashing function is pre-specified.

Neural networks are generally 'trained' using the back propagation algorithm. 'Training' really means determining values for the parameters so that the predicted outputs $g(x, \hat{\theta})$ are good estimates of the true outputs. A training data set therefore must consist of both the inputs and the outputs for a set of cases. The back propagation algorithm iteratively estimates $\hat{\theta}$ via a gradient descent method. The training of the network starts by choosing starting values $\hat{\theta}_0$. An updated

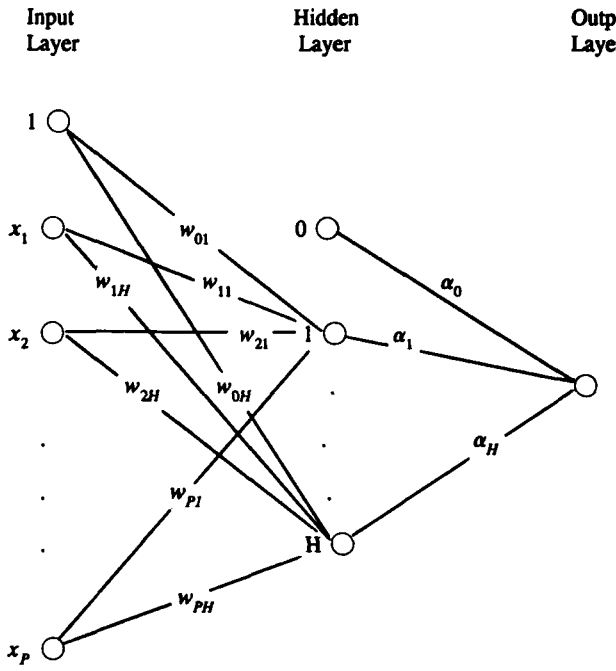

Figure 1. Single hidden layer neural network

estimate is determined as observations (y_i, x_i) are considered one at a time. Multiple passes through the data are usually needed before the estimates converge. For further discussion of back-propagation methods see, for example, Wasserman.[17]

3. NEURAL NETWORK SURVIVAL MODELS

The most commonly used model for censored data with covariates is Cox's proportional hazards (PH) model.[13] The hazard function depends on time t and a vector of covariates x_i through

$$h(t,x_i) = h_0(t)\exp\{\beta x_i\}. \tag{2}$$

The vector of parameters β is estimated by maximizing the partial likelihood

$$L_c(\beta) = \prod_{i\in u} \frac{\exp(\beta x_i)}{\sum_{j\in R_i} \exp(\beta x_j)} \tag{3}$$

using the Newton–Raphson method. The product in (3) is taken over the uncensored observations. For more details see, for example, Miller.[18]

Consider replacing the linear functional βx_i in (2) by the output $g(x_i,\theta)$ of the network. The proportional hazards model becomes $h(t,x_i) = h_0(t)\exp[g(x_i,\theta)]$ and the function to be maximized becomes

$$L_c(\theta) = \prod_{i\in u} \frac{\exp\left\{\sum_{h=1}^{H} \alpha_h/(1+\exp(-w_h'x_i))\right\}}{\sum_{j\in R_i} \exp\left\{\sum_{h=1}^{H} \alpha_h/(1+\exp(-w_h'x_j))\right\}} \tag{4}$$

Note that in the proportional hazards case, we do not include the constant α_0. Also, when replacing βx_i with the output $g(x_i,\theta)$ of the network, we omit the squashing function that is usually applied to the output of the network, so that like βx_i the output will be on the whole real line and not confined to the interval (0, 1).

Maximum likelihood estimates (MLE's) of the parameters of the neural network are obtained using the Newton–Raphson method to maximize the partial likelihood. The important feature of our approach is the use of a specific statistical model and MLE's determined by maximizing the likelihood function of this model. The use of likelihood functions and maximum likelihood to estimate the network parameters will enable us to utilize standard statistical tests and procedures to evaluate the network. Our use of the Newton–Raphson method rather than gradient descent is conceptually unimportant.

This way of constructing the neural network model extends to other models for censored data for example the accelerated failure time model[14] and the Buckley–James model.[15] These models also assume that the covariates influence survival through a linear functional βx. Using neural network topology with these models requires replacing (βx) by a neural network non-linear function $g(x,\theta)$ and proceeding with the algorithms that are commonly used to estimate the parameters in those models.

4. OPTIMIZATION METHODS

We employ Newton–Raphson iterations to maximize the partial likelihood function (4) using GAUSS[19] maximum likelihood software. Both the first and second derivatives were approximated numerically. The program is available from the authors upon request.

Since the network function is a non-linear one with many parameters involved, the error surface is highly convoluted and, as others have pointed out, many local optima may be present. To have reasonable assurance that we found the global maximum, we used multiple starting values to initiate the Newton–Raphson iterations. Other problems with neural networks include over-fitting the data[20] and failure of convergence of parameter estimates due to the algorithm getting trapped in regions with large parameter values at the edges of the surface. One way of overcoming the latter problem is to optimize a function that incorporates a penalty for large parameter values, as discussed by Ripley.[21] To obtain starting values we used Newton–Raphson to maximize a penalized likelihood function

$$L_c(\theta) + \lambda \sum \theta_k^2 \tag{5}$$

with the summation taken over the components of the parameter vector θ. We experimented with various values of the shrinkage parameter λ and used multiple starting values for optimizing the penalized likelihood function. The estimates resulting from the penalized likelihood function were used as starting values for maximizing $L_c(\theta)$ by Newton–Raphson. The largest local maximum obtained was taken as the estimate of the global maximum.

5. AN EXAMPLE

We have used a prostatic cancer data set[22] to illustrate the method described above. The data relate to 506 patients with stage 3 or 4 prostatic cancer who entered a four-group clinical trial comparing the effect of 0·2, 1 or 5 mg of diethylstilbestrol (DES) to placebo. Data on twelve covariates, beside the treatment group, were available, in addition to survival time and survival status. For simplicity we have combined the placebo group with the low dose of DES and compared these to the two groups receiving the higher doses of DES. We have chosen a subset of

the covariates for the analysis comprising one binary factor (stage of disease) and two continuous factors (age and weight). The outcome event in the analysis was death from any cause. The number of patients in the analysis was reduced to 475 by omitting those with missing data.

We fitted four models:

(a) First-order PH model (4 parameters);
(b) Second-order (interactions) PH model (10 parameters);
(c) Neural network model with two hidden nodes (12 parameters);
(d) Neural network model with three hidden nodes (18 parameters).

Parameter estimation was done on a randomly selected half of the data (238 patients). Data on the other 237 patients were used for validating the models. Table I gives summary statistics for the complete data set and for the training and validation subsets. As seen in Table I, the distributions of the factors are similar among the training and validation subsets. The search for the maximum likelihood estimators for the neural network models started with very small negative numbers as initial values and a penalized log-likelihood (5). The values for the parameters obtained at convergence were then used in a non-penalized log-likelihood search to obtain the final MLE's. As for the first- and second-order PH model the final MLE's were obtained by maximizing (3). Table II summarizes the results obtained using the four models. Results are given both for the half of the data used for parameter estimation and the half used for validation. Two measures of fit are given; the log partial likelihood and a measure of concordance c suggested by Harrell *et al.*[23] The statistics c is calculated by taking all possible pairings of patients. For a given pair, the predictions are said to be concordant with the outcome if the patient having a higher predicted probability of survival lived longer. If both patient's survival times are censored, or if only one died and the follow-up duration of the other was less than the survival time of the first, the pair is not counted. The c index is the proportion of predictions that are concordant out of all pairs of patients for which ordering of the survival times can be determined. Hence, $c = 0{\cdot}5$ represents the level of concordance expected when the model is not predictive. When outcome is dichotomous, c is the area under the receiver operating characteristic (ROC) curve.[24] Hence, the c statistics can be viewed as a generalization of the ROC area under the curve for censored data.

Since the addition of hidden nodes generate neural network models that are nested, likelihood based techniques can be used to select the appropriate number of nodes. We considered using the likelihood ratio tests and the Akaike criterion:[12]

$$AIC(s) = -\max \log(\textit{likelihood}) + s \tag{6}$$

where s is the number of parameters in the evaluated model. The Akaike approach selects the model that minimizes (6).

With the data from the training subset, the neural network model with three hidden nodes (18 parameters) yielded a log-likelihood of $-794{\cdot}9$ compared to $-801{\cdot}2$ for the neural network with two hidden nodes (12 parameters). This increase was not statistically significant at the 5 per cent level. The improvement in the Akaike criterion between the two models is marginal (0·3). These marginal improvements did not carry over to the validation subset. There was a very large decrease in the log-likelihood for the neural network model with three hidden nodes ($-860{\cdot}0$). This decrease in the likelihood demonstrates that when a mode such as the neural network model is used, overparameterization is possible and validation of the model is crucial. For the first- and second-order models, the hypothesis that the second-order terms of the PH model are all simultaneously zero was rejected at the 0·05 significance level using the likelihood ratio test on the training subset. Nevertheless, the second-order model does not show an improvement in the likelihood over the linear model in the validation subset. These results demonstrate the difference

Table I. Summary statistics for the factors included in the models

	Complete data	Training set	Validation set
Sample size	475	238	237
Stage 3	47·5%	47·6%	47·4%
4	52·5%	52·4%	52·6%
Median age	73 years	73 years	73 years
Median Weight	98·0	97·0	99·0
Treatment: Low	49·9%	48·3%	51·5%
High	50·1%	51·7%	48·5%
Median survival	33 months	33 months	34 months
% censoring	28·8%	29·8%	27·8%

Table II. Log-likelihoods and c statistics for first-order, second-order and neural network proporational hazards models

Model		Training data		Test data	
	Number of parameters s	Log lik	c	log lik	c
First-order PH	4	− 815·3	0·608	− 831·0	0·607
Second-order PH	10	− 805·6	0·648	− 834·8	0·580
Neural network $H = 2$	12	− 801·2	0·646	− 834·5	0·600
Neural network $H = 3$	18	− 794·9	0·661	− 860·0	0·582

between statistical significance and accurate prediction. The difference is presumably due to the increased variance of prediction in estimating additional parameters, which is not sufficiently offset by the reduction in bias resulting from the incorporation of additional covariates. With covariates that are more strongly associated with the outcome the results could be different.

To interpret and compare the meaning of the different models we defined 16 cells defined by two levels of the three covariates and a treatment indicator. The levels were disease stage (3, 4), age in years (65, 75), weight in kilograms (85, 105) and the treatment indicator (placebo-low dose, high dose of DES). Using the parameter estimates obtained from the training set, predictions for the 16 different cells under the four different models were calculated. These predictions were then used as responses and 2^4 factorial design[25] contrasts were used to estimate the main effects and higher order interactions for each of the four models. For example, to estimate the treatment × age interaction, the sum of the eight predictors that had treatment and age of opposite levels was subtracted from the sum of the eight predictors for which both treatment and age were at the same level and the result was divided by 8. Table III shows the estimates of the effects for each model. By construction all interactions are zero for the first-order PH model and third- and fourth-order interactions are zero for the second-order model. Since the neural network models involve non-linear transformations of the covariate values we obtain non-zero estimates for all the effects.

The neural network model with two hidden nodes has 12 parameters so, under this model, the 16 estimated contrasts are not independent. However, they still provide some means of comparing the different models. All four models had similar main effects. The neural network models detected the two-way interactions with the same signs and similar magnitudes as the

Table III. Estimation of the main effects and higher order interactions using 2^4 factorial design contrasts and the predictions obtained from the different models

Effects	PH 1st order	PH 2nd order	Neural network $H = 2$	Neural network $H = 3$
Stage	0·300	0·325	0·451	0·450
Rx*	− 0·130	− 0·248	− 0·198	− 0·260
Age	0·323	0·315	0·219	0·278
Weight	− 0·249	− 0·238	− 0·302	− 0·581
Stage × Rx	0	− 0·256	− 0·404	− 0·655
Stage × Age	0	− 0·213	− 0·330	− 0·415
Stage × Wt*	0	− 0·069	− 0·032	− 0·109
Rx × Age	0	0·293	0·513	0·484
Rx × Wt	0	− 0·195	− 0·025	0·051
Age × Wt	0	− 0·128	− 0·228	− 0·070
Stage × Rx × Age	0	0	0·360	0·475
Stage × Rx × Wt	0	0	0·026	0·345
Stage × Age × Wt	0	0	− 0·024	0·271
Rx × Age × Wt	0	0	0·006	− 0·363
Stage × Rx × Age × Wt	0	0	0·028	− 0·128

* Rx = Treatment
Wt = Weight

second-order PH model. Past analysis of these data has indicated that the treatment × age interaction is important.[22] The neural network with two hidden nodes found this interaction effect to be the largest among all effects calculated. This neural network model also suggested the existence of a third-order interaction (Stage × Treatment × Age), which the second-order PH model could not include. Figure 2 shows the relationship between predictions of the PH models (βx) (x-axes) and predictions of the neural network models ($g(x,\theta)$) (y-axes) in the validation data subset. Similar results are seen for the training subset. The closest association appears between the second-order PH model and the neural network with two hidden nodes.

6. DISCUSSION

The purpose of this paper is to introduce a neural network model for censored survival data and to indicate the broad applicability of this approach. The approach of replacing the functional βx by the output function $g(x,\theta)$ of a single hidden layer feed-forward neural network is applicable to other survival models, such as accelerated failure time models. This approach can also be used to investigate neural network predictors in the context of any generalized linear model. The linear predictor is replaced by $g(x,\theta)$ for any specified link function and error distribution. In neural network models, emphasis should be given to predictive power rather than inference about model parameters since the latter are generally difficult to interpret. Interpretation of model predictions can be facilitated by use of the factorial design approach of Box *et al.*[25] as illustrated in our example.

Our example is an illustration of the proposed method. It is not intended as an evaluation of the predictive accuracy of neural network models versus additive proportional hazard models. We have illustrated our method on one set of data and comparisons of neural network models to other models in a variety of applications will be necessary to evaluate the utility of this approach. The neural network models and the additive proportional hazard models are non-nested

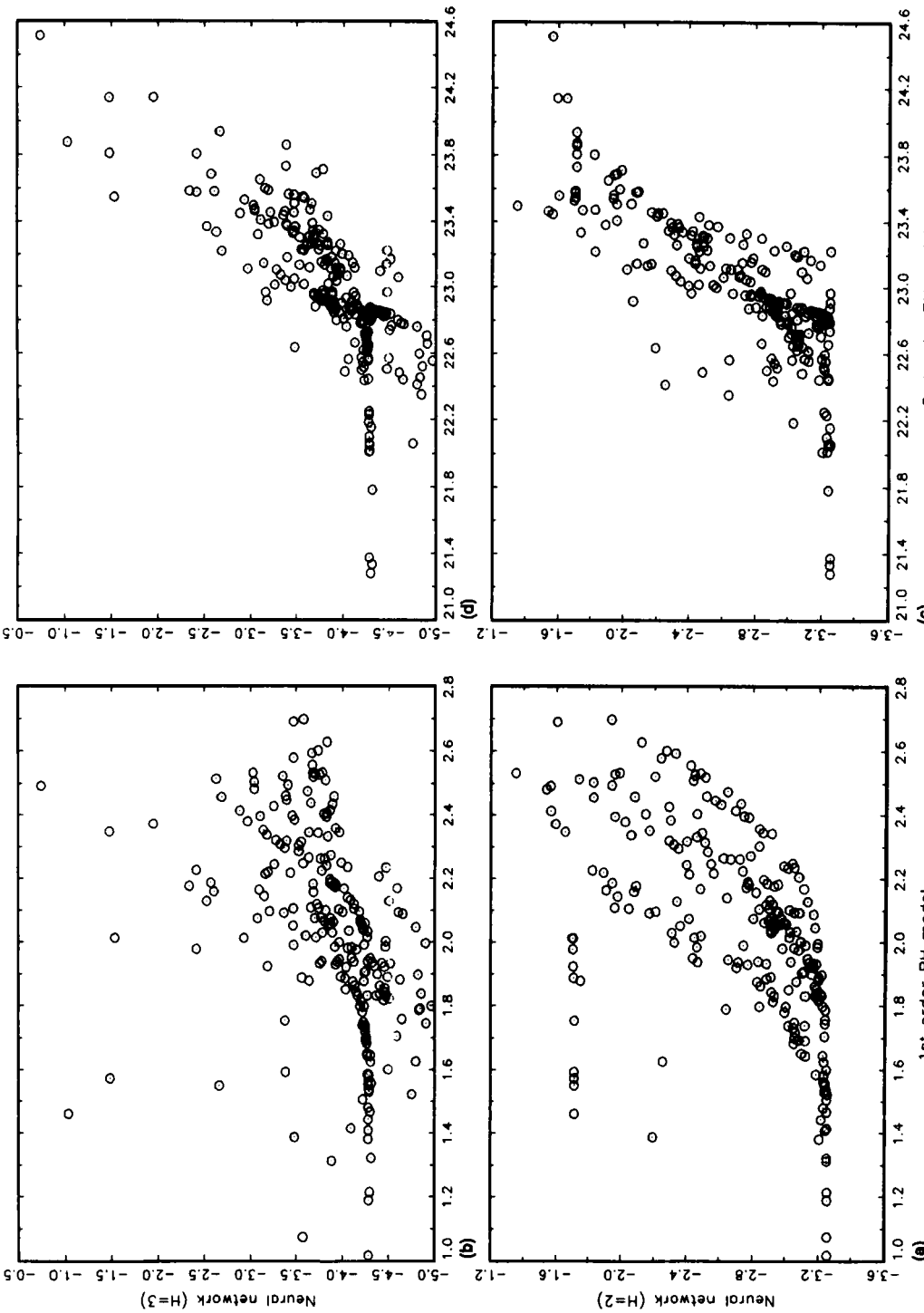

Figure 2. Relationship between predictions of the additive proportional hazards models (PH) and the neural network models (NN) in the validation set: (*a*) βx from first-order PH model versus $g(x, \theta)$ from NN($H = 2$); (*b*) βx from first-order PH model versus $g(x, \theta)$ from NN($H = 3$); (*c*) βx from second-order PH model versus $g(x, \theta)$ from NN($H = 2$); (*d*) βx from second-order PH model versus $g(x, \theta)$ from NN($H = 3$)

models.[26] One approach to evaluating whether one model predicts better than the other for a given set of data, is use of the bootstrap to construct a confidence interval for the difference in the c statistics obtained from the two models. This would require re-fitting the neural network for each bootstrap sample and would therefore be computationally intensive. Asymptotic variances for the c statistics even for individual models are not, to our knowledge, available because of the functional dependence of the predicted outcomes on the survivals through the estimated parameters. Although computationally demanding, the bootstrap would also provide more precise estimate of predictive accuracy for each model than our use of a single data split as pointed out by Efron.[27]

This neural network generalization of the proportional hazard model has several potential advantages. It allows the topology of a single hidden layer feed-forward network to be used for representing the relationship between the hazard function and covariates. It does this in the context of a specific statistical model for the data and hence standard statistical methods for evaluating estimators, selecting sub-models and evaluating predictors are available. Because the likelihood function may not be unimodal, however, careful attention to computational issues is necessary. In this paper, we have presented a simple form of neural networks with only a few hidden nodes. A standard optimization algorithm is used to maximize the model likelihood function with respect to the network parameters. A different approach would be to use a more complicated network function, with many more nodes in the hidden layer or more than one hidden layer. However, overfitting the data then becomes a major concern. To overcome this problem curtailed training and cross-validation are commonly used. The point of termination is determined by an estimate of prediction error computed from a validation set of observations not part of the set used for training. That is, the iterative procedure used for maximizing the fit of predictions to observed responses is terminated before convergence is reached. Other training algorithms could also be used, for example, back-propagation or simulated annealing.[28] These methods can be used with censored survival data using the approach we have outlined and warrant future investigation.

We have tried to emphasize what we believe are the most important aspects of the neural network; its functional representation and its ability to be used for prediction and classification with a wide range of statistical models. We have adapted a simple neural network architecture for use in conjunction with a statistical model for survival data. The neural network literature is now voluminous and algorithms for training the network play a major role. It is important to realize that the back propagation method of training the network is not a part of the network architecture, but merely one of many algorithms that can be used to optimize an error (or likelihood) function with regard to some unspecified parameters.

The approach described here can also be generalized to adapt other new modelling approaches, for example, the multivariate adaptive regression spline (MARS) procedure,[29] to survival data. The linear predictor βx of the survival model (for example, proportional hazards model or accelerated failure time model) could be replaced by the non-linear MARS predictor rather than by the neural network output function. The model selection criterion could then be based on the model likelihood function rather than on the squared errors of predictions.

ACKNOWLEDGEMENT

The authors thank the referees for helpful comments.

REFERENCES

1. Rumelhart, D. E. Hinton, G. E. and Williams, R. J. 'Learning internal representations by error propagation', in *Parallel Distributed Processing. Vol. 1*, MIT Press, Cambridge, MA 1986, pp. 318–362.

2. Sejnowski, T. J. and Rosenberg, L. R. 'Parallel networks that learn to pronounce english test', *Complex Systems*, **1**, 145–168 (1987).
3. Daponte, J. S. and Sherman, P. 'Classification of ultrasonic image texture by satistical discriminate analysis of Neural Network', *Comput. Med. Image. Graph.*, **15**, 3–9 (1991).
4. Mann, N. H. I. and Brown, M. D. 'Artificial intelligence in the diagnosis of low back pain', *Orthopedic Clinics of North America*, **22**, 303–314 (1991).
5. Holley, L. H. and Karplus, M. 'Protein structure prediction with a neural network', *Proceedings of the National Academy of Science, U.S.A.* **86**, 152–156 (1989).
6. Weinstein, J. N., Kohn, K. W., Grever, M. R., Viswanadhan, V. N., Rubinstein, L. V., Monks, A. P, Scudiero, D. A., Welch, L., Koutsoukos, A. D., Chiausa, A. J. and Paull, K. D. 'Neural computing in cancer drug development predicting mechanism of action', *Science*, **258**, 447–451 (1992).
7. Ebell, M. H. 'Artificial neural network for predicting failure to survive following in-hospital cardiopulmonary resuscitation', *The Journal of Family Practice*, **36**, 297–303 (1993).
8. Davis, G. E., Lowell, W. E. and Davis, G. L. 'A neural network that predicts psychiatric length of stay', *M. D. Computing*, **10**, 87–92 (1993).
9. Tu, J. V. and Guerriere, M. R. J. 'Use of a neural network as a predictive instrument for length of stay in the intensive care unit following cardiac surgery', *Computers and Biomedical Research*, **26**, 220–229 (1993).
10. Ravdin, P. M., Clark, G. M., Hilsenbeck, S. G., Owens, M. A., Vendely, P., Pandian, M. R. and McGuire, W. L. 'A demonstration that breast cancer recurrence can be predicted by neural network analysis,' *Breast Cancer Research and Treatment*, **21**, 47–53 (1992).
11. Ravdin, P. M. and Clark, G. M. 'A practical application of neural network analysis for predicting outcome of individual breast cancer patients', *Breast Cancer Research and Treatment*, **22**, 285–293 (1992).
12. Akaike, H. 'Information theory and an extension of the maximum likelihood principle', Proceedings of the Second International Symposium on Information Theory, Patrov, B. N. and Csaki, F. (eds.), Akademia Kiado, Budapest, 1973, pp. 267–281.
13. Cox, D. R. 'Regression models and life tables (with discussion)', *Journal of the Royal Statistical Society, Series, B*, **34**, 187–220 (1972).
14. Prentice, R. L. and Kalbfleisch, J. D. 'Hazard rate models with covariates', *Biometrics*, **35**, 25–39 (1979).
15. Buckely, J. and James, I. 'Linear regression with censored data', *Biometrika*, **66**, 429–436 (1979).
16. Friedman, J. H. and Stuetzle, W. 'Projection pursuit regression', *Journal of the American Statistical Association*, **76**, 817–823 (1981).
17. Wasserman, P. D. *Neural Computing Theory and Practice*, Van Nostrand Reinhold, New York, 1989.
18. Miller, R. G. *Survival Analysis*, Wiley, New York, 1981.
19. Gauss. *Maximum Likelihood Gauss Applications*, Aptech Systems, Inc. 1993.
20. Geman, S., Bienstock, E. and Doursat, R. 'Neural networks and the bias/variance dilemma', *Neural Computation*, **4**, 1–58 (1992).
21. Ripley, B. D. Statistical aspects of neural networks', Brandorff-Nielsen, O. E., Jensen, J. L. and Kendall, W. S. (eds.), *Network and Chaos – Statistical and Probabilistic Aspects*, Chapman and Hall, 1993.
22. Byar, D. P. and Green, D. K. 'The choice of treatment for cancer patients based on covariates information: application to prostate cancer', *Bulletin of Cancer, Paris*, **67**, 477–488 (1980).
23. Harrell, F. E., Lee, K. L., Califf, R. M., Pryor, D. B. and Rosati, R. A. 'Regression modelling strategies for improved prognostic prediction', *Statistics in Medicine*, **3**, 143–152 (1984).
24. Hanley, J. A. and McNeil, B. L. 'The meaning and use of the area under a receiver operating characteristic (ROC) curve', *Radiology*, **143**, 29–36 (1982).
25. Box G. E. P., Hunter, W. G. and Hunter, J. S. *Statistics for Experimenters*, Wiley, New York, 1978.
26. Efron, B. 'Comparing non-nested models', *Journal of the American Statistical Association*, **79**, 791–803 (1984).
27. Efron, B. *The Jackknife the Bootstrap and Other Resampling Plans*, SIAM, Philadelphia, 1982.
28. Bohachevsky, I. O., Johnson, M. E. and Stein, M. L. 'Generalized simulated annealing for function optimization', *Technometrics*, **28**, 209–217 (1986).
29. Friedman, J. H. 'Fitting to noisy data in high dimensions', *Computer Science and Statistics: Proceedings of the 20th Symposium on the Interface*, 13–43 (1988).