Hindawi Publishing Corporation
Journal of Probability and Statistics
Volume 2013, Article ID 753930, 7 pages
http://dx.doi.org/10.1155/2013/753930

Research Article

Nonlinear Survival Regression Using Artificial Neural Network

Akbar Biglarian,[1] Enayatollah Bakhshi,[1] Ahmad Reza Baghestani,[2] Mahmood Reza Gohari,[3] Mehdi Rahgozar,[1] and Masoud Karimloo[1]

[1] *Department of Biostatistics, University of Social Welfare and Rehabilitation Sciences (USWRS), Tehran 1985713834, Iran*
[2] *Department of Biostatistics, Faculty of Paramedical Sciences, Shahid Beheshti University of Medical Sciences, Tehran 1971653313, Iran*
[3] *Hospital Management Research Center, Tehran University of Medical Sciences (TUMS), Tehran 1996713883, Iran*

Correspondence should be addressed to Akbar Biglarian; abiglarian@gmail.com

Received 9 May 2012; Revised 21 November 2012; Accepted 23 November 2012

Academic Editor: Shein-chung Chow

Survival analysis methods deal with a type of data, which is waiting time till occurrence of an event. One common method to analyze this sort of data is Cox regression. Sometimes, the underlying assumptions of the model are not true, such as nonproportionality for the Cox model. In model building, choosing an appropriate model depends on complexity and the characteristics of the data that effect the appropriateness of the model. One strategy, which is used nowadays frequently, is artificial neural network (ANN) model which needs a minimal assumption. This study aimed to compare predictions of the ANN and Cox models by simulated data sets, which the average censoring rate were considered 20% to 80% in both simple and complex model. All simulations and comparisons were performed by R 2.14.1.

1. Introduction

Many different parametric, nonparametric, and semiparametric regression methods are increasingly examined to explore the relationship between a response variable and a set of covariates. The choice of an appropriate method for modeling depends on the methodology of the survey and the nature of the outcome and explanatory variables.

A common research question in medical research is to determine whether a set of covariates are correlated with the survival or failure times. Two major characteristics of survival data are censoring and violation of normal assumption for ordinary least squares multiple regressions. These two characteristics of time variable are reasons that straightforward multiple regression techniques cannot be used. Different parametric and semiparametric models in survival regression were introduced which model survival or hazard function. Parametric models, for instance, exponential or weibull, predict survival function while accelerated failure time models are parametric regression methods with logarithm failure time as dependent variable [1, 2].

Choosing an appropriate model for the analysis of the survival data depends on some conditions which are called the underlying assumptions of the model. Sometimes, these assumptions may not be true, for example: (a) lack of independence between consequent waiting times to occurrence of an event or nonproportionality of hazards in semiparametric models, (b) lack of independency of censoring or the distribution of failure times in the case of parametric models [1–3].

Although, the Cox regression model is an efficient strategy in analyzing survival data, but when the assumptions of this model are fail, the free assumption methods could be suitable.

Artificial neural network (ANN) models, which are completely nonparametric, have been used increasingly in different areas of sciences. Although analyzing the data using ANN methodology is usually more complex than traditional approaches, ANN models are more flexible and efficient when our main aim is prediction or classification of an outcome using different explanatory variables [4–17].

Note that when several covariates and complex interactions are of concern the best method is ANN; otherwise, based on model assumptions simple regression models can be appropriately used.

In this study, simulated data sets with different rates of censoring were used to predict the outcome using ANN and

traditional Cox regression models, and then the results of predictions were compared.

2. Methods

2.1. Cox Regression Model. Suppose that T denotes a continuous nonnegative random variable describing the failure time of an event (i.e., time-to-event) in a system. The probability density function of t; that is, the actual survival time is $f(t)$. The survival function, $S(t)$, is probability that the failure occurs later than time t. The related hazard function, $h(t)$, denotes the probability density of an event occurring around time t, given that it has not occurred prior to time t.

As we know, an inherent characteristic of survival data is censoring. Right censored data which is the commonest form of censoring occurs when survival times are greater than some defined time point [1, 2]. The generated data used in this study contains right-censored data.

Proportional hazards model, which also called Cox regression, is a popular method in analysis of survival data. This model is presented as

$$h\left(t \mid x\right) = h_0\left(t\right) \exp\left(\boldsymbol{\beta}'\mathbf{x}\right). \tag{1}$$

In this model, $\boldsymbol{\beta}$ is the regression coefficients vector, $\mathbf{x}$ is a vector of covariates, and $h_0(t)$ is the baseline hazard which is unspecified and a function of time. The likelihood and partial likelihood function for right censored survival data [18], given by (2) and (3):

$$\begin{aligned} L &= \prod_{i=1}^{N} f\left(t_i, x_i\right)^{\delta_i} S\left(t_i, x_i\right)^{1-\delta_i} \\ &= \prod_{i=1}^{N} \left(\frac{f\left(t_i, x_i\right)}{S\left(t_i, x_i\right)}\right)^{\delta_i} S\left(t_i, x_i\right) \\ &= \prod_{i=1}^{N} h\left(t_i, x_i\right)^{\delta_i} S\left(t_i, x_i\right), \end{aligned} \tag{2}$$

$$L = \prod_{i=1}^{D} \frac{\exp\left(\sum_{k=1}^{p} \beta_k X_{(i)k}\right)}{\sum_{j \in R(t_i)} \exp\left(\sum_{k=1}^{p} \beta_k X_{(i)k}\right)}, \tag{3}$$

δ_i, is censoring status where $\delta_i = 1$ if the observation is complete and $\delta_i = 0$ if it is censored. $R(t_i)$ is defined as the risk set at time t_i.

To fit the Cox regression and estimate $\boldsymbol{\beta}$, the partial likelihood in (3) is maximized using iteratively reweighted least squares to implement the Newton-Raphson method. However, in the high-dimensional case, this approach cannot be used to estimate $\boldsymbol{\beta}$; even $\boldsymbol{\beta}$ is not unique.

2.2. Neural Networks Model. An ANN consists of several layers. Layers are interconnected group of artificial neurons. In addition, each layer has a weight that indicates the amount of the effect of neurons on each other. Usually an ANN model has three layers that called input, hidden (middle), and output. The input layer contains the predictors. The hidden layer contains unobservable nodes and applies a nonlinear transformation to the linear combination of input layer. The value of each hidden node is a function of the predictors. The output layer contains the outcome which is some functions of the hidden units. In hidden and output layers, the exact form of the function depends on the network type and user definition (based on response variable).

There are different methods for learning in the NN. For example, in multiple layers perceptron (MLP), which is the most commonly used, the learning performs with minimization of the mean square error of the output and by back-propagation learning algorithm [16, 19, 20].

In this paper, we use the activation transfer function (g_h) as sigmoid function in hidden and in output layers (g_o). The ith response y_i for the predictor values H_{ih} is a nonlinear function as

$$\begin{aligned} y_i &= g_o\left(\mathbf{H}_i'\boldsymbol{\beta}\right) + \varepsilon_i, \\ H_{ih} &= g_h\left(\mathbf{X}_i'\boldsymbol{\alpha}_h\right). \end{aligned} \tag{4}$$

Note that $\mathbf{X}_i'$ is ith row of the input data matrix $\mathbf{X}$, H_{ih} is a nonlinear function of linear combination of input data, $\boldsymbol{\beta}$ is the vector weights of the hidden to the output units, and $\boldsymbol{\alpha}$ is the matrix weights of the input to the hidden units. Equation (4) together yield the MLP model:

$$y_i = g_o\left(\mathbf{H}_i'\boldsymbol{\beta}\right) + \varepsilon_i = g_o\left[\beta_0 + \sum_{h=1}^{H-1} \beta_h g_h\left(\mathbf{X}_i'\boldsymbol{\alpha}_h\right)\right] + \varepsilon_i. \tag{5}$$

By the sigmoid activation function, (5) can be written as below which is a nonlinear regression:

$$\begin{aligned} y_i &= \left[1 + \exp\left(-\mathbf{H}_i'\boldsymbol{\beta}\right)\right]^{-1} + \varepsilon_i \\ &= \left[1 + \exp\left(-\beta_0 - \sum_{h=1}^{H-1} \beta_h\left[1 + \exp\left(-\mathbf{X}_i'\boldsymbol{\alpha}_j\right)\right]^{-1}\right)\right]^{-1} + \varepsilon_i \\ &= g\left(\mathbf{X}_i, \boldsymbol{\beta}, \boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2, \ldots, \boldsymbol{\alpha}_H\right) + \varepsilon_i, \end{aligned} \tag{6}$$

where: $\boldsymbol{\beta}, \boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2, \ldots, \boldsymbol{\alpha}_H$ are unknown parameter vectors, $\mathbf{X}_i$ is a vector of known constants, and ε_i are residuals. The parameters (weights) can be estimated by optimizing some criterion function such as maximizing the log-likelihood function or minimizing the sum of squared errors.

In an MLP framework, a serious problem is overfitting. To control of the overfitting, usually a penalty term is added to the optimization criterion. To this, penalized least squares criterion for parameter estimation is given by [1]:

$$\begin{aligned} E^* &= \sum_{i=1}^{n} \left(y_i - \widehat{y}_i\right)^2 + p_\lambda\left(\boldsymbol{\beta}, \boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2, \ldots, \boldsymbol{\alpha}_H\right) \\ &= \sum_{i=1}^{n} \left(y_i - g\left(\mathbf{X}_i, \boldsymbol{\beta}, \boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2, \ldots, \boldsymbol{\alpha}_H\right)\right)^2 \\ &\quad + p_\lambda\left(\boldsymbol{\beta}, \boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2, \ldots, \boldsymbol{\alpha}_H\right), \end{aligned} \tag{7}$$

where the penalty term is $p_\lambda(\boldsymbol{\beta}, \boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2, \ldots, \boldsymbol{\alpha}_H) = \lambda(\sum \beta_i^2 + \sum \alpha_{ij}^2)$.

In likelihood schema which is often used in shrinkage method, an adaption of (7) is [8, 21]

$$L = -\log \text{likelihood} + \lambda\left(\sum \beta_i^2 + \sum \alpha_{ij}^2\right). \tag{8}$$

The penalty weight λ regulates between over- and underfitting. A best value of λ is between 0.001 and 0.1 and is chosen by cross-validation [1, 8]. In this paper, we use (8) to get the parameter estimated. It is mentioned that, for an outcome (the response variable) with two classes $\delta = (0/1)$, p_i is probability of event for the ith patient, and the error function provides the cross-entropy error function as

$$E = -\sum \left\{\delta_i \log\left(\widehat{p}_i\right) + \left(1 - \delta_i\right) \log\left(1 - \widehat{p}_i\right)\right\}. \tag{9}$$

An ANN can be modeled as a generalized linear modeling with nonlinear predictors [8–11]. Bignazoli et al. [8] introduced a method called partial logistic ANN, and Lisboa [22] developed it with fit smooth estimates of the discrete time hazard in structure. It is similar to MLP [23] with additional covariate, namely, time as an input and given by

$$\frac{h_p\left(x_p, t_k\right)}{1 - h_p\left(x_p, t_k\right)} = \exp\left(\sum_{h=1}^{N_h} w_h g\left(\sum_{i=1}^{N_i} w_{ih} x_{pi} + w_k t_k + b_h\right) + b\right), \tag{10}$$

where N_i and N_h denote the number of the input and the hidden nodes, respectively. b_h and b denote bias term in the hidden and output layers, respectively. After the estimation of the network weights, $\mathbf{w}$, a single output node estimates conditional failure probability values from the connections with the middle units, and the survivorship is calculated from the estimated discrete time hazard by multiplying the conditionals for survival over time interval. Then –log(likelihood) statistics could be obtained as [8]

$$E = -\sum_{p=1}^{\text{no of patient}} \sum_{k=1}^{t_1} \left[\delta_{pk} \log\left(h_p\left(x_p, t_k\right)\right) + \left(1 - \delta_{pk}\right) \log\left(1 - h_p\left(x_p, t_k\right)\right)\right], \tag{11}$$

where δ is the censoring indicator function.

2.3. Model Fitting. The ultimate goal of the learning process is to minimize the error by net. In training step, to fit the model by a fixed number of hidden nodes, we use penalized likelihood as

$$E^* = E + \lambda\left(\sum \beta_i^2 + \sum \alpha_{ij}^2\right). \tag{12}$$

By using this, we improve the convergence of the optimization and also control overfitting problem [1, 8, 9, 16, 21].

To identify the number of the hidden nodes and then model selection, Bayesian Information Criterion (BIC) and Network Information Criterion (NIC) [8, 23, 24], that is, generalization of Akaike Information Criterion (AIC), are calculated:

$$\begin{aligned} \text{BIC} &= -2 \times \log \text{likelihood} + \log(N) \times P, \\ \text{NIC} &= 2E^* + 2P, \end{aligned} \tag{13}$$

where P is the number of the parameters estimated, and N is the number of observations in training set. The best model is with the smallest value of these criterions. In addition, to assess prediction accuracy in validation (testing) group, we calculated classification accuracy and mean square error (MSE).

The best model is selected with the smallest value of MSE. The models considered 2, 3, 4, 5, 10, 15, and 20 hidden nodes. The weight decay was considered 0.012 which is chosen based on some empirical study [25].

At finally, in order to comparison of the Cox and ANN predictions, classification accuracy and concordance indexes were calculated. All simulations and comparisons were performed by R 2.14.1.

3. Simulation

In order to compare the accuracy of the predictions by ANN and Cox regression, four different simulation schemes, based on Monte Carlo simulation, were used. In each schema, hazard at any time t was considered as exponential form [26], namely, λ (Table 1). For each schema, 1,000 independent random observations were generated and then with the based on the relationship between exponential parameter and independent variables, survival times were generated. Afterward, the survival times were transformed as right censored. In this context, if generated time t_i is greater than the quantile of exponential function with parameter of λ, it is considered as censorship. This process was repeated 100 times. To access the accuracy of predictions, each sample is randomly divided to two parts. The first part, the training group, was consisting of 700 observations, and the 300 remainder observations were allocated to second group, that is, the testing group. Furthermore, in all simulation, the average rates of censorship were considered equal to 20%, 30%, 40%, 50%, 60%, 70%, and 80%. In addition, the models were considered with the main effects and without/with any interaction terms as simple and complex model, respectively.

In simulation 1 and 2, two covariates were used which was generated randomly from binomial and standard normal distributions. The models of these simulations were consisting of any and one interaction terms. In simulation 3, three covariates were used which was generated randomly from binomial and standard normal distributions, respectively. The model of this simulation has had two interaction terms. In simulation 4, four covariates were used which was generated randomly from binomial and standard normal distributions. The models of these simulations are complex and consist of two-, three- and four-interaction terms (Table 1).

Table 1: Describing of models* and its censoring status.

Model	Censoring status
Model (I): $\lambda = \exp(X_1 + 0.2X_2)$, $X_1 \sim \text{Ber}(0.5)$, $X_2 \sim N(0,1)$	Average censoring for training and testing sets equal to 20%, 30%, 40%, 50%, 60%, 70%, and 80%
Model (II): $\lambda = \exp(X_1 + 0.2X_2 + 0.2X_1X_2)$, $X_1 \sim \text{Ber}(0.5)$, $X_2 \sim N(0,1)$	Average censoring for training and testing sets equal to 20%, 30%, 40%, 50%, 60%, 70%, and 80%
Model (III): $\lambda = \exp(X_1 + X_2 + 0.5X_3 + 0.5X_1X_1 + 0.25X_1X_3 + 0.25X_1X_3)$, $X_1 \sim \text{Ber}(0.25)$, $X_2 \sim \text{Ber}(0.5)$, $X_3 \sim N(0,1)$	Average censoring for training and testing sets equal to 20%, 30%, 40%, 50%, 60%, 70%, and 80%
Model (IV): $\lambda = \exp[2(X_1 + X_2) + 1(X_3) + 0.5(X_4) + 1.0(X_1X_2 + X_1X_3 + X_1X_4) + 0.5(X_2X_3 + X_2X_4 + X_3X_4) + 0.25(X_1X_2X_3 + X_1X_2X_4 + X_1X_3X_4 + X_2X_3X_4) + 0.5(X_1X_2X_3X_4)]$, $X_1 \sim \text{Ber}(0.25)$, $X_2 \sim \text{Ber}(0.5)$, $X_3 \sim N(0,1)$, $X_4 \sim N(0,1)$,	Average censoring for training and testing sets equal to 20%, 30%, 40%, 50%, 60%, 70%, and 80%

*The hazard at any time t was considered as exponential form, and then the survival time was generated.

Table 2: Results of ANN models for simulated data.

# of H	MLP (H = 2)		MLP (H = 3)		MLP (H = 4)		MLP (H = 5)		MLP (H = 10)		MLP (H = 15)		MLP (H = 20)	
Model*	BIC**	MSE***	BIC	MSE	BIC	MSE	BIC	MSE	BIC	MSE	BIC	MSE	BIC	MSE
I:														
20%	0.213	1.055	0.203	1.148	0.205	1.252	0.214	1.302	0.223	1.405	0.254	1.387	0.278	2.235
30%	0.213	1.145	0.212	1.149	0.210	1.241	0.211	1.249	0.219	1.411	0.239	1.422	0.269	2.222
40%	0.212	1.134	0.217	1.163	0.209	1.321	0.209	1.392	0.228	1.433	0.249	1.473	0.283	2.314
50%	0.227	1.243	0.209	1.187	0.234	1.449	0.203	1.218	0.231	1.429	0.251	1.521	0.279	2.543
60%	0.256	1.289	0.217	1.251	0.239	1.443	0.211	1.325	0.241	1.511	0.262	1.611	0.281	2.468
70%	0.281	1.259	0.226	1.263	0.259	1.398	0.232	1.281	0.224	1.534	0.273	1.597	0.280	2.413
80%	0.278	1.337	0.232	1.258	0.266	1.527	0.223	1.216	0.249	1.617	0.261	2.018	0.266	2.731
II:														
20%	0.229	1.255	0.223	1.248	0.235	1.452	0.234	1.402	0.323	2.005	0.264	1.987	0.269	2.315
30%	0.254	1.431	0.282	1.428	0.235	1.665	0.243	1.723	0.259	1.835	0.263	2.650	0.276	2.123
40%	0.231	1.233	0.218	1.219	0.221	1.332	0.220	1.381	0.249	1.531	0.257	1.583	0.291	2.411
50%	0.237	1.342	0.222	1.317	0.241	1.539	0.213	1.318	0.251	1.689	0.266	1.591	0.280	2.561
60%	0.256	1.381	0.245	1.359	0.248	1.471	0.232	1.425	0.255	1.533	0.282	1.635	0.279	2.471
70%	0.281	1.259	0.236	1.263	0.263	1.411	0.223	1.279	0.263	1.576	0.291	1.608	0.291	2.409
80%	0.278	1.337	0.255	1.358	0.287	1.597	0.269	1.221	0.247	1.523	0.287	2.107	0.273	2.729
III:														
20%	0.283	1.261	0.235	1.274	0.250	1.408	0.241	1.299	0.240	1.539	0.277	1.607	0.291	2.433
30%	0.269	1.344	0.241	1.262	0.260	1.532	0.249	1.284	0.252	1.623	0.258	2.208	0.269	2.751
40%	0.367	1.204	0.259	1.888	0. 202	1.883	0.202	2.329	0.247	2.428	0.222	2.222	0.264	1.895
50%	0.241	1.358	0.232	1.321	0.252	1.541	0.220	1.327	0.263	1.699	0.277	1.603	0.288	2.593
60%	0.261	1.421	0.255	1.362	0.259	1.485	0.244	1.432	0.274	1.543	0.295	1.688	0.280	2.528
70%	0.289	1.371	0.267	1.274	0.271	1.422	0.253	1.269	0.292	1.552	0.286	1.662	0.303	2.687
80%	0.286	1.345	0.273	1.366	0.281	1.622	0.265	1.482	0.295	1.538	0.302	2.134	0.333	2.869
IV:														
20%	0.243	1.388	0.269	1.422	0.239	1.365	0.241	1.350	0.270	1.708	0.288	1.694	0.293	2.585
30%	0.267	1.435	0.259	1.471	0.230	1.488	0.237	1.499	0.280	1.638	0.285	1.679	0.311	2.569
40%	0.290	1.387	0.273	1.365	0.282	1.433	0.268	1.328	0.288	1.595	0.296	1.675	0.298	2.701
50%	0.292	1.667	0.281	1.621	0.283	1.560	0.271	1.634	0.279	1.807	0.288	1.632	0.302	2.325
60%	0.298	1.832	0.204	1.834	0.199	2.386	0.161	2.912	0.158	2.126	0.159	2.325	0.173	2.667
70%	0.238	2.181	0.239	2.174	0.247	2.392	0.237	2.403	0. 253	2.359	0.233	2.249	0.256	2.411
80%	0.300	2.087	0.301	1.902	0.321	2.099	0.292	2.077	0.295	2.364	0.289	2.630	0.322	2.716

*Four models with different rates of censoring.
**Average of BIC for learning set (700 cases with 100 replications).
***Average of MSE for testing set (300 cases with 100 replications).

Table 3: Results of concordance indexes of simulation study in testing subset (300 cases with 100 replications).

Model*	ANN	Cox Reg.
I:		
20%	0.821 ± 0.028	0.820 ± 0.031
30%	0.819 ± 0.027	0.820 ± 0.030
40%	0.824 ± 0.028	0.823 ± 0.030
50%	0.823 ± 0.027	0.823 ± 0.029
60%	0.823 ± 0.027	0.822 ± 0.030
70%	0.825 ± 0.027	0.822 ± 0.032
80%	0.826 ± 0.030	0.823 ± 0.031
II:		
20%	0.818 ± 0.029	0.817 ± 0.031
30%	0.817 ± 0.034	0.815 ± 0.046
40%	0.818 ± 0.032	0.818 ± 0.031
50%	0.816 ± 0.035	0.814 ± 0.032
60%	0.816 ± 0.031	0.813 ± 0.032
70%	0.814 ± 0.033	0.811 ± 0.031
80%	0.812 ± 0.034	0.808 ± 0.036
III:		
20%	0.808 ± 0.030	0.798 ± 0.035
30%	0.799 ± 0.033	0.790 ± 0.033
40%	0.812 ± 0.043	0.795 ± 0.038
50%	0.805 ± 0.028	0.795 ± 0.030
60%	0.804 ± 0.028	0.790 ± 0.033
70%	0.805 ± 0.028	0.791 ± 0.034
80%	0.803 ± 0.028	0.790 ± 0.032
IV:		
20%	0.764 ± 0.033	0.759 ± 0.036
30%	0.773 ± 0.029	0.762 ± 0.033
40%	0.777 ± 0.030	0.759 ± 0.035
50%	0.779 ± 0.035	0.764 ± 0.034
60%	0.812 ± 0.039	0.790 ± 0.036
70%	0.764 ± 0.036	0.744 ± 0.035
80%	0.766 ± 0.035	0.741 ± 0.037

*Four models with different rates of censoring.

The model selection is based on BIC for learning set and SSE criterion for the testing subset data as a verification. The results in Table 2 show that the simple model performs with less hidden node but complex model performs better with more hidden nodes. The MSE values confirm these results (Table 2).

In the next step, to compare of ANN and Cox regression predictions, concordance indexes were calculated from classification accuracy table in testing subset. Concordance index was reported as a generalization of the area under receiver operating characteristic curve for censored data [27, 28]. This index means that the proportion of the cases that are classified correctly in noncensored (event) and censored groups and 0 to 1 values indicated as the ability of the models accuracy. The concordance index of ANN and Cox regression models was reported in Table 3. The results of simulation study in simpler model showed that there was not any different between the predictions of Cox regression and NN models. But NN predictions were better than Cox regression predictions in complex model with high rates of censoring.

4. Conclusion

In this paper, we presented two approaches for modeling of survival data with different degrees of censoring: Cox regression and neural network models. A Monte-Carlo simulation study was performed to compare predictive accuracy of Cox and neural network models in simulation data sets.

In the simulation study, four different models were considered. The rate of censorship in each of these models was considered from 20% up 80%. These models were considered with the main effects and also with the interaction terms. Then the ability of these models in prediction was evaluated. As was seen, in simple models and with less censored cases, there was little difference in ANN and Cox regression models predictions. It seems that for simpler models, the levels of censorship have no effect on predictions, but the predictions in more complex models depend on the levels of censorship. The results showed that the NN model for more complex models was provided better predictions. But for simpler models predictions there was not any different in results. This result was consistent with the finding from Xiang's study [26]. Therefore, NN model is proposed in two cases of: (1) occurrence of high censorship (i.e., censoring rate of 60% and higher) and/or (2) in the complex models (i.e., with many covariates and any interaction terms). This is a very good result and can be used in practical issues which often are faced on with many numbers of variables and also many cases of censorship. For that reason, in these two cases the ANN strategy can be used as an alternate of traditional Cox model. Finally, it is mentioned that there are some flexible alternative methods, such as piecewise exponential and grouped time models which can be used for survival data and then its ability compared with ANN model.

Acknowledgment

The authors wish to express their special thanks to referees for their valuable comments.

References

[1] E. T. Lee and J. W. Wang, *Statistical Methods for Survival Data Analysis*, Wiley Series in Probability and Statistics, Wiley-Interscience, Hoboken, NJ, USA, 3rd edition, 2003.

[2] V. Lagani and I. Tsamardinos, "Structure-based variable selection for survival data," *Bioinformatics*, vol. 26, no. 15, pp. 1887–1894, 2010.

[3] M. H. Kutner, C. J. Nachtsheim, and J. Neter, *Applied Linear Regression Models*, McGraw-Hill/Irwin, New York, NY, USA, 4th edition, 2004.

[4] W. G. Baxt and J. Skora, "Prospective validation of artificial neural networks trained to identify acute myocardial infarction," *Lancet*, vol. 347, pp. 12–15, 1996.

[5] B. A. Mobley, E. Schecheer, and W. E. Moore, "Prediction of coronary artery stenosis by artificial networks," *Artificial Intelligence in Medicine*, vol. 18, pp. 187–203, 2000.

[6] D. West and V. West, "Model selection for medical diagnostic decision support system: breast cancer detection case," *Artificial Intelligence in Medicine*, vol. 20, pp. 183–204, 2000.

[7] F. Ambrogi, N. Lama, P. Boracchi, and E. Biganzoli, "Selection of artificial neural network models for survival analysis with genetic algorithms," *Computational Statistics & Data Analysis*, vol. 52, no. 1, pp. 30–42, 2007.

[8] E. Biganzoli, P. Boracchi, L. Mariani, and E. Marubini, "Feed forward neural networks for the analysis of censored survival data a partial logistic regression approach," *Statistics in Medicine*, vol. 17, pp. 1169–1186, 1998.

[9] E. Biganzoli, P. Boracchi, and E. Marubini, "A general framework for neural network models on censored survival data," *Neural Networks*, vol. 15, pp. 209–218, 2002.

[10] E. M. Bignazoli and P. Borrachi, "The Partial Logistic Artificial Neural Network (PLANN): a tool for the flexible modelling of censored survival data," in *Proceedings of the European Conference on Emergent Aspects in Clinical Data Analysis (EACDA '05)*, 2005.

[11] E. M. Bignazoli, P. Borrachi, F. Amborgini, and E. Marubini, "Artificial neural network for the joint modeling of discrete cause-specific hazards," *Artificial Intelligence in Medicine*, vol. 37, pp. 119–130, 2006.

[12] R. Bittern, A. Cuschieri, S. D. Dolgobrodov et al., "An artificial neural network for analysing the survival of patients with colorectal cancer," in *Proceedings of the European Symposium on Artificial Neural Networks (ESANN '05)*, Bruges, Belgium, April 2005.

[13] K. U. Chen and C. J. Christian, "Using back-propagation neural network to forecast the production values of the machinery industry in Taiwan," *Journal of American Academy of Business, Cambridge*, vol. 9, no. 1, pp. 183–190, 2006.

[14] C. L. Chia, W. Nick Street, and H. W. William, "Application of artificial neural network-based survival analysis on two breast cancer datasets," in *Proceedings of the American Medical Informatics Association Annual Symposium (AMIA '07)*, pp. 130–134, Chicago, Ill, USA, November 2007.

[15] A. Eleuteri, R. Tagliaferri, and L. Milano, "A novel neural network-based survival analysis model," *Neural Networks*, vol. 16, pp. 855–864, 2003.

[16] B. D. Ripley and R. M. Ripley, "Neural networks as statistical methods in survival analysis," in *Clinical Applications of Artificial Neural Networks*, pp. 237–255, Cambridge University Press, Cambridge, UK, 2001.

[17] B. Warner and M. Manavendra, "Understanding neural networks as statistical tools," *Amstat*, vol. 50, no. 4, pp. 284–293, 1996.

[18] J. P. Klein and M. L. Moeschberger, *Survival Analysis: Techniques for Censored and Truncated Data*, Springer, New York, NY, USA, 2th edition, 2003.

[19] J. W. Kay and D. M. Titterington, *Statistics and Neural Networks*, Oxford University Press, Oxford, UK, 1999.

[20] E. P. Goss and G. S. Vozikis, "Improving health care organizational management through neural network learning," *Health Care Management Science*, vol. 5, pp. 221–227, 2002.

[21] R. M. Ripley, A. L. Harris, and L. Tarassenko, "Non-linear survival analysis using neural networks," *Statistics in Medicine*, vol. 23, pp. 825–842, 2004.

[22] P. J. G. Lisboa, H. Wong, P. Harris, and R. Swindell, "A Bayesian neural network approach for modeling censored data with an application to prognosis after surgery for breast cancer," *Artificial Intelligence in Medicine*, vol. 28, no. 1, pp. 1–25, 2003.

[23] C. M. Bishop, *Neural Networks for Pattern Recognition*, The Clarendon Press Oxford University Press, New York, NY, USA, 1995.

[24] B. D. Ripley, *Pattern Recognition and Neural Networks*, Cambridge University Press, Cambridge, UK, 1996.

[25] N. Fallah, G. U. Hong, K. Mohammad et al., "Nonlinear Poisson regression using neural networks: a simulation study," *Neural Computing and Applications*, vol. 18, no. 8, pp. 939–943, 2009.

[26] A. Xiang, P. Lapuerta, A. Ryutov et al., "Comparison of the performance of neural network methods and Cox regression for censored survival data," *Computational Statistics & Data Analysis*, vol. 34, no. 2, pp. 243–257, 2000.

[27] F. E. Harrell, R. M. Califf, D. B. Pryor et al., "Evaluating the yield of medical tests," *The Journal of the American Medical Association*, vol. 247, pp. 2543–2546, 1982.

[28] F. E. Harrell, K. L. Lee, R. M. Califf et al., "Regression modeling strategies for improved prognostic prediction," *Statistics in Medicine*, vol. 3, pp. 143–152, 1984.