Training artificial neural networks directly on the concordance index for censored data using genetic algorithms.

Kalderstam, Jonas; Edén, Patrik; Bendahl, Pär-Ola; Forsare, Carina; Fernö, Mårten; Ohlsson, Mattias

Published in:
Artificial Intelligence in Medicine

DOI:
10.1016/j.artmed.2013.03.001

2013

Document Version:
Other version

Training artificial neural networks directly on the concordance index for censored data using genetic algorithms

Jonas Kalderstam[a,*], Patrik Eden[a], Pär-Ola Bendahl[b], Carina Strand[b], Mårten Fernö[b], Mattias Ohlsson[a]

[a] *Computational Biology and Biological Physics group, Department of Astronomy and Theoretical Physics, Lund University, Sölvegatan 14A, SE-22362 Lund, Sweden*
[b] *Department of Oncology, Clinical Sciences Lund, Lund University, Skåne University Hospital, SE-22185 Lund, Sweden*

Abstract

Objective: The concordance index (c-index) is the standard way of evaluating the performance of prognostic models in the presence of censored data. Constructing prognostic models using artificial neural networks (ANNs) is commonly done by training on error functions which are modified versions of the c-index. Our objective was to demonstrate the capability of training directly on the c-index and to evaluate our approach compared to the Cox proportional hazards model.

Method: We constructed a prognostic model using an ensemble of ANNs which were trained using a genetic algorithm. The individual networks were trained on a non-linear artificial data set divided into a training and test set both of size 2000, where 50% of the data was censored. The ANNs were also trained on a data set consisting of 4 042 patients treated for breast cancer

*Corresponding author. Tel.: +46 222 34 94
Email address: jonask@thep.lu.se (Jonas Kalderstam)

spread over five different medical studies, 2/3 used for training and 1/3 used as a test set. A Cox model was also constructed on the same data in both cases. The two models' c-indices on the test sets were then compared. The ranking performance of the models are additionally presented visually using modified scatter plots.

Results: Cross validation on the cancer training set did not indicate any non-linear effects between the covariates. An ensemble of 30 ANNs with one hidden neuron was therefore used. The ANN model had almost the same c-index score as the Cox model (c-index = 0.70 and 0.71 respectively) on the cancer test set. Both models identified similarly sized low risk groups with at most 10% false positives, 49 for the ANN model and 60 for the Cox model, but repeated bootstrap runs indicate that the difference was not significant. A significant difference could however be seen when applied on the non-linear synthetic data set. In that case the ANN ensemble managed to achieve a c-index score of 0.90 whereas the cox model failed to distinguish itself from the random case (c-index = 0.49).

Conclusions: We have found empirical evidence that ensembles of ANN models can be optimized directly on the c-index. Comparison with a Cox model indicates that near identical performance is achieved on a real cancer data set while on a non-linear data set the ANN model is clearly superior.

Keywords: Survival analysis, genetic algorithms, artificial neural networks, concordance index, breast cancer recurrence

---

1. Introduction

Given a binary classification task and a machine learning model that provides a score (which together with a threshold value can split the data set into two classes), the area under the receiver operating characteristic (ROC) curve is a very common performance measure. It gives the probability that a randomly chosen case from class 1 has a higher score than a randomly chosen case from class 0 [1].

In survival analysis, the focus is to analyze or predict the *survival time* – the time to occurrence of the event of interest, e.g. death or recurrence of cancer. A property of survival data is the existence of censored cases, where the full survival time is unknown, e.g. if an event occurs for a reason other than the one of interest, or if a patient reaches a pre-defined follow-up time. Censored data adds the information that the patient was event-free until the time of censoring. Removing censored data therefore introduces a bias.

To measure the performance of prognostic survival models the concordance index (c-index) [2] is very common and represents a natural extension of the area under the ROC curve for survival data. The c-index is the fraction of all *usable* pairs for which the predictions and the outcomes are in concordance. Usable pairs are either two non-censored entries with different survival times or one censored and one non-censored entry where the survival time is shorter than the censored follow-up time. A c-index of 1.0 indicates a perfect ordering and a value of 0.5 is no better than random ordering.

One of the conventional approaches when dealing with censored survival data is the Cox proportional hazards model [3]. It is based on the assumption of proportional hazards, meaning that the significance of a covariate

is assumed to be a multiplicative of a base hazard. Because the underlying hazard function is common for all patients, their respective prognoses are constrained to be proportional [4] which becomes a greater limitation the more heterogeneous the data set becomes. The standard Cox model is also unable to include non-linear relations between covariates. Multiple ideas have been presented to introduce non-linearity in the Cox model including fractional polynomials [5, 6] and artificial neural networks (ANNs) [7]. A more flexible approach can be found in the work of Biganzoli et al. [8] where the hazard function is expressed as a set of discrete hazard rates, modeled by ordinary multilayer ANNs. Here the hazard function depends (non-linearly) on both time and the covariates. This approach has also been extended to include cause-specific hazards [9] and to use the Bayesian framework when training the ANNs, which allows for a natural ranking of the covariates [10, 11].

In this study the focus will be on the c-index itself, specifically models providing a prognostic index which directly maximizes the c-index. The purpose of such a prognostic index is not to predict survival times, but rather to order patients according to survival. In many clinical applications it is common to divide patients into high- and low-risk groups as a basis for therapy or triage. One can also find clinical settings where it is important to be able to predict whether an event will eventually occur or not (e.g. recurrence of breast cancer). Both these situations can be accomplished using a prognostic index, thereby avoiding modeling of actual survival times.

To allow for non-linear interactions the models will be based on ANNs and deployed in ensembles to increase generalization. Many machine learning techniques use gradients during training, and are therefore ill suited to

maximize the rank-based c-index. Yan et al. [12] overcame this by introducing a smooth approximation to the step function. Van Belle et al. have developed support vector machines for survival analysis including c-index optimization [13]. Another approach can be found by Raykar et al. [14] where bounds were derived for the c-index and used in the optimization. Our approach for optimizing on the c-index is based on choosing an optimization algorithm, more specifically a genetic algorithm, that does not require the computation of any gradients.

The application used in this paper is a study on the recurrence of breast cancer. We aimed at investigating if biomarkers such as age, tumor size and the amount of oestrogen receptors, can be used to construct a prognostic index for distant recurrence of cancer. The patient population used is large and heterogeneous. A simulated data set was also used to demonstrate the non-linear capabilities of the proposed model.

2. Materials and methods

2.1. Study population

The real cancer data set used is a compilation of data from five breast cancer studies, four Swedish and one Danish [15–19]. Time from diagnosis of primary breast cancer to first distant recurrence or last follow-up without distant recurrence was available for all patients. The inclusion criteria for the different sub-studies varied with regard to age, lymph node status and treatment. Common criteria were: no adjuvant chemotherapy and complete data on follow-up, number of positive lymph nodes in the axilla, tumor size and age at diagnosis. This resulted in a cohort of 4 042 patients, with 73%

censored cases with a maximum follow-up time of 10 years.

2.2. Covariates

In the study, data was provided for age, oestrogen and progesterone receptor, lymph node status, tumor size, histological grade and HER2 status. However, there were many missing values. Covariates with more than 50% missing were excluded (see Table 1), leaving a final list of five unique covariates (see Table 2). For the included covariates, the simplest possible imputation of missing values was used: the mean value of known values. In an analysis where the relative importance of different covariates is assessed, this sub-optimal imputation penalizes covariates with many missing values, which we considered a desired feature. Furthermore, we also included a set of computed covariates to possibly aid the modeling. Although non-linear machine learning models should be able to find needed transformations during training, simple transformations such as the logarithm and commonly used dichotomisations were added to the list of covariates. The dichotomisations were part of the individual studies. One such dichotomisation, `ErPos`, which was a binarization of the `Er` covariate, actually had less missing data than `Er`. Using both `Er` and `ErPos` can therefore add information. The reason for the mismatch between the number of `Er` and `ErPos` data is explained by the fact that they entered in the patient records as separate variables. In summary 12 covariates were used in the breast cancer data set (Table 2).

2.3. Simulated data

The simulated data is generated to specifically demonstrate a case where non-linear modeling is needed to obtain a good c-index. Any monotonically

Table 1: Covariates that were excluded due to more than 50% missing data, where `HistGrad` = histological grade (in 3 steps), `MitoticGrade` = subjective estimate of cell proliferation, `TubularGrade` = subjective estimate of amount of tubular formations, `NuclearGrade` = subjective estimate of irregularity of cell nuclei, `HER2Pos` = HER2 status measured by the Fluorescence In Situ Hybridization (FISH) test and `Ki67` = Ki-67 reactivity measured by immunohistochemical (IHC) staining. The type of variable, the mean and the standard deviation (SD) are also presented.

Covariate	Type	Mean	SD	Missing data
`HistGrad`	Real	2.10	(0.69)	56%
`MitoticGrade`	Real	1.80	(0.80)	72%
`TubularGrade`	Real	2.57	(0.62)	72%
`NuclearGrade`	Real	2.40	(0.59)	72%
`Ki67`	Real	18.21	(20.00)	73%
Covariate	Type	Ones	Zeros	Missing data
`Her2Pos`	Binary	120	755	78%

increasing function can be modeled perfectly using a linear model like the Cox model, when evaluated using the c-index. The simulated data model is therefore highly non-monotonic, to make sure that any linear model will fail to produce accurate results.

For the simulated data we used 10 covariates, similar to the number for the breast cancer data, and the survival time t was generated according to the following model

$$t = (x_0+x_1+x_2-15)^2+(x_3+x_4-10)^2+(x_5+x_6+x_7-15)^2+(x_8+x_9-10)^2 \quad (1)$$

where all covariates x_i were drawn from a continuous uniform distribution between 0 and 10. Random noise was added to both the survival time and

Table 2: Summary of the covariates used in the modeling, where `Age` = Age in years at diagnosis of primary cancer, `TumorSize` = tumor size in mm, `NumLymph` = number of positive lymph nodes in the axilla, `ER` = oestrogen receptor (fmol/mg protein), `PgR` = progesterone receptor (fmol/mg protein), Apart from the logarithm transformations the following dichotomisations were also used: `Size20` = 1 if `TumorSize` > 20, `LymphPos` = 1 if `NumLymph` > 0, `ERPos` = 1 if `Er` > 25, `PgRPos` = 1 if `PgR` > 25.

Covariate	Type	Mean	SD	Missing data
`Age`	Real	60.04	(10.72)	0%
`TumorSize`	Real	23.57	(10.95)	0%
`NumLymph`	Real	2.08	(3.65)	0%
`log(1+NumLymph)`	Real	0.72	(0.82)	0%
`Er`	Real	197.71	(291.58)	41%
`log(1+ER)`	Real	4.03	(2.01)	41%
`PgR`	Real	180.96	(350.96)	44%
`log(1+PgR)`	Real	3.31	(2.40)	44%
Covariate	Type	Ones	Zeros	Missing data
`Size20`	Binary	2220	1822	0%
`LymphPos`	Binary	2191	1851	0%
`ERPos`	Binary	2049	858	28%
`PgRPos`	Binary	1277	1010	43%

the covariates with an exponential probability distribution,

$$p(\epsilon) = \frac{1}{2\beta} \exp\left(-\frac{|\epsilon|}{\beta}\right) \tag{2}$$

where $p(\epsilon)$ denotes the probability of adding noise with a value of ϵ. β was 0.3 for covariate noise and 1.0 for survival time noise. Survival times were kept positive by ignoring cases where the noise would make them negative.

Furthermore, 50% of the data was censored, where follow-up times were determined by uniformly random numbers between 0 and the corresponding survival time.

Using the above model for the simulated data, we generated a training data set and an independent test data set, both of size 2000, where all survival times were positive.

2.4. The concordance index

To define the c-index we introduce the survival time t_j for patient j. In the case of a censored patient, t_j is the follow-up time. Let p_j be the prognostic index for patient j, with the aim of sorting patients according to actual survival times. A pair (p_i, p_j) is said to be in concordance if $p_i > p_j$ and $t_i < t_j$, assuming non-censored events, meaning that a higher prognostic index corresponds to a shorter survival time. If patient j was censored, a comparison with patient i can only be made if the follow-up time t_j was larger than the event time t_i. Again, such a pair is in concordance if $p_i > p_j$. No comparison can be made if both patients were censored. The c-index is simply the fraction of comparable pairs in concordance.

Formally, let Ω be the set of usable pairs of patients. A pair (i, j) is usable if both patients had an event with the condition $t_i < t_j$, or patient j was censored with a follow-up time larger than the event time of patient i. Given Ω the c-index is computed as

$$\text{c-index} = \frac{1}{|\Omega|} \sum_{(i,j)\in\Omega} I(p_i, p_j) \tag{3}$$

where $|\Omega|$ is the number of patient pairs in Ω and the indication function I

is defined as

$$I(p_i, p_j) = \begin{cases} 1 & \text{if } p_i > p_j, \\ 0 & \text{otherwise.} \end{cases}$$

The interpretation of the c-index follows naturally from its definition as the estimate that a patient with a higher prognostic index will have an event within a shorter time than a patient with a smaller prognostic index.

2.5. The prognostic index model

Given a set of K covariates $\mathbf{x}_i = (x_{i1}, x_{i2}, \ldots, x_{iK})$ for each patient i, the task is then to compute a prognostic index $p(\mathbf{x}_i)$ such that a large index will indicate a high risk of an event. We will model $p(\mathbf{x})$ using ANNs, specifically multilayer perceptrons with one hidden layer. Such a model is given by

$$p(\mathbf{x}) = \sum_{j=1}^{J} \omega_j \cdot \varphi \left(\sum_{k=1}^{K} \tilde{\omega}_{jk} x_k + \tilde{\omega}_{j0} \right) + \omega_0 \tag{4}$$

where ω_j, $\tilde{\omega}_{jk}$ are called weights and are parameters in the model. The integer J is the number of hidden neurons and $\varphi()$ a non-linear function, here set to the hyperbolic tangent function tanh(). This ANN provides means of modeling complex relations between the covariates by increasing the J parameter (number of hidden neurons). It can also easily be turned into a linear model by setting $J = 1$ and $\varphi(x) = x$.

The weights $(\omega_j, \tilde{\omega}_{jk})$ are determined by minimizing an objective function. Usually this objective function is differentiable with respect to the weights, which allows for gradient-based optimization methods. However, in our case our objective function is the c-index, which cannot be differentiated with respect to the weights. This limits the number of minimization methods one can use.

2.6. Training using genetic algorithms

To treat survival analysis as a ranking problem, we have chosen to utilize a *genetic algorithm* which allows us to train directly on the c-index without requiring gradient information. A genetic algorithm optimizes a solution by mimicking evolution: simulating mutation and sexual reproduction.

Many possible implementations of genetic algorithms exist. Montana and Davis [20] experimented with many different variations and evaluated their relative effectiveness for some cases. Our implementation is based on what they determined were the most effective procedures, which we found performed well also in our case. The procedure is as follows:

Initialize the population

To begin with, 50 ANNs are initialized with random weights from the exponential distribution

$$p(\epsilon) = 2.5 \exp\left(-5|\epsilon|\right), \tag{5}$$

thus favoring smaller weights while allowing for larger weights in some cases. Smaller weights generally make training faster but sometimes larger weights are required to achieve better results. To keep the procedure as simple as possible, the actual architecture of the ANNs is fixed. As a last step in the initialization, the ANNs are evaluated and sorted according to their performance. This sorting is maintained throughout the entire training.

Create a new generation

New ANNs are created by crossover where the child ANN inherits each weight from one of its two parents. An ANN with rank k, when sorted by

performance, is selected as parent with the probability $p(k)$,

$$p(k) \propto (0.95)^{k-1}. \tag{6}$$

This results in a 90% probability to select a rank of 35 or less. Once a new ANN is created it is subject to mutation. Each weight ω is modified with probability $\frac{1}{4}$ according to $\omega = \omega + \epsilon$ where ϵ is a random number from the same distribution as for the initial weights (see equation 5).

This "new-born" ANN is now evaluated and inserted in the population. Then, the ANN with the worst rank is deleted. This keeps the population size constant and weeds out poor performing ANNs. A generation is elapsed when the number of generated children equals the population size. The genetic algorithm runs for a fixed number of generations.

2.7. Ensembles of prognostic models

A common approach to counter over-fitting, and also to increase the performance, is to use an ensemble of ANNs instead of a single one. Often, an ensemble result is merely the average output of its members. Averaging clearly only works if the members are different, then the ensemble result will often perform better than any of the individual ANNs. To promote this needed diversity we trained ANNs on different parts of the training data by dividing it into three random parts of equal size. From this partition, three new smaller training data sets were created by combining two of the three parts, thus resulting in three member ANNs. This procedure was then repeated a number of times to obtain the desired ensemble size.

With a rank-based objective function, the ensemble result cannot be generated by direct averaging of individual member outputs, since these outputs

need not lie within any defined range or conform to any joint scale. Thus, outputs can be expected to differ wildly between ANNs even if they are equivalent in terms of the c-index. To be able to average outputs, they will first be transformed into ranks by comparing with training data outputs for each ANN. Let N_i be the number of training data that was used to train ensemble

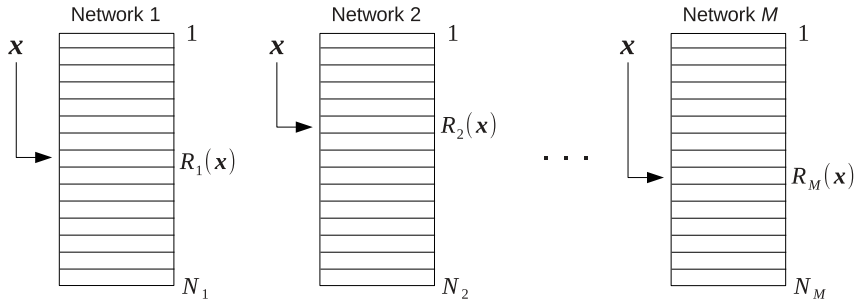

Figure 1: A new patient $\mathbf{x}$ obtains a rank number $R_i(\mathbf{x})$ for each ANN i by comparing with the training data output list for each ANN. The rank numbers are transformed into a *normalized relative rank* by normalizing with the size of the training data set. The final ensemble output is the average of these relative ranks.

member i. The output $y_i(\mathbf{x})$ for ANN i and patient $\mathbf{x}$ will give rise to a rank $R_i(\mathbf{x})$. This rank is determined by inserting the output $y_i(\mathbf{x})$ into the sorted list of training data outputs for ANN i. The rank $R_i(\mathbf{x})$ is now simply the position of $y_i(\mathbf{x})$ in the sorted list (see Figure 1). To allow for ANNs trained with different sizes of training sets, the rank is divided by $N_i + 1$, yielding a number between 0 and 1 called the **normalized relative rank** $\tilde{R}_i(\mathbf{x})$. From a c-index point of view, ANN output $y_i(\mathbf{x})$ and $\tilde{R}_i(\mathbf{x})$ are completely equivalent. Computing an ensemble output $y_c(\mathbf{x})$ is now straightforward and

is the average of normalized relative ranks,

$$y_c(\mathbf{x}) = \frac{1}{M} \sum_{i=1}^{M} \tilde{R}_i(\mathbf{x}) \tag{7}$$

where M is the size of the ensemble. In summary the prognostic index of each ANN has now been combined into an ensemble prognostic index.

2.8. Model selection and performance evaluation

For both simulated and clinical data, the performance was measured on a separate test data set. For the breast cancer data, a test set of one third of the data (1347 patients) was used. This was stratified with respect to censoring and individual studies. It was kept out of reach during the development phase and only used once during evaluation. For the simulated data, an independent test data set of size 2000 was generated with the same parameters as for the training data.

As with any modeling we must consider the possibility of over-fitting. One can formally argue that a rank-based objective function reduces the risk of over-fitting, since it only measures how well the network can sort the data. Once an ANN has been optimized to produce a perfect sorting (c-index = 1) there will be no further weight updates. This is different from the classification case where the training continues after perfect sorting, until the error function is zero. It is of course still possible to over-fit on the noise and in order to prevent that we limited the number of hidden neurons (model size). The best model size was selected using the K-fold cross validation scheme where a number of different model sizes were tested and the one with the best cross validation performance was selected. The

combined performance evaluation and model selection procedure, including the ensemble learning, is illustrated in Figure 2.

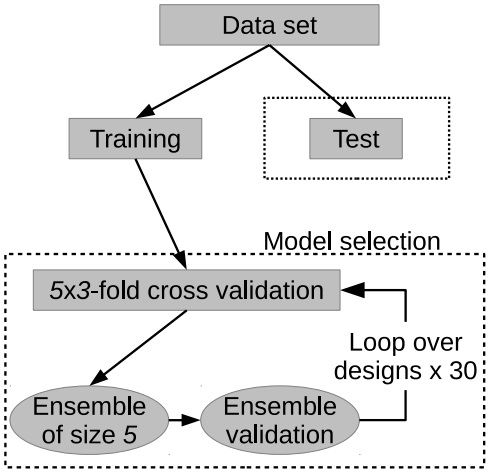

Figure 2: The performance of the models are evaluated using a separate test set, while the model selection is carried out using K-fold cross validation. For each model validated, called a design, 15 networks were trained using a 5x3-fold cross validation loop, in turn removing the validation part from the data set when training. By this scheme each data point will be part of the validation set 5 times, thus producing an ensemble validation result for the given design. This procedure was repeated 30 times with different random 5x3-fold divisions to average out random effects. Once the optimal model was selected, a final 10x3 ensemble was trained using the full training set.

2.9. Visualizing results

The results of the prognostic index model could be visualized using scatter plots, using the survival time on the x-axis and the prognostic index on the y-axis. If we assume non-censored data, a high c-index would imply all points to follow a monotonic curve with few "off-diagonal" points. A simple model would be to fit a straight line in the scatter plot and identify points above

this line as cases with overestimated risk and points below the line as cases with underestimated risk.

However, the presence of censored data makes it difficult to interpret such scatter plots, since there may be a very low correlation between a censored patients follow-up time and the prognostic index, still resulting in a good c-index. Put in other words, it is the very definition of (right) censored data that there are no valid comparisons to make regarding any potential overestimation.

We therefore introduced a modified scatter plot where the predicted prognostic index of a censored patient cannot be plotted above the straight line, which we defined by a least-squares linear fit to non-censored data. All censored data points above this line were instead plotted on the line.

2.10. Implementation details

The ANNs and genetic training procedures were implemented in Python with some computationally expensive procedures implemented in C. The running time scales as $O(n^2)$ with respect to the size of the training data set. The procedure was parallelized and training an ensemble of 30 networks using 30 CPU cores took some 20 minutes.

2.11. Comparing with the Cox model

For comparison, a Cox proportional hazards model was trained on the same data sets as the ANN models. We used the standard Cox model without any extensions (e.g. non-linear effects or time-varying covariates). Training and testing of this model was accomplished using the survival package in the R environment.

3. Results

Cancer data

The cross validation results for the breast cancer data can be found in Figure (3). This figure shows the cross validation c-index as a function of the number of hidden neurons (top graph). The best validation c-index obtained was 0.72, corresponding to models with one non-linear hidden neuron (lower graph). There were only small differences in cross validation performance when changing the model size, indicating limited non-linear interactions between the covariates. A final ensemble of 30 ANNs with one non-linear hidden neuron was trained on the full training data set. This model was then tested on the separate test set, resulting in a c-index of 0.70. The corresponding results for the Cox model, trained on the full training set, was 0.71.

To see if any difference between low risk groups from the ANN model and the Cox model existed, we first identified the largest group of patients in the test data with at least 90% survival after 10 years using the prognostic index from both models. The group identified by the Cox model had 60 patients (Figure 5), while the ANN model identified a group of 49 patients (Figure 4). The 10 year survival within the high risk group was 68% for both models. The groups are quite small and are probably not relevant for clinical use, but it should be noted that histological grade was not used in our modeling and this is a powerful prognostic marker.

To verify if there was any significance to the difference in group sizes, the models were tasked with identifying risk groups in the same fashion but with data that was bootstrapped from the test data. The results showed no significant difference between group sizes.

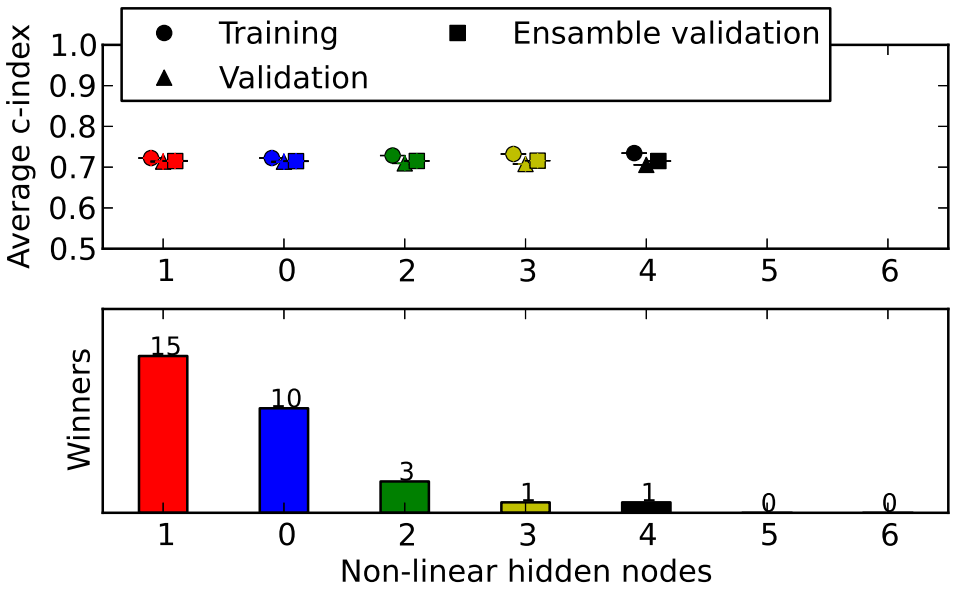

Figure 3: The c-index (cross validation results) as a function of the number of (non-linear) hidden neurons in the ANNs (top graph) where the left marker indicates average ANN training result, the middle marker represents average ANN validation result and the right marker indicates the average validation result for the ensemble. The plot also shows maximum and minimum deviations from the averages but the deviations are very small. The lower graph shows the number of times a given model complexity obtained the largest ensemble validation c-index. The winning complexity was 1 non-linear hidden neuron.

To visualize the correlation between the prognostic indices provided by the ANN model and the actual survival times, scatter plots were used. Figure 6 (left graph) shows the unmodified scatter plot for the test data, corresponding to a c-index of 0.70. The modified scatter plot, where overestimation of censored data is removed (see section 2.9) is shown in Figure 6 (right graph).

To assess the importance of the different covariates for the trained model, a similar idea to that of Nord et al. [21] was used. The c-index value for

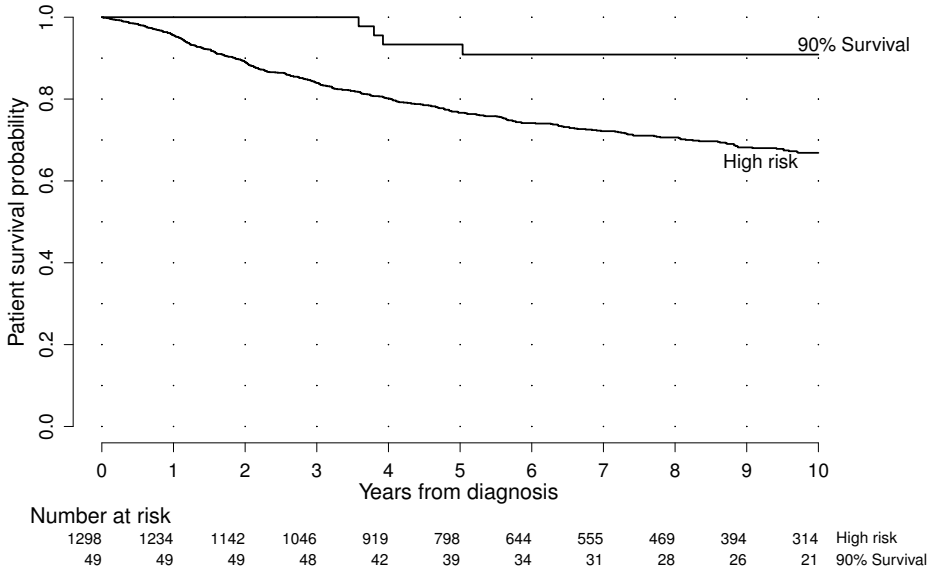

Figure 4: Kaplan-Meier plots for two groups of patients, based on the ANN prognostic index (test data).

the test set was used as a reference value. The importance of a covariate was then measured as the drop in c-index, compared to the reference, when the covariate was replaced by its mean value for the whole test set. This procedure was sequentially repeated for all covariates. The result can be seen in Table 3. The three most important covariates were the number of positive lymph nodes in the axilla, the logarithm of progesterone receptor measurements and age.

Simulated data

The cross validation results for the simulated data suggested a model size of 12 hidden neurons. An ensemble of 30 ANNs was used. The cross valida-

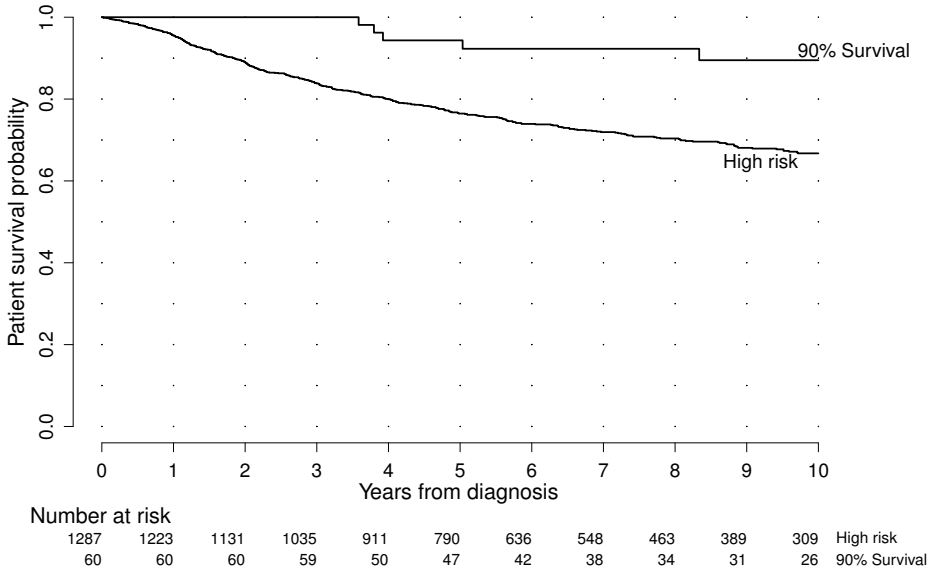

Figure 5: Kaplan-Meier plots for two groups of patients, based on the Cox model (test data).

tion c-index for this model was 0.89. For the simulated data an independent test set was generated using the same parameters as for the training data. The test set performance for the above model was 0.90 and the resulting scatter plots are shown in Figure 7. The left graph shows the raw scatter plot where the issue with censored data occurs again. Using the modified scatter plot the agreement is much better (right graph).

An advantage of using simulated data is that we know the correct survival times before any censoring was performed. This allows us to compare the predicted prognostic index with the *true* survival times (no censoring), for a model that was trained on censored data. This comparison can never be performed with real data sets. Figure 8 (left graph) shows the scatter plot

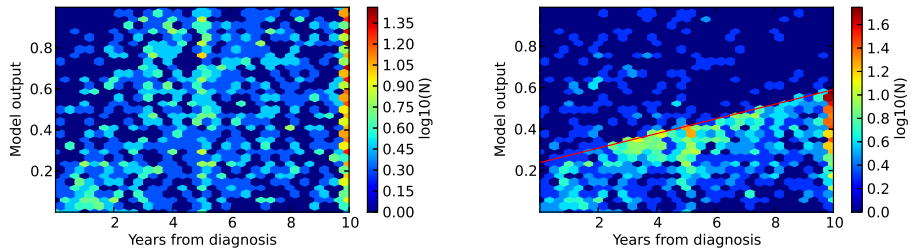

Figure 6: (Left graph) Scatter plot between the prognostic index and the actual survival times for the test data (c-index = 0.70). The colors indicate the amount of patients according to the scale on the right. (Right graph) Scatter plot of the same data with the added rule that a prognostic index for a censored patient can not be plotted above the red line. This is because as far as the c-index is concerned such an index would be considered (approximately) correct.

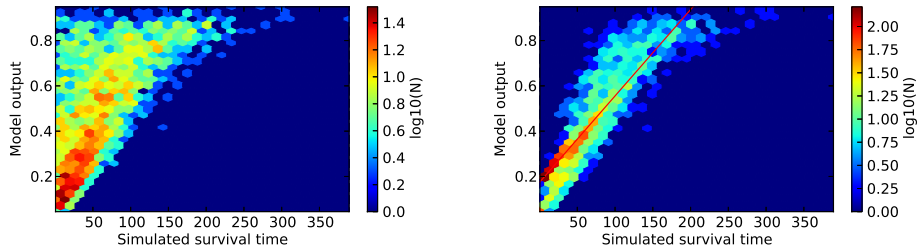

Figure 7: Scatter plots for the simulated data with a c-index of 0.90. The left graph demonstrates the impracticability of doing scatter plots with censored data. The right graph compensates for this by only plotting censored points below the diagonal which is also shown. A much more pleasing and visually accurate result is achieved.

between the prognostic index and the true survival times for the ANN model. The corresponding c-index was 0.87 in this case.

The result for the Cox proportional hazards model on the simulated data is, as expected, not good: c-index = 0.49. For random values, repeated com-

Table 3: Covariate importance as measured by the effect a given covariate has on the model when being replaced by its mean value. The drop in c-index was used to measure this effect and is presented in this table for all covariates in the model. The top most important ones were: number of positive lymph nodes in the axilla and the logarithm of progesterone receptor measurements.

Covariate(s)	Drop in c-index	Change relative to `NumLymph`
`NumLymph`	5.88	1.000
`log(1+PgR)`	1.97	0.336
`Age`	0.93	0.158
`TumorSize`	0.78	0.133
`PgRPos`	0.56	0.096
`log(1+NumLymph)`	0.22	0.038
`LymphPos`	0.20	0.033
`Size20`	0.14	0.023
`log(1+ER`	-0.05	-0.009
`ER`	0.04	0.007
`ERPos`	-0.03	-0.005
`PgR`	0.00	0.001

puter simulations of 2000 values (the size of the synthetic test set) indicate that the mean c-index is 0.50 and the standard deviation is 0.01. So it performs no better than random on the synthetic data set. This model cannot handle the square covariate dependency that is used to define the survival time for the simulated data (see equation 1)

Figure 8 (right graph) shows the corresponding scatter plot for the simulated data (test set), again using the true uncensored survival times.

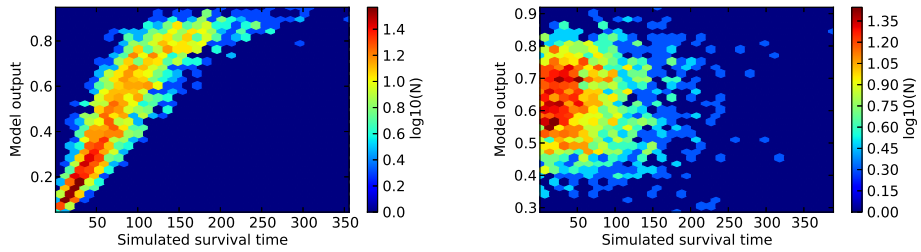

Figure 8: True scatter plots can only be generated when the uncensored information is available. The left graph shows the ANN model which achieves a c-index of 0.87 on the uncensored data. The right graph demonstrates the inability of the Cox model to perform better than random: c-index is 0.49.

4. Discussion

We have developed a prognostic index model for survival data based on an ensemble of ANNs that optimizes directly on the concordance index used for survival data.

Previous work concerning c-index optimization in survival analysis has largely been dominated by different approaches to approximate the c-index with a differentiable objective function, in order to utilize gradient techniques. We have avoided this step since optimization techniques such as genetic algorithms are very much able to achieve good results with the advantage of training directly on the metric of interest. Genetic algorithms are computationally more expensive than gradient techniques and this can be seen as a disadvantage. However, it is practically not a problem since computational resources available today are more than adequate even in a single machine and the resources are only required during the training procedure. A computational problem may arise for for very large data sets as calculating

the c-index is a $O(n^2)$ operation and must be calculated for each generation of networks in the genetic training.

In this study the prognostic index model was tested on a real world cancer data set of 4 042 patients comprised of five individual studies. The results show a negligible difference between our genetically trained ensemble of ANNs compared to the traditional Cox model. A direct difference between Cox modeling and our prognostic index is the fact that the Cox model is based on optimizing the partial likelihood and not the c-index. However, Raykar et al. [14] show a close connection between partial likelihood and c-index optimization. Furthermore, to analyze the ANN model with respect to the used covariates (cancer data only), a variable ranking procedure was employed. This showed that only a few covariates are important for the prediction, where the number of positive lymph nodes in the axilla came out as the most important one. Interestingly, training a model with the five most important covariates (according to Table 3) resulted in only a marginal reduction of the test c-index, indicating that for the purpose of maximizing the c-index only a subset of the covariates are needed. Feature selection using rank based measurements, such as the c-index, may have limitations, as pointed out by Cook et al. [22]. In light of such discussions the above ranking results may differ from other feature selection methods. Our ranking results should only be interpreted in terms of optimizing the c-index using the proposed ANN model.

In addition to comparing the c-index of the ANN and Cox models, we used the models to identify a low risk group defined as the largest possible group with the highest predicted survival chance with at most 10 percent

false positives. For both models the group sizes were less than 5 percent of the test set. By repeatably identifying groups with bootstrapped data, we determined that no significant difference between identified groups of the ANN and Cox models could be observed. It should be noted that one of the strongest factors for predicting recurrence, the histological grade, was not used in the modeling due to missing data. Histological grade is used in prognostic tools for breast cancer, such as the Nottingham Prognostic Index [23] and Adjuvant! Online [24]. Using histological grade in the current models would probably have improved the result for both models, as well as their ability to find low risk groups.

To challenge the proposed model on non-linear data, a simulated data set was used where the generated survival times depended non-linearly and non-monotonically on the covariates. The simulated data was designed to be impossible to handle using a Cox model, resulting in a c-index no better than random (see Figure 8, right graph). The ANN model could easily produce a prediction with the required non-linear relations between the covariates, resulting in a c-index of 0.90. 50% of the data was censored, but since the data was generated we had access to the uncensored data. The c-index between the model output and the uncensored simulated survival times was 0.87 indicating that our ANN model was able to approximate the underlying function despite the large fraction of censoring used during training (see Figure 8, left graph). The decrease from 0.90 in the censored case, to 0.87 in the uncensored case, also illustrates the inherent bias of the c-index itself to overestimate the performance of a prognostic model on censored data.

Further improvements could be made to the ANN training procedure.

The genetic algorithm itself is a rather standard approach and we have not endeavored to tweak the training parameters to their absolute best. The performance on non-linear data could be improved by finding the optimum parameters but it is unlikely that it would result in any significant benefit on the real cancer data which seems to be predominately linear. One idea for improvement would be to evolve the structure (e.g. complexity) of the ANNs together with the weights as in the NEAT algorithm [25].

5. Conclusion

In this paper we have proposed a prognostic index model for survival data that maximizes the concordance index. The model is based on ANN ensembles and is trained using a genetic algorithm. A normalized relative rank was developed to allow for an ensemble of individually c-index optimized models. We have explored ways of visualizing the correlation between prognostic indices and survival times in presence of censored data, using modified scatter plots. The model was tested on a breast cancer data set originating from five different studies and one simulated data set. Cox modeling was used for comparison and the results for the cancer data set shows near identical performance between the ANN and Cox models. The ANN was however able to correctly model a synthetic non-linear data where the Cox model could not perform better than random.

6. Acknowledgements

We are indebted to Signe Borgquist, Gunilla Chebil, Anna-Karin Falck, Dorthe Grabau, Martin Bak, Karin Jirström, Marie Klintman, Per Malm-

ström, Hans Olsson, Lisa Rydén, and Olle Stål for providing clinical data and follow-up. The study was supported by funds from the Swedish Foundation for Strategic Research (CREATE Health), Swedish Cancer Society, the Swedish Research Council, the Gunnar Nilsson Cancer Foundation, the Mrs. Berta Kamprad Foundation, The Anna and Edwin Berger's Foundation, Skåne University Hospital Research Foundations, Skåne County Council's Research and Development Foundation and Governmental Funding of Clinical Research within the National Health Service.

References

[1] Hanley, J.A., McNeil, B.J.. The meaning and use of the area under a receiver operating characteristic (ROC) curve. Radiology 1982;143(1):29–36.

[2] Harrell, F.E., Lee, K.L., Mark, D.B.. Multivariable prognostic models: issues in developing models, evaluating assumptions and adequacy, and measuring and reducing errors. Statistics in medicine 1996;15(4):361–87.

[3] Cox, D.R.. Regression models and life-tables. Journal of the Royal Statistical Society Series B (Methodological) 1972;34(2):187–220.

[4] Kay, R., Kinnersley, N.. On the use of the accelerated failure time model as an alternative to the proportional hazards model in the treatment of time to event data: a case study in influenza. Drug information journal 2002;36(3):571–9.

[5] Royston, P., Altman, D.G.. Regression Using Fractional Polynomials of Continuous Covariates: Parsimonious Parametric Modelling.

Journal of the Royal Statistical Society Series C (Applied Statistics) 1994;43(3):429–67.

[6] Royston, P., Sauerbrei, W.. A new approach to modelling interactions between treatment and continuous covariates in clinical trials by using fractional polynomials. Statistics in medicine 2004;23(16):2509–25.

[7] Faraggi, D., Simon, R.. A neural network model for survival data. Statistics in Medicine 1995;14(1):73–82.

[8] Biganzoli, E., Boracchi, P., Mariani, L., Marubini, E.. Feed forward neural networks for the analysis of censored survival data: a partial logistic regression approach. Statistics in Medicine 1998;17(10):1169–86.

[9] Biganzoli, E.M., Boracchi, P., Ambrogi, F., Marubini, E.. Artificial neural network for the joint modelling of discrete cause-specific hazards. Artificial intelligence in medicine 2006;37(2):119–30.

[10] Lisboa, P.J.G., Wong, H., Harris, P., Swindell, R.. A Bayesian neural network approach for modelling censored data with an application to prognosis after surgery for breast cancer. Artificial Intelligence in Medicine 2003;28(1):1–25.

[11] Taktak, A.F., Eleuteri, A., Aung, M.S., Lisboa, P.J., Desjardins, L., Damato, B.E.. Survival analysis in cancer using a partial logistic neural network model with Bayesian regularisation framework: a validation study. International Journal of Knowledge Engineering and Soft Data Paradigms 2009;1(3):277.

[12] Yan, L., Verbel, D., Saidi, O.. Predicting prostate cancer recurrence via maximizing the concordance index. In: Kohavi, R., editor. Proceedings of the 2004 ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '04. New York, NY, USA: ACM Press; 2004, p. 479.

[13] Van Belle, V., Pelckmans, K., Suykens, J.A.K., Van Huffel, S.. Support Vector Machines For Survival Analysis. In: Ifeachor, E., Anastasiou, A., editors. Proceedings of the third international conference on Computational Intelligence in Medicine and Healthcare (CIMED). 2007, p. 1–8.

[14] Raykar, V., Steck, H., Krishnapuram, B., Dehing-Oberije, C., Lambin, P.. On ranking in survival analysis: Bounds on the concordance index. In: Platt, J., Koller, D., Singer, Y., Roweis, S., editors. Advances in Neural Information Processing Systems 20; vol. 20. MIT Press, Cambridge, MA; 2008, p. 1209–16.

[15] Rydén, L., Jönsson, P.E., Chebil, G., Dufmats, M., Fernö, M.r., Jirström, K., et al. Two years of adjuvant tamoxifen in premenopausal patients with breast cancer: a randomised, controlled trial with long-term follow-up. European journal of cancer 2005;41(2):256–64.

[16] Chebil, G., Bendahl, P.O., Idvall, I., Fernö, M.. Comparison of immunohistochemical and biochemical assay of steroid receptors in primary breast cancer–clinical associations and reasons for discrepancies. Acta oncologica 2003;42(7):719–25.

[17] Falck, A.K., Bendahl, P.O., Ingvar, C., Lindblom, P., Lövgren, K., Rennstam, K., et al. Does Analysis of Disseminated Tumor Cells in Bone Marrow Give Additional Prognostic Information in Primary Breast Cancer? Analysis of Disseminated Tumor Cells in Bone Marrow — Report of a Prospective Study with 5 Years Follow-Up. Cancer Research 2010;70(24):105–6.

[18] Hansen, S., Grabau, D.A., Sørensen, F.B., Bak, M., Vach, W., Rose, C.. The prognostic value of angiogenesis by Chalkley counting in a confirmatory study design on 836 breast cancer patients. Clinical cancer research 2000;6(1):139–46.

[19] Swedish Breast Cancer Cooperative, . Randomized trial of two versus five years of adjuvant tamoxifen for postmenopausal early stage breast cancer. Swedish Breast Cancer Cooperative Group. Journal of the National Cancer Institute 1996;88(21):1543–9.

[20] Montana, D.J., Davis, L.. Training feedforward neural networks using genetic algorithms. In: Sridharan, N.S., editor. Proceedings of the 11th international joint conference on Artificial intelligence - Volume 1. IJCAI'89; San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.; 1989, p. 762–7.

[21] Nord, L.I., Jacobsson, S.P.. A novel method for examination of the variable contribution to computational neural network models. Chemometrics and Intelligent Laboratory Systems 1998;44(1-2):153–60.

[22] Cook, N.R.. Use and misuse of the receiver operating characteristic curve in risk prediction. Circulation 2007;115(7):928–35.

[23] Haybittle, J.L., Blamey, R.W., Elston, C.W., Johnson, J., Doyle, P.J., Campbell, F.C., et al. A prognostic index in primary breast cancer. British journal of cancer 1982;45(3):361–6.

[24] Ravdin, P.M., Siminoff, L.A., Davis, G.J., Mercer, M.B., Hewlett, J., Gerson, N., et al. Computer program to assist in making decisions about adjuvant therapy for women with early breast cancer. Journal of clinical oncology 2001;19(4):980–91.

[25] Stanley, K., Miikkulainen, R.. Evolving neural networks through augmenting topologies. Evolutionary computation 2002;10(2):99–127.