Application of Neural Networks and Sensitivity Analysis to improved prediction of Trauma Survival

Hunter, A., Kennedy, L., Henry, J. and Ferguson, R.I.

Corresponding author, and proofs to:
Dr. Andrew Hunter, Senior Lecturer, University of Sunderland.
Dept. Computer Science, St. Peter's Campus, Sunderland, Tyne and Wear, England.
+44 191 515 2778 fx: +44 191 515 2781 Andrew.Hunter@sunderland.ac.uk

Prof. Lee Kennedy, Dept. Medicine, Sunderland Royal Hospital, Kayll Road, Sunderland, SR4 7TP. L.Kennedy@sunderland.ac.uk

Ms. Jenny Henry Scottish Trauma Audit Group Royal Infirmay of Edinburgh, Lauriston Place, Edinburgh, EH3 9YW.

Dr. Ian Ferguson, Senior Lecturer, University of Sunderland, Dept. Computer Science St. Peter's Campus, Sunderland, Tyne and Wear, England. Ian.Ferguson@sunderland.ac.uk

Abstract

The performance of trauma departments is widely audited by applying predictive models that assess probability of survival, and examining the rate of unexpected survivals and deaths. Although the TRISS methodology, a logistic regression modelling technique, is still the de. facto. standard, it is known that neural network models perform better.

A key issue when applying neural network models is the selection of input variables. This paper proposes a novel form of sensitivity analysis, which is simpler to apply than existing techniques, and can be used for both numeric and nominal input variables. The technique is applied to the audit survival problem, and used to analyse the TRISS variables. The conclusion discusses the implications for the design of further improved scoring schemes and predictive models.

Keywords: Sensitivity Analysis, Neural Networks, TRISS, Trauma, Survival Analysis.

Introduction

This paper discusses the application of Neural Networks to the audit of Trauma Survival data. Trauma Survival is usually audited using the TRISS methodology [12]. This uses the well-known logistic regression model to estimate the probability of survival of patients based on clinical data gathered during admission. A number of authors have observed that the survival estimates yielded by TRISS can be improved by locally re-optimising the TRISS coefficients [4], by including factors omitted from the original TRISS methodology, such as admission pH [6] or pre-existing medical conditions [4]. Other results have indicated that comparable results may be gained by replacing TRISS with simpler models using logistic regression, either using the same variables as used in TRISS or using alternative variables [2]. Some preliminary results have also indicated that prediction accuracy may be improved by using alternative modelling approaches, such as neural networks [9] in place of logistic regression.

This paper confirms many of the above results, using a large database of some 15,000 cases (from STAG – the Scottish Trauma Audit Group), with progressively improved results yielded by locally re-optimising TRISS, by using a simpler logistic regression model than TRISS, and by using a simple neural network. We have further experimented with a novel approach to sensitivity analysis of neural network inputs, and can thus speculate usefully about the most important variables in Trauma survival assessment. This study of sensitivity agrees with other authors on the importance of some variables, but contradicts others.

Problem Description

Our objective is to assess the probability of Survival (PS) of patients admitted to Trauma units, based on information gathered when the patient is admitted. Predictions are used for audit purposes – to identify unexpected deaths or survivals, so that the causes of any regional patterns can be identified. We wish to determine whether the results obtained using the TRISS methodology can be improved upon, and in particular whether more complex neural network models are justified in place of the traditional logistic regression approach.

The TRISS model consists of two logistic regression equations, one of which is applied if the patient has a Penetrating injury, the other for Blunt injuries. The TRISS equations [2] each use three predictor variables: the Revised Trauma Score (hereafter abbreviated as RTS), the Injury Severity Score (ISS), and the age encoded as a binary variable (AGE55, indicating whether patients are aged less than 55 years, or 55 years and over). The RTS variable in turn is derived from the patient's Blood Pressure (BP) and Respiratory Rate (RESP), and the Glasgow Coma Scale (GCS). GCS in turn in derived by scoring Eye movement (EYE), Motor coordination (MOTOR) and Verbal response (VERBAL). The ISS variable is derived by scoring injury severity in six body areas (A1-A6), and combining these together.

Experimental Approach

We conducted a number of experiments to improve prediction accuracy. Our database consisted of 15,055 cases, gathered from Scottish Trauma departments in the period 1992-1996. The data were gathered by the Scottish Trauma Audit Group from a range of teaching, district and community hospitals. Information is gathered for patients who are admitted for three or more days, or who die within hospital. The data is initially entered by Accident and Emergency nursing staff; injury scoring is performed by a highly-trained regional coordinator. The data set contains almost no missing values. However, if the initial clinical observation is missing and the patient was managed in a resuscitation room and/or died, normal physiological variables are allocated and the patient is added to the database. If these observations are missing and the patient surrvives and is not managed in a resuscitation room, the patient is omitted from the database. The data set was divided into two subsets: the training set, containing 7,224 cases from 1992-1994; and the test Set, containing 7,831 cases gathered from 1995-1996. In all cases, the new models were optimised using the training Set, so there is an element of prospective testing as results are reported on the test set. A small number of additional cases (15) were discarded because some variable values were missing.

The two major modelling techniques used were logistic regression and neural networks. The first of these is already used in the TRISS methodology, and is widely known in the medical field. Neural

networks are less well known in medicine, and are sometimes regarded as an unproven and "mysterious" model. It is worth discussing the relationship between the two modelling techniques in some details, as they are much more closely related than is commonly realised.

In logistic regression, a weighted sum of the input variables is taken, and to this sum a bias value is added. The resulting value is passed through the logistic function:

$$act = b + \sum_i w_i x_i$$

$$PS(x) = \frac{1}{1 + e^{-act}}$$

The output of the logistic function is a value between 0 and 1, which is interpreted as a probability of class membership.

The theoretical justification for the use of logistic regression rests on the fact that, in the case where the two classes (live or die) have probability density functions following the multivariate normal distribution, with equal covariance matrices, the posterior probability function will indeed follow the logistic functional form.

Of course, if the probability density functions are significantly non-normal, or have different covariance matrices, then the performance of logistic regression will be non-optimal. It is therefore desirable to consider more complex forms of model.

Neural networks are a family of semi-parametric models which often exhibit better performance than logistic regression on classification problems with significant non-normality, or differing class covariance matrices [1]. In the most common form of Neural Network, the Multilayer Perceptron (MLP), two layers of units (sometimes called "neurons") are commonly used, although more layers are occasionally seen. Each unit forms a weighted, biased sum of its inputs, and then applies an activation function to this sum. One common activation function is the logistic function. A single unit

in the first layer (the *hidden layer*) of an MLP actually forms a logistic response function, identical to that used in logistic regression. However, in the MLP, a number of these logistic response surfaces are recombined in the output layer to form the actual probability estimate. The outputs units also use a biased logistic function. Thus, a logistic regression model is identical to an extremely simple neural network model (a neural network with no hidden layer, and a single output unit), and neural networks can be viewed as a generalisation of logistic regression. The primary advantage of neural networks over logistic regression is that they can model non-normal class distributions, and it is therefore not surprising to find that they are often capable of improved performance when compared to the logistic regression approach.

It is easy to adjust the model complexity in neural networks, simply by increasing or decreasing the number of hidden units – this allows valuable modelling flexibility within a single framework. An important feature of neural networks is that as the number of units in the network increases, so the model complexity increases, but also the difficulty of fitting the model and the amount of data required increase. A good design therefore incorporates the minimum size of network necessary to adequately model the problem.

Both logistic regression and neural network models have to be optimised using sampled data. We used the same optimisation strategy for both type of model. First, the logistic regression coefficients or neural network weights were randomly assigned. They were then optimised against the training set using a two-stage process – on-line gradient descent with momentum (Back Propagation, learning rate 0.1, momentum coefficent 0.3) for a moderate number of epochs (30), followed by Quasi-Newton BFGS optimisation for 100 epochs [7]. The Multilayer Perceptrons were optimised against the cross-entropy error function, which is equivalent to maximum likelihood optimisation of the logistic regression models [1]. The network error function in this case is:

$$E = \sum_i \ln(o_i)t_i + \ln(1-o_i)(1-t_i)$$

Where t_i is the target output for training case i, and o_i the actual output of the output neuron. This error function is used instead of the usual sum-squared error function. Despite the apparently more complex form, when used to modify the back propagation algorithm, and in conjunction with the

logistic activation function, the derivative used to calculate the error correction is actually *simplified* by omission of the *o(1-o)* term. Details are given in [1].

The two-stage optimisation process described above is little known, and we have adopted it as our experience indicates that second-order non-linear optimisation algorithms such as Quasi-Newton are substantially more reliable if they are preceded by a short burst of gradient descent (they are far less likely to become stuck in a local minima). Other authors have also reported this effect [8]. We suspect that this improved performance is due to greater stability in the error function Hessian matrix once the gradient-descent algorithm has located a reasonable starting point for second-order descent. We have also found it beneficial to scale the input variables into a consistent range [0,1] using the Minimax approach, as this aids the gradient descent stage (this is not necessary if optimisation is solely using Quasi-Newton).

Optimisation was quite straightforward, indicating that the data is well structured. Each model was optimised ten times, with the best resulting model selected. Results were remarkably consistent, showing little of the variation in performance which is common in neural network training – each model settled quickly to a consistent error level in the vast majority of cases (with one or two local minima, which were discarded), with no significant over-training. As a consequence, we were able to use the entire Training Set for optimisation of neural network models with no cross validation. The results reported below are from one arbitrarily selected model of each type.

Initially we evaluated a variety of types of Neural Network, including Radial Basis Function, Multilayer Perceptrons and Probabilistic Neural Networks [10], and with various numbers of hidden units. Since RBF and PNN networks proved consistently inferior to MLPs on this dataset, we conducted the full set of experiments using MLPs only, and report only these results.

Models were assessed for performance by calculating the Receiver-Operating-Characteristic (ROC) curves [12], and comparing the areas under the curves. The ROC curve and the area below were calculated using the Test Set (1995-1996 cases), which was not used in optimising the models. When

applying a classifier, there is an inevitable trade-off between type one and type two errors (false positives and false negatives), and the trade-off can be managed by adjusting the decision threshold. The ROC curve summarises the performance of a classifier across the range of possible decision thresholds. The area corresponds to the probability that a randomly selected patient who actually survives will be judged more likely to survive than a randomly selected patient who actually dies. In the case where a perfect classifier is possible (that is, where the cases from the two classes are separable) an area of 1.0 is achievable; in practical problems, where there is some overlap between classes, the upper threshold achievable is somewhere below 1.0. However, we do not know *a priori* what the optimum achievable result is.

Improved Prediction of Survival Probability

We conducted a number of experiments using the same variables as TRISS.

As a benchmark, we first calculated survival probabilities using the original TRISS coefficients. The area under the ROC curve in this case is 0.941.

We then re-optimised the TRISS coefficients using the available training data set, improving the result slightly to 0.943. This is consistent with results reported by other authors [4,6], which indicate that local re-optimisation is beneficial.

We next built simpler logistic regression models, in particular incorporating AGE directly as an input variable, and training a single model (either with TYPEINJ as an input, or ignored entirely). A model using a single logistic regression equation, with the Injury Type (TYPEINJ, Blunt or Penetrating), the Revised Trauma Scale, Injury Severity Score, and the Age of the patient entered directly (after rescaling), rather than encoded into a binary variable as in TRISS, achieved an area under the ROC curve of 0.953 - a noticeable improvement. Interestingly, a second logistic regression model with only three inputs (RTS, ISS and AGE) performed nearly as well, at 0.952. Since the type of injury is known to have a significant impact on the chance of survival (survival rates are far lower for those with Penetrating injuries, although these are far less common) it initially seems surprising that this

very simplified model can out-perform re-optimised TRISS. The explanation lies in the treatment of the AGE variable. In TRISS, this is reduced to a binary code indicating whether the patient is above or below 55. However, our sensitivity analysis experiments (discussed in more detail in the next section) consistently identify age as the most important independent predictor of survival in our data set.

Finally, we built simple Multilayer Perceptron neural network models, using the same range of input variables as in our logistic regression models. With the number of hidden units in the (single hidden layer) MLP varying between two and eight, the area under the ROC curve was remarkably consistent, varying between 0.9536 and 0.9548. The TYPEINJ variable proved irrelevant and was discarded, so that the MLP inputs were: RTS, ISS and AGE. The optimum architecture had only two hidden units, with performance falling off gradually as more units were added. Performance was equally good with and without the type of injury as an input to the network, so this variable was discarded.

To confirm our speculations regarding the importance of the encoding of the AGE variable, we experimented with an alternative MLP network with two hidden units, but with the TRISS-style binary-encoded age, rather than the actual age, as input. In this case, performance dropped to 0.948, still better than TRISS, but inferior to all our other models. Even then, sensitivity analysis confirms that age is the most important independent predictor in the model.

The experimental results are summarised in table 1; figure 1 shows the ROC curves for the best neural network and the original TRISS model.

Experiment	**ROC Area (test set)**
TRISS	0.9411
Reoptimised TRISS	0.9426
Logistic Regression, inc. Age and Type of Injury	0.9534
Logistic Regression, inc. Age	0.9521

Multilayer Perceptron, 2 hidden units, inc. Age	0.9548
Multilayer Perceptron, 2 hidden units, Age < 55 nominal-encoded	0.9475

Table 1: Performance of Logistic Regression and Neural Network Classifiers

As the ROC area can at best be increased to 1, the performance of the best neural network reported corresponds to a 23% improvement in the classifier's performance (i.e. 23% of the remaining discrepancy in ROC area from perfect performance is removed, when compared with TRISS). We can, for example, maintain the same sensitivity, but improve specificity so that approximately 500 cases that were reported as unexpected survivors are instead classified as expected survivals. This is a very significant improvement, making it far more feasible to follow up the audit with detailed study of those cases showing unexpected outcomes.

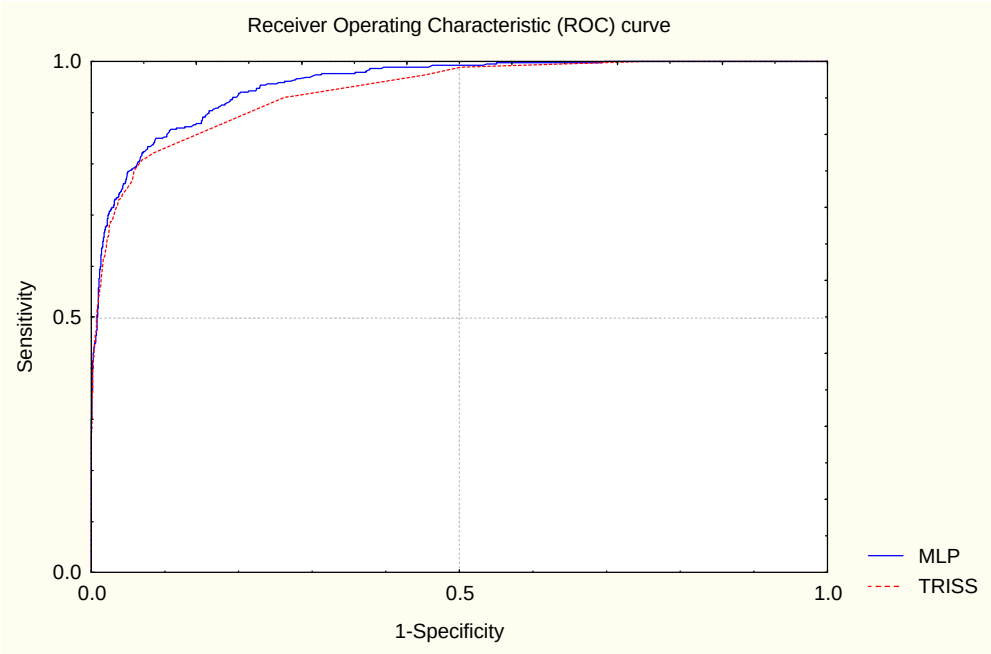

Figure 1: Comparison of ROC Curves for best Multilayer Perceptron and original TRISS

Sensitivity Analysis

A key question when modelling such problems is this: which variables are the most important predictors? It is desirable to assign some rating of importance to each variable. Any procedure to do

this is, however, fundamentally flawed, as variables are not in general independent. There may be some variables that are of use only in conjunction with others, and others that encode the same information and are therefore partially redundant (different arbitrary subsets of mutually redundant variables may be equally acceptable). Strictly speaking, variable selection is a subset selection problem [5], and as for N variables there are 2^N possible subsets, is it not usually practical to evaluate all subsets. Notwithstanding these theoretical objections, it is useful to assign ratings to variables within the context of a particular model. This is known as *sensitivity analysis*.

One approach is to add noise to each input variable, and to observe the effect upon the overall error [3]. There are drawbacks to this approach for feature ranking, however. First, one must make an essentially arbitrary decision about how much noise to add (usually, to add some multiple of the sample standard deviation of the variable). Second, it is not possible to add noise to nominal variables (i.e. variables which take one of a number of discrete values). Although an alternative approach of introducing random errors with some determined frequency could be adopted, applications of sensitivity analysis tend to be in domains where all inputs are numeric. In our case, the Type of Injury is a nominal variable.

A second approach is to analyse the derivatives of the fan-out weights of the input units, with a high derivative indicating high sensitivity [11]. This approach also has its drawbacks, as it is relatively complex to calculate the sensitivity, and the rating for each variable is necessarily a composite value derived from a number of derivatives, which compromises its reliability.

We propose an alternative form of sensitivity analysis, based on our approach to the *missing value problem*. In many problem domains, it is common to find the values of some variables missing. The STAG data set is actually very well-defined, so that missing values are extremely rare, and such cases have been omitted from these experiments. In problem domains where missing values are encountered more frequently, it becomes critical to have a method to “fill in” the missing values in partially-complete cases.

There are a number of approaches to this problem. Perhaps the simplest is to substitute the sample mean from the training set for a missing value, in the case of a continuous variable, or the *a priori* probabilities (or an estimate of these using the distribution in the training set, if *a priori* values are not known) for a nominal variable. This is the approach we commonly use. More sophisticated approaches include *data imputation* where one attempts to predict the unknown input variables conditioned upon those which are known, by constructing auxiliary models - the imputed values are used to fill in the gaps before the main model is used to predict the output. Any form of missing value substitution is compatible with our approach to sensitivity analysis, although we would expect (but have not confirmed) that data imputation would better isolate the "independent" contribution of each variable.

We analyse sensitivity by replacing each variable in turn with missing values, and assessing the effect upon the output error (i.e. the RMS of the individual cross-entropy errors of the test cases). A variable that is relatively important will cause a correspondingly large deterioration in the model's performance. This approach is somewhat analogous to a single step in the backwards feature selection procedure [5], except that in the latter case alternative models are actually built, whereas we simply alter the input to our existing model.

We submitted all of our models to this form of sensitivity analysis. In *all* cases, the variables were ranked in importance as follows: Age, Injury Severity Score, Revised Trauma Scale. Type of Injury, when included, was ranked fourth and of marginal importance. Precise errors (as opposed to ranking) vary from model to model.

This result may seem surprising – in particular, one might expect that the age of a patient is a less powerful predictor of trauma survival than any sensible measurement of trauma severity. This is a good example of how sensitivity analysis must be treated with caution. Age is the most significant *independent* contributor to the models – there is clearly a great deal of interdependence between the ISS and RTS variables. If either of these is ranked against AGE without including the other, it is far more significant.

Using Sensitivity Analysis to Examine the Composite Variables

Since the major input variables used in the TRISS equations are themselves composites of a number of simpler variables, it is interesting to consider which of these base variables are most influential. Some of these variables are extremely expensive to gather, requiring an expert assessment of the patients, and the volume of data recorded is a significant cost for all trauma units. There is therefore a great deal of motivation to reduce the number of variables gathered, or to use variables that are less expensive to gather.

To assess the relative importance of the base variables, we built a Multilayer Perceptron model using fourteen variables as inputs, and submitted it to sensitivity analysis. The variables were the A1-A6 scores used in the ISS Injury Severity Score; the Blood Pressure and Respiratory Rate used in the RTS Revised Trauma Scale, and the Eye, Motor and Verbal response variables used in the Glasgow Coma Scale, which is also in turn encoded in RTS, plus the Injury Type (Blunt or Penetrating), Age and Sex of the patients.

This model achieves a ROC area of only 0.951, somewhat inferior to our other models. This appears to reflect both the higher dimensionality of the resulting model, and the fact that the scoring schemes used in TRISS do recode the information in a more effective fashion. Nonetheless, the results are interesting.

Table 2 below summarises the sensitivity of the variables in this model; the errors compare with a base-line error of 0.407. It can be seen that three variables (Sex, Respiratory Rate and Eye Response) actually slightly degrade the model, and a further one (A2) is of no benefit. The insignificance of parameter A2 is not surprising (it rates facial injuries, which do not greatly effect the chance of survival). The irrelevance of gender is interesting; this contradicts the findings of Hannan, although he considered only the specific case of trauma in victims of low falls [4]. However, it is very surprising to find that Respiratory Rate and Eye motion are irrelevant – this no doubt reflects mutual

redundancy with other variables, and is again indicative of the dangers of placing too naïve an interpretation on the results of a sensitivity analysis.

Variable	Meaning	Error when Omitted	Ranking
AGE		0.603	1
MOTOR	Motor Movement	0.478	2
A1	Head Injury	0.465	3
A6	External Trauma	0.430	4
A3	Chest Injury	0.427	5
TYPEINJ	Blunt or Penetrating	0.421	6
A4	Abdominal Injury	0.420	7
VERBAL	Verbal Response	0.410	8
BP	Systolic Blood Pressure	0.409	9
A5	Extremities	0.408	10
A2	Facial Injury	0.407	11 *
RRATE	Respiratory Rate	0.406	12 *
EYE	Eye Movement Detected	0.406	13 *
SEX		0.399	14 *

* These variables may be omitted entirely without degrading the model.

The benchmark error (before omitting variables) was 0.407

Table 2: Sensitivity Analysis of Base Variables

Building a second model that used only the ten inputs ranked as most important in the above-experiment, we found that performance was not affected, whereas removing further variables degraded performance. This confirms the results of the sensitivity analysis.

Although this ten-input model does not perform as well as our models using the composite variables derived from the TRISS approach, it does confirm our suspicion that some of the information used in TRISS is not of great significance. We therefore conclude that there is some value in revisiting the

scoring methods used to produce those composite variables, a viewpoint which is borne out by the work of other authors [2] using quite different approaches.

It is possible to build quite effective models using only a small subset of the available variables. For example, with the first five inputs listed above we built a multilayer perceptron with a ROC area of 0.9445 – significantly inferior to the models using a more complete set of input variables, but still superior to TRISS, even when its coefficients are reoptimised. This opens up the possibility of using simplified models for triage, as this information requires relatively little expertise to gather, and is available from an initial examination of the patient (age might need to be estimated if the patient cannot respond to verbal questions).

Conclusion

We have evaluated the TRISS methodology and alternative approaches, including neural networks, on a large database (STAG) which has not previously been used for this purpose. Our experiments confirm that neural networks can yield better results than logistic regression, and that locally-reoptimised models are to be preferred. We have also confirmed that recoding the Age variable in a more straightforward fashion improves the performance of the models.

We have conducted a novel form of sensitivity analysis on our models. This is simple to apply, and effective in identifying key variables, allowing an important form of knowledge extraction to be applied to the neural network. The technique is equally applicable to any modelling approach, including those that are more sensitive to the inclusion of irrelevant inputs, such as fuzzy logic systems.

The sensitivity analysis confirms the importance of the Age variable. Further sensitivity analysis reveals that gender is irrelevant, and that a number of the basic variables used in the TRISS methodology contribute little.

The results of our sensitivity analysis suggest that further work should be undertaken in simplifying the scoring schemes used to produce the composite variables in TRISS. It should be possible to produce a system with a smaller number of base variables and with performance equal to our best models. For future work, we intend a more thorough investigation of alternative scoring schemes, together with an in-depth investigation of other variables that are commonly available, but which have not been included in this analysis.

References

[1] C. Bishop, *Neural Networks for Pattern Recognition* (Oxford University Press, 1995).

[2] H.R. Champion *et.al*, Improved Predictions from a Severity Characterization of Trauma (ASCOT) over Trauma and Injury Severity Score (TRISS): Results of an Independent Evaluation. *Journal of Trauma: Injury, Infection and Critical Care, 40 (1),* 1996.

[3] T.D. Gedeon, Data Mining of Inputs: Analysing Magnitude and Functional Measures. *Int. Journal of Neural Systems, 8 (2),* 1997, 209-218.

[4] E.L. Hannan *et. al.*, Multivariate Models for Predicting Survival of Patients with Trauma from Low Falls: The Impact of Gender and Pre-existing Conditions. *Journal of Trauma, Injury, Infection and Critical Care 38 (5),* 1995, 697-704.

[5] A. Jain and D. Zongker Feature Selection: Evaluation, Application and Small Sample Performance *IEEE Trans. Pattern Analysis and Machine Intelligence, 19 (2),* 1997.

[6] F.H. Millham *et.al.* Predictive Accuracy of the TRISS Survival Statistic Is Improved by a Modification that Includes Admission pH. *Arch. Surg. 130,* March 1995.

[7] W.H. Press, S.A. Teukolsky, W.T. Vetterling and B.P. Flannery. *Numerical Recipes in C: the Art of Scientific Computing (Second Ed.).* (Cambridge University Press, 1992).

[8] B.D. Ripley, *Pattern Recognition and Neural Networks.* (Cambridge University Press, 1996).

[9] R. Rutledge, O. Turner, S. Emery and S. Kromhout-Schiro. The end of the Injury Severity Score (ISS) and the Trauma and Injury Severity Score (TRISS): ICISS, an … *Journal of Trauma: Injury, Infection and Critical Care, 44 (1)*, 1996.

[10] D.F. Speckt, Probabilistic Neural Networks. *Neural Networks 3 (1)*, 1990, 109-118.

[11] J.M. Zurada, A. Malinowski and I. Cloete, Sensitivity Analysis for Minimization of Input Data Dimension for Feedforward Neural Network, *IEEE International Symposium on Circuits and Systems*, London, May 30-June 3, 1994.

[12] M.H. Zweig and G. Campbell Receiver-Operating Characteristic (ROC) Plots: A Fundamental Evaluation Tool in Clinical Medicine. *Clin. Chem 39 (4)*, 1993, 561-577.