



PREDICTION FOR TRAFFIC ACCIDENT SEVERITY: COMPARING THE ARTIFICIAL NEURAL NETWORK, GENETIC ALGORITHM, COMBINED GENETIC ALGORITHM AND PATTERN SEARCH METHODS

Mehmet Metin Kunt¹, Iman Aghayan², Nima Noii³

^{1, 2}Dept of Civil Engineering, Eastern Mediterranean University, Gazimagusa KKTC, Mersin 10, Turkey

³School of Civil Engineering and Surveying, University of Portsmouth, Portsmouth,
Hampshire, PO1 3AH, United Kingdom

E-mails: ¹metin.kunt@emu.edu.tr (corresponding author);

²iman.aghayan@cc.emu.edu.tr; ³nima.noii@gmail.com

Submitted 28 October 2010; accepted 31 July 2011

Abstract. This paper focuses on predicting the severity of freeway traffic accidents by employing twelve accident-related parameters in a genetic algorithm (GA), pattern search and artificial neural network (ANN) modelling methods. The models were developed using the input parameters of driver's age and gender, the use of a seat belt, the type and safety of a vehicle, weather conditions, road surface, speed ratio, crash time, crash type, collision type and traffic flow. The models were constructed based on 1000 of crashes in total that occurred during 2007 on the Tehran–Ghom Freeway due to the fact that the remaining records were not suitable for this study. The GA evaluated eleven equations to obtain the best one. Then, GA and PS methods were combined using the best GA equation. The neural network used multi-layer perceptron (MLP) architecture that consisted of a multi-layer feed-forward network with hidden sigmoid and linear output neurons that could also fit multi-dimensional mapping problems arbitrarily well. The ANN was applied during training, testing and validation and had 12 inputs, 25 neurons in the hidden layers and 3 neurons in the output layer. The best-fit model was selected according to the *R*-value, root mean square errors (RMSE), mean absolute errors (MAE) and the sum of square error (SSE). The highest *R*-value was obtained for the ANN around 0.87, demonstrating that the ANN provided the best prediction. The combination of GA and PS methods allowed for various prediction rankings ranging from linear relationships to complex equations. The advantage of these models is improving themselves adding new data.

Keywords: artificial neural network, genetic algorithm, pattern search, prediction, traffic accident severity.

1. Introduction

As the world population grows and cars become increasingly common, the number of traffic crashes worldwide is increasing. Traditional measures to reduce crashes include improved geometric design, congestion management strategies and better driver education and enforcement. While these measures are generally effective, they are often not feasible or prohibitively expensive to implement. Many factors are involved in traffic crashes, and some of those have a profound impact on one another, thus preventing transportation safety designers from using only one parameter to fully explain traffic accident severity. Studying parameters involved in traffic crashes using combined modern models that include the interactions of input and output variables can lead to a decrease in the number of traffic crashes. The model

of crash prediction (also called the safety performance function) is one of the most important techniques for investigating the relationship between crash occurrence and risk factors associated with various traffic entities. More than 28000 people are killed per year on Iranian roads with economic and social consequences. Factors with a profound impact on traffic accident severity include the demographic or behavioural characteristics of the driver (vehicle speed, driver's age and gender, seat belt use), environmental factors and roadway conditions at the time of the crash (crash time, weather conditions, road surface, crash type, collision type, traffic flow) and technical characteristics of the vehicle itself (vehicle type and safety). The primary goal of this study is to compare various models and select the most accurate one to predict traffic accident severity based on selected parameters; in addition, this research provides a possibility

of modifying itself with new added data with regard to twelve parameters and three injury severity levels selected as input and output variables.

This paper investigates three modelling techniques for achieving high predictive accuracy. Artificial neural networks are capable of capturing highly nonlinear relationships between predictor variables (crash factors) and the target variable (severity level of injuries). This aspect of neural networks is particularly useful when the relationship between the variables is unknown or complex and therefore difficult to handle statistically.

The second model is a genetic algorithm used for solving both constrained and unconstrained optimization problems based on natural selection, which is the process that drives biological evolution. The third model we investigate is a model combining the genetic algorithm (GA) and pattern search (PS) models. The use of GA and PS models in transportation safety studies is relatively new; therefore, we are going to combine these models in order to improve prediction accuracy.

Past research analyzing accident frequencies has mainly relied on statistical models such as linear regression models, Poisson regression and/or negative binomial regression models because the occurrence of accidents on a highway section can be regarded as a random event.

2. Background

The main focus of the prior studies has been to identify a defensible statistical relationship between crash counts and exposure. The negative binomial (NB) model arises mathematically (and conveniently) by assuming that unobserved crash heterogeneity (variation) across sites (intersections, road segments, etc.) is Gamma distributed while crashes within sites are Poisson distributed (Washington *et al.* 2010). Bayesian empirical methods have also been developed (Mahalel *et al.* 1982; Ng, Sayed 2004; Wright *et al.* 1988). Poisson, Poisson-Gamma (NB) and other related models are called generalized linear models. Hosseinlou and Aghayan (2009) used fuzzy logic to predict traffic accident severity on the Tehran-Ghom freeway in Iran.

Artificial neural networks (ANN) have been verified to be efficient in many fields. Neural networks are commonly used for non-linear modelling and forecasting. In traffic safety, some studies have applied ANNs to predicting crash rates and analyzing crashes, but none have used twelve parameters, including important factors with detail. Thus, this study attempted to incorporate all relevant parameters into the models to achieve a high percentage of crash forecasting. Mussone *et al.* (1999) applied artificial neural networks to analyze vehicular crashes that occurred at an intersection in Milan, Italy. A number of studies have attempted to identify groups of drivers at a greater risk of being injured or killed in traffic crashes (Zhang *et al.* 2000; Valent *et al.* 2002). Bédard *et al.* (2002) applied multivariate logistic regression analysis to investigate the effects of a driver, crash and vehicle characteristics on fatal crashes. Ivan

et al. (2000) investigated single and multi-vehicle highway crash rates and their relationships with traffic density while controlling for land use, the time of the day and light conditions. Temporal effects were also considered for single-vehicle crashes. Lord *et al.* (2005) conducted analysis on the relationship among crash, density (vehicles per km per lane) and v/c ratio. They found that along with an increase in v/c ratio, fatal and single-vehicle crashes decreased after some point, and crash rates followed U-shaped relationship. Artificial neural networks have scarcely been used as a modelling approach in the analysis of crash-related injury severity. More recent applications in the transportation field using the ANN have included traffic prediction (Yin *et al.* 2002; Zhong *et al.* 2004), the estimation of traffic parameters (Tong, Hung 2002), traffic signal control (Zhang *et al.* 2001), incident detection (Jin *et al.* 2002; Yuan, Cheu 2003), travel behaviour analysis (Subba Rao *et al.* 1998; Hensher, Ton 2000; Vythoulkas, Koutsopoulos 2003) and traffic accident analysis (Mussone *et al.* 1996, 1999; Sohn, Lee 2003; Abdel-Aty, Pande 2005). For example, Abdelwahab and Abdel-Aty (2001) used artificial neural networks for modelling the relationship between driver injury severity and crash factors related to the driver, vehicle, roadway, and environmental characteristics. Their study focused on classifying accidents into one of three injury severity levels using the readily available crash factors. These authors limit their domain of study to two vehicle accidents that occurred at intersections with signals. The predictive performance of a multi-layer perceptron (MLP) neural network was compared to the performance of the ordered logit model. The obtained results showed that MLP achieved better classification (correctly classifying 65.6 and 60.4% of cases for training and testing phases respectively) than the ordered logit model (correctly classifying 58.9 and 57.1% of cases for training and testing phases respectively). Abdel-Aty and Pande (2005) applied a probabilistic neural network (PNN) model for predicting crash occurrence on the Interstate-4 corridor in Orlando, Florida. The average and standard deviation from speed around crash sites were extracted from loop data as input variables. The results of this analysis showed that at least 70% of the crashes could be correctly identified by the proposed PNN model.

Genetic algorithms are powerful stochastic search techniques based on the principle of natural evolution. These algorithms were first introduced and investigated by Holland (1992). According to Chang and Chen (2000), regression models generated by genetic programming (GP) are also independent of any model structure. According to Deschaine and Francone (2004), the GP is observed to perform better than classification trees with lower error rates and also outperforms neural networks in regression analysis. Several studies (Park *et al.* 2000; Ceylan, Bell 2004; Teklu *et al.* 2007) have used GP methods in the traffic signal system and network optimization.

3. Methodology

3.1. Artificial Neural Network

Neural networks are composed of simple elements operating in parallel inspired by biological nervous systems. As in nature, connections between elements largely determine the network function. A neural network can be trained to perform a particular function by adjusting the values of connections (weights) between elements.

We used the architecture of a multi-layer perceptron (MLP) neural network that consisted of a multi-layer feed-forward network with sigmoid hidden neurons and linear output neurons. Multi-layers of neurons and the non-linear transfer function allow the network to learn non-linear and linear relationships between input and output vectors. The linear output layer allows the network to produce values outside the range from -1 to +1 so that this network with biases, a sigmoid layer and a linear output layer are capable of approximating any function with a finite number of discontinuities. This network can fit multi-dimensional mapping problems arbitrarily well given consistent data and enough neurons in its hidden layer. The network will be trained applying Levenberg-Marquardt back propagation algorithm. This structure essentially consists of a collection of non-linear neurons organized and connected to each other in a feed-forward multi-layer structure using directed arrows as coefficients (commonly called weight and bias in neural network terminology). The structure usually consists of input nodes, a hidden layer including some neurons and output nodes. The hidden layer is the network layer, which is not connected to the network output (for instance, the first layer of a two-layer feed forward network). This pattern is known to be well-suited to prediction and classification problems.

3.2. Genetic Algorithm

A genetic algorithm is a method for solving both constrained and unconstrained optimization problems and is based on natural selection, the process that drives biological evolution. Genetic algorithms repeatedly modify a population of individual solutions. At each step, the genetic algorithm selects individuals at random from the current population to be parents and uses them to produce children for the next generation. Over successive generations, the population ‘evolves’ toward an optimal solution. Genetic algorithms can be applied to solve a variety of optimization problems that are not well-suited to standard optimization algorithms, including problems in which the objective function is discontinuous, non-differentiable, stochastic or highly nonlinear. This meth-

od was developed by Holland (1992) over the course of the 1960s and 1970s and was finally popularized by one of his students, Goldberg, who was able to solve a difficult problem for his dissertation involving the control of gas-pipeline transmission (Goldberg 1989). Holland was the first to try to develop a theoretical basis for GAs through his schema theorem. The work of De Jong (1975) demonstrated the usefulness of GAs for function optimization and was the first concerted effort to optimize GA parameters.

GA operators are mutation (changes in a randomly chosen bit of a chromosome) and crossover (exchanging randomly chosen slices of a chromosome).

Fig. 1 shows a genetic cycle of the GA where the best individuals are continuously selected and operated on by crossover and mutation.

3.3. Pattern Search

Direct search is a method of solving optimization problems and does not require any information about the gradient of the objective function. Unlike more traditional optimization methods that use information about the gradient or higher derivatives to search for an optimal point, a direct search algorithm searches a set of points around the current point, looking for one point where the value of the objective function is lower than the value at the current point. Direct search can be used for solving problems when the objective function is not differentiable or even not continuous. Pattern search algorithms are direct search methods well-suited for the global optimization of highly nonlinear, multi-parameter and multimodal objective functions (Lewis, Torczon 1999). The current paper tests a pattern search algorithm based on GPS Positive Basis 2N (Lewis, Torczon 1999; Audet, Dennis 2003).

Pattern Search functions include two main algorithms called the generalized pattern search (GPS) algorithm and the mesh adaptive search (MADS) algorithm. Both are pattern search (PS) algorithms that compute a sequence of points that approach an optimal point. Pattern search algorithms are direct search methods that are capable of solving global optimization problems of irregular, multimodal objective functions without the need to calculate any gradient or curvature information, especially to address problems for which the objective functions are not differentiable, stochastic or even discontinuous (Torczon 1997).

At each step, the algorithm searches for a set of points called a mesh around the current point that was computed in the previous step of the algorithm.

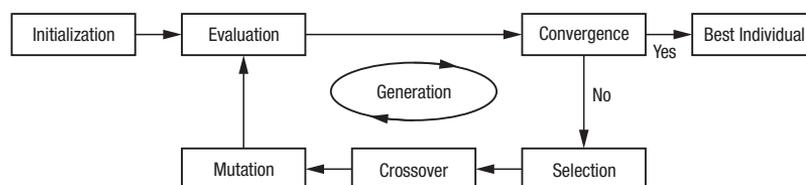


Fig. 1. The general structure of genetic algorithms

The mesh is formed by adding the current point to a scalar multiple of a set of vectors called a pattern. If the pattern search algorithm finds a point in the mesh that improves the objective function at the current point, the new point becomes the current point in the next step of the algorithm. The MADS algorithm is a modification of the GPS algorithm. The algorithms differ in how the mesh is computed. The GPS algorithm uses fixed direction vectors, whereas the MADS algorithm uses a random selection of vectors to define the mesh. The MADS algorithm uses the relationship between mesh size Δ^m and an additional parameter called the poll parameter, Δ^p , to determine stopping criteria.

For positive basis $N+1$, the poll parameter is $N\sqrt{\Delta^m}$, and for positive basis $2N$, the poll parameter is $\sqrt{\Delta^m}$. The relationship for the MADS stopping criterion is $\Delta^m \leq$ mesh tolerance, where Δ^m is the mesh size.

At each iteration pattern search polls the points in the current mesh by computing the objective function at the mesh points to see if any points have function values less than the current value. The pattern that defines the mesh is specified by the poll method option. GPS positive basis $2N$ consists of the following $2N$ directions, where N is the number of independent variables for the objective function. Pattern searches sometimes run faster using GPS positive basis $Np1$ as the poll method rather than GPS positive basis $2N$ because the algorithm searches fewer points at each of the iterations. MADS positive basis $Np1$ is also faster than MADS positive basis $2N$ (Lewis, Torczon 2002).

4. Measures for Goodness-of-Fit Regression Model

Goodness-of-fit (GOF) statistics is useful for comparing results across multiple studies, for examining competing models within a single study and for providing feedback on the extent of knowledge about uncertainty involved in the phenomenon of interest. Four measures of the GOF model are discussed: the sum of squares due to error (SSE), root mean square error (RMSE), correlation coefficient (R), MAE (mean absolute error) (Draper, Smith 1998).

4.1. Sum of Squares Due to Error

This statistics measures the total deviation of response values from fit to response values. It is also called the summed square of residuals and is usually labelled as SSE by Eq. (1) in which y_i is response value (target output) and \hat{y}_i is prediction response value:

$$SSE = \sum_{i=1}^n w_i (y_i - \hat{y}_i)^2. \quad (1)$$

SSE value closer to 0 indicates that the model has a smaller random error component and that the fit will be more useful for prediction.

4.2. Root Mean Squared Error

This statistics is also known as the fit standard error and the standard error of regression. RMSE is an estimate of a standard deviation from the random component in data and is defined as Eq. (2):

$$RMSE = S = \sqrt{MSE}, \quad (2)$$

where: MSE is the mean square error or the residual mean square, Eq. (3):

$$MSE = \frac{SSE}{\nu}. \quad (3)$$

Just as with SSE, MSE value closer to 0 indicates a fit more useful for prediction, and the root mean square error (RMSE) is a frequently-used measure of differences between the values predicted by a model or an estimator and the observed values.

4.3. Mean Absolute Error (MAE)

The average error of estimator $f_k(\vec{x})$ with respect to estimated parameter y_k is defined as the mean of the absolute difference between the estimator and real value, Eq. (4):

$$\frac{1}{n} \sum_{k=1}^n |f_k(\vec{x}) - y_k|. \quad (4)$$

4.4. Correlation Coefficient (R)

The correlation coefficient matrix represents the normalized measure of the strength of the linear relationship between variables.

Matrix \mathbf{R} of correlation coefficients was calculated from input matrix \mathbf{X} the rows of which are observations and columns are variables. Matrix \mathbf{R} is related to covariance matrix $\mathbf{C} = \text{cov}(\mathbf{X})$ by Eq. (5):

$$R(i, j) = \frac{C(i, j)}{\sqrt{C(i, i)C(j, j)}}. \quad (5)$$

The correlation coefficients range from -1 to 1 , where values close to 1 suggest that there is a positive linear relationship between data columns. The values close to -1 suggest that one column of data has a negative linear relationship to another column of data (*anti-correlation*), and the values close to or equal to 0 suggest that no linear relationship exists between data columns (Bevington, Robinson 2002).

5. Typical Steps in Designing a Model

Fig. 2 describes the principles of the employed models. Initially, 1000 records collected from police records were used for constructing objective functions for these models. Then, the models were able to modify the objective function with regard to each of those 1000 records added to preliminary data. In addition, the optimum coefficients of the objective function (for new records) were the initial optimum vector in combined GA and PS models (for the last records). To achieve optimal results from the ANN model, new weights and biases were calculated from the preliminary weight matrix and bias vector. Therefore, the ANN and GA as well as combined GA and PS models were able to find the minimum even with less than optimum choice for the initial range. Finally, the errors of objective functions were calculated

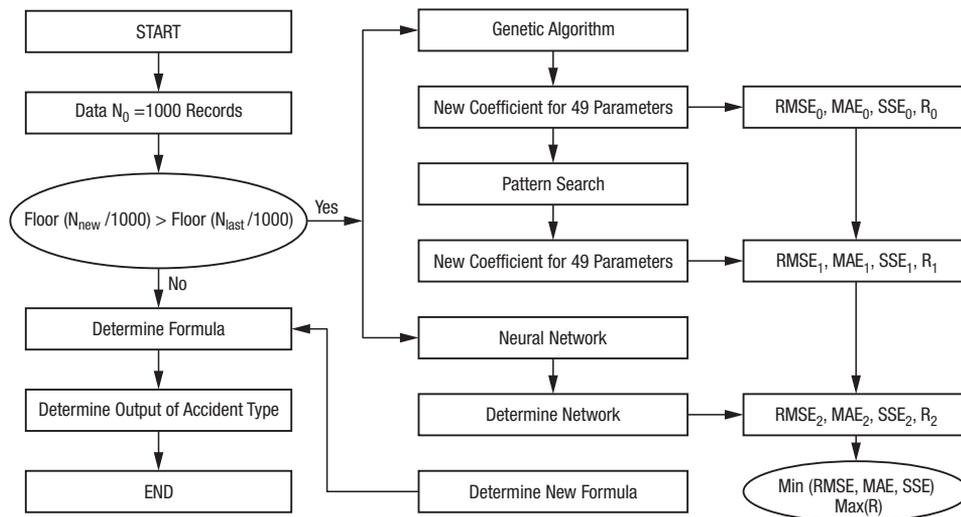


Fig. 2. The flowchart of the processes carried out in a typical run

applying these models, and the most appropriate error with respect to its type in each model was selected to determine the final objective function. The advantage of this structure is the ability of the model to improve itself with new added data.

6. Data Description

The dataset used in this study was derived from a total of 1063 reported traffic crashes in Tehran, the capital of Iran. We selected these crashes from the total number of crashes that occurred on the Tehran–Ghom freeway in 2007 because these were the only complete crash records. These data were used as training and testing data for the artificial neural network, genetic algorithm and combined GA and PS methods. The predictions of these three models were compared. The majority of crashes (74.8%) involved two vehicles. The distribution of driver

injuries made 14% of fatal injuries, 38.4% of evident injuries and 47.6% contained no injuries.

Three injury levels were considered for this study (i.e. no injury, evident injury or disabling injury/fatality), and twelve variables were selected from the obtained data. The vehicle speed in police reports was calculated by a camera or breaking distance. Speed ratio was used as one of the input variables defined as the ratio of estimated speed at the time of a crash to posted speed limit at the crash location. Road geometry parameters were not taken into consideration because the selected road had a desirable geometry common to all crashes in the dataset. The input variables have either numerical or dummy values to be used in the program. Table 1 shows coding input and output variables. MATLAB software was used for comparing the performance of three modelling approaches (ANN, GA, and combined GA and PS) discussed earlier.

Table 1. A description of study variables

Input Variables		Variable	Coding/Values	Data
Parameters	Subdivided Parameters			
1	2	Driver's Gender	Man= (1, 0)	97.56%
			Woman= (0, 1)	2.44%
2	1	Driver's Age	20–34=39%	
			35–49=44%	
			50–64=10%	
			65–79=7%	
3	2	Use of Seat Belt	In use= (1, 0)	78.66%
			Not in use= (0, 1)	21.34%
4	3	Type of Vehicle	Passenger car= (1, 0, 0)	83.54%
			Bus= (0, 1, 0)	2.44%
			Pick-up= (0, 0, 1)	14.02%

End of Table 1

Input Variables		Variable	Coding/Values	Data
Parameters	Subdivided Parameters			
5	2	Safety of Vehicle	High standard= (1, 0)	31.71%
			Low standard= (0, 1)	68.29%
6	4	Weather Condition	Clear= (1, 0, 0, 0)	56.71%
			Snowy= (0, 1, 0, 0)	7.93%
			Rainy= (0, 0, 1, 0)	10.37%
			Cloudy=(0, 0, 0, 1)	25%
7	3	Road Surface	Dry= (1, 0, 0)	75%
			Wet= (0, 1, 0)	17.68%
8	1	Speed Ratio	km/hr / km/hr	
			Day= (1, 0)	65.85%
9	2	Crash Time	Night= (0, 1)	34.15%
			With vehicle= (1, 0)	74.81%
10	2	Crash Type	With multiple vehicles= (0, 1)	25.19%
			Rear-end= (1, 0, 0)	51.95%
11	3	Collision Type	Right-angle= (0, 1, 0)	30.24%
			Sideswipe= (0, 0, 1)	17.80%
			Traffic Flow	veh/h
Output variables				
1	3	Driver Injury Severity	Fatality= (1, 0, 0)	14.02%
			Evident injury= (0, 1, 0)	38.41%
			No injury= (0, 0, 1)	47.56%

7. Models Used For Analysis

7.1. Multilayer Perceptron Neural Networks

The MLP model consisted of two layers having weight matrix W, bias vector b and output vector p^i that $i > 1$. Fig. 3 shows the selected final model for each of these layers in the MLP model. The number of the layer was appended as a superscript to the variable of interest.

Superscripts were used for identifying the source (second index) and destination (first index) of various weights and other elements of the network.

The weight matrix connected to input vector p^1 was labelled as input weight matrix ($IW^{1,1}$) having source 1 (second index) and destination 1 (first index). The elements of layer 1 such as its bias, net input and output have superscript 1 to represent that they were associated with the first layer.

The matrices of layer weight (LW) and input weight (IW) were used in the MLP model. Data were randomly divided into three parts: training, testing and validating. The MLP model had 12 inputs, 25 neurons in the first

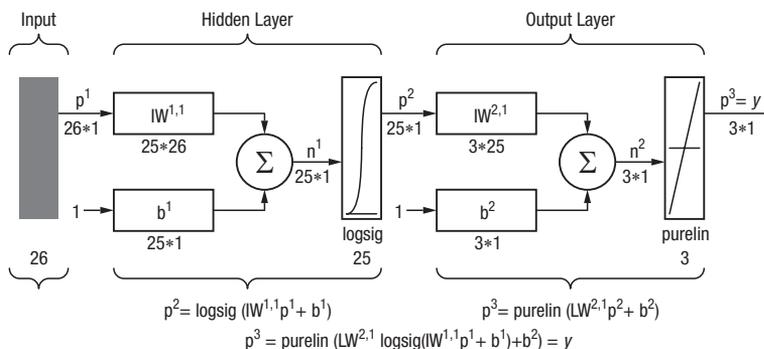


Fig. 3. The structure of the final Multi-Layer Perceptron neural network model

layer and 3 neurons in the second layer. The output layer of the MLP model consisted of three neurons representing three levels of injury severity. 70% of the original data were used in the training phase. Validation and testing data sets each contained 15% of the original data.

Constant input 1 was fed to the bias of each neuron. Note that the outputs of each intermediate layer were the inputs to the following layer. Thus, layer 2 can be analyzed as one-layer network having 25 inputs, 3 neurons and 3×25 weight matrix W^2 ; under such circumstances, input layer 2 is p^2 . All the vectors and matrices of layer 2 have been identified. The layer can be treated as a single-layer network on its own. The layers of a multi-layer network play different roles in the prediction process. This kind of two-layer network was used extensively in backpropagation. This study suggested that the output of the second layer, p^3 , was the network output of interest and was labelled as y (Rumelhart *et al.* 1986).

The objective of this network is to reduce error e , which is the difference between t and p^i in which $i > 1$ and t is the target vector. The perceptron learning rule calculates desired changes (target output) in the weights and biases of the perceptron, given input vector p^1 and associated error e . Thus, the goal is to minimize the average of the sum of these errors. The Least Mean Square Error (LMS) algorithm adjusts the weights and biases of the linear network so as to minimize this mean square error.

The error at output neuron j at iteration t can be calculated by the difference between the desired output (target output) and the corresponding real output, $e_j(t) = d_j(t) - y_j(t)$. Accordingly, Eq. (6) is the total error energy of all output neurons.

$$\varepsilon(t) = \frac{1}{2} \sum_{j \in c} e_j^2(t). \tag{6}$$

Referring to Fig. 3, the output of the k -th neuron in the l -th layer can be calculated by Eq. (7) in which $f_2 = \log \text{sig}$ and $f_3 = \text{purelin}$:

$$y_k^l = f_k \left(\sum_{j=1}^{n^{l-1}} w_{jk}^l \cdot y_j^{l-1} \right), \tag{7}$$

where: $1 \leq l \leq 3$, n^l refers to the number of neurons in layer l . For the input layer thus holds $l=1$, $y_j^1 = x_j$, for the output layer $-l=3$, $y_j^3 = y_j$. The mean square error (MSE) of the output can be computed by:

$$E = \frac{1}{2} \sum_{j=1}^3 (d_j - y_j)^2 = \frac{1}{2} \sum_{j=1}^3 \left[d_j - f_3 \left(\sum_{i=1}^{25} w_{ij}^3 \cdot y_i^2 \right) \right]^2. \tag{8}$$

The steepest descent of MSE can be used to update weights by Eq. (9) (Yeung *et al.* 2010):

$$w_{ij}^3(t+1) = w_{ij}^3(t) - \eta \frac{\partial E}{\partial w_{ij}^3}. \tag{9}$$

The mean square error performance index for the linear network is a quadratic function as shown in Eq. (8). Thus, the performance index will either have one global minimum, weak minimum or no minimum, depending on the characteristics of input vectors. Specifically, the characteristics of input vectors determine whether or not a unique solution exists (Hagan *et al.* 1996).

The results of the MLP model are presented in Table 2 in the form of a prediction table. Table 2 depicts the prediction level of injury severity patterns in training, testing and validation phases.

Table 2. Prediction table of the MLP model

R	No Injury	Evident Injury	Fatality	Overall
Training	0.9091	0.9029	0.8966	0.9125
Validation	0.8187	0.7613	0.6974	0.7863
Test	0.8372	0.6936	0.7587	0.7737
All	0.8849	0.8513	0.8372	0.8731

Fig. 4 shows regression plots for the output with respect to training, validating and testing data. The value of the correlation coefficient (R) for each phase was calculated. The R -value was around 0.87 for the total response in the MLP model.

Fig. 5 plots training errors, validation errors and testing errors to find validation error in the training window. The best validation performance occurred at iteration 7, and the network at this iteration was returned. The plot in Fig. 5 shows the mean squared error of the network starting at a large value and decreasing to a smaller value, which means that network learning is improving. The plot has three lines, because 1000 input and target vectors were randomly divided into three sets. 70% of the vectors were used for training the network. 15% of those were used for validating how well the network was generalized. Training vectors continues as long as training reduces the network error on validation vectors. After the network memorized the training set (at the expense of generalizing more poorly), training is stopped. This technique automatically avoids the problem of over fitting, which plagues many optimization and learning algorithms. Finally, the last 15% of the vectors provide an independent test of network generalization about data that the network has never seen.

7.2. Genetic Algorithm

The genetic algorithm (GA) is an optimization and search technique based on the principles of genetics and natural selection. The genetic algorithm starts with a population of solutions (chromosomes) represented by coded strings (typically 0 and 1 binary bits) as the underlying parameter set of the optimization problem. GAs generate successively improved populations of solutions (better generations) by applying three main genetic operators: selection, crossover and mutation. The selection function chooses parents for the next generation based on their scaled values from the fitness scaling function where the stochastic uniform selection function was used. Crossover is achieved by exchanging coding bits between two mated strings. The chromosomal material of different parents can be combined to produce an individual that could benefit from the strength of both parents. In this case, the applied crossover function was scattered.

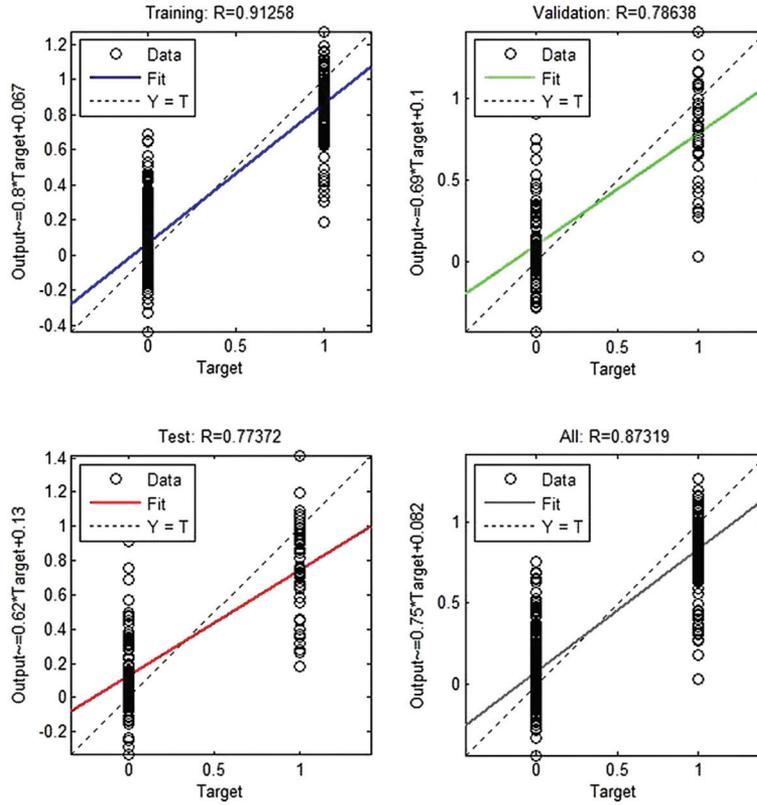


Fig. 4. Regression plots for training, testing and validation phases and the total response in the MLP model

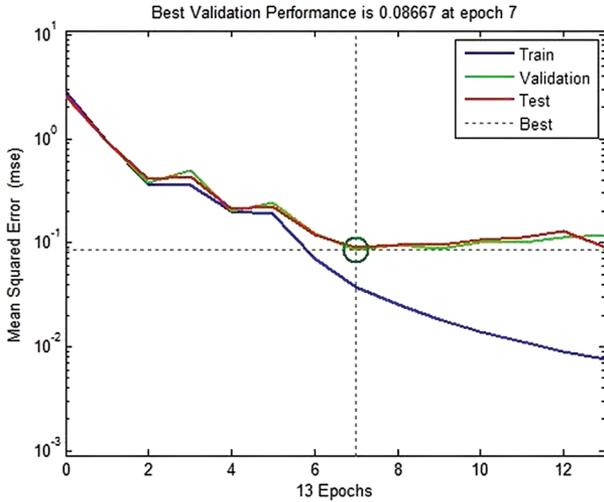


Fig. 5. The validation error in the MLP model

Mutation occasionally provides and recovers useful material for chromosomes through the random alteration of the value of a string bit (in the binary case, from 0 to 1 and vice versa). In our case, Gaussian mutation function was used. The following formula was obtained from 1000 police records, and therefore the system was able to modify the formula based on the added records. The goal is to find the solution in the set with the highest (optimum) performance according to our measure of 'goodness'. An objective function can be defined to represent the severity of traffic crash and prediction target that we seek to optimize. The objective functions were

selected by checking the values of R , MAE RMSE, and SSE as shown in Table 3.

Thus, we conclude that the objective function given in Eq. (6) has the best results for the GA model, with the R -value around 0.78 because the GA starts up creating a random initial population that contains an individual vector related to the population. The GA process stops when stopping criteria such as the maximum number of generation, stall time, stall generation and fitness limit are met or reach function tolerance values (1.0×10^{-6}). In Table 3, the objective function having higher R is in the first row, and therefore we can change it. By checking the optimized objective function having different initial populations, vectors and stopping criteria, we can get better coefficients related to our model. After checking the multiple of these situations for getting better results of the coefficient, we received the R -value of 0.79.

$$F = \sum_{k=1}^{n=1000} \left| X_{13} + \sum_{i=1}^{12} (X_{13+i} \sin(X_i b_{i,k})) + \sum_{i=1}^{12} (X_{24+2i} \sin(X_{25+2i} b_{i,k})) - Out_k \right|, \quad (10)$$

where: x is the coefficient of the optimized objective function and b and out parameters are related to input and output variables respectively. Table (4) presents modified coefficients of the objective function.

Fig. 6 displays the best and mean values of the fitness function at each generation. In addition, the best and mean values in the current generation are shown at the top of Fig. 6.

Table 3. Objective functions used in the GA model

F	R	MAE	RMSE	SSE	F	R	MAE	RMSE	SSE
$w_0 + \sum_{i=1}^{12} \sin(w_i X_i) + \sum_{i=1}^{12} \sin(v_i X_i)$	0.78689	0.33002	0.43949	178.308	$w_0 + \sum_{i=1}^{12} \sin(w_i X_i)$ $v_0 + \sum_{i=1}^{12} \sin(v_i X_i)$	0.76533	0.34574	0.46290	197.1453
$w_0 + \sum_{i=1}^{12} \sin(w_i X_i) + \sum_{i=1}^{12} \cos(v_i X_i)$	0.74474	0.34955	0.48068	209.6778	$w_0 + \sum_{i=1}^{12} \sin(w_i X_i)$ $v_0 + \sum_{i=1}^{12} \cos(v_i X_i)$	0.74999	0.34192	0.47364	203.5874
$w_0 + \sum_{i=1}^{12} w_i X_i$	0.60020	0.44124	0.57711	302.2494	$w_0 + \sin\left(w_{13} + \sum_{i=1}^{12} w_i X_i\right)$	0.46702	0.52028	0.70868	455.7767
$w_0 + \sum_{i=1}^{12} w_i X_i^{P_i}$	0.70776	0.39912	0.51465	240.3676	$w_0 + \cos\left(w_{13} + \sum_{i=1}^{12} w_i X_i\right)$	0.41690	0.54515	0.75001	510.4594
$w_0 + \sum_{i=1}^{12} e^{w_i X_i}$	0.46653	0.53863	0.64319	375.4189	$2 + \sin\left(\frac{1}{\exp\left(-\left(1 + \sum_{i=1}^{12} w_i X_i\right)\right)}\right)$	0.408693	0.48124	0.70213	447.3826
$\frac{w_0 + \sum_{i=1}^{12} w_i X_i}{v_0 + \sum_{i=1}^{12} v_i X_i}$	0.58782	0.45016	0.59606	322.4268					

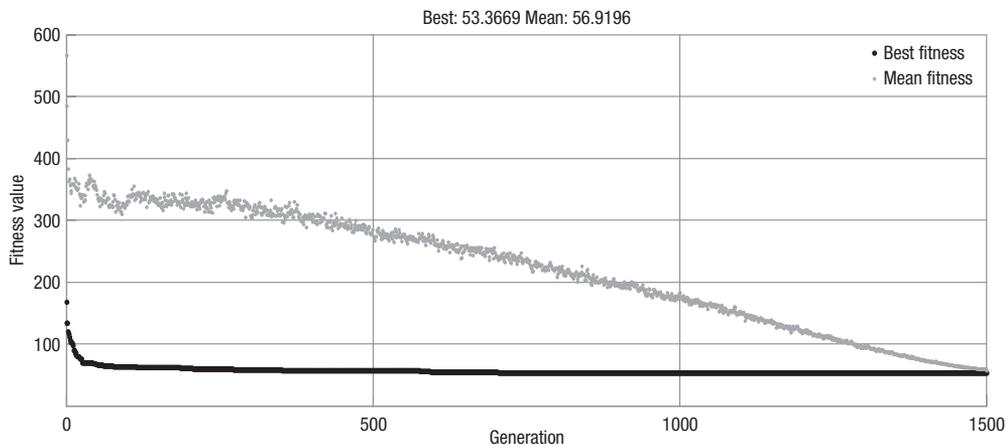


Fig. 6. The best and mean values of the fitness function at each generation in the GA model

7.3. Combination of the Genetic Algorithm and Pattern Search

We combined GA and PS models to determine whether this combined method would achieve better results than the genetic algorithm. This paper is based on GPS Positive Basic 2N, which enhances the performance of pattern search algorithms.

The initial point of this method was obtained from the optimum point of the GA shown in Table 4. Table 5 presents the modified coefficients of the combined model. The combined GA and PS model has the R -value of around 0.79.

Fig. 7 shows the value of the objective function at the best point considering each of the iterations. Typical-

ly, the value of the objective function improves rapidly in early iterations and then level off as they approach the optimal value. The initial point of this graph is the optimum final result of the GA.

The convergence curve in Fig. 7 is typical of pattern search algorithms. The initial convergence occurred after the first 800 iterations, followed by progressively slower improvements as the optimal solution was approached.

Fig. 8 displays mesh size at each iteration as it increased after each successful and decreased after each unsuccessful iteration. The best point did not change following an unsuccessful poll.

As a result, the algorithm halves mesh size with a contraction factor set to 0.5. The computed objective

Table 4. Modified coefficients of the objective function in the GA model

x_1	-0.10386	x_8	-1.61021	x_{15}	-0.1684	x_{22}	0.67988	x_{29}	-0.14108	x_{36}	0.07376	x_{43}	-1.41873
x_2	-1.18334	x_9	1.24933	x_{16}	-1.84944	x_{23}	0.26354	x_{30}	0.13037	x_{37}	4.31879	x_{44}	0.16222
x_3	0.30521	x_{10}	-0.63851	x_{17}	0.79854	x_{24}	-0.97961	x_{31}	-0.57707	x_{38}	0.91677	x_{45}	-0.29329
x_4	0.80627	x_{11}	0.20228	x_{18}	0.43804	x_{25}	-0.20209	x_{32}	-0.26776	x_{39}	-0.28983	x_{46}	0.64982
x_5	-0.61428	x_{12}	-0.40444	x_{19}	0.41867	x_{26}	0.78213	x_{33}	0.86287	x_{40}	0.69897	x_{47}	0.15646
x_6	0.55561	x_{13}	0.04129	x_{20}	0.87691	x_{27}	0.49914	x_{34}	-1.98046	x_{41}	2.90065	x_{48}	0.2271
x_7	0.81175	x_{14}	2.74527	x_{21}	-2.6484	x_{28}	0.20184	x_{35}	0.10735	x_{42}	-0.04085	x_{49}	0.17168

Table 5. Modified coefficients of the objective function in the combined GA and PS model

x_1	-0.10374	x_8	-1.61021	x_{15}	-0.17632	x_{22}	0.67988	x_{29}	-0.14139	x_{36}	0.06993	x_{43}	-1.69779
x_2	-1.18334	x_9	1.24933	x_{16}	-1.84944	x_{23}	0.26354	x_{30}	0.12699	x_{37}	4.31879	x_{44}	0.18301
x_3	0.30910	x_{10}	-0.62458	x_{17}	0.79854	x_{24}	-0.97961	x_{31}	-0.57707	x_{38}	0.91677	x_{45}	-0.32155
x_4	0.80627	x_{11}	0.20228	x_{18}	0.43804	x_{25}	-0.20335	x_{32}	-0.27866	x_{39}	-0.28983	x_{46}	0.64787
x_5	-0.60150	x_{12}	-0.40445	x_{19}	0.41916	x_{26}	0.78213	x_{33}	0.86268	x_{40}	0.71681	x_{47}	0.15646
x_6	0.55622	x_{13}	0.07327	x_{20}	0.87691	x_{27}	0.49914	x_{34}	-1.98046	x_{41}	2.90065	x_{48}	0.21438
x_7	0.81175	x_{14}	2.74527	x_{21}	-2.64840	x_{28}	0.18443	x_{35}	0.10735	x_{42}	-0.03879	x_{49}	0.17168

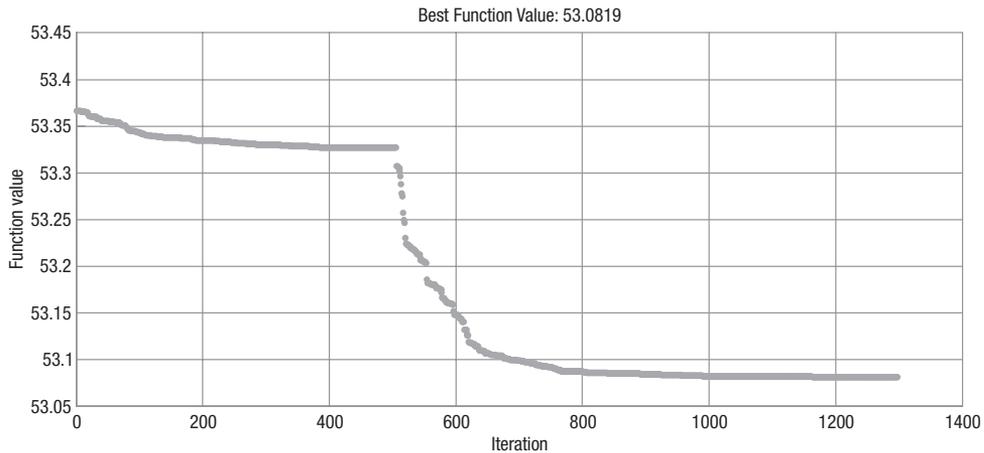


Fig. 7. The function value at each iteration in the combined GA and PS model

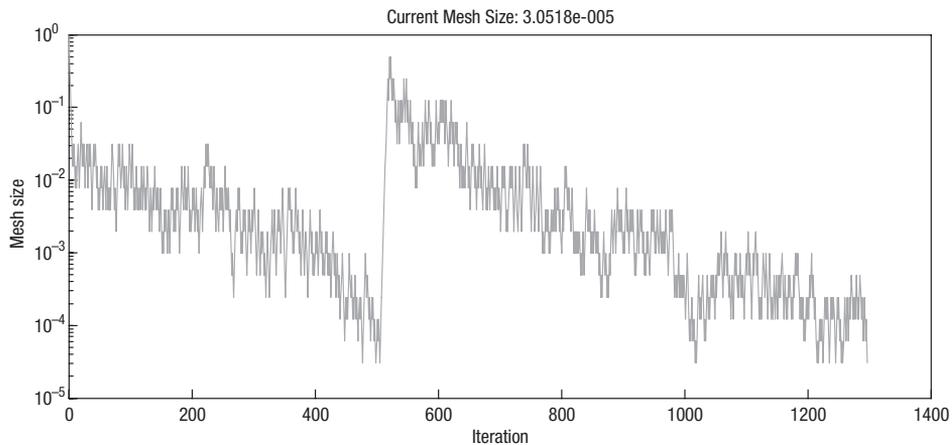


Fig. 8. Mesh size at each iteration in the combined GA and PS model

function value at iteration 2 was less than the value at iteration 1 in Fig. 1, which indicates that the poll at iteration 2 is successful. Thus, the algorithm doubles mesh size with the expansion factor set to 2 in Fig. 8. Clearly, the poll at iteration 4 was unsuccessful. As a result, the function value remained unchanged from iteration 3, and mesh size was halved.

As shown in Fig. 9, after 1297 iterations were completed, the pattern search algorithm performed approximately 98000 function evaluations to locate the most promising region in the solution space containing the global minima.

8. Discussion

This study used an artificial neural network, a genetic algorithm, combined genetic algorithm and pattern search method to predict the severity of traffic accidents. The final results showed that the ANN performed better than the GA and combined GA and PS models.

Table 6 presents correlation coefficient (R), mean absolute error (MAE), RMSE and SSE values. These results demonstrate that the constructed ANN is promising for modelling traffic injury severity.

Fig. 10 compares the real output values of crash severity with the predicted values of three models tested in our case. This graphical presentation depicts a considerable overlap between real and predicted graphs showing that the models successfully predict traffic accident severity with high accuracy.

Fig. 11 shows regression plots for the output with regard to fatality, evidence injury and no-injury; in addition, the value of correlation coefficient (R) for each level of crash severity was estimated. The R value of no-injury was higher than others which means that the results were compatible with the number of records.

Table 6. The final results of the objective function in each model

Error	Algorithm		
	GA	GA-PS	ANN
R	0.792411	0.793479	0.87319
MAE	0.323436	0.321709	0.16178
RMSE	0.43992	0.437782	0.22979
SSE	175.628	173.9248	123.4373

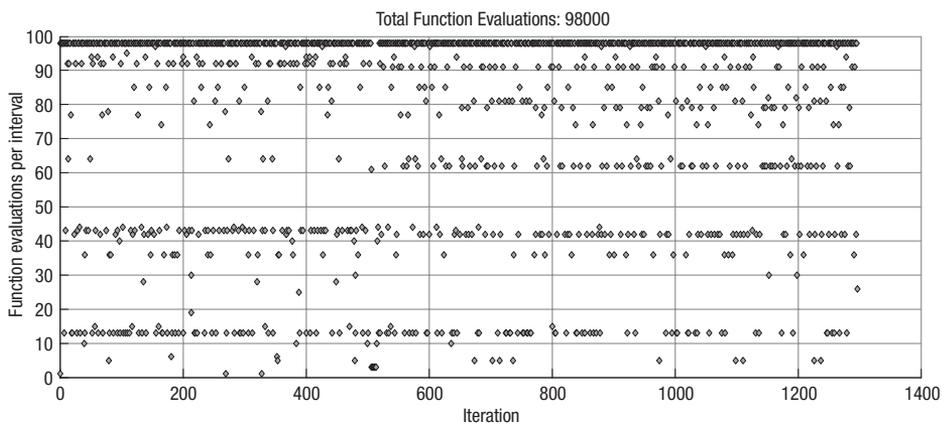


Fig. 9. Function evaluation per interval in the combined GA and PS model

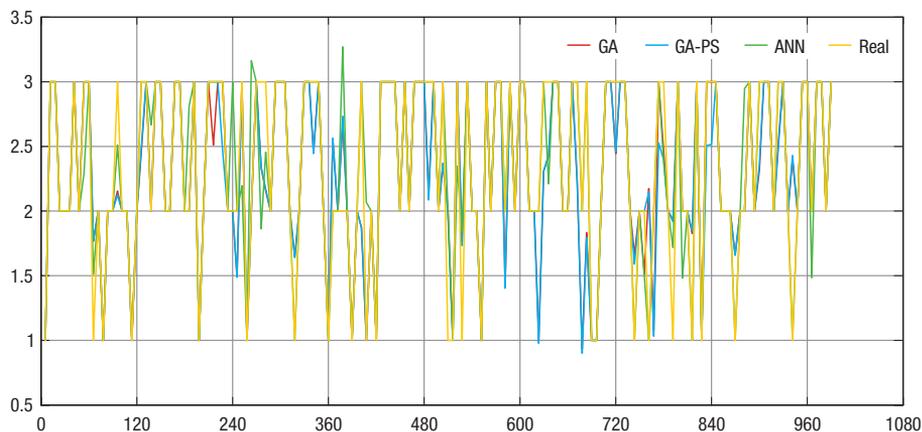


Fig. 10. Comparing real and predicted values

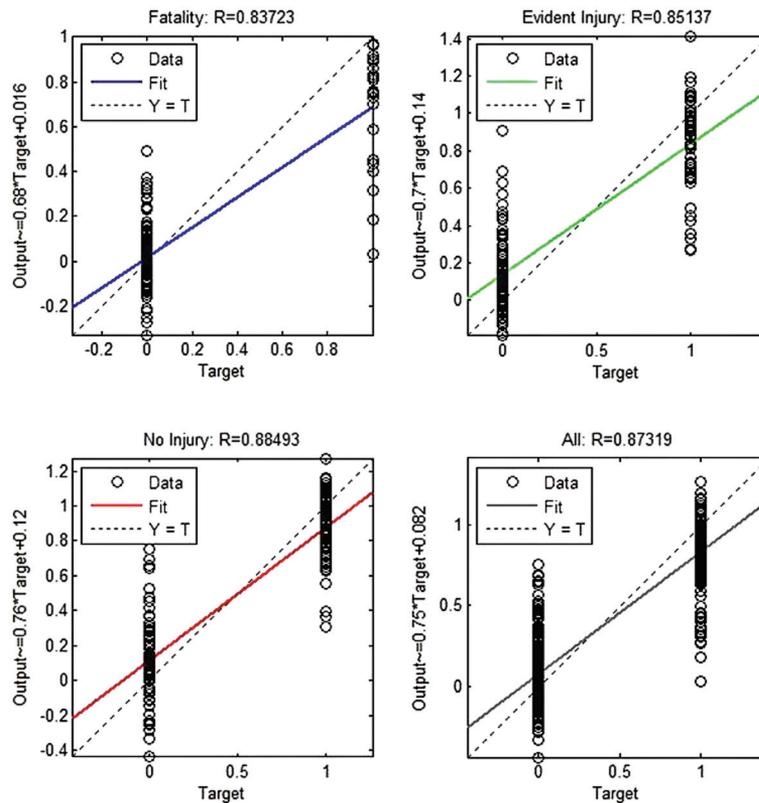


Fig. 11. Regression plots for each level of crash severity in the MLP model

9. Conclusions

1. This study used the GA, combined GA and PS, and the ANN with MLP architecture to predict traffic injury severity using twelve input parameters and three levels of injury severity. The performance of these methods was compared to find the most suitable method for predicting crash severity at three levels: fatality, evident injury, and no injury.
2. The ANN was applied for training, testing and validation and had 12 inputs, 25 neurons in the hidden layers and 3 neurons in the output layer. Data on training, validation and testing of applying the ANN represented 70%, 15% and 15% of all data on crashes, respectively. The R -value of the ANN was around 0.87.
3. The GA alone as well as combined with the PS model were used for predicting accident severity. The ANN provided the highest prediction accuracy with the R -value of around 0.87 followed by the combination of the GA and PS with the R -value of around 0.79 and GA of 0.79. Therefore, for this dataset, the ANN constructs a better relationship between twelve input parameters of the model and crash severity. On the other hand, the advantage of using the GA or the combined GA and PS model is that the functions and coefficients of relationships are known. Thus, each model has its own advantage, and therefore using more than one method may provide a better understanding of the relationship between input and output variables.

4. The constructed models were able to incorporate additional data. Moreover, the optimum coefficients of the objective function are the initial optimum vector in the combined GA and PS model. In order to reach optimum results using the ANN model, new weight and bias are calculated from the preliminary weight matrix and bias vector.
5. The use of more than one model suggested in this research provided a complete understanding of the relationship between input and output variables (combination of the GA and PS) and allowed for high prediction accuracy (ANN).

References

- Abdel-Aty, M.; Pande, A. 2005. Identifying crash propensity using specific traffic speed conditions, *Journal of Safety Research* 36(1): 97–108. doi:10.1016/j.jsr.2004.11.002
- Abdelwahab, H. T.; Abdel-Aty, M. A. 2001. Development of artificial neural network models to predict driver injury severity in traffic accidents at signalized intersections, *Transportation Research Record* 1746: 6–13. doi:10.3141/1746-02
- Audet, C.; Dennis, J. E. 2003. Analysis of generalized pattern searches, *SIAM Journal on Optimization* 13(3): 889–903. doi:10.1137/S1052623400378742
- Bédard, M.; Guyatt, G. H.; Stones, M. J.; Hirdes, J. P. 2002. The independent contribution of driver, crash, and vehicle characteristics to driver fatalities, *Accident Analysis and Prevention* 34(6): 717–727. doi:10.1016/S0001-4575(01)00072-0
- Bevington, P.; Robinson, D. K. 2002. *Data Reduction and Error Analysis for the Physical Sciences*. 3rd edition. McGraw-Hill Science/Engineering/Math. 336 p.

- Ceylan, H.; Bell, M. G. H. 2004. Traffic signal timing optimisation based on genetic algorithm approach, including drivers' routing, *Transportation Research Part B: Methodological* 38(4): 329–342. doi:10.1016/S0191-2615(03)00015-8
- Chang, N.-B.; Chen, W. C. 2000. Prediction of PCDDs/PCDFs emissions from municipal incinerators by genetic programming and neural network modeling, *Waste Management and Research* 18(4): 341–351. doi:10.1034/j.1399-3070.2000.00141.x
- De Jong, K. A. 1975. *Analysis of the behavior of a class of genetic adaptive systems*. Technical Report No 185. The University of Michigan. 256 p. Available from Internet: <http://deepblue.lib.umich.edu/bitstream/2027.42/4507/5/bab6360.0001.001.pdf>.
- Deschaine, L. M.; Francone, F. 2004. *White Paper: Comparison of Discipulus™* (Linear Genetic Programming Software with Support Vector Machines, Classification Trees, Neural Networks and Human Experts). Available from Internet: <http://www.rmltech.com>.
- Draper, N. R.; Smith, H. 1998. *Applied Regression Analysis*. 3rd edition. Wiley-Interscience. 736 p.
- Goldberg, D. E. 1989. *Genetic Algorithms in Search, Optimization, and Machine Learning*. 1 edition. Addison-Wesley Professional. 432 p.
- Hosseiniou, M. H.; Aghayan, I. 2009. Prediction of traffic accident severity based on fuzzy logic, in *8th International Congress on Civil Engineering*, 11–13 May 2009, Shiraz, Iran, 243–248.
- Hagan, M. T.; Demuth, H. B.; Beale, M. H. 1996. *Neural Network Design*. PWS Publishing. 736 p.
- Hensher, D. A.; Ton, T. T. 2000. A comparison of the predictive potential of artificial neural networks and nested logit models for commuter mode choice, *Transportation Research Part E: Logistics and Transportation Review* 36(3): 155–172. doi:10.1016/S1366-5545(99)00030-7
- Holland, J. H. 1992. *Adaptation in Natural and Artificial Systems: an Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence*. A Bradford Book. 211 p.
- Ivan, J. N.; Wang, C.; Bernardo, N. R. 2000. Explaining two-lane highway crash rates using land use and hourly exposure, *Accident Analysis and Prevention* 32(6): 787–795. doi:10.1016/S0001-4575(99)00132-3
- Jin, X.; Cheu, R. L.; Dipti, S. 2002. Development and adaptation of constructive probabilistic neural network in freeway incident detection, *Transportation Research Part C: Emerging Technologies* 10(2): 121–147. doi:10.1016/S0968-090X(01)00007-9
- Lewis, R. M.; Torczon, V. 1999. Pattern search algorithms for bound constrained minimization, *SIAM Journal on Optimization* 9(4): 1082–1099. doi:10.1137/S1052623496300507
- Lewis, R. M.; Torczon, V. 2002. A globally convergent augmented lagrangian pattern search algorithm for optimization with general constraints and simple bounds, *SIAM Journal on Optimization* 12(4): 1075–1089. doi:10.1137/S1052623498339727
- Lord, D.; Manar, A.; Vizioli, A. 2005. Modeling crash-flow-density and crash-flow-V/C ratio relationships for rural and urban freeway segments, *Accident Analysis and Prevention* 37(1): 185–199. doi:10.1016/j.aap.2004.07.003
- Mahalel, D.; Hakkert, A. S.; Prashker, J. N. 1982. A system for the allocation of safety resources on a road network, *Accident Analysis and Prevention* 14(1): 45–56. doi:10.1016/0001-4575(82)90006-9
- Mussone, L.; Rinelli, S.; Reitani, G. 1996. Estimating the accident probability of a vehicular flow by means of an artificial neural network, *Environment and Planning B: Planning and Design* 23(6): 667–675. doi:10.1068/b230667
- Mussone, L.; Ferrari, A.; Oneta, M. 1999. An analysis of urban collisions using an artificial intelligence model, *Accident Analysis and Prevention* 31(6): 705–718. doi:10.1016/S0001-4575(99)00031-7
- Ng, J. C. W.; Sayed, T. 2004. Effect of geometric design consistency on road safety, *Canadian Journal of Civil Engineering* 31(2): 218–227. doi:10.1139/03-090
- Park, B.; Messer, C. J.; Urbanik II, T. 2000. Enhanced genetic algorithm for signal-timing optimization of oversaturated intersections, *Transportation Research Record* 1727: 32–41. doi:10.3141/1727-05
- Rumelhart, D. E.; Hinton, G. E.; Williams, R. J. 1986. Learning internal representations by error propagation, in *Parallel Distributed Processing: Explorations in the Microstructure of Cognition: Foundations* (D. E. Rumelhart et al. Eds.) 1: 318–362.
- Subba Rao, P. V.; Sikdar, P. K.; Krishna Rao, K. V.; Dhingra, S. L. 1998. Another insight into artificial neural networks through behavioural analysis of access mode choice, *Computers, Environment and Urban Systems* 22(5): 485–496. doi:10.1016/S0198-9715(98)00036-2
- Sohn, S. Y.; Lee, S. H. 2003. Data fusion, ensemble and clustering to improve the classification accuracy for the severity of road traffic accidents in Korea, *Safety Science* 41(1): 1–14. doi:10.1016/S0925-7535(01)00032-7
- Teklu, F.; Sumalee, A.; Watling, D. 2007. A genetic algorithm approach for optimizing traffic control signals considering routing, *Computer-Aided Civil and Infrastructure Engineering* 22(1): 31–43. doi:10.1111/j.1467-8667.2006.00468.x
- Tong, H. Y.; Hung, W. T. 2002. Neural network modeling of vehicle discharge headway at signalized intersection: model descriptions and results, *Transportation Research Part A: Policy and Practice* 36(1): 17–40. doi:10.1016/S0965-8564(00)00035-5
- Torczon, V. 1997. On the convergence of pattern search algorithms, *SIAM Journal on Optimization* 7(1): 1–25. doi:10.1137/S1052623493250780
- Valent, F.; Schiava, F.; Savonitto, C.; Gallo, T.; Brusaferrero, S.; Barbone, F. 2002. Risk factors for fatal road traffic accidents in Udine, Italy, *Accident Analysis and Prevention* 34(1): 71–84. doi:10.1016/S0001-4575(00)00104-4
- Vythoulkas, P. C.; Koutsopoulos, H. N. 2003. Modeling discrete choice behavior using concepts from fuzzy set theory, approximate reasoning and neural networks, *Transportation Research Part C: Emerging Technologies* 11(1): 51–73. doi:10.1016/S0968-090X(02)00021-9
- Washington, S. P.; Karlaftis, M. G.; Mannering, F. L. 2010. *Statistical and Econometric Methods for Transportation Data Analysis*. 2nd edition. Chapman and Hall/CRC. 544 p.
- Wright, C. C.; Abbess, C. R.; Jarrett, D. F. 1988. Estimating the regression-to-mean effect associated with road accident black spot treatment: towards a more realistic approach, *Accident Analysis and Prevention* 20(3): 199–214. doi:10.1016/0001-4575(88)90004-8
- Yeung, D. S.; Cloete, I.; Shi, D.; Ng, W. W. Y. 2010. Introduction to Neural Networks, in *Sensitivity Analysis for Neural Networks: Natural Computing Series*, 1–15. doi:10.1007/978-3-642-02532-7_1

- Yin, H.; Wong, S. C.; Xu, J.; Wong, C. K. 2002. Urban traffic flow prediction using a fuzzy-neural approach, *Transportation Research Part C: Emerging Technologies* 10(2): 85–98. doi:10.1016/S0968-090X(01)00004-3
- Yuan, F.; Cheu, R. L. 2003. Incident detection using support vector machines, *Transportation Research Part C: Emerging Technologies* 11(3–4): 309–328. doi:10.1016/S0968-090X(03)00020-2
- Zhang, J.; Lindsay, J.; Clarke, K.; Robbins, G.; Mao, Y. 2000. Factors affecting the severity of motor vehicle traffic crashes involving elderly drivers in Ontario, *Accident Analysis and Prevention* 32(1): 117–125. doi:10.1016/S0001-4575(99)00039-1
- Zhang, H. M.; Ritchie, S. G.; Jayakrishnan, R. 2001. Coordinated traffic-responsive ramp control via nonlinear state feedback, *Transportation Research Part C: Emerging Technologies* 9(5): 337–352. doi:10.1016/S0968-090X(00)00044-9
- Zhong, M.; Lingras, P.; Sharma, S. 2004. Estimation of missing traffic counts using factor, genetic, neural, and regression techniques, *Transportation Research Part C: Emerging Technologies* 12(2): 139–166. doi:10.1016/j.trc.2004.07.006