PREDICTING TRAFFIC CRASHES USING REAL-TIME TRAFFIC SPEED PATTERNS

Mohamed Abdel-Aty*, Anurag Pande

Abstract

Despite of the recent advances in traffic surveillance technology and ever-growing concern over traffic safety, there have been very few research efforts establishing links between the real-time traffic flow parameters and crash occurrence. This study aims at the identification of the patterns in the freeway loop detector data, which potentially precede traffic crashes. This would have important implications for Advanced Traffic Management Centers (ATMC). ATMCs could then be able to predict the potential for crashes on freeways and take action to reduce this hazard by warning drivers or introducing variable speed limits. Solution approach to this research problem essentially involves classification of traffic speed patterns emerging from the loop detector data. Historical crash and loop detector data from Interstate-4 corridor in Orlando metropolitan area has been used for this study. The classification methodology adopted here is the probabilistic neural network (PNN): neural network implementation of well-known Bayesian-Parzen classifier. The PNN, not being a training based classifier, has strong statistical basis. The inputs to the model found best suitable for classification were logarithms of the coefficient of variation in speed obtained from three stations, namely, station of the crash (i.e. station nearest to the crash location) and two stations immediately preceding it in the upstream direction, during 5 minute time slice of 10-15 minutes prior to the crash time. The results showed that about 70% of the crashes on the evaluation dataset could be identified using the classifiers developed here.

1. Introduction

During the past few decades, tremendous growth has been observed in advanced traffic management and information systems (ATMIS). Although the concern over traffic safety has grown in this period, there were no efforts devoted to prevent crashes using these systems until few very recent studies.

The conventional approach to traffic safety analysis has been to establish relationships between the traffic characteristics (e.g. flow, speed), roadway and environmental conditions (e.g. geometry of the freeway, weather conditions) and driver characteristics (e.g. gender, age) and crash occurrence. The problem with most of the models developed using this approach is that they rely upon aggregate measures of traffic speed (e.g. speed limit) and volume (e.g. AADT or hourly volumes) and hence are not sufficient to identify the "black spots" (i.e.

locations having high probability of crashes), created due to the ambient traffic conditions, using the real-time variables (speed, flow and occupancy) obtained from the loop detectors in an ATMIS environment.

In this study, the problem of predicting crashes using the loop data has been approached as a classification problem in which we categorize the real-time traffic conditions as measured by loop detectors into either leading or not leading to a crash. The identification of parameters to be used as inputs to the classification algorithm (Probabilistic Neural Network; in this case) is also a part of this study.

2. Background

The idea of applying loop data for traffic safety research in order to predict crashes in real-time is still in preliminary stages. However, in the recent past there have been some efforts in this field. Lee et al. (2002) introduced the concept of "crash precursors" and hypothesized that the likelihood of a crash is significantly affected by short-term turbulence of traffic flow. They came up with factors like speed variation along the length of the roadway (i.e., difference between the speeds upstream and downstream of the crash location) and also across the three lanes at the crash location. Another important factor identified by them was traffic density at the instant of the crash. Weather, road geometry and time of the day were used as external controls. With these variables, a crash prediction model was developed using log-linear analysis. In a later study Lee et al. (2003) continued their work along the same lines and modified the aforementioned model. They incorporated an algorithm to get a better estimate of time of the crash and the length of time slice (prior to the crash) duration to be examined. It was found that the average variation of speed difference across adjacent lanes doesn't have direct impact on crashes and hence was eliminated from the model. They also concluded that variation of speed has relatively longer-term effect on crash potential rather than density and average speed difference between upstream and downstream ends of roadway sections.

A study by Oh et al. (2001) also showed the five minutes standard deviation of speed value to be the best indicator of "disruptive" traffic flow leading to a crash as opposed to "normal" traffic flow. They used the Bayesian classifier to categorize the two possible traffic flow conditions. Since Bayesian classifier requires probability distribution function for each class, they fitted their crash and non-crash speed standard deviation data to non-parametric distribution functions using Kernel smoothing techniques. Due to lack of crash data (only 52 crashes) their model remains far from being implemented in the field. It is also important to note that if a crash prediction model has to be useful in preventing crashes we need to identify the crash prone conditions much ahead of the crash occurrence time and not just 5-minutes prior; so that Regional Transportation Management Center (RTMC) has some time for analysis, prediction and dissemination of the information.

Although these studies do indicate the potential of applying real-time loop detector data to identify "alarming" traffic patterns on freeways, the biggest shortcoming of their analysis is that the data used in these studies were coming from just one station downstream and/or upstream of the crash location. Alarming conditions leading to crashes on a freeway might actually originate far upstream and "travel" with traffic platoons until they culminate into a crash at certain downstream location. To account for this possibility here we would be examining data from several stations upstream of the crash location at several time periods leading to the crash. This will also serve the purpose of identifying how far in advance ahead

(in terms of both time and distance) of a crash occurrence certain freeway segment may be flagged real-time due to high potential of a crash.

3. Methodology: theoretical background of PNN

As explained earlier the solution approach to the research problem essentially involves classification of traffic speed patterns emerging from the loop detector data. This section provides theoretical overview of probabilistic neural network (PNN) based classifiers used for the analysis. The PNN is a neural network implementation of the well-established multivariate Bayesian classifier, using Parzen estimators to construct the probability density functions of different classes (Specht, 1996).

3.1. Bayesian classifier

The PNN is strongly based on Bayes' method, which is arguably the single most popular classification paradigm. If we have a collection of random samples from K $(k = 1,2, \ldots\ldots, K)$ populations and each of these samples is a vector $x = [x_1, x_2, \ldots\ldots x_m]$. For a general case, if we allow for the possibility that the different populations have different probabilities to deliver random samples to us (k^{th} class has the prior probability h_k). When we misclassify a case that truly belongs to class k, the cost associated with this misclassification is c_k. It may be proved that if we happen to know the true probability density functions $f_k(\boldsymbol{x})$, then there exists a Bayes optimal decision rule resulting in a classification algorithm whose expected misclassification cost is minimum based on the available sample. Any unknown sample will be classified as population class i if:

$$h_i c_i f_i(x) > h_j c_j f_j(x) \qquad \forall j \neq i$$

Essentially this rule favors a class if it has high density in the vicinity of the pattern of unknown class, as the density $f_k(\boldsymbol{x})$ corresponds to the concentration of class k cases around the pattern of unknown class. The problem with the above rule is that we generally don't know the probability density functions and it should be estimated from the random samples available from K populations (Masters, 1995).

3.2. Parzen estimator

Parzen estimator uses the weight function $W(d)$ (frequently referred to as potential function or a kernel) having largest value at $d=0$ and it decreases rapidly as the absolute value of "d" increases. The weight functions are centered at each training sample point with the value of each sample's function at a given abscissa is being determined by the distance "d" between x and that sample point. The pdf estimator is the scaled sum of that function for all the sample cases. The method can be stated mathematically using the following equation:

$$g(x) = \frac{1}{n\sigma} \sum_{i=1}^{n} W\left(\frac{x - x_i}{\sigma}\right)$$

The scaling parameter σ defines the width of the bell curve that surrounds each sample point. As we will see later the value of this parameter might have a profound influence on the performance of a PNN. While the too small values will cause individual training cases to have too much of an influence, losing the benefit of aggregate information, the large values will cause so much blurring that the details of density will be lost (Masters, 1995).

3.3. Multivariate bayesian discrimination and classical PNN

The accuracy of the decision boundaries' estimation and the subsequent classification depends on the accuracy with which the underlying PDFs are estimated. A nice feature of this approach and the related PNN implementation is estimation consistency. Consistency implies that the error in estimating the PDF from a limited sample gets smaller as the sample size increases. The estimated PDF (the class estimator) collapses on the unknown true PDF as more patterns in the sample become available.

An example of the Parzen estimation of the PDFs (described in the preceding section) is given below for the special case that the multivariate kernel is a product of the univariate kernels. In the case of the Gaussian kernel, the multivariate estimates can be expressed as:

$$f_k(X) = \frac{1}{(2\pi)^{p/2}\sigma^p}\frac{1}{m}\sum_{i=1}^{m}\exp\left[\frac{-(X-X_{ki})'(X-X_{ki})}{2\sigma^2}\right]$$

where k is the class or category; i the pattern number; m the total number of training patterns; X_{ki} the ith training pattern from category or population π_k; σ the smoothing parameter and p the dimensionality of feature (input) space.

Note that the estimated PDF for a given class, say $f_1(x)$, is the sum of small multivariate Gaussian distributions centered at each training sample. However, the sum is not necessarily Gaussian. It can, in fact, approximate any smooth density function. The smoothing factor σ can alter the resulting PDF. Larger values of σ cause a vector X to have about the same probability of occurrence as the nearest training vector. The optimal σ can be easily determined experimentally (Abdulhai and Ritchie, 1999).

The network in Figure 1 shows p dimensional inputs to be classified into two classes. The pattern layer contains one neuron for each training case while the summation layer has one neuron for each class. Execution starts by simultaneously presenting the input vector to all pattern layer neurons. Each pattern neuron then computes a distance measure (Euclidean in the case of a classical PNN) between the input and the training case represented by that neuron. It then subjects that distance measure to the neuron's activation function that is essentially the Gaussian Parzen window. The following layer contains summation units having a modest task. Each summation neuron is dedicated to a single class. It just sums up the pattern layer neurons corresponding to the members of that summation neuron's class. The attained activation of summation neuron is the estimated density function value of this population class. The output neuron is merely a threshold discriminator and decides which of its inputs from the summation units is the maximum (Masters, 1995).

3.4. Statistical distance and the modified PNN

The PNN uses Euclidean distance as a measure of nearness among different patterns. Euclidean distance is statistically unsatisfactory for some applications because it does not account for differences in variations along the axes nor the presence of correlation among the variables constituting the pattern vector. To overcome this deficiency, Abdulhai and Ritchie (1999) proposed modification in the classical PNN algorithm.

To replace the employed Euclidean distance with the preferred statistical distance principal components rather than the original variables may be used. Algebraically, principal components are particular linear combinations of the original set of random variables.

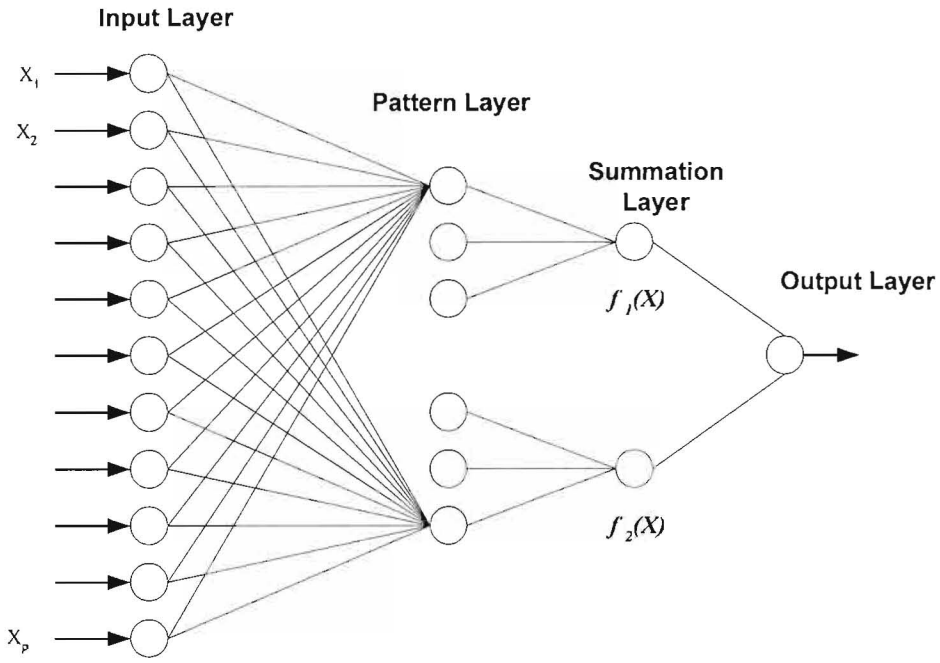

Figure 1: The traditional PNN architecture for a two-class classification problem

The original input vector X is transformed into the rotated vector Y using the eigenvectors (e_{ij}) of the covariance matrix Σ associated with random vector X. The component variables of the vector in terms of the rotated axes are then divided by their standard deviations $(\lambda_i)^{0.5}$ to equalize the variances and obtain a new set of inputs free of the effects of correlation and widely varying variances.

Figure 2 shows the modified version of the PNN (referred to as PNN2) that takes the above transformations into account. Two layers replace the previous input layer of the PNN: an input layer and a transformation layer. The weights between the input layer and the transformation layer are the eigenvectors of the sample covariance matrix. The transfer function in the units of the transformation layer simply divides the weighted input to the unit by the standard deviations $(\lambda_i)^{0.5}$. Beyond this transformation layer processing of PNN2 is identical to the original PNN described earlier (Abdulhai and Ritchie, 1999).

4. Explorations with the loop detector data

There are several studies which have concluded that the crash occurrences are related to variation in vehicle speeds (e.g., Shinar, 1999 and Garber and Ehrhart, 2000). It has been argued that as individual vehicles speeds deviate more and more from the average speed of the traffic stream the probability of having a crash increases. The data emanating from several consecutive loop detectors on a freeway section has been used here as a surrogate for the detailed vehicle movement data in order to capture the variance in vehicle speeds.

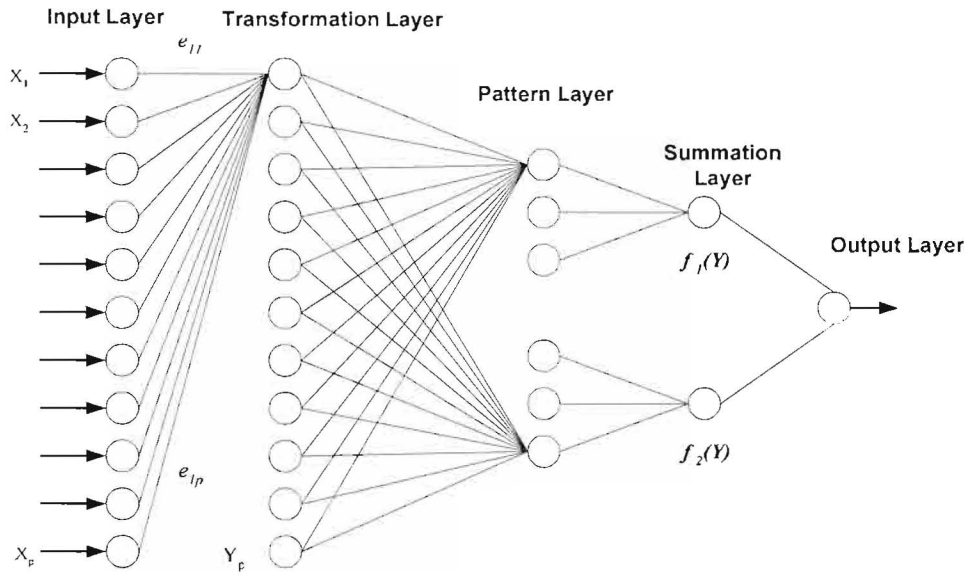

Figure 2: The modified PNN for a two-class classification problem (Abdulhai and Ritchie, 1999)

4.1. Study area and data collection procedure

The study was conducted on the corridor of Interstate-4 (I-4) in Orlando. The freeway section under consideration is 11.2 miles long and has a total of 25 loop detector stations, spaced out at nearly half a mile. Each of these stations consists of three dual loops in each direction and measures average speed, occupancy and volume over 30 seconds period on each of the through travel lane. This freeway stretch is under the jurisdiction of the Orlando police department (OPD) and hence OPD was the source of crash data for this study.

First, the location for each of the 670 crashes that occurred in the study area during the period of April 1999 to November 1999 were identified. The remaining months of that year had to be excluded, as no loop data was available for those months. For every crash, the loop detector station nearest to its location was determined. This station is referred to as the station of the crash from here on. The next step was to extract pre-crash speed data from the archived loop detector database. As mentioned earlier our focus is on comparison and classification of crash and non-crash traffic flow variables, therefore if a crash is reported to occur on April 12, 1999 (Monday) 6:00 PM, I-4 Eastbound and the nearest loop detector was at station 30, data was extracted from station 30, five loops upstream and one loop downstream of station 30 for half an hour period prior to the reported time of the crash for all the Mondays of the year at the same time. So this crash will have loop data table consisting of the speed values for all three lanes from the loop stations 25-31 (on eastbound direction) from 5:30 PM to 6:00 PM for all the Mondays of the year 1999, with one of them being the day of crash. This data was available for only 377 (out of 670) crashes, during the time of remaining crashes none of the loops, from which data was required, were functioning.

The loop detectors suffer from intermittent hardware problems that result in unreasonable values of speed, volume and occupancy. These values include Occupancy>100, speed=0 or >100, flow>25, and flow =0 with speed>0 and were removed from raw 30-second data. From the "cleaned" data tables the average and standard deviation of speed were extracted over each lane for six, 5-minute intervals recorded prior to the crash on the station nearest to the crash location (referred to as station of the crash), five stations upstream and one station downstream of the station of the crash. It requires creation of 252 fields (7 stations*6time slices*3 lanes*2 variables, i.e., average and standard deviation of speed) in the database for each crash. The same 252 fields were extracted for all the "corresponding" non-crash days as well.

The nomenclature procedure adopted for defining the station and time slice to which the average and standard deviation belongs is shown in Figure 3. All the stations were named as "A" to "G", with "A" being farthest station upstream and so on. It should be noted that "F" is the station of the crash and "G" will be the station downstream of the crash location since we have collected data from 5 upstream stations, station of the crash itself and one downstream station. Similarly the 5-minute intervals were also given "ID" from 1 to 6. The interval between time of the crash and 5 minutes prior to the crash was named as slice 1, interval between 5 to 10 minutes prior to the crash as slice 2, and interval between 10 to 15 minutes prior to the crash as slice 3 and so on.

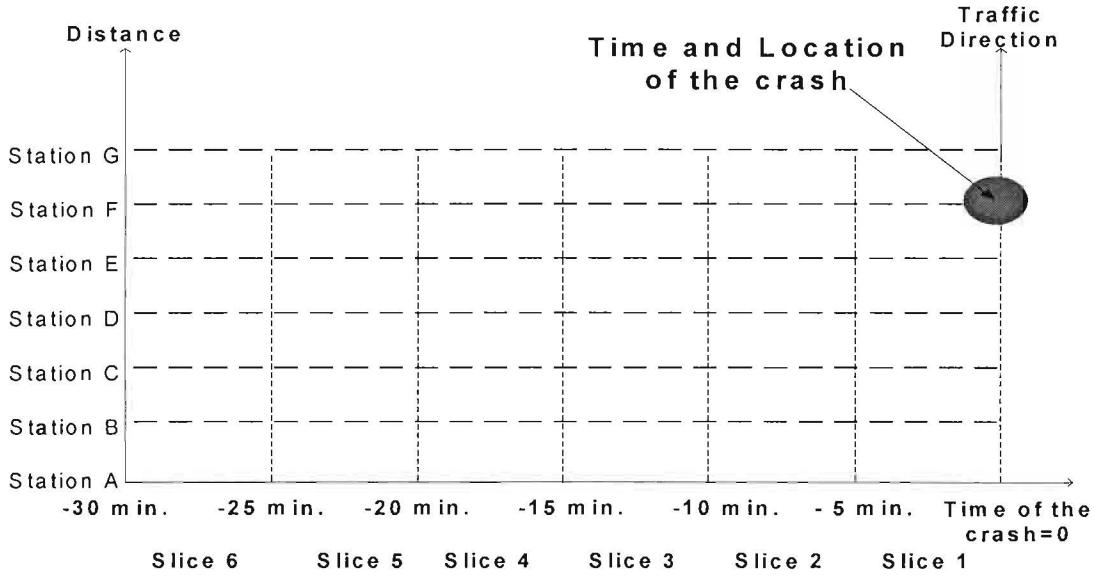

Figure 3: The nomenclature for defining the station and time slice to which any "effect" belongs

4.2. Exploratory analysis

Due to malfunctioning of certain loops the speed values over the three lanes were rarely available simultaneously. To overcome the problems due to missing data, it was decided to replace the values on three lanes with one value that was the average over three lanes.

Averaging was preferred over imputation of missing values because imputation procedures would have been very time consuming and beyond the scope of this study.

To justify the averaging over the lanes, Pearson's tests were carried out to detect the correlation between 5-minute average (and standard deviation) of speed across the three lanes for crash and non-crash cases separately. The test detected significant correlation at all the stations and time-slices. On the basis of these results from hereon the values used for spot speed are values averaged over the three lanes.

To detect the trends in 5-minute averages and standard deviations of speed at various time slices and stations their averages over all the crash and non-crash cases were obtained. Table 1 provides the average of 5-minute standard deviation of speeds over all the crash cases (Columns with Y=*1*) and non-crash cases (Columns with Y=*0*). Similarly, Table 2 provides the average of 5-minute averages of speeds. The values are obtained at all 42 (7 stations * 6 slices) time-slice and station combinations. It should be noted that the average over non-crash cases is observed over much more data points than the crash cases.

Table 1: Average values of 5-minute standard deviation of speeds observed at various time slice-station combinations

	Time Slice											
	1		2		3		4		5		6	
	Y		Y		Y		Y		Y		Y	
	0	1	0	1	0	1	0	1	0	1	0	1
Station												
A	5.38	5.63	5.39	5.39	5.36	5.46	5.36	5.61	5.31	5.41	5.30	5.21
B	5.33	5.67	5.37	5.69	5.29	5.65	5.31	5.59	5.34	5.60	5.31	5.51
C	5.38	5.58	5.36	5.71	5.36	5.71	5.34	5.73	5.34	5.44	5.33	5.42
D	5.23	6.00	5.27	5.70	5.26	5.69	5.26	6.05	5.25	5.59	5.24	5.47
E	5.27	6.00	5.30	5.55	5.22	5.63	5.24	5.51	5.26	5.86	5.23	5.63
F	5.33	5.89	5.33	5.79	5.34	5.89	5.33	5.89	5.30	5.85	5.27	5.42
G	5.14	5.50	5.20	5.67	5.15	5.26	5.20	5.60	5.17	5.55	5.17	5.56

Observing Table 1 closely it may be realized that the crash case variance (*Y=1*) is higher than the non-crash (*Y=0*) counterpart at all the stations during every time slices except for station A (that is 5 stations upstream of the station of the crash) during time slice 6 (25-30 minutes prior to the crash). Another interesting aspect is that as we "approach" the time and location of the crash the difference in standard deviation increases. Also the difference during all the time slices at station A and during time slice 6 at all the stations is relatively smaller and insignificant. It justifies the selection of 5 stations upstream and half an hour period.

From Table 2 we may observe that the crash case speeds are lower than their non-crash counterparts and the differences again become larger as we approach the time and location of the crash. Since for each crash, non-crash cases were chosen such that if a crash occurred on Monday 5:30 P.M. non-crash data consists of all the other available Mondays of that year at the same time and location. The day of the week, time of the day and location are the parameters affecting flow the most, once controlled the flow may be assumed to be the same for crash and non-crash cases. It indicates that the crash days had lower speeds at what is supposed to be more or less the same flow. According to the basic traffic flow theory it

implies that the density on the crash days was higher than the non-crash days. The higher standard deviation and lower average speeds on the crash cases indicated that 5-minute coefficient of variation (standard deviation / mean) in speed may be used to account for the trends observed.

Table 2: Average values of 5-minute average speeds observed at various time slice-station combinations

	Time Slice											
	1		2		3		4		5		6	
	Y		Y		Y		Y		Y		Y	
	0	1	0	1	0	1	0	1	0	1	0	1
Station												
A	49.24	46.81	49.15	46.30	49.11	45.82	49.11	46.63	49.13	47.01	49.24	46.72
B	46.96	43.80	46.95	43.55	47.06	43.57	47.06	43.94	47.08	44.20	47.15	44.65
C	46.62	42.59	46.62	42.76	46.79	42.34	46.86	42.57	46.97	42.94	47.12	42.86
D	47.23	41.56	47.20	42.27	47.41	42.73	47.66	42.81	47.78	43.57	47.89	43.43
E	46.23	40.18	46.27	41.00	46.39	41.47	46.50	41.30	46.62	42.20	46.79	42.80
F	45.71	40.08	45.71	39.93	45.92	39.88	46.01	39.38	46.21	40.18	46.38	41.02
G	48.09	42.60	48.10	42.43	48.21	41.69	48.38	41.67	48.49	41.61	48.66	42.89

5. Preliminary matched case control logistic regression

A basic matched case-control analysis, where the crashes are taken as case and all the corresponding non-crash data is used as the control, was performed. In this analysis the value of "Hazard ratio" for the data combined over three lanes was derived.

In a logistic regression setting the function of dependent variables yielding a linear function of the independent variables would be the logit transformation.

$$g(x) = \ln\left[\frac{\pi(x)}{1-\pi(x)}\right] = \beta_0 + \beta_1 x$$

Where $\pi(x) = E(Y|x)$ *is* the conditional mean of Y (dummy variable representing crash occurrence in our case) given x when the logistic distribution is used. Under the assumption that the logit is linear in the continuous covariate x the equation for the logit would be $g(x) = \beta_0 + \beta_1 x$. It follows that the slope coefficient, β_1, gives the change in the log odds for an increase of 1 unit in x, i.e. $\beta_1 = g(x+1) - g(x)$ for any value of x. Hazard ratio is defined as the exponential of this coefficient (Agresti, 2002).

Figure 4 depicts the trends shown by the values of "hazard ratio" when logcvs, i.e., logarithms of coefficient of variation in speed at all possible time slice-station combination is used one at a time as the risk factor (i.e. independent variable) in the matched case-control logistic regression analysis. Note that the crashes are treated as cases while all available corresponding non-rash cases act as controls.

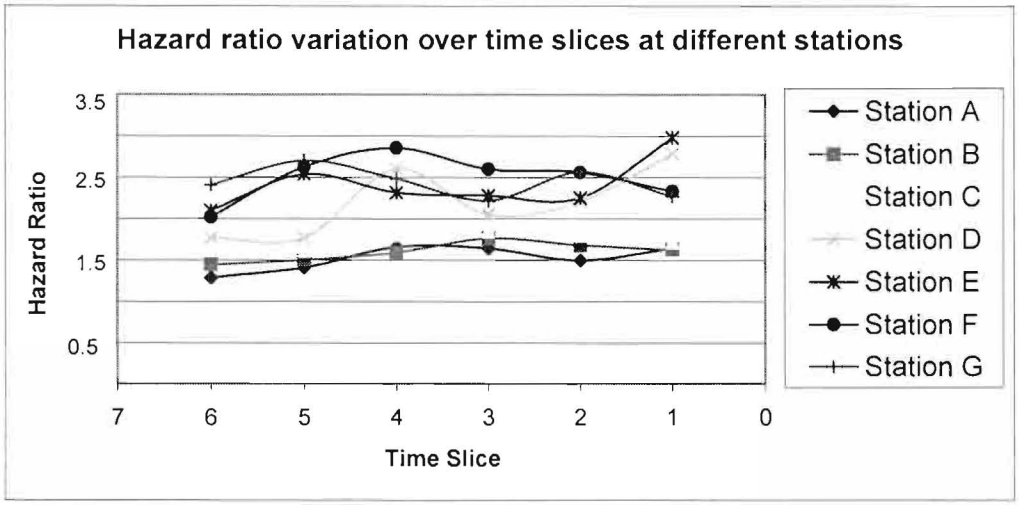

Figure 4: Hazard Ratio variation over time slices observed at different stations

The "hazard ratio" essentially represents the factor with which the risk of a crash occurring will increase when the corresponding "risk factor" (i.e., the covariate used as independent variable) is increased by one unit. It implies that the time slice-station combination with higher value of "hazard ratio" will affect the probability of crash occurrence more significantly. It may be seen that the values observed for stations "D" "E" "F" and "G" are higher than those observed for stations "A" "B" and "C" during all the time slices. The higher value of the hazard ratio is an important consideration while selecting which of the logarithms of coefficient of variation in speed (Logcvs) will become input to the PNN models.

6. Development of classification models

The variables (Logcv) with maximum hazard ratio are chosen to become inputs to the PNN models, but this was not the only consideration. If the Logcvs during time slice 1 and 2 (i.e. 0-5 and 5-10 minutes prior to the crash), despite of having maximum hazard ratio, were to become inputs to the model the prediction will come out too late to predict a crash and warn the drivers about it, once the model is applied on-line. Hence it was decided to work with variables, which are observed at least 10-15 minutes prior to the crash. Also, all the Logcvs to be fed into a model for training and testing should belong either to the same time slice duration or to the same station. This was required from a field application stand point since if a model uses data from different detectors at different time slices and classifies a real-time pattern as "alarming" it would be difficult to determine exactly which section should be flagged as a potential crash location.

The horizontal curves in the area of the study are not of widely varying radii and hence the alignments all along the freeway corridor under consideration were divided into straight and curved sections. To incorporate this into the PNN models the population classes were increased from two to four, i.e. crash on curved section, crash on straight section, non-crash on curved section and non-crash on straight section, instead of just having crash and non-crash.

6.1. Preparation of training and evaluation datasets

As described in the previous section loop detector data was obtained for 377 crashes. This data was then used to calculate Logcvs at various time slice- station combinations. To classify these data through a neural network based classifier all the Logcvs to be fed in the model should be simultaneously available. Based on this consideration, due to poor availability of data we were left with 148 (out of 377) crash and 2857 non-crash data points. From both categories (crash and non-crash) two-third (66%) of the data points were used for creation of the networks and one-third for evaluation. The data belonging to crash category was heavily under represented, hence it was necessary to balance the dataset in order to have equal crash and non-crash data points used for the creation of PNNs.

First, 100 crash data points (66% of the total 148) were randomly selected from the available crashes. Subtractive clustering procedure was then used in order to reduce 1883 non-crash data points (66% of the total 2857; to be used for creation of PNN) into 100 cluster centers. The procedure essentially involved identifying an appropriate cluster radius such that 100 points (out of 1883) are selected as cluster centers representing all the points lying within that particular radius. With randomly selected 100 crash data points and 100 non-crash cluster centers the dataset for creation of PNNs was ready. It should be noted, however, that the evaluation data was not clustered and was used as is. Hence, we had a total of 1022 test data points having 48 crashes and the rest belonging to non-crash category in the evaluation set.

Since the crashes during late night and early morning hours may be attributed mostly to human errors rather than ambient traffic conditions, a reduced dataset (referred to as "time-limited") was prepared in which only the crashes (and corresponding non-crash data points) occurred during 7:00 AM to 10:00 PM were included. The number of data points available for training and testing was obviously reduced in the time-limited dataset.

The structure of the datasets is shown in Table 3. The figure in parenthesis in the column containing non-crash training data points is the number of patterns from which the cluster centers, equal to the number of crash data points, are obtained.

Table 3: The number of patterns in the datasets created for training and evaluation of neural networks

Data set	Crash data points	Non-crash data points	Number of training data points		Number of evaluation data points	
			Crash	Non-crash	Crash	Non-crash
Complete	148	2857	100	100(1883)	48	974
Time-limited	116	2289	78	78(1526)	38	763

6.2. Classification models: Results and discussion

First experiment with PNN was for deciding on the combination of Logcvs to be used as inputs. Based on the hazard ratio values for the variables and the practical consideration described earlier, various combinations of Logcvs were used as PNN inputs and the resulting performance of the models on the evaluation dataset was carefully examined. It was observed that the three-dimensional input pattern involving the Logcvs at stations D, E and F (which are 2 stations upstream and the station of the crash itself, respectively) during time slice 3 (10-15

minutes prior to the time of the crash) meets the requirement of providing the optimal classification accuracy on the evaluation data set. Hence the final models utilized this three-dimensional input pattern to represent the real-time traffic characteristics. The accuracy of the models was evaluated in terms of two parameters, namely, percentage of overall (crash and non-crash) patterns classified correctly on the test dataset and percentage of crash identification over the test dataset. The criterion for the optimal model was the maximum overall classification accuracy for at least 70 % of crashes identified correctly.

Table 4 shows the results of the model utilizing the aforementioned 3-dimesional input patterns and classifying them as crash or non-crash over a range of σ values. The optimal performance based on the criterion adopted is highlighted in the table.

It may be seen that at very small spread values (e.g. 0.005) the model has very high accuracy for crashes (above 95%) but the overall classification accuracy is poor (less than 20 %). What this essentially means is that most of the data points from the test data set are being classified as crashes and would lead to excessive "false alarms" from a practical point of view. The reason for the same lies in the fact that at near zero spread values the PNNs act as nearest neighbor classifier. It is not highly unlikely to have non-crash data near to at least one of the crash data points (the reason being that sometimes even the alarming conditions may not culminate into a crash due to driver's ability). Hence if for a non-crash case its "nearest neighbor" lies in the crash category at near zero spread value it will be classified as crash even though it is nearer to many more non-crash cases. Once the value of spread parameter was increased gradually (i.e., with an increment of 0.005) and it was found that although the overall classification accuracy increases, the percentage of crashes correctly identified decreases, which means that at even higher spread values such a network will classify everything as non-crash and achieve high overall accuracy but will be of no use to forecast, as the aim is to identify the crashes correctly. The reason for missing out on crashes is because so much blurring is caused by the high spread parameter value that it looses the details of density function of the crash data. Therefore, an appropriate spread value providing optimal classification based on the 70 % crash identification criterion should be chosen.

Table 4: PNN models employed on the complete dataset with only considering real-time traffic speed patterns

Spread Value	Result parameters for classical PNN (%)		Result parameters for modified PNN (%)	
	Overall classification accuracy (test crash and non-crash data)	Accuracy on test crash data	Overall classification accuracy (test crash and non-crash data)	Accuracy on test crash data
0.005	18.9	97.5	19.9	98.0
0.01	21.5	97.5	20.0	97.5
0.015	27.9	90.0	25.5	90.8
0.02	37.8	87.5	34.2	88.5
0.025	48.6	85.0	46.7	84.2
0.03	56.2	77.5	54.3	76.3
0.035	62.1	72.5	59.8	73.7
0.04	66.7	67.5	63.9	68.4
0.045	70.0	65.0	67.7	65.8
0.05	72.5	62.5	70.3	63.2

Two more PNN models were created and evaluated, incorporating the horizontal alignment at the crash location and time of the day when crash occurred, respectively, in the classes to be identified. A model classifying the speed patterns into crash and non-crash was also developed using the time-limited dataset. Time limited data set was used because inclusion of time of the day into the classes to de identified doesn't improve efficiency of PNN, however, a careful analysis of the missed (i.e. unidentified) crashes led to the conclusion that most of these crashes occurred during late night hours. In all, four PNN classifiers and the optimal results obtained from them are depicted in Table 5. Note that the input patterns to all these models PNN models are three-dimensional, consisting of Logcv-D3, Logcv-E3 and Logcv-F3 (i.e. logarithms of coefficient of variation in speed (Logcvs) at stations D, E and F during time slice 3).

To compare across various models we may observe that the PNN model improves its performance (i.e. reasonable crash identification rate at moderate false alarm rate) once the geometry is incorporated into the classes to be identified (results shown in second row, Table 5). The topology of this network is shown in the Figure 5. The classification performance also improved when Time-limited dataset was used for classification between crash and non-crash, i.e. without including the horizontal alignment (results shown in forth row, Table 5).

Table 5: The optimal classification performances by various PNN models

Dataset used for training and evaluation	**Horizontal alignment in the classes to be identified**	**Time of the day in the classes to be identified**	**Parameters for classical PNN**			**Parameters for modified PNN**		
			Spread Value	Overall accuracy (test crash and non-crash data)	Accuracy on test crash data	Spread Value	Overall accuracy (test crash and non-crash data)	Accuracy on test crash data
Complete	×	×	0.035	62.1 %	72.5 %	0.035	59.8 %	73.7 %
Complete	√	×	0.045	74.6 %	71.7 %	0.045	73.2 %	71.6 %
Complete	×	√	0.015	17.8 %	72.3 %	0.035	18.8 %	72.0 %
Time-limited	×	×	0.050	80.0%	70.1 %	0.045	72.6 %	73.9 %

It was not possible to develop a time-limited model that accounts for the horizontal alignment using this data since on the time-limited dataset separating the crashes belonging to straight and curved sections would have resulted in insufficient evaluation sample size. Another point to be noted here is that there is no marked difference between the performances of classical and modified PNN. It implies that on this data set whether the Euclidian or statistical distance is applied as a measure of nearness in the PNN models no difference is observed. The reason might be that the three Logcvs applied as inputs are equally important and explain the variance in the data in almost equal proportions.

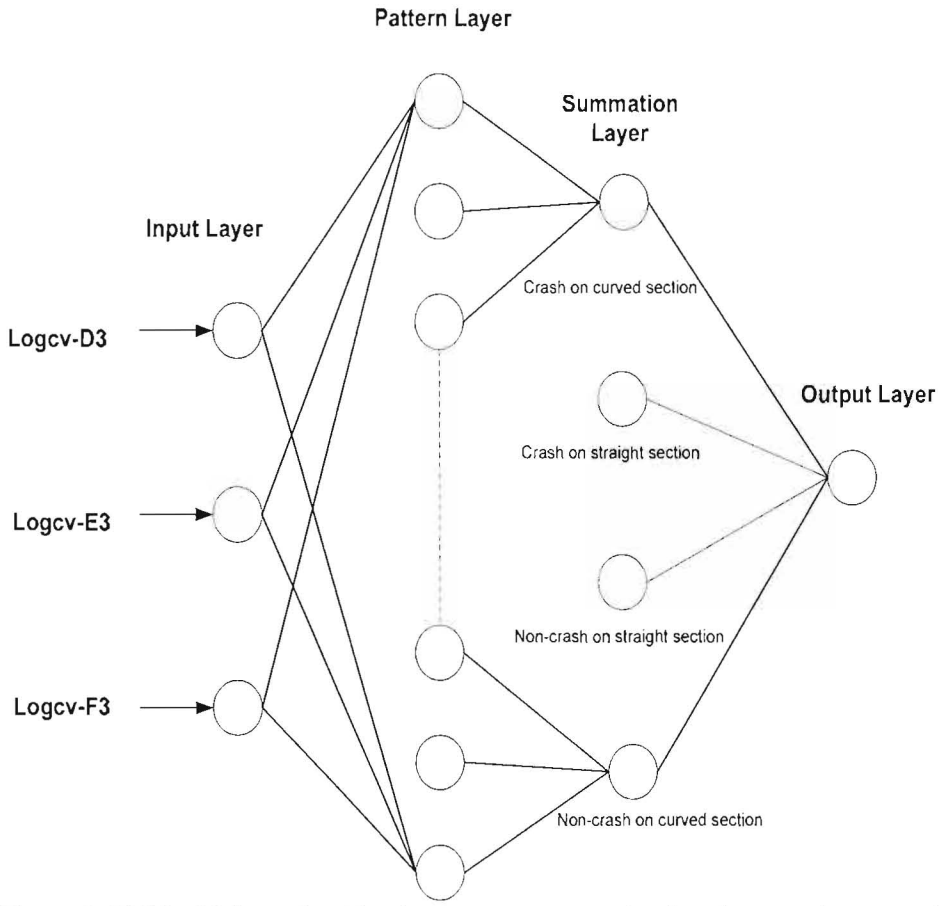

Figure 5: PNN with best classification accuracy on evaluation dataset when complete crash and non-crash data is used for creation and evaluation

7. Proposed real-tme application

The results of the PNN based classifier show that it is possible to identify more than 70 % of the crashes at a reasonable "false alarm" rate based on the loop data coming out from 3 loop detectors, 10-15 minute prior to the crashes. The models developed here may be applied in real-time very easily. On a stretch of a freeway one may collect data from sets of 3 consecutive detector stations, e.g. a series of 10 loop detectors on a freeway section, may be divided into sets of three detectors as (1,2 and 3), (2,3 and 4), (3,4 and 5) and so on. The 5-minute logarithm of coefficient of variation in speed can be calculated from the data emanating from these set of detectors and subjected to the PNN models. If patterns emerging from any set of detectors is classified as crash, the freeway section in the vicinity of station that is the most downstream in the set of three (as it will correspond to station "F"; station of the crash), may be flagged as potential crash location. Warnings could be conveyed to the drivers through variable message signs (VMS). Also, the concept of variable speed limits could be used to intervene and reduce the variation in speeds.

7.1. Identification of "Level of Threat"

In PNN architecture summation layer neurons precede the threshold discriminator output neuron. Whichever neuron in the summation layer has maximum activation, the class corresponding to that neuron becomes the output of PNN. Observation of differences between the individual activations of summation layer neurons, in addition to the resulting output class, will provide us with a measure of reliability for the PNN output. If in fact, the activation of pattern neuron belonging to crash category is higher (meaning a crash warning is impending) and the difference between the activations is quite large; it would mean a severe "threat" of crash. A reduced "level of threat" will be observed if the difference is smaller. The case of very small difference between the activations of pattern neurons will give rise to an additional "don't know" answer, which would enhance the reliability of these models and make them more suitable for an application as sensitive as crash prediction.

8. Conclusions

After examining several available combinations of Logcvs (logarithms of coefficient of variation in speed) it was concluded that Logcvs, observed during 10-15 minutes prior to crash at three stations namely; station of the crash and two stations immediately preceding the station of the crash in the upstream direction, when used as inputs lead to a PNN achieving the best classification performance. The performance further improved, when additional information regarding the horizontal alignment at the crash location was provided to the model through increasing the number of classes. Inclusion of time of the day (day time or late night) doesn't improve the performance of the models. While once a time-limited dataset (excluding late night crashes) is used for training and evaluation of neural networks, the best model in terms of overall classification accuracy is achieved. This leads us to infer that it may be very difficult to "predict" late night crashes as they mostly are caused by human errors while the loop data patterns are not necessarily alarming.

The study demonstrates the applicability of loop detector data for predicting freeway crashes. Once a potential crash location is identified in real-time, measures for reducing the speed variance may be taken in order to reduce the risk. The strategy for such measures, however, should be carefully investigated prior to such field application.

Acknowledgement

The authors wish to thank the Florida Department of Transportation for funding this research. All opinions and results are those of the authors.

References

Abdulhai, B., and Ritchie, S.G., 1999. Enhancing the universality and transferability of freeway incident detection using a Bayesian-based neural network. Transportation Research Part C: Emerging Technologies, 7(5), 261-280.

Agresti, A. 2002. Categorical data analysis, 2nd Ed. John Wiley and Sons, Inc.

Garber, N., and Ehrhart, A., 2000. The effect of speed, flow, and geometric characteristics on crash frequency for two-lane highways. Transportation Research Record, No. 1717, Transportation Research Board, National Research Council, Washington, D.C., 76-83.

Lee, C., Saccomanno, F., and Hellinga, B., 2002. Analysis of crash precursors on instrumented freeways. Presented at the 81st annual meeting of Transportation Research Board, Washington, D.C.

Lee, C., Saccomanno, F., and Hellinga, B., 2003. Real-time crash prediction model for the application to crash prevention in freeway traffic. Presented at the 82nd annual meeting of Transportation Research Board, Washington, D.C.

Masters, T. 1995. Advanced algorithms for neural networks: A C++ sourcebook. John Wiley and Sons, Inc.

Oh, C., Oh, J., Ritchie, S., and Chang, M., 2001. Real time estimation of freeway accident likelihood. Presented at the 80th annual meeting of Transportation Research Board, Washington, D.C.

Shinar, D., 1999. Speed and crashes: A controversial topic and an elusive relationship. Traffic Eng. 41, 52–55.

Specht, D.F., 1996. Probabilistic neural networks and general regression neural networks. In: Chen, C.H. (Ed.), Fuzzy Logic and Neural Network Handbook. McGraw-Hill, Berlin, 3.1–3.37.