Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2013, Article ID 524861, 8 pages
http://dx.doi.org/10.1155/2013/524861

Research Article

Using Principal Component Analysis to Solve a Class Imbalance Problem in Traffic Incident Detection

Changjiang Zheng,[1] Shuyan Chen,[2] Wei Wang,[2] and Jian Lu[2]

[1] *College of Civil and Transportation Engineering, Hohai University, Nanjing 210098, China*
[2] *Transportation School, Southeast University, Nanjing 210096, China*

Correspondence should be addressed to Changjiang Zheng; zhenghhu@sina.com

Received 21 August 2013; Revised 6 November 2013; Accepted 12 November 2013

Academic Editor: Wuhong Wang

High imbalances occur in real-world situations when a detection system needs to identify the rare but important event of a traffic incident. Traffic incident detection can be treated as a task of learning classifiers from imbalanced or skewed datasets. Using principal component analysis (PCA), a one-class classifier for incident detection is constructed from the major and minor principal components of normal instances. Experiments are conducted with a real traffic dataset collected from the A12 highway in The Netherlands. The parameters setting, including the significance level, the percentage of the total variation explained, and the upper bound of the eigenvalues for the minor components, is discussed. The test results demonstrate that this method achieves better performance than partial least squares regression. The method is shown to be promising for traffic incident detection.

1. Introduction

Early detection of traffic incidents can minimize the delay experienced by drivers, wasted fuel, emissions, and lost productivity, while also reducing the likelihood of secondary collisions [1]. Traffic incident detection is thus a critical issue and it is important to develop mechanisms to detect traffic incidents as early as possible. The incident detection problem has received great interest from researchers and many incident detection techniques have been developed. Black and Sreedevi [2] have extensively reviewed many approaches to incident detection. Existing incident detection methods fall into the following major categories: pattern recognition, time series analysis, Kalman filters, partial least squares regression [3, 4], and data mining technologies, which include neural networks, fuzzy logic, support vector machines (SVM) [5, 6], rough set [7], ensemble learning [8, 9], and decision tree learning [10]. Data mining technologies have been shown to be the most promising techniques of the incident detection methods.

The typical classifiers in data mining, such as decision tree inductive systems or neural networks, are designed to optimize overall accuracy without accounting for the relative distribution of each class. As a result, these classifiers tend to neglect small classes while focusing on classifying the large classes accurately [11]. Unfortunately, these classifiers perform poorly in incident detection because the real-world traffic data suffers from class imbalances that typically contain much fewer incident cases than incident-free cases. Such situations pose challenges for these classifiers. Many solutions to the class imbalance problem have been previously proposed both at the data and algorithmic levels. At the data level, resampling methods are commonly used to address the class imbalance problem [12, 13]. Although such approaches can be very simple to implement, tuning resampling methods to be effective in an application is not an easy task. At the algorithmic level, cost-sensitive learning [14], one-class classifiers [15], and ensemble-based classifiers, such as Boosting and Adaboost [16, 17], are very well-known approaches for solving dataset imbalance problems.

Typically, in a conventional multiclass classification problem, data from two (or more) classes are available and the decision boundary is supported by the presence of examples from each class. However, a one-class classifier completely neglects one of the classes and learning is accomplished using examples from a single class only at the learning stage.

Different researchers have used other terms to present similar concepts, such as outlier detection, novelty detection or concept learning [18]. One-class approaches to solving classification problems may be superior to discriminative (two-class) approaches, such as decision trees or neural networks [19]. Raskutti and Kowalczyk [15] demonstrated that one-class learning with positive-class examples can be a very robust classification technique when dealing with extremely unbalanced datasets composed of a high-dimensional noisy feature space.

When a traffic incident occurs, the associated traffic data change dramatically, so that the incident observation is different from most of the traffic data. Thus, we can detect traffic incidents by recognizing that the traffic incident data deviate significantly from normal traffic data. From this point of view, one-class classifiers or outlier detection can be employed to address the incident detection problem. A one-class classifier can be built from normal data to detect any deviation from the normal model in the observed data. Given a set of normal data to train from, together with a new piece of test data, the goal is to determine whether the test data belong to "normal" or anomalous behavior. For traffic incident detection, the task is to build an incident detector from the traffic data, that is, a predictive model capable of distinguishing between abnormal traffic states (called incidents) and normal states.

Shyu et al. [20] proposed a novel outlier detection scheme, based on principal components that was applied to intrusion detection. The underlying assumption of such a method is that intrusions appear as outliers in the normal data. Similarly, traffic incident cases also appear as outliers in the normal traffic data. In this paper, this principal component-based approach was used to detect traffic incidents. We used this approach for incident detection mainly due to the advantages of the principal component approach over many anomaly detection methods. First, the principal component approach does not make any distributional assumption. Second, this approach is typically used with high-dimensional datasets. Another benefit of this scheme is that the statistics can be computed in a shorter time during the detection stage so that the scheme can be used in real time. Last but not least, the detection model can be built using only normal cases, thereby avoiding the class imbalance problem.

This paper is organized as follows. Section 2 provides background on principal component analysis (PCA) and outlier detection. Section 3 provides details of the datasets used in the experiments, followed by an analysis of the results; the results are discussed in Section 4. Finally, the work concludes in the last section.

2. Anomaly Detection Scheme

2.1. Principal Component Analysis (PCA). Principal component analysis (PCA) describes the variance-covariance structure of a set of variables in terms of fewer new variables that are linear combinations of the original variables. The new variables are easily obtained from eigenanalysis of the covariance matrix or the correlation matrix of the original data. If the variables are measured on scales with widely different ranges or if the units of measurement are not commensurate, it is preferable to perform PCA on the sample correlation matrix. In our study, we perform PCA on the correlation matrix of the normal group because the features of the data, such as time, volume, or speed, are measured on different scales.

Let the original dataset $X = [X_1 \ X_2 \ \cdots \ X_p]$ be an $n * p$ data matrix of n observations, each consisting of p variables, and let $R = [R_1 \ R_2 \ \cdots \ R_p]$ be the correlation matrix of X. If $(\lambda_1, \mathbf{e}_1), (\lambda_2, \mathbf{e}_2), \ldots, (\lambda_p, \mathbf{e}_p)$ are the p eigenvalue-eigenvector pairs of the matrix $\mathbf{R}$, then the ith principal component is

$$y_i = ze_i = z_1 e_{1i} + z_2 e_{2i} + \cdots + z_p e_{pi}, \quad (i = 1, 2, \ldots, p), \tag{1}$$

where $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_p \geq 0$, $e_i = (e_{1i}, e_{2i}, \ldots, e_{pi})'$ is the ith eigenvector and $z = (z_1, z_2, \ldots, z_p)$ is the standardized matrix defined as

$$z_k = \frac{x_k - \overline{x}_k}{s_k}, \quad (k = 1, 2, \ldots, p), \tag{2}$$

where $\overline{x}_k$ and s_k are the average and standard deviation of the observations in the kth dimension of X, respectively.

2.2. Outlier Detection. Most datasets often contain one or a few samples that do not conform to the general behavior of the dataset and which are called outliers. When an observation is different from most of the data or is sufficiently unlikely under the assumed probability model for the data, the observation is considered to be an outlier. With data on a single feature, outliers are those that are either very large or very small relative to the others. Many features correspond to a complex situation. In high dimensions, all features need to be considered together using a multivariate approach.

The procedure commonly used to detect multivariate outliers is to measure the distance of each observation from the center of the data using the Mahalanobis distance. Any observation larger than a threshold value is considered to be an outlier. The threshold is typically determined from the empirical distribution of the distances.

PCA has long been used for multivariate outlier detection. Consider the sample principal components y_i $(i = 1, 2, \ldots, p)$ of an observation x, the sum of squares of the standardized principal component scores is then given by

$$\sum_{i=1}^{p} \frac{y_i^2}{\lambda_i} = \frac{y_1^2}{\lambda_1} + \frac{y_2^2}{\lambda_2} + \cdots + \frac{y_p^2}{\lambda_p}, \tag{3}$$

which is equivalent to the Mahalanobis distance of the observation x from the sample mean [20]. It is customary to examine the individual principal components or some functions of the principal components for outliers. Hawkins [21] obtained superior performance for a scheme that used statistics derived from principal components to detect errors in multivariate data.

Because the sample principal components are uncorrelated, under the normal assumption and assuming the sample size is large, the major components are given as follows:

$$\sum_{i=1}^{q} \frac{y_i^2}{\lambda_i} = \frac{y_1^2}{\lambda_1} + \frac{y_2^2}{\lambda_2} + \cdots + \frac{y_q^2}{\lambda_q}, \quad q \le p, \tag{4}$$

corresponding to a chi-square distribution with q degrees of freedom. For a given significance level α, an observation x is an outlier if the following criterion is satisfied:

$$\sum_{i=1}^{q} \frac{y_i^2}{\lambda_i} > \chi_q^2(\alpha), \tag{5}$$

where $\chi_q^2(\alpha)$ is the upper α percentage point of the chi-square distribution with q degrees of freedom. The value of α indicates the error or false alarm probability in classifying a normal observation as an outlier. The number of major components, q, can be determined from the amount of the variation in the training data that are described by these components. Based on experiments, Shyu et al. suggested using q major components that can explain approximately 50 percent of the total variation of the standardized features.

In addition to the major components, Shyu et al. [20] proposed that the minor components, $\sum_{i=p-r+1}^{p}(y_i^2/\lambda_i)$, should be used to detect observations that do not conform to the normal correlation structure. The value of r can be determined by examining those components whose variance or eigenvalue is less than 0.20.

An observation is an outlier with respect to the correlation structure if

$$\sum_{i=p-r+1}^{p} \frac{y_i^2}{\lambda_i} > \chi_r^2(\alpha), \tag{6}$$

where $\chi_r^2(\alpha)$ is the critical value for a chi-square distribution with r degrees of freedom testing at a given significance level α.

The anomaly detection scheme differs from other existing approaches in the use of both the major and minor components of the data. A clear advantage of this scheme over others is that outliers can be detected based on being extreme values or not having the same correlation structure as the normal data.

2.3. Incident Detection Scheme. In this paper, the principal component classifier (PCC) mentioned above is used to detect traffic incidents. The procedure for traffic incident detection is as follows.

Step 1. Divide the original dataset into a training set and a testing set. Note that the outlier detection model is constructed from only the normal instances in the training set that correspond to nonincident traffic data.

Step 2. Perform PCA on the correlation matrix of this training set. Determine the number of major components q and the number of minor components r according to the obtained eigenvalues.

Step 3. Determine the outlier thresholds, c_1 and c_2, for a given error probability in the distribution of the test statistics.

As previously mentioned, the collected data are assumed to conform to a multivariate normal distribution, such that the thresholds are the inverse of the chi-square cumulative distribution function; that is,

$$c_1 = \chi_q^2(\alpha_1), \qquad c_2 = \chi_r^2(\alpha_2). \tag{7}$$

In practice, the normality assumption seldom holds true. Therefore, we opt to set the outlier thresholds based on empirical distributions of the test statistics rather than on the chi-square distribution. The values of α_1 and α_2 are chosen to reflect the relative importance of the types of outliers we would like to detect. Without loss of generality, we choose $\alpha_1 = \alpha_2$.

Step 4. Compute the principal component scores for the major components and the minor components for each observation x in the testing dataset.

Step 5. Classify x in the testing set as an outlier corresponding to an incident state if

$$\sum_{i=1}^{q} \frac{y_i^2}{\lambda_i} > c_1 \quad \text{or} \quad \sum_{i=p-r+1}^{p} \frac{y_i^2}{\lambda_i} > c_2. \tag{8}$$

3. Data Description

In this study, we investigated PCC performance in incident detection, as applied to real-life loop detector traffic data collected from the A12 freeway in The Netherlands. The distance between two adjacent detectors installed on A12 is approximately equal to 500 m. Individual vehicle data are collected by the detectors. More specifically, the passing time, the speed, the occupancy time, and the lane in which the vehicle is being driven are recorded for every car passing the detector.

To assess PCC performance in incident detection, two different datasets were employed. The first dataset consisted of loop detector data from many neighboring detectors installed on the A12 Dutch freeways collected during December 2007. The second data source contained information on all the registered incidents that occurred over the period considered on the A12 freeway.

3.1. Loop Detector Data. The loop detector data, in the form of lane-specific traffic volume and speed, were collected at 60-second intervals during normal conditions and incident conditions. Because one incident occurred on the road rather than in a lane, we computed the average volume and speed over all the lanes.

We cannot expect all collected data to be of a high quality. Suspicious or dirty data may be buried in a dataset. Upon close examination of the data, cases where the traffic volume and speed changed dramatically were easily found, but these cases did not always correspond to an incident case registered over that time period in the incident database. If no incident occurred during the respective time period, dramatic changes

in the traffic volume or speed may have been due to a faulty detector or transmission distortion. For anomaly detection to function effectively, such nonpertinent data must be captured and removed from the dataset to improve the veracity and reliability of the traffic information. This procedure is called data quality control or data cleaning. Various methods may be used for this purpose. More details on these methods can be found in the literature [22–24].

3.2. Incident Data. The database contained information on the incidents that occurred during December 2007 as mentioned above; the database included the following information for each incident:

(i) the location of the incident,

(ii) the "approximate" starting time and ending time of the incident,

(iii) a short description of the incident (the lanes in which the incident occurred, a qualitative description of the incident, etc.).

Note, however, that the exact times (start times as well as end times) of the incidents were unknown. The database contained only the time at which the incident was reported and the time at which the end of the incident was reported. Thus, we must keep in mind that the actual starting time would have been a little earlier than the reported starting time.

We constructed the incident dataset using all the incidents with duration between 20 and 180 minutes. We obtained a total of 95 incidents that occurred on the A12 freeways during December 2007. There were 6 columns in this dataset, including the date, direction, location, start time, end time, and duration of the traffic incident.

3.3. Construction of Training and Testing Sets. Incident detection was based on section-related traffic data, which means that the traffic data were collected from two adjacent detectors: an upstream detector and a downstream detector. Each incident instance included the following items:

(i) time (reported as hh + mm/60) of data collection,

(ii) traffic volume and speed from the upstream detector,

(iii) traffic volume and speed from the downstream detector,

(iv) traffic state,

where the item "traffic state" is a label with a value of −1 or 1, which denotes that there was no incident or that an incident occurred, respectively, as determined by the incident dataset.

Each instance in the dataset contained 5 feature values and was labeled as either a normal traffic state or an incident state. The entire dataset was divided into two parts, a training set and a testing set, which were used for calibration and testing, respectively. The training set was composed of 99961 nonincident instances and 2751 incident instances (58 incident cases), collected from 1 to 14 December; the testing set was composed of 67791 samples, including 65957 nonincident instances and 1834 incident instances (33 incident cases), collected from 15–31, December.

The outlier thresholds are determined from the normal instances in training data, and they were used to decide the label of each instance in the testing set; thus the model's detection ability can be computed. The incident instances in the training set were of no use in the PCC. However, the incident instances were used to build other incident detection models for comparison with PCC, such as the partial least squares (PLS) model.

4. Case Studies

Incident detection results are typically evaluated using the following criteria: the detection rate (DR), the false alarm rate (FAR), and the mean time to detection (MTTD). The classification rate (CR) is an additional performance measure of interest. Detailed information on these criteria can be found in references [3–10].

The experiments were conducted within the following framework:

(i) only major components were used to detect outliers,

(ii) both major components and minor components were used,

(iii) PCC was compared with PLS.

Matlab subroutines were written for these procedures and all the algorithms were run on a computer with a 1.50 GHz Intel Pentium processor and 1024 MB of memory.

4.1. Use of Only Major Components. To determine the appropriate number of major components to be used in PCC, we conducted a preliminary study by varying the percentage of total variation explained by the major components. In the distribution of the test statistics, the major components scores were closer to a log-normal distribution than to any other distribution.

The major components were considered to account for 50% up to 100% of the total variation at 5% increments and a 0.04 significance level; thus, we obtained 11 classifiers with different numbers of major components, ranging from 2 to 5. All these classifiers were evaluated using the same testing set. The results showed that as the percentage of the variation explained increased, corresponding to increasing the number of major components used, the values of the four measurements, DR, FAR, MTTD, and CR, appeared to stay the same until 90% of the variation was explained. It means that the PCC method based on the major components that can explain from 50% to 90% of the total variation has the same ability for incident detection.

When more than 90% of the variation was explained, corresponding to all the major components being included in the classifiers, the measurement values changed: DR and FAR increased, while MTTD and CR decreased. Thus, PCC with more major components yielded a higher false alarm rate and a lower classification rate, which is undesirable in incident detection. Figure 1 illustrates the changes in DR,

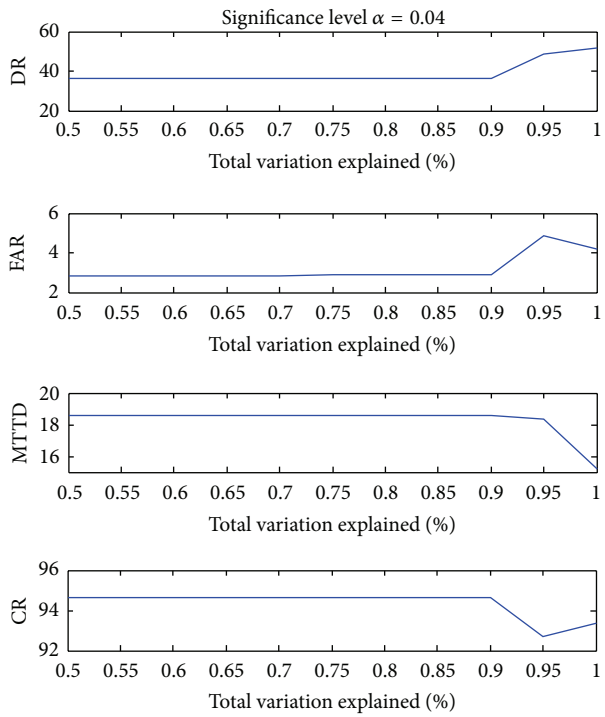

Figure 1: DR, FAR, MTTD, and CR versus the percentage of the variation explained.

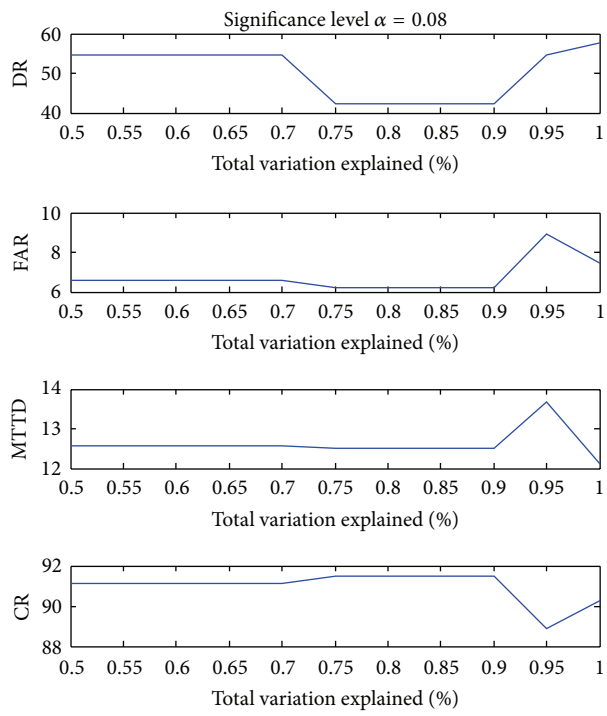

Figure 2: DR, FAR, MTTD, and CR versus the percentage of the variation explained.

FAR, MTTD and CR, as a function of the percentage of the variation explained for a significance level of 0.04.

The significance level was then increased from 0.01 to 0.10 in 0.01 increments and the process described above was repeated. A total of 110 classifiers were obtained. Applying all these classifiers to the same testing set, we observed that when the significance level exceeded 0.06, the PCC models with major components explaining 75% to 90% of total variation performed poorly, as evidenced by a rapid decrease in the detection rate. This result is illustrated in Figure 2 for a significance level of 0.08.

The results suggest that a value of q of 2 or 3 should be used at lower significance levels, while a q value of 2 should be used at a significance level larger than 0.06, when the classifier is only constructed using the major components.

4.2. Use of Both Major Components and Minor Components. We next constructed outlier models using PCC with both the major and minor components; the performance of these models when applied to traffic incident detection was then evaluated. First, the percentage of the variation explained was set to 70% and the upper bound of the eigenvalues for the minor components was set to 0.2 [20]; thus, the numbers of the major and minor components were found to be 2 and 1, respectively. We constructed a series of outlier models with significance levels ranging from 0.01 to 0.10 in 0.01 increments. We always used the same values for α_1 and α_2 in (7), as we did not know in advance which type of outliers we should pay more attention to.

In the distribution of the test statistics, the major components scores followed a log-normal distribution with a freedom degree of 2, while the minor components scores followed a Weibull distribution with a freedom degree of 1: these distributions fit the test statistics more closely than any other distribution.

Figure 3 presents the testing results of these classifiers, showing how the four measurements changed with the significance level. Within increasing significance levels, DR increased while the MTTD decreased, which is desirable for an incident detection model. However, the FAR values were too high to apply the classifiers at a high significance level.

Keeping the percentage of the variation explained the same; the upper bound of the eigenvalues for the minor components was increased to 0.3 and the number of the minor components was increased from 1 to 2; we then constructed another 10 classifiers and tested their performance.

Next, we let the major components account for 90% of the total variation and let the upper bound of the eigenvalues for the minor components range from 0.2 to 0.3; we repeated the experiments above again.

Table 1 shows the testing results obtained for 4 classifiers that were built with different numbers of major components and minor components due to different values of the parameters setting. The significance level for all 4 classifiers was set to 0.08. The first column is the parameters setting; for example, (70%, 0.2) indicated that the major components accounted for 70% of the total variation and that the upper bound of the eigenvalues for the minor components was 0.2. The column "numbers" corresponds to the number of major and minor components used.

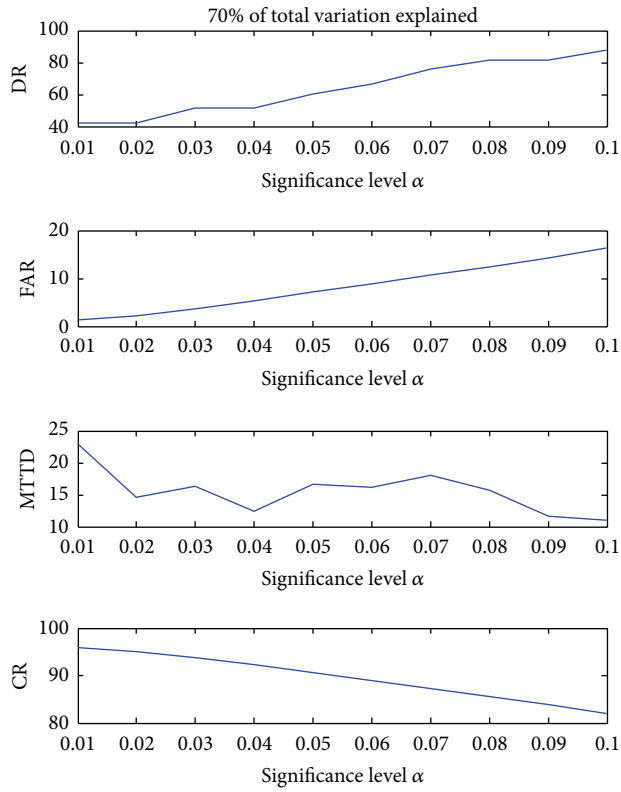

Figure 3: DR, FAR, MTTD and CR versus significance level.

Table 1: Comparison of the performance of classifiers built with different parameters.

Classifiers		DR	FAR	MTTD	CR
Parameters	Numbers				
(70%, 0.2)	(2, 1)	**81.82**	12.48	15.63	85.59
(70%, 0.3)	(2, 2)	72.73	12.04	12.21	85.97
(90%, 0.2)	(3, 1)	72.73	12.10	16.29	85.93
(90%, 0.3)	(3, 2)	63.64	**11.65**	**11.76**	**86.32**
Only major components		54.55	6.60	12.56	91.11

In this table, the best result in each column is shown in bold (not including the last row). The classifiers with the parameter pair (90%, 0.3) outperformed the other classifiers in terms of the FAR, MTTD, and CR values. However, the detection rate for the parameter pair (90%, 0.3) did not perform acceptably. The table shows that as the percentage of the variation explained increased, DR tended to decrease, while FAR and CR exhibited small fluctuations and performed at a similar level. Another observation is that a classifier yielded good MTTD values when more minor components were used by increasing the upper bound of the eigenvalues for the minor components.

For comparison, we also list the testing results achieved by a classifier that only used the major components in the last line of Table 1. The significance level was also set to 0.08 and the major components accounted for 70% of the total variation. Although this classifier achieved the best values for FAR and CR, the DR values were too low to be used in practice. A classifier using minor components can thus dramatically improve the detection rate of models, as has been proven by Shyu et al. [20].

Table 2: Performance of PLSR built with a different proportion of incident instances.

Proportion	DR	FAR	MTTD	CR
35	12.12	0.90	5.00	96.49
40	36.36	4.22	15.00	93.42
45	**69.70**	**13.07**	**11.04**	**85.12**
50	93.94	32.50	7.03	66.72
55	100.00	53.80	2.58	46.69

4.3. Comparison with PLS. Wang et al. [3, 4] developed automatic incident detection (AID) models based on partial least squares regression, which were compared to the use of support vector machines for freeway incident detection. Here, we compare PCC and PLS for incident detection.

The training dataset had 102712 instances, with only 2751 incident instances compared to 99961 nonincident instances. The class distribution was therefore highly skewed as the frequency of the main class was more than 97%. The PLSR model is sensitive to imbalanced training data; that is, the proportion of incident samples in the training set strongly influences the detection performance [3]. To address this problem, we discarded random nonincident instances while retaining all the incident instances to increase the proportion of incident instances. In this way, we obtained a series of new training datasets with the proportion of incident instances ranging from 35% to 55% in 5% increments. Then, we built a PLSR model for each new set. Next, we tested the detection performance of the PLSR models with the testing dataset mentioned above.

Table 2 shows the testing results obtained with the PLSR model. Using the dataset containing 40% to 50% of incident instances to build the PLSR resulted in relatively good model performance. If there were too few incident instances, the PLSR model would produce DR values that would be too low; if there were too many incident instances, the PLSR model would produce FAR values that would be too high. Taking all these criteria into consideration, we chose the classifier modeled on the training set with 45% incident instances for further study. The testing results of this classifier are shown in bold.

Twelve classifiers were then constructed by allowing the significance level to range from 0.07 to 0.09, adjusting the percentage of the variation explained from 0.7 to 0.9, and increasing the upper bound of the eigenvalues for the minor components from 0.2 to 0.3. The detection performances of the 12 classifiers are presented in Table 3, along with the average detection performance; the PLSR performance is listed in the last row for comparison.

The results show the variation in classifier performance. Table 3 shows that the upper bound of the eigenvalues for the minor components strongly influenced the DR and MTTD values. To increase its value, the MTTD values decreased,

Table 3: Performance comparison between PCC and PLSR.

Classifiers			DR	FAR	MTTD	CR
Significance	Explained	Bound				
0.7	70%	0.2	75.76	10.78	18.08	87.16
		0.3	69.70	10.75	12.65	87.17
	90%	0.2	69.70	10.40	18.57	87.53
		0.3	63.64	10.31	12.29	87.58
0.8	70%	0.2	81.82	12.48	15.63	85.59
		0.3	72.73	12.04	12.21	85.97
	90%	0.2	72.73	12.10	16.29	85.93
		0.3	63.64	11.65	11.76	86.32
0.9	70%	0.2	81.82	14.37	11.70	83.81
		0.3	72.73	13.42	11.92	84.68
	90%	0.2	75.76	13.93	11.64	84.22
		0.3	66.67	13.01	11.18	85.05
Average of PCC			75.76	13.93	11.64	84.22
PLSR			69.70	13.07	11.04	85.12

which was a desirable result. Unfortunately, the DR values also decreased as the MTTD values decreased. Comparing the PCC and PLSR model results, PCC appeared to exhibit poorer performance in terms of the FAR, MTTD, and CR values, but yielded higher average DR values. However, we should keep in mind that the highest performing PLSR model was compared with the PCC model results. If the average PLSR performance was compared to PCC performance, PLSR performance would be observed to be slightly inferior to PCC performance.

5. Conclusion

The detection of traffic incidents, congestion, and other traffic operational problems is a very important component of traffic system operation. The task of incident detection can be regarded as constructing classifiers from imbalanced or skewed datasets. In such problems, almost all the instances are labeled as a nonincident class, while far fewer instances are labeled as an incident class, which is usually the more important class. The learned classifier determines whether an incident occurs using traffic flow measurements, so that classifying the traffic state is any efficient means of using the class imbalance problem to solve the incident detection problem.

In this study, we determined the applicability of principal component analysis. PCA is often applied to reduce the dimensionality of a problem, as well as to detect outliers. In this paper, PCA was used to construct a simple classifier to detect incidents. This classifier consisted of two simple functions, the major components and the minor components. Only a few parameters, the significance level, the percentage of the total variation explained by the major components, and the upper bound of the eigenvalues for the minor components, needed to be retained for future detection. The influence of these parameters on detection performance was discussed in our experiments.

PCA performance was tested on a real dataset collected from the A12 freeway in The Netherlands. The PCC results were compared to the results of the standard linear PLSR method. The experimental results showed that PCC outperformed the PLSR model for AID.

Although the test results showed that PCC can achieve good incident detection performance, there is still room for improvement; the FAR values were too high and the MTTD values were too high with PCC. These results may be attributed to the poor quality of real traffic data. Real traffic data are always easily contaminated by a high noise level in the dataset, due to missing values, transcription errors, incomplete information, and the absence of standard formats. Learning from noisy data is a challenging and practical issue for real-world data mining applications. Common practices include data cleaning, error detection, and classifier ensembles [25, 26].

Refinements in data quality show promise in terms of improving incident detection performance. Note that common data cleaning methods are not suitable for AID data because these methods recognize incident instances as noise and consequently delete the incidents based on their rarity. Therefore, further research should focus on developing suitable algorithms for data cleaning.

Acknowledgments

The authors would like to express their gratitude to Dr. Sascha Hoogendoorn in DVS Center for Transport and Navigation, Ministry of Transport, Public Works and Water Management, The Netherlands, and Dr. Yusen Chen at the Delft University of Technology for their assistance in obtaining traffic data. This work was supported by the National Natural Science Foundation of China under Grant no. 61074141, the National High Technology Research and Development Program of China (863 Program) under Grant no. 2012AA112304, and the Natural Science Foundation of Jiangsu Province, China, under Grant no. BK2011745.

References

[1] B. Hellinga and G. Knapp, "AVI based freeway incident detection," in *Proceedings of the TRB Annual Meeting*, pp. 1–26, 2000.

[2] J. Black and I. Sreedevi, "Detection Algorithms," http://www.calccit.org/itsdecision/serv_and_tech/Incident_management/incident_management_overview.html.

[3] W. Wang, S. Chen, and G. Qu, "Incident detection algorithm based on partial least squares regression," *Transportation Research Part C*, vol. 16, no. 1, pp. 54–70, 2008.

[4] W. Wang, S. Chen, and G. Qu, "Comparison between partial least squares regression and support vector machine for freeway incident detection," in *Proceedings of the 10th International IEEE Conference on Intelligent Transportation Systems, ITSC 2007*, pp. 190–195, Seattle, Wash, USA, October 2007.

[5] F. Yuan and R. L. Cheu, "Incident detection using support vector machines," *Transportation Research Part C*, vol. 11, no. 3-4, pp. 309–328, 2003.

[6] R. L. Cheu, D. Srinivasan, and E. T. Teh, "Support vector machine models for freeway incident detection," in *Proceedings of Intelligent Transportation Systems*, vol. 1, pp. 238–243, 2003.

[7] S.-Y. Chen, W. Wang, and G.-F. Qu, "Traffic incident detection based on rough sets approach," in *Proceedings of the 6th International Conference on Machine Learning and Cybernetics (ICMLC '07)*, pp. 3734–3739, August 2007.

[8] S. Chen, W. Wang, G. Qu, and J. Lu, "Application of neural network ensembles to incident detection," in *Proceedings of the IEEE International Conference on Integration Technology (ICIT '07)*, pp. 388–393, Shenzhen, China, March 2007.

[9] S. Chen, W. Wang, and H. van Zuylen, "Construct support vector machine ensemble to detect traffic incident," *Expert Systems with Applications*, vol. 36, no. 8, pp. 10976–10986, 2009.

[10] S. Chen and W. Wang, "Decision tree learning for freeway automatic incident detection," *Expert Systems with Applications*, vol. 36, no. 2, pp. 4101–4105, 2009.

[11] S. Kotsiantis, D. Kanellopoulos, and P. Pintelas, "Dimitris kanellopoulos, and panayiotis pintelas. Handling imbalanced datasets: a review," *GESTS International Transactions on Computer Science and Engineering*, no. 1, pp. 25–36, 2006.

[12] N. Japkowicz and S. Stephen, "The class imbalance problem: a systematic study," *Intelligent Data Analysis*, vol. 6, no. 5, pp. 429–450, 2002.

[13] A. Estabrooks, T. Jo, and N. Japkowicz, "A multiple resampling method for learning from imbalanced data sets," *Computational Intelligence*, vol. 20, no. 1, pp. 18–36, 2004.

[14] C. Elkan, "The foundations of cost-sensitive learning," in *Proceedings of the 17th International Joint Conference on Artificial Intelligence*, pp. 973–978, Seattle, Wash, USA, 2001.

[15] B. Raskutti and A. Kowalczyk, "Extreme rebalancing for svms: a case study," *SIGKDD Explorations*, vol. 6, pp. 60–69, 2004.

[16] R. E. Schapire, "A brief introduction to boosting," in *Proceedings of the 16th International Joint Conference on Artificial Intelligence*, pp. 1401–1406, 1999.

[17] Y. Freund and R. Schapire, "Experiments with a new boosting algorithm," in *Proceedings of the 13th International Conference on Machine Learning*, M. Kaufmann, Ed., pp. 148–156, SanFrancisco, Calif, USA, 1996.

[18] S. S. Khan and M. G. Madden, "A survey of recent trends in one class classification," *Lecture Notes in Computer Science*, vol. 6206, pp. 188–197, 2010.

[19] N. Japkowicz, "Concept-learning in the presence of between-class and within-class imbalances," in *Proceedings of the Fourteenth Conference of the Canadian Society for Computational Studies of Intelligence*, pp. 67–77, 2001.

[20] M. -L. Shyu, S. -C. Chen, K. Sarinnapakorn et al., "A novel abnormal detection scheme based on principal component classifier," in *The IEEE International Conference on Data Mining series (ICDM ,03)*, 2003.

[21] D. M. Hawkins, "The detection of errors in multivariate data using principal components," *Journal of the American Statistical Association*, vol. 69, no. 346, pp. 340–344, 1974.

[22] S. Turner, L. Albert, B. Gajewski, and W. Eisele, "Archived intelligent transportation system data quality: preliminary analyses of San Antonio TransGuide data," *Transportation Research Record*, no. 1719, pp. 77–84, 2000.

[23] J. Han and M. Kamber, *Data Mining: Concepts and Techniques*, Morgan Kaufmann Publishers, 2001.

[24] S. Chen, W. Wang, and H. van Zuylen, "A comparison of outlier detection algorithms for ITS data," *Expert Systems with Applications*, vol. 37, no. 2, pp. 1169–1178, 2010.

[25] Y. Zhang, X. Zhu, X. Wu, and J. P. Bond, "ACE: an aggressive classifier ensemble with error detection, correction and cleansing," in *Proceedings of the 17th IEEE International Conference on Tools with Artificial Intelligence (ICTAI '05)*, pp. 310–317, chn, November 2005.

[26] V. Brusic, J. Zeleznikow, T. Sturniolo, E. Bono, and J. Hammer, "Data cleansing for computer models: a case study from immunology," in *Proceedings of 6th International Conference on Neural Information Processing (ICONIP '99)*, pp. 603–6609, 1999.