Civil Engineering Infrastructures Journal, 51(1): 55 – 86, June 2018
Print ISSN: 2322-2093; Online ISSN: 2423-6691
DOI: 10.7508/ceij.2018.01.004

Identification of Structural Defects Using Computer Algorithms

Yasi, B.[1] and Mohammadizadeh, M.R.[2*]

[1] M.Sc., Department of Civil Engineering, Faculty of Engineering, University of Hormozgan, Bandar Abbas, Iran.
[2] Assistant Professor, Department of Civil Engineering, Faculty of Engineering, University of Hormozgan, Bandar Abbas, Iran.

Received: 09 May 2017; Revised: 24 Feb. 2018; Accepted: 25 Feb. 2018

ABSTRACT: One of the numerous methods recently employed to study the health of structures is the identification of anomaly in data obtained for the condition of the structure, e.g. the frequencies for the structural modes, stress, strain, displacement, speed, and acceleration) which are obtained and stored by various sensors. The methods of identification applied for anomalies attempt to discover and recognize patterns governing data which run in sharp contrast to the statistical population. In the case of data obtained from sensors, data appearing in contrast to others, i.e. outliers, may signal the occurrence of damage in the structure. The present research aims to employ computer algorithms to identify structural defects based on data gathered by sensors indicating structural conditions. The present research investigates the performance of various methods including Artificial Neural Networks (ANN), Density-Based Spatial Clustering of Applications with Noise (DBSCAN), Manhattan Distance, Curve Fitting, and Box Plot in the identification of samples from damages in a case study using frequency values related to a cable-support bridge. Subsequent to the implementation of the methods in the datasets, it was shown that the ANN provided the optimal performance.

Keywords: Artificial Neural Networks, Damage Identification, Frequency, Manhattan Distance, Structures.

INTRODUCTION

Structures are always subject to various risks which might endanger their health. Structural damage may lead to loss of function and unavailability. Many factors adversely affect the structure causing damage. These factors include imbalance, manufacturing problems, corrosion, scaling and deposits on system components, looseness, tear and wear, abrasion and erosion of components, use of systems beyond their intended service, lack of experience on the part of the operator, various forces being exerted on the system such as hydraulic and aerodynamic forces, fatigue, surface defects, failure of components, impact and collision, and deformation of elements.

Various methods are available for assessing the health and damages sustained by structures. These methods predict the occurrence of defects preventing their progress from going unnoticed. These form part of the condition monitoring system.

* Corresponding author E-mail: mrzmohammadizadeh@yahoo.com

Condition monitoring aims to obtain the signs and indicators revealing the condition of the structure in operation such that the damaged structure can be repaired and maintained safely and economically. Most structural defects are accompanied by signs and indicators by means of which the occurrence of the defect can be predicted. This procedure is more cost-effective than predictive or regular maintenance or repair preventing the early replacement of parts.

Numerous methods have been proposed and applied in the literature to detect anomalous data. The following are some of these methods.

Loureiro et al. (2004) employed the clustering method to identify false international contract data gathered by the Portuguese Statistical Association INE. The identification of the false data was a very demanding, time-consuming, and significant task. The present research utilized hierarchical clustering method to categorize false data which were presented to the inspectors (Loureiro et al., 2004). Among the research studies conducted on the clustering method one can refer to Alguliyev et al. (2017), Bai et al. (2011), Gagolewski et al. (2016), Jiang et al. (2016) and Zhu et al. (2018). Massert and Smeyers (2005) showed that plots can be employed to analyze data drawing box plots for real data and describing the reasoning for box plots. They used curve fitting robust nonlinear regression to clearly identify false data. The results of this research showed that the false discover rate in this method is less than one percent. Thus, the low average false discover rate could be considered as a successful step towards identifying false data using curve fitting (Motulsky and Brown, 2006). Zhuang et al. (2007) used the DBSCAN to solve the problems inherent in the quality of data observed in Ocean Biogeographic Information System (OIBS). This method was used successfully to identify, categorize and cluster OIBS data related to remote geographical places (Zhuang et al., 2004). Beliakov et al. (2011) asserted that, in linear and nonlinear models, to discover false data under conditions where data present an unnatural behavior or do not follow the assumption they cannot perform satisfactorily identifying a large number of true data as being false. They proposed the implementation of the ANN for false data discovery showing that the method is more accurate (Beliakov et al., 2011). Sinwar and Kaushik (2014) compared two real and artificial datasets and two Euclidean and Manhattan distance criteria for false data discovery purposes. The data were initially organized in clusters. Then, using the theoretical analysis and experimental results it was shown that the Euclidean acts far better than the Manhattan distance (Sinwar and Kaushik, 2014). Among the research studies conducted on the other methods to detect outliers can be referred to Bai et al. (2016); Tang and He (2017); Huang et al. (2016) and Alguliyevet al. (2017).

The review of literature revealed a paucity of research on false and anomalous data detection and discovery. Consequently, it may not be possible to rely on the results obtained. Therefore, the present research investigates the discovery of false data using the five method of Artificial Neural Network (ANN), Density-Based Spatial Clustering of Applications with Noise (DBSCAN), Manhattan Distance, Curve Fitting, and Box Plot. Then, the efficiency and performance of the afore-mentioned methods are compared.

Damage detection methods ranging from visual methods to more advanced approaches such as the use of digital instruments in conjunction with novel computer algorithms can enhance the accuracy of health monitoring of structures. The present research aims to employ the Anomaly Detection Algorithm in machine learning for the purpose of health monitoring structures.

The methods based on anomaly detection refer to the comparison of normal conditions of the system with the observed conditions so as to detect serious differences which normally occur during the sustainment of damages. This difference is then compared with the threshold limit. The threshold limit refers to the value representing candidacy for damage where the difference between the data for the observed conditions exceeds the value for normal conditions. Health monitoring systems based on anomaly detection algorithms are well documented in the literature and represent the appearance of conditions governing various structural elements. From among these one can refer to the frequencies measured through condition monitoring of the structure (Gaffney and Ulvila, 2001). These precedents emerge as a result of exploring and recording the performance of the system in a given period and time interval. For instance, these documents can refer to the displacements occurring in a structure during several years. Health monitoring tools are intended to measure the characteristics of the existing conditions comparing them with the threshold stored in the system records. For instance, the ratio of structural displacement might exceed the threshold. As a result, it may be inferred that the structure has sustained damages.

The main advantage of health monitoring systems acting on the basis anomaly detection algorithms is that they can detect various hitherto-unknown damages whose patterns have not been already observed. The background and the precedence for the structure employed in these systems is recorded and investigated in a learning phase which might take days or weeks. These records may be of a fixed nature or change comparatively with the passage of time. In the first procedure, the data remain constant unless the learning phase is resumed by the system manager due to varying system conditions as time passes. In the comparative profile method, not many problems occur as time passes. Nevertheless, the possibility still exists that the damage gradually induces the changes intended and lapse of time may make the damage detection system treat this behavior as being normal. Another issue concerning anomaly detection methods is the fact that, due to the varied nature of structural complexities and behaviors, the establishment of data precedence and background necessitates high accuracy. Furthermore, the accurate detection of the exact cause of the anomaly is not possible. It may be possible for system data updating, which requires data transfer and numerous connections, not to be contemplated in the learning phase. Thus, during system operation, such changes spontaneously cause the appearance of false and pointless warnings. In situations where the system produces warnings, it may be extremely difficult to ascertain whether the warning is justified. Nevertheless, the determination of damage type on the basis of measured parameters is a very demanding task (Karim et al., 2014).

MATERIALS AND METHODS

Datasets under Study

In a paper entitled “Structural Health Monitoring of Cable-supported Bridges Based On Vibration Measurements” an attempt was made to eliminate the effect of vibration-based damage and temperature detection. Parametric and nonparametric procedures were introduced to observe the effects in vibration-based damage detection with illustrative examples given.

Five long span bridges were investigated in Hong Kong. These were Tsing Ma (suspension), Kap Shui Mun (cable-stayed), Ting Kau (cable-stayed), Western Corridor (cable-stayed),and Stonecutters (cable-stayed). The health monitoring system was

employed in these bridges to instantaneously measure the four sets of parameters used. These parameters were (Ni, 2014):

1. Environmental conditions: wind, temperature, vibrations and seismic situation, humidity, corrosion status, etc.

2. Operational loads: highway traffic and railway traffic

3. Bridge features: including static features such as influence coefficients and dynamic features such as modal parameters

4. Bridge responses: geometrical profile, cable force displacement/detection, strain/stress histories, cumulative fatigue damage, etc.

Figure 1 shows the health monitoring systems employed in the bridges. In this research, frequency data for the initial five modes of the Ting Kau bridge were studied. Figure 2 depicts the data related to the measurements.

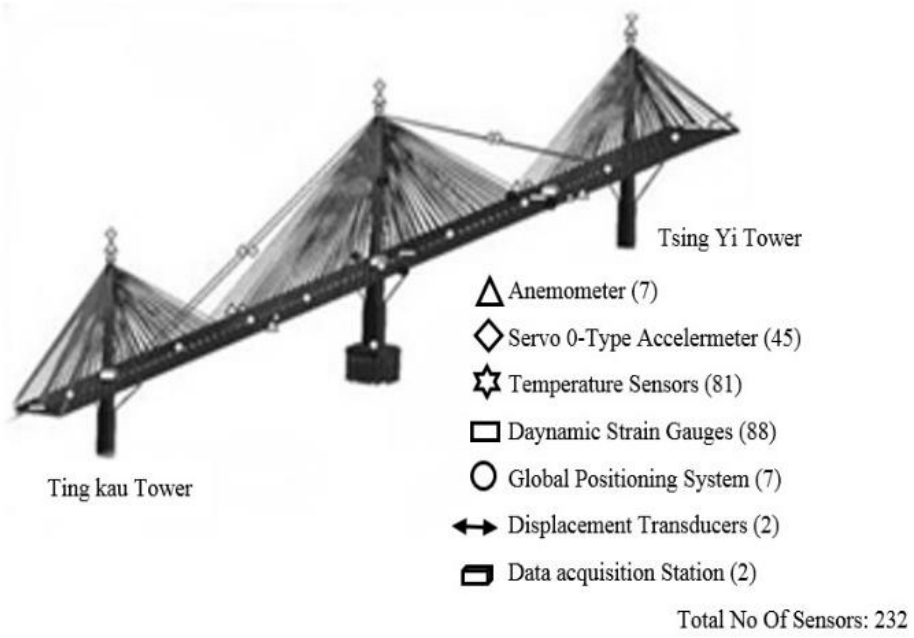


Fig. 1. The Ting Kau Bridge (TKB) health monitoring system (Ni, 2014)

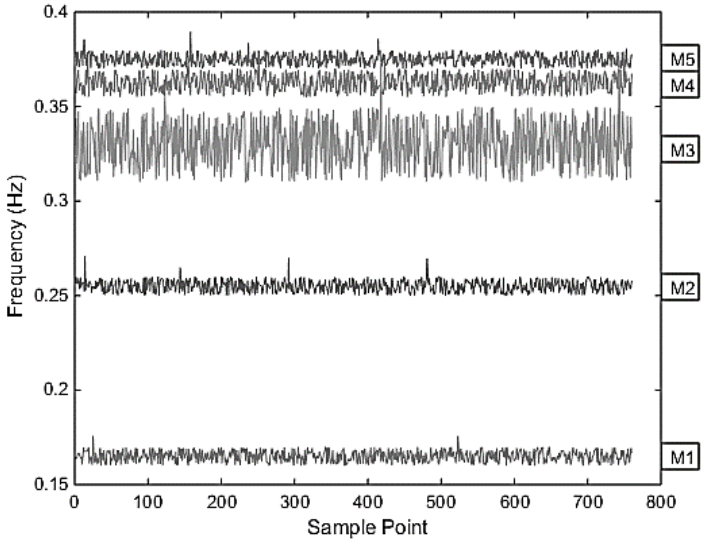


Fig. 2. Modal frequencies measured for the initial five modes (Ni, 2014)

In this section, damage data detection methods used in the present study are described. These methods are: Artificial Neural Networks, DBSCAN, Manhattan Distance, Curve Fitting, and Box Plot.

Artificial Neural Network (ANN)

Artificial Neural Networks are among the most frequently used and practical methods for modelling complex and wide-ranging problems. Artificial Neural Networks can be employed for problems concerning clustering and categorization (with the output being a class or a category) or regression (with the output being a numeric value). Each neural network incorporates an input layer where each node represents a prediction variable. The nodes of the middle layer are referred to as nodes of the hidden layer. Each input node is connected to all hidden layer nodes. The nodes in the hidden layer can make connections to the nodes of another hidden layer or the nodes in the output layer. The output layer consists of one or several output variable (Hand et al., 2001). Figure 3 shows the afore-mentioned concepts.

Each connection joining, say X and Y nodes possesses a weight represented by Wxy. These weights are used in intervening layer calculations where each node in an intervening layer possesses several inputs from different connections each having a specific weight (Figure 4).

Each node in the intervening layer allows the multiplication of the input value and the weight of the respective connection summing up these products. Then, a predetermined function, known as the activation function, operates on the summation, dispatching the result as output to the nodes in the subsequent layer.

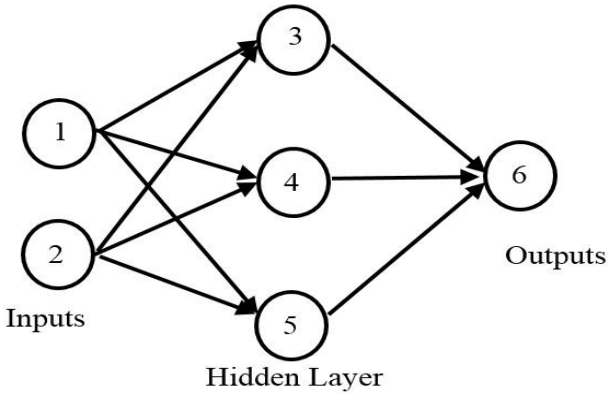


Fig. 3. A neural network having one hidden layer (Hand et al., 2001)

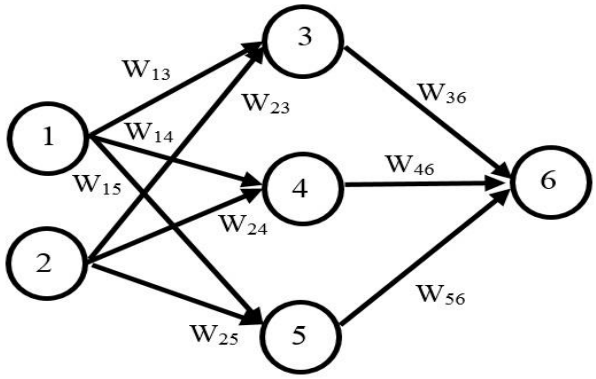


Fig. 4. Wxy represents the weight of connection between X and Y (Hand et al., 2001)

The weight of connections are unknown parameters determined by the training function and data supplied to the system. The number of nodes and hidden layers and the manner in which these are interconnected specifies the architecture or topology of the artificial network. The user or the neural network software needs to specify the number of nodes and hidden layers, the activation function, and the limitations and constraints governing the weight of connections. One of the most important types of ANN is the Feed Forward Backpropagation which is employed in the present research. The following is a brief description of Feed Forward Backpropagation neural network:

Feed Forward: The term "Feed Forward" signifies that fact that the value for output parameter is determined on the basis of input variables and primary weightings. Input variables combine and are used in the hidden layers and the values in the hidden layers combine to calculate the output values.

Back propagation: Comparing the input value with the value obtained from test data, the output error is calculated. This value is utilized to correct the network and to change the weights of the connections. This process starts from the output node with the calculations proceeding backwards to the input node. This procedure is repeated for each record in the data bank. Each implementation of this algorithm for all the data existing in the data bank is referred to as an "epoch". Epochs continue so long as the value of error does not change.

There are various criteria for the assessment of the performance of the ANN in predicting the model. In the present research, the Mean Square Error (MSE) and the Correlation Factor (R) are utilized for this purpose. In general, the closer MSE gets to zero and R to 1 the more the network is optimized and efficient. The values for these parameters can be obtained from the following relations:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(x_i - y_i)^2 \quad (1)$$

$$R = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2 \sum_{i=1}^{n}(y_i - \bar{y})^2}} \quad (2)$$

where y_i: represents the predicted data, x_i: is the measured data, $\bar{y}$: is the mean of predicted data, $\bar{x}$: is the mean of measured data, and n: denotes the number of measured data.

To detect the data which represent candidates for damage the artificial network model is used to establish a model for normal behavior. The normal model is one in which data which are not damaged are specifically introduced residing there. Any data whose difference exceeds the value from the prepared normal model represents a candidate for damage.

Density-Based Spatial Clustering of Applications with Noise

Clustering can be considered as an unsupervised spontaneous automatic learning approach whereby data are divided into categories or clusters whose members are similar. Thus, a cluster refers to a dataset with similar data which are dissimilar to data in other clusters. The DBSCAN was first introduced by (Ester et al., 1996). This algorithm is a method for clustering based on data density. In this method, to estimate the density of distribution of points two parameters, i.e. neighborhood radius (ε) and the minimum points to form a cluster (MinPts), were employed. This algorithm begins with any arbitrary point. The points in the neighborhood of this point and having a distance of less than ε are counted. If the number of points exceeds MinPts, a cluster is formed. Otherwise, the point under study is considered an outlier. The important concept in this approach is that this point may be classified under another cluster at later stages. The other advantage lies in the possibility of identifying and differentiating outliers from

other data. To implement the clustering method of DBSCAN the following terms need to be defined:

Local point density at point p:

If p is defined as the core point of a neighborhood and ε the neighbourhood radius for point p, then neighbourhood to the radius ε can be defined in the following way:

$$N_{\varepsilon} = \{q \; in \; data \; set \; D | dist(p,q) \leq \varepsilon\} \quad (3)$$

where N_{ε}: represents the number of points lying within a neighborhood, D: is the dataset and q: is a member of the data set and dist denotes the function of measurement of distance between the points (e.g. Euclidean distance).

The number of points lying within a neighborhood is referred to as the density of the points of the neighborhood. For instance, in Figure 5, the data density in the neighborhood of (ε) from core p is equal to 5.

Directly-Reachable Density for Point p

The data point is said to be directly reachable by density q if it lies within the neighborhood ε from the core q. Figure 6 depicts this concept:

Density-Connected p: Datum p is said to be density-connected if there exist data such as o which are reachable by both densities p and q. Figure 7 depicts this concept:

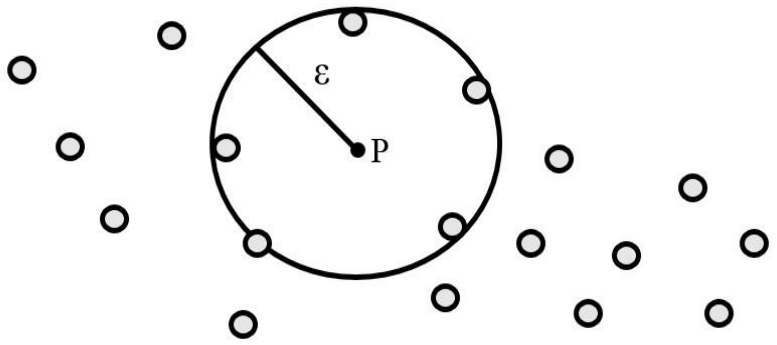


Fig. 5. The datum p having a density of 5 in the neighborhood radius ε (Ester et al., 1996)

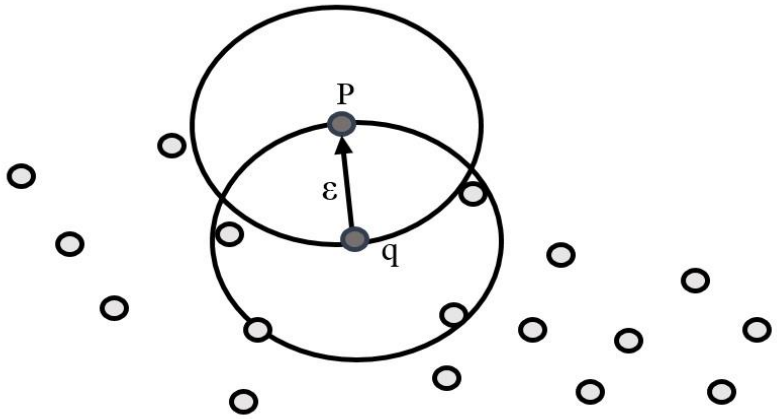


Fig. 6. Datum p is reachable by data density q (Ester et al., 1996)

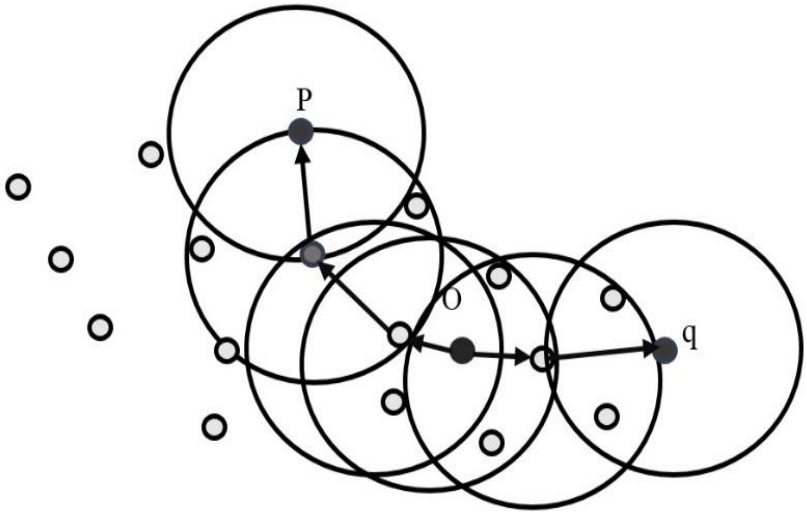


Fig. 7. *p* is density-connected *q* (Ester et al., 1996)

Density-based cluster: This represents a nonempty set (S) from the dataset (D) satisfying the following two conditions:

- If *p* lies within S and q is reachable by density *q*, then *q* also belongs to S.
- Each data point within S is density-connected to other data within S.

Density-Based Clustering

Density-based clustering on the dataset D represents the set {S1, …, Sn, N} such that:

- S1, …, Sn represents all the density-connected clusters within D.
- N is referred to as the noise set.

Figure 8 depicts the afore-mentioned concept.

DBSCAN

In this clustering method, each data belonging to the cluster C (the productive cluster) is available to densities for other data belonging to the cluster but is not available to any other data. The quasi code for this algorithm is outlined in Figure 9. Figure 10 depicts an example for DBSCAN.

During the detection of outliers using clustering algorithm, if the data do not belong to any clusters, it can be classified as outliers.

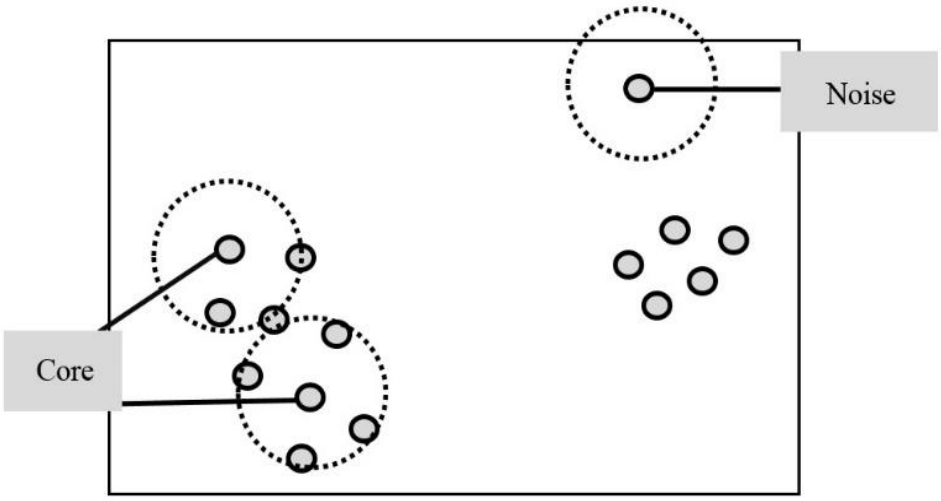


Fig. 8. Density-based clustering (Ester et al., 1996)

```
For each o ∈ D do
   if o is not yet classified then
      if o is a core-object then
         collect all objects density-reachable from o
               and assign them to a new cluster
      else
               assign o to NOISE
```

Fig. 9. Quasi code for the DBSCAN algorithm (Ester et al., 1996)

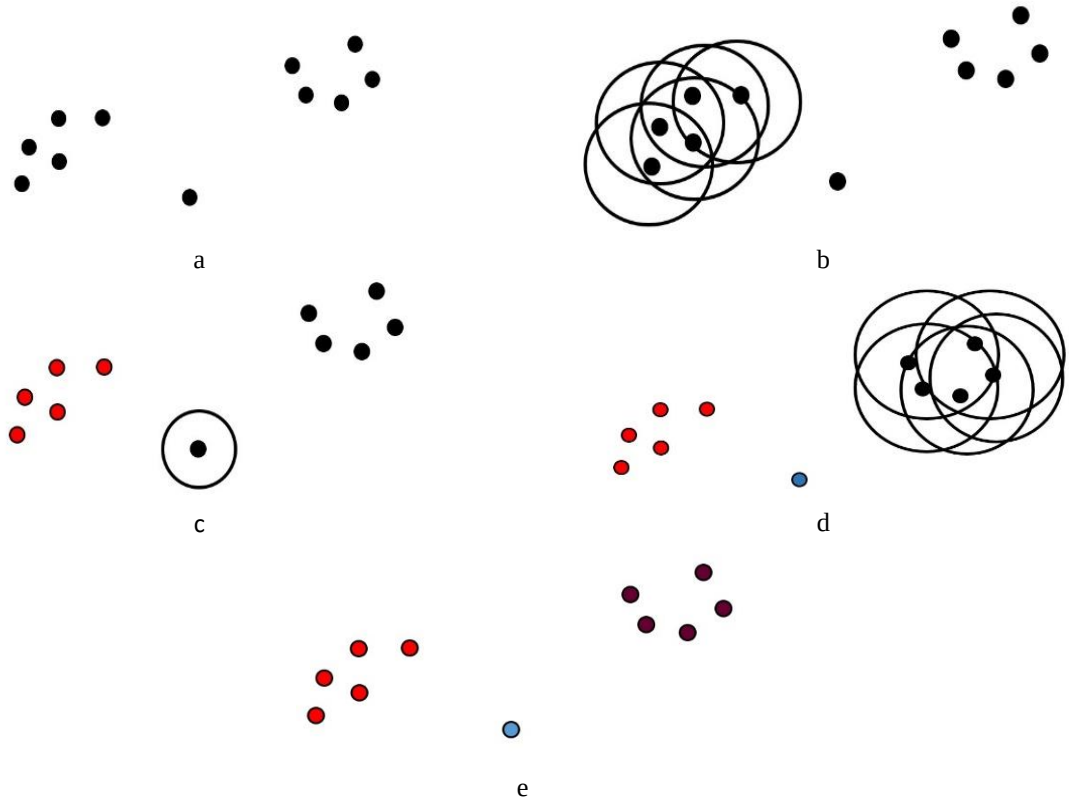


Fig. 10. An example for data clustering using DBSCAN (Ester et al., 1996)

Manhattan Distance

Manhattan Distance is a parametric criterion which depends on the estimation of distribution of multivariate parameters and data covariance (Johnson and Wichern, 1992). If the next dim dataset (the dim variable represents the number of dimensions of the set of input data) possesses n observations (the variable n represents the number of measured samples), $\bar{x}$ the average vector and cov the covariance matrix for the dataset, then we have:

$$Cov = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \overline{x})(x_i - \overline{x})^T \qquad (4)$$

The covariance for a multivariate dataset is a matrix but in the present study the result is a scalar numerical value as input variables are univariate. The scalar values are also of the one by one matrix type. As a result, the Manhattan distance is calculated using the following relation:

$$M_i = \sqrt{(x_i - \overline{x})^T Cov^{-1}(x_i - \overline{x})} \qquad (5)$$

If the value calculated for M_i for the test data x_i is larger than the threshold, then x_i observed can be a candidate for an outlier.

Curve Fitting

Another method for the outlier detection is curve fitting the data which can be used for both univariate and multivariate data. To detect outliers using the above method, residuals (difference between real and estimated values) are initially calculated. Then, larger values are selected as candidates for outliers. There are a number of methods for detecting outliers using residuals. One of these methods is standardized residuals which is described below:

$$d_i = \frac{e_i}{\sqrt{MSE}} \quad (6)$$

The expression $e_i = y_i - \hat{y}_i$ represents error prediction and MSE stand for Mean Square Error. Large d_i values (usually $d_i > 3$) represent outliers (Montgomery et al., 2012).

Numerous methods are available for curve fitting data. In the present research, the method of sum of sines is used to curve fit the data:

$$y = \sum_{i=1}^{n} a_i \sin(b_i x + c_i) \quad (7)$$

in this method, the number of (n) series is initially determined. Then, using Least Squares and Trust Region algorithm, the coefficients for a_i, b_i, and c_i are calculated. Least Squares is a statistical method to solve the set of equations in which the number of equations exceeds the number of unknowns. This method is mainly used in regression. In the present research, a_i represents amplitude, b_i is the frequency, and c_i is the phase constant for each sinusoidal term.

Box Plot Method

In descriptive statistics, the box plot describes the variation of data. In this statistical tool, a box is used to show the distance between the first and the third quartiles with a line in the box representing the median (second quartile). The minimum and the maximum data values represents outside the box. In fact, the box plot is a graphic method describing the distribution of data using five major characteristics (Frigge et al., 2012):

- the minimum normal observation (min)
- the lower quartile (Q1)
- the median
- the upper quartile (Q3)
- the maximum normal observation (Max)

The value (Q3-Q1) represents the Inter Quartile Range (IQR). Using the parameter, it is possible to examine whether data are normal or not (outlier). Data which are $1.5 \times$ IQR times smaller than Q1 or $1.5 \times$ IQR times larger than Q3 can be considered as candidates for outliers. The concepts enumerated above are depicted in Figure 11.

Performance Validation Criteria

Data anomaly detection algorithms are usually validated using detection rate and false alarm rate (Latecki et al., 2007). The following four parameters are employed to define these criteria:

- The TP parameter: The number of real anomalous data which are correctly detected as anomalous data.
- The FN parameter: The number of real anomalous data which are incorrectly detected as normal data.
- The FP parameter: The number of real normal data which are incorrectly detected as anomalous data.
- The TN parameter: The number of real normal data which are correctly detected as normal data.

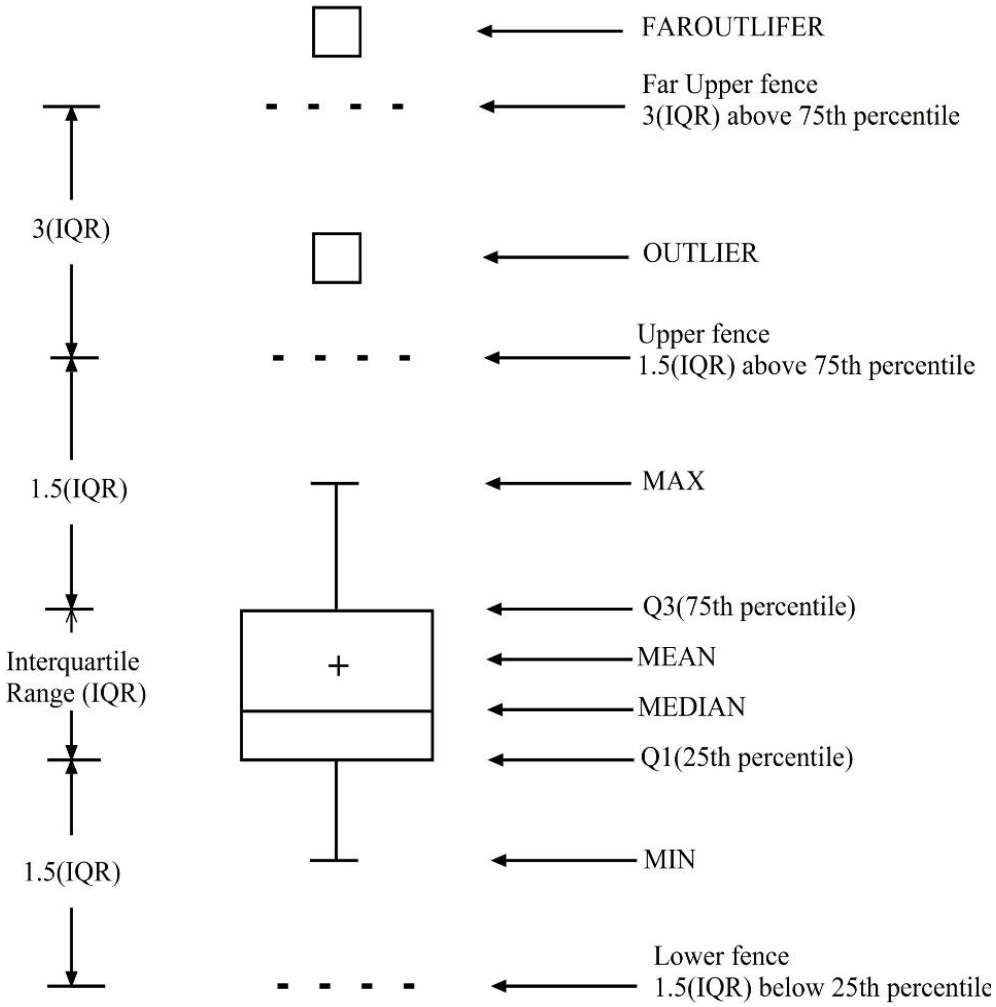


Fig. 11. Concepts related to the box plot (Benjamini, 2012)

The criteria for detection rate and false alarm rate can be calculated using the following relations:

$$Detection\ Rate = \frac{TP}{TP + FN} \quad (8)$$

$$False\ Alarm\ Rate = \frac{FP}{FP + TN} \quad (9)$$

The detection rate criterion supplies information on the relative number of correctly detected anomalous data while the false alarm rate represents the relative number of anomalous data which have been incorrectly detected as normal. The closer the detection rate to 1 and the closer the false alarm rate to 0 the more efficient the method.

RESULTS AND DISCUSSION

In this section, the results from different methods concerning the datasets under study are discussed. If these methods can function appropriately in detecting damage data they can be employed as a suitable tool for structure health monitoring in future research. To implement all the tests, MATLAB codes were produced. To validate the ability of the methods under study in detecting damage data the tests are monitored signifying that damage data in each dataset are predetermined such that the efficiency of the methods in detecting damage data is validated. Thus, a method which is capable of detecting a larger number of anomalies with

the least error possesses more efficiency as compared with the other methods. In the results section M1 represents mode 1, M2 is mode 2, M3 denotes mode 3, M4 represents mode 4, and M5 denotes mode 5. Table 1 outlines data related to the occurrence of damage in the datasets under study.

Results from Artificial Neural Networks

As a three-layered network is capable of approximating any nonlinear function, the present research makes use of a feed forward three-layered network to detect damage data. Input data for the ANN can be elements from any sets and the target data are a set having equal dimensions with the input data dimensions composed of 0 and 1 elements. A 0 value shows the normality of the data and the 1 value represents damage data. Thus, the outputs from the network can be assigned only 0 or 1 values. As it is possible for the network output for new samples not to be exactly equal to 0 or 1, the output is rounded to the nearest integer. To design the network, it should be subjected to training using parameters effective in its performance eventually selecting the most optimized parameter leading to the best response, i.e. the error value approaches zero and the correlation coefficient approaches 1. Table 2 outlines the general specifications of the neural network under study for all the datasets and Figure 12 shows the architecture of the optimized networks.

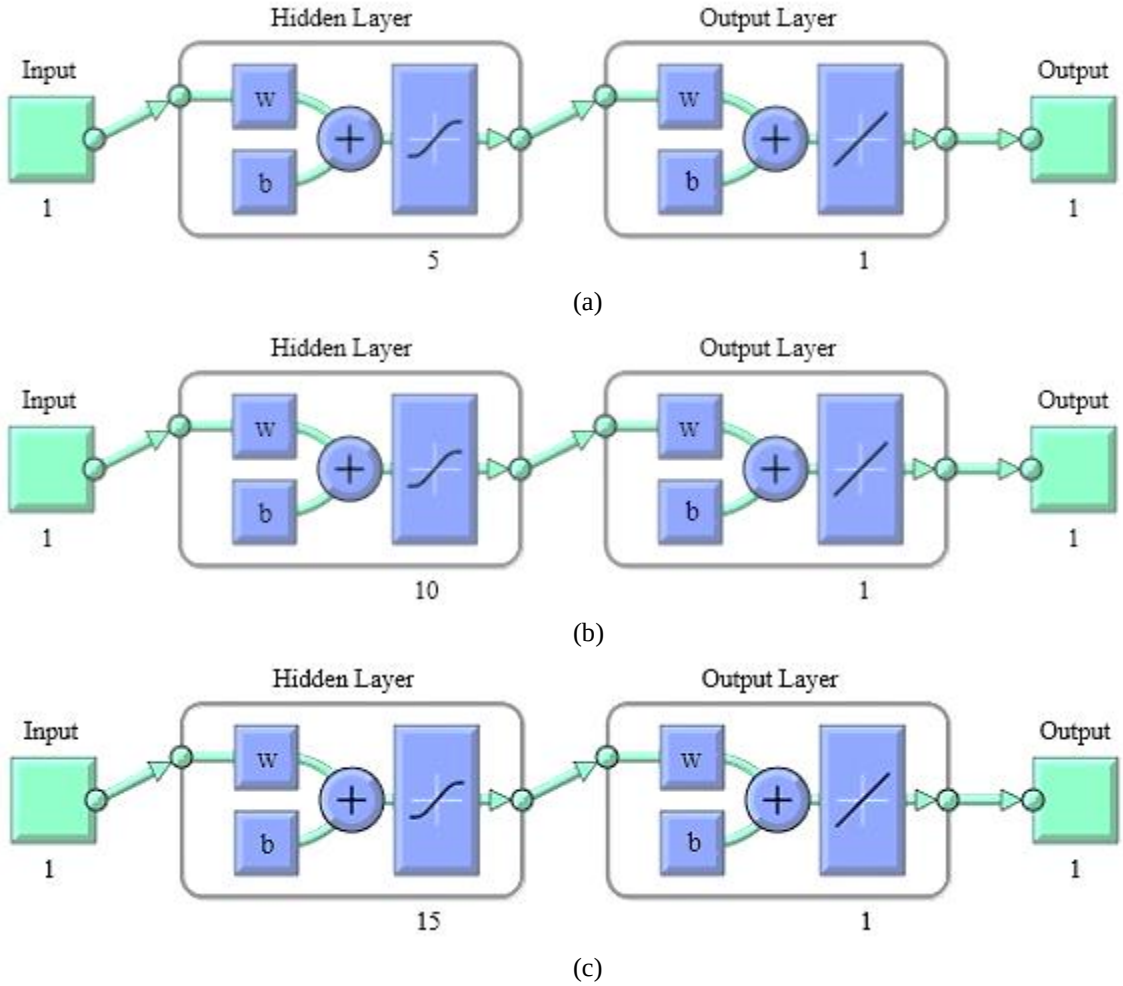


Fig. 12. Architecture of optimized networks: a) having 5 neurons in the hidden layer, b) having 10 neurons in the hidden layer, c) having 15 neurons in the hidden layer

Table 1. Damage occurrence data in Ting Kau cable-supported bridge dataset

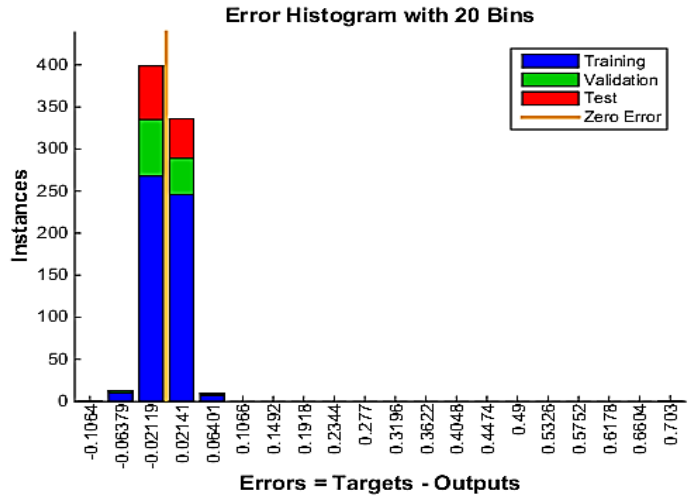
Mode number	Description of Dataset	Data for Damaged Structure	Number of Damage Data
M5	Mode 5 Frequency	753, 414, 237, 158, 13	5
M4	Mode 4 Frequency	750, 423, 251, 155, 18	5
M3	Mode 3 Frequency	743, 418, 123	3
M2	Mode 2 Frequency	481, 292, 144, 14	4
M1	Mode 1 Frequency	523, 25	2

Table 2. Architecture of the optimized network

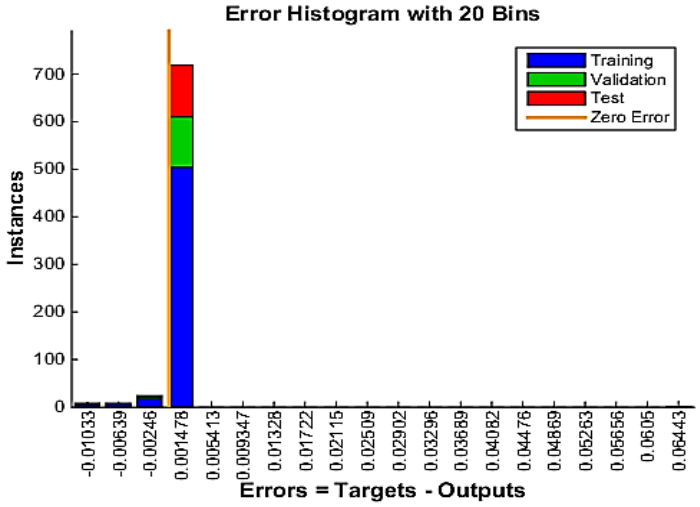
Network Types	Three-layered, Feed Forward
Number of input variables	1
Number of Neurons in input layer	1
Number of neurons in middle layers	5, 10, 15
Number of output variables	1
Number of output layer Neurons	1
Number of training data	70% of all the data
Number of validation data	15% of all the data
Number of test data	15% of all the data
Training algorithm	Levenberg-Marquardt
Activation function for the hidden layer	Hyperbolic tangent sigmoid
Activation function for the output layer	Linear
Error Measurement Function	Error measurement functions
Correlation Factor	Pearson R

Subsequent to training each model and deriving the best results, the output can be used to detect damage in the datasets. Damage detection is a process whereby new sample data are considered as model input and in case the network output is equal to 0 this demonstrates the normality of the data else it shows damage. Table 3 outlines the Root Mean Square Error and the correlation coefficient R for each dataset. The closer the RMSE approaches 0 and the coefficient R approaches 1 the more trained and efficient the model is considered to be. Also, in the present research, the number of hidden layer neurons is considered 5, 10, and 15 to better visualize the model and to show the effective role of the number of neurons in the hidden layer on the efficiency of the method. For instance, the graphs representing frequencies for five modes of the cable-supported bridge are presented. The graph representing the frequency of training data error, validation data, and test data for performance of the optimized network in Figure 13 the number of neurons in the hidden layer 10. Furthermore, the graphs for correlation coefficient for training data, validation data, test data, and all data to the optimized network in Figure 14 show the number of neurons in the hidden layer 10 for all the frequencies for cable-supported bridge.

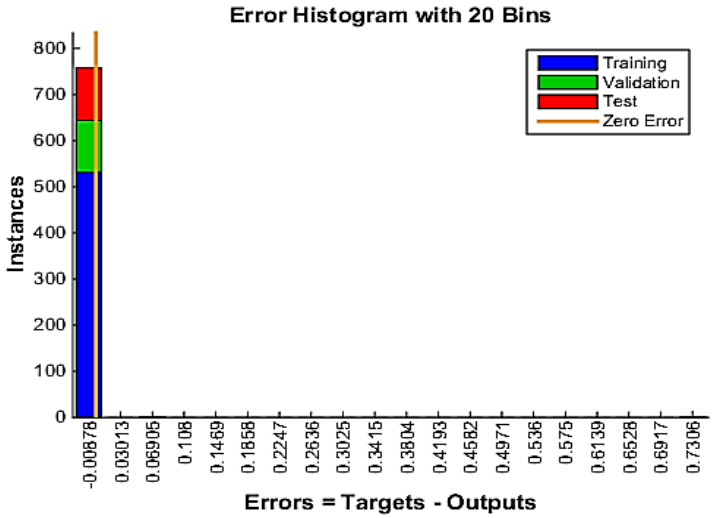
Subsequent to the establishment of the neural network model for each dataset, the result can be used to detect candidates for damage data in the structure. Data differing largely from the normal data can be considered as damage data .Table 4 outlines the results from implementation of neural network on the datasets under study. Furthermore, Figure 15 show the graphic representation of program output subsequent to detection of candidates for damage data in the dataset related to the cable-supported bridge. As can be observed in Table 4 the neural network has been able to detect damage data using ten neurons in the hidden layer. The desirable performance of the neural network in detecting damage using ten neurons in the hidden layer is due to the suitability of its training phase (Table 3). If the network cannot be trained appropriately (which is the case with number of neural networks hidden layer neurons 5 and 15) it cannot detect damage data even to the point of detecting normal data as damage data.


Error Histogram with 20 Bins
Training
Validation
Test
Zero Error
Instances
Errors = Targets - Outputs

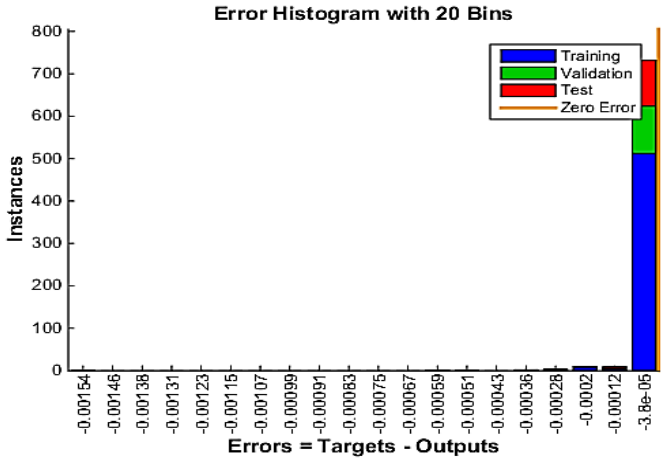
(a)


Error Histogram with 20 Bins
Training
Validation
Test
Zero Error
Instances
Errors = Targets - Outputs

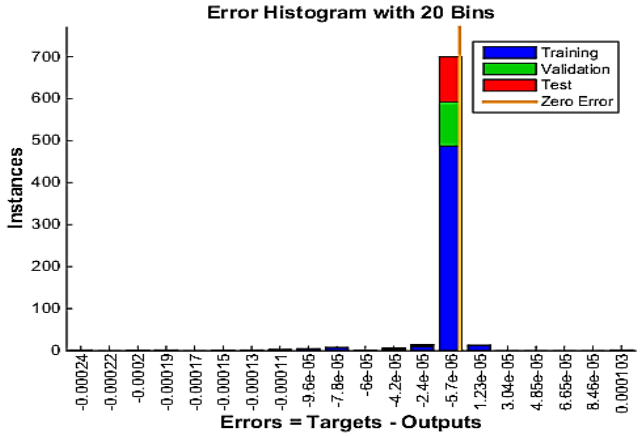
(b)


Error Histogram with 20 Bins
Training
Validation
Test
Zero Error
Instances
Errors = Targets - Outputs

(c)



(d)

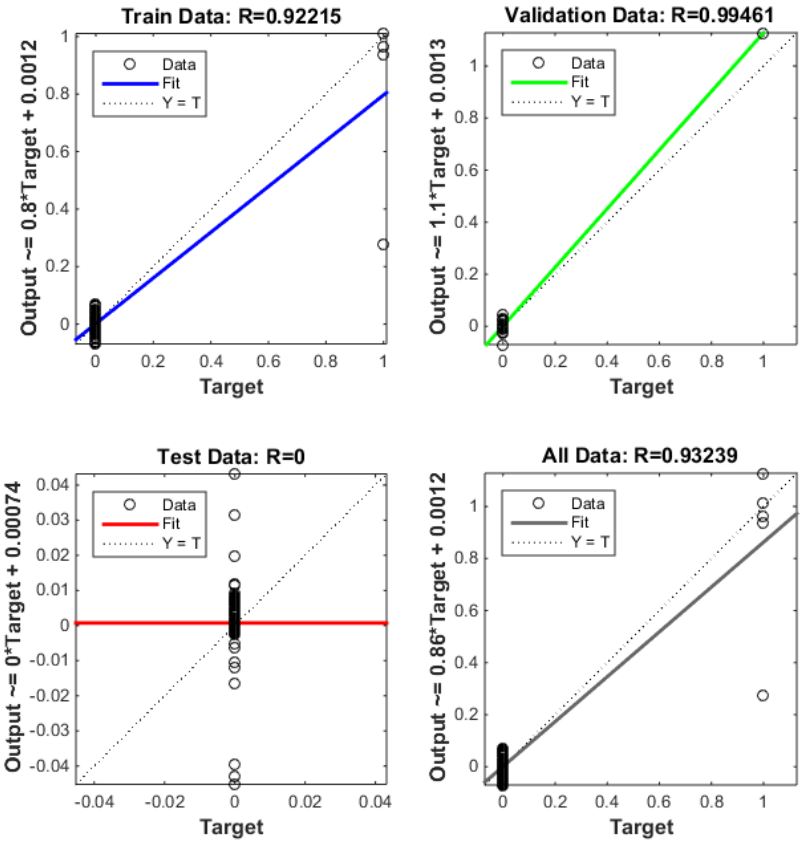


(e)

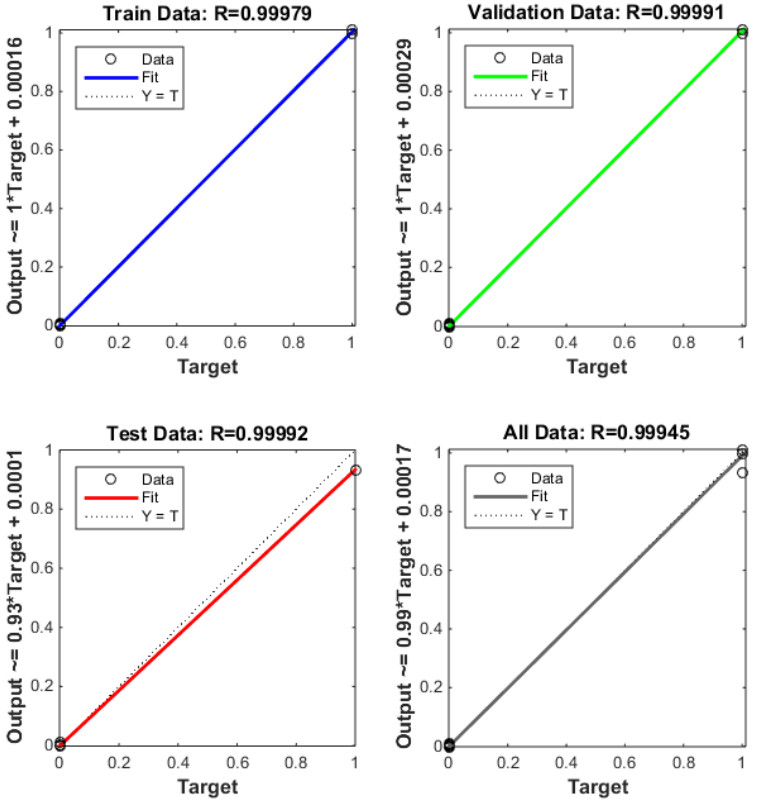
Fig. 13. Graphs for frequency of error in training data, validation data, and test data for performance of the optimized network related to the number of neurons in the hidden layer 10 for cable-supported bridge: a) M5 (Frequency for mode 5), b) M4 (Frequency for mode 4), c) M3 (Frequency for mode 3), d) M2 (Frequency for mode 2), e) M1 (Frequency for mode 1)

Table 3. Performance parameters for the optimized network for datasets

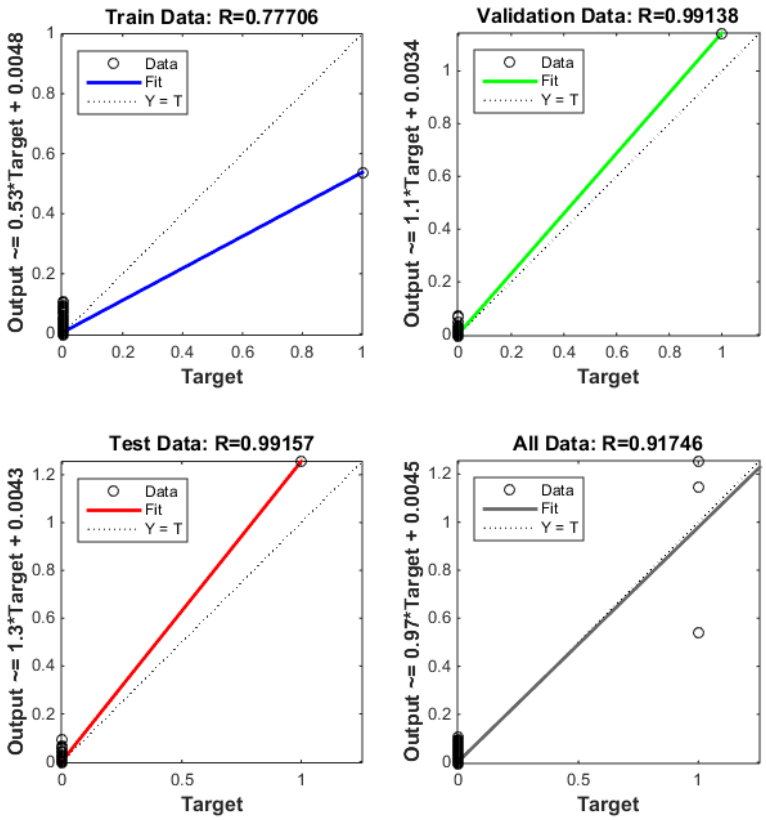
Sample Under Study	Datasets	Number of Neurons in Hidden Layer	RMSE	R
Cable-Supported Bridge	M5 Frequency	5	0.2024	0.8338
		10	0.1080	0.9323
		15	0.2012	0.6506
	M4 Frequency	5	0.3118	0.9993
		10	0.1140	0.9994
		15	0.2196	0.9674
	M3 Frequency	5	0.2570	0.9043
		10	0.1218	0.9174
		15	0.3012	0.7963
	M2 Frequency	5	0.3272	1
		10	0.1876	1
		15	0.3240	0.9992
	M1 Frequency	5	0.2642	1
		10	0.1564	1
		15	0.3714	1


Train Data: R=0.92215
Output ~= 0.8*Target + 0.0012
Validation Data: R=0.99461
Output ~= 1.1*Target + 0.0013
Test Data: R=0
Output ~= 0*Target + 0.00074
All Data: R=0.93239
Output ~= 0.86*Target + 0.0012
Data
Fit
Y = T
Target

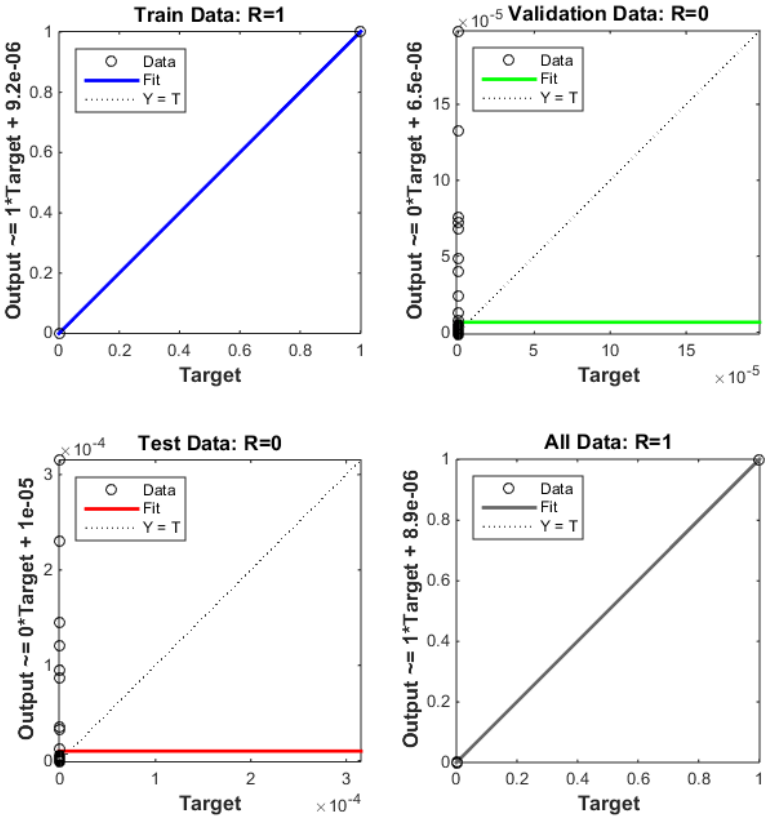
(a)


Train Data: R=0.99979
Output ~= 1*Target + 0.00016
Validation Data: R=0.99991
Output ~= 1*Target + 0.00029
Test Data: R=0.99992
Output ~= 0.93*Target + 0.0001
All Data: R=0.99945
Output ~= 0.99*Target + 0.00017
Data
Fit
Y = T
Target

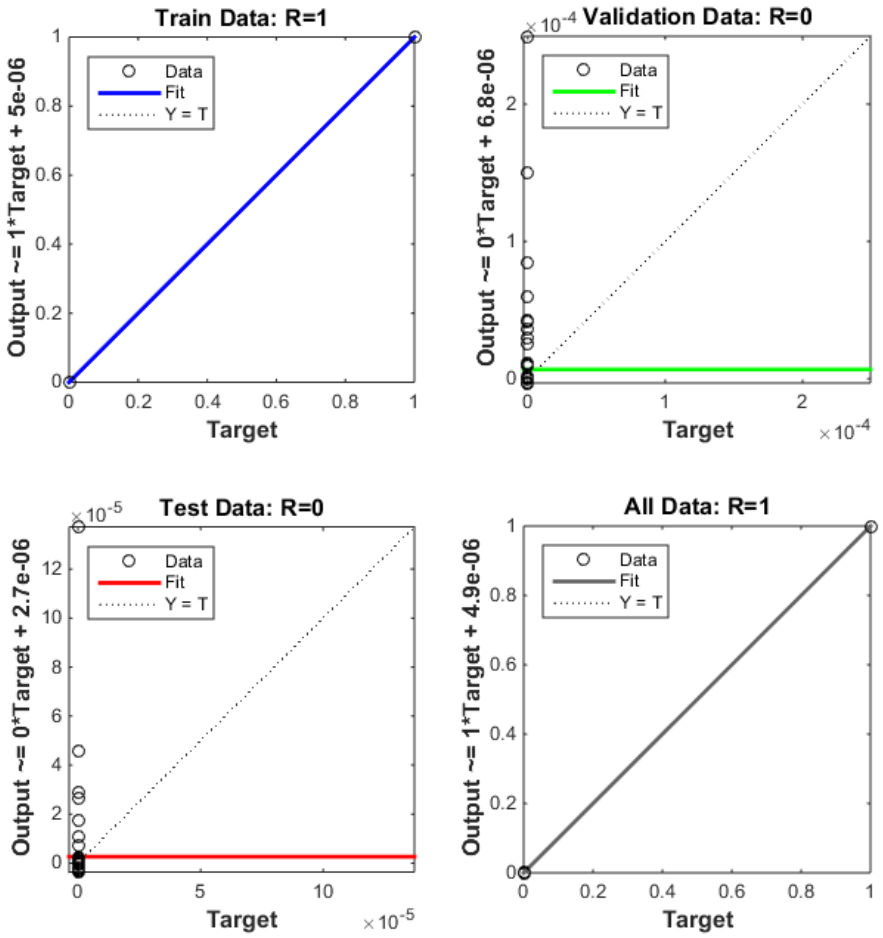
(b)


Train Data: R=0.77706
Validation Data: R=0.99138
Test Data: R=0.99157
All Data: R=0.91746
Data
Fit
Y = T
Target
Output ~= 0.53*Target + 0.0048
Output ~= 1.1*Target + 0.0034
Output ~= 1.3*Target + 0.0043
Output ~= 0.97*Target + 0.0045

(c)


Train Data: R=1
Validation Data: R=0
Test Data: R=0
All Data: R=1
Data
Fit
Y = T
Target
Output ~= 1*Target + 9.2e-06
Output ~= 0*Target + 6.5e-06
Output ~= 0*Target + 1e-05
Output ~= 1*Target + 8.9e-06

(d)

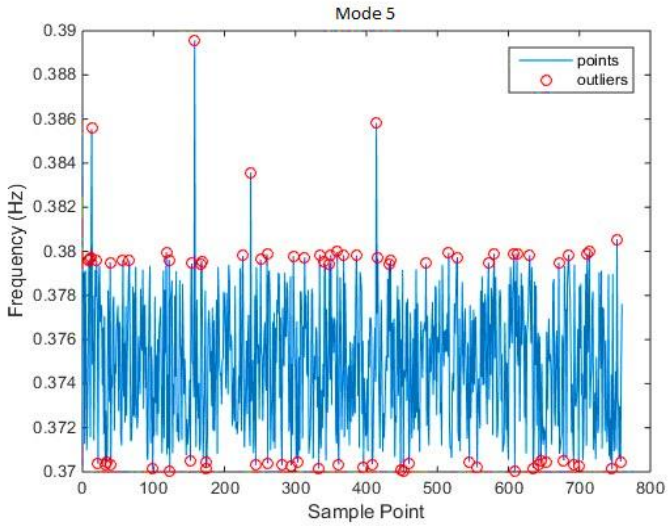


(e)

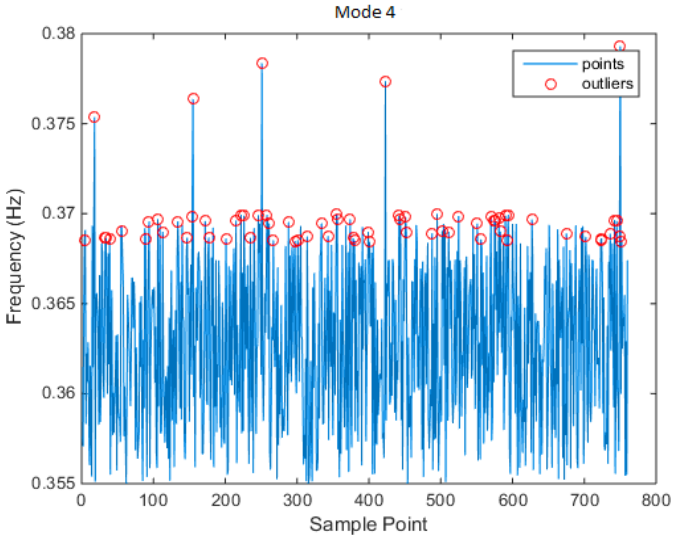
Fig. 14. Graphs for correlation coefficients R for training data, validation data, test data, and all the data related to performance of the optimized network related to the number of neurons in the hidden layer 10 for the cable-supported bridge: a) M5 (Frequency for mode 5), b) M4 (Frequency for mode 4), c) M3 (Frequency for mode 3), d) M2 (Frequency for mode 2), e) M1 (Frequency for mode 1)

Table 4. Results from implementation of ANN on datasets

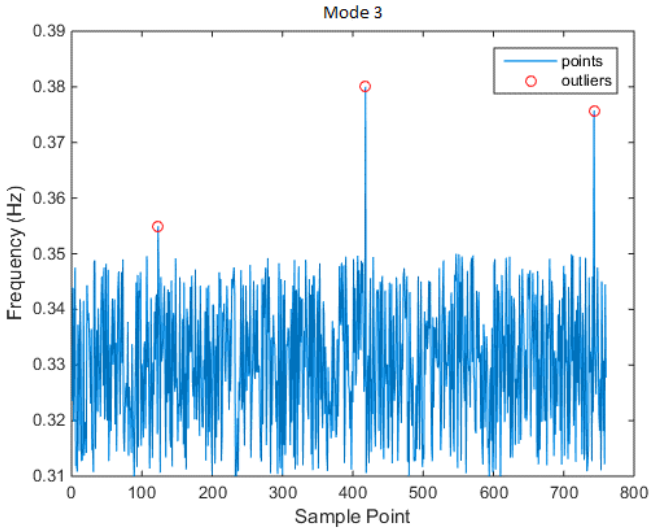
Sample Under Study	Datasets	Number of Hidden Layer Neurons	Detection Rates (%)	False Alarm Rate (%)
Cable-Supported Bridge	M5 Frequency	5	100	11.92
		10	100	9.54
		15	80	0
	M4 Frequency	5	60	0
		10	100	8.61
		15	100	19.47
	M3 Frequency	5	100	13.61
		10	100	0
		15	66.67	0
	M2 Frequency	5	100	10.76
		10	100	0.66
		15	100	5.29
	M1 Frequency	5	100	6.62
		10	100	3.83
		15	100	3.83


Mode 5
points
outliers
Frequency (Hz)
Sample Point

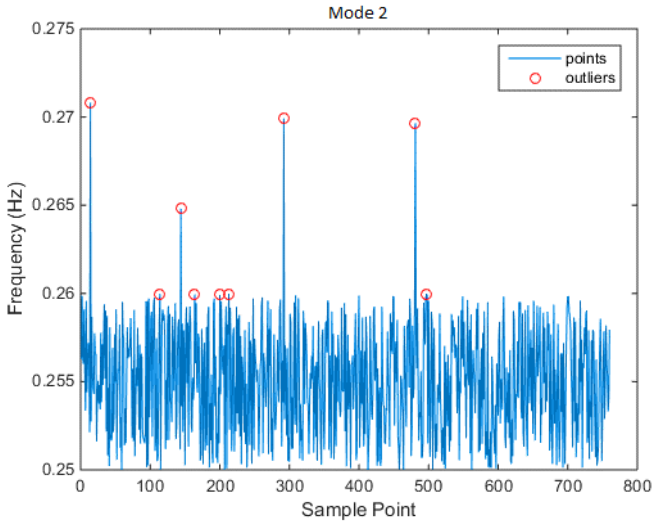
(a)


Mode 4
points
outliers
Frequency (Hz)
Sample Point

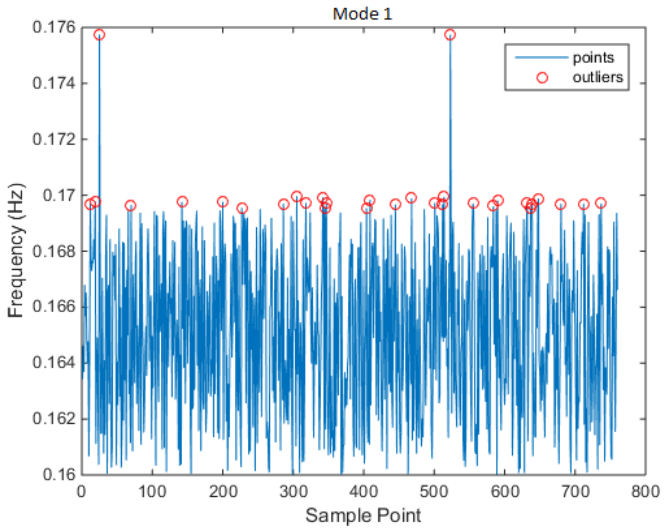
(b)


Mode 3
points
outliers
Frequency (Hz)
Sample Point

(c)



(d)



(e)

Fig. 15. Data detected as candidates for occurrence of damage for cable-supported bridge datasets using feed forward ANN with 10 hidden layer neurons: a) M5 (Frequency for mode 5), b) M4 (Frequency for mode 4), c) M3 (Frequency for mode 3), d) M2 (Frequency for mode 2), e) M1 (Frequency for mode 1)

RESULTS FROM THE CLUSTERING ALGORITHM DBSCAN

The implementation of the DBSCAN algorithm on the dataset requires the determination of the values for the two parameters Minpts (the minimum number of points existing in the neighborhood radius) and ε (the neighborhood radius). The selection of these values depends on the nature of input data which are usually obtained through trial and error. For the purposes of this study, for all the datasets the value for the Minpts parameter is considered as 10% of the total number of dataset samples ε the mean of points. Subsequent to the establishment of the cluster, the clusters whose number of members is substantially lower than those of other clusters are considered as outlier clusters or damage candidates. Table 5 outlines the results from implementation of the method on the datasets for each dataset.

Table 5. Results from implementing the DBSCAN clustering algorithm on datasets

Sample Under Study	Datasets	Detection Rate	False Alarm Rate
Cable-Supported Bridge	M5 Vibrations	-	-
	M4 Vibrations	-	-
	M3 Vibrations	-	-
	M2 Vibrations	-	-
	M1 Vibrations	-	-

As cab be observed in Table 5, the DBSCAN method is incapable of detecting damage data candidates in any datasets under study which is due to the low sensitivity of the method to outliers with the method detecting as outliers only data substantially distant from others. Thus, it can be inferred that in cases where outlier data differ slightly from normal data, the method cannot be relied on as an efficient and useful one.

Results from Manhattan Distance

The implementation of Manhattan Distance Algorithm on datasets for the purpose of detecting damage requires the determination of threshold values. In fact, if the distance calculated for a specific datum exceeds the threshold, the data is detected as damage data. In this approach, the value for this parameter is selected using trial and error. In other words, the results from implementation of the method are selected on the basis of various threshold values and the best value. Table 6 outlines the results from implementation of the method on datasets for the value of threshold parameter 1.5 times each dataset. Also, Figure 16 depicts the graphic sample of the program output after detecting candidates for damage data in datasets related to the cable-supported bridge. Candidates for damage data are encircled.

Results for Sum-of-Sines Curve Fitting

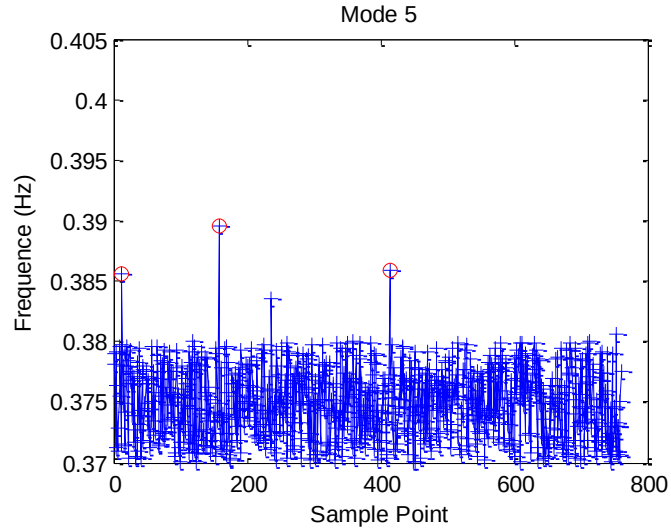
The implementation of Sum-of-Sines curve fitting on datasets requires the determination of the threshold parameter value and the number of series terms. In the present research, the value for the threshold parameter is considered as being equal to 3 for all datasets. In fact, if for a given data the value obtained from Eq. (6) exceeds 3 it can be considered as a candidate for damage. Also, in the present research the number of series terms is equal to 3, 4, 5, and 6 so as to show the effect of the number of sinusoidal terms on correct curve fitting of the data. Using the Least Squares method and the Trust-Region Algorithm the coefficients for a_i, b_i and c_i are calculated. Table 7, for instance, shows the coefficients for a_i, b_i and c_i related to series 4 for five cable-supported bridge modes. Table 8 depicts the results from implementing the method on datasets for each dataset. Sum-of-Sines Curve Fitting show Figure 17 the implementation of the sum of sines with the number of series term ranging 3 on datasets of the cable-supported bridge.

Figure 18 show damage data for the cable-supported bridge.

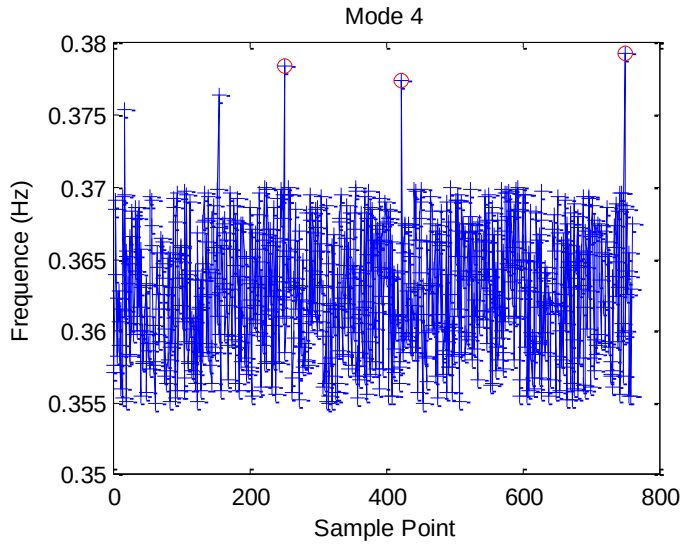
Considering the results outlined in Table 9 it can be observed that the best curve fitting is related to the n = 5 series terms.

Table 6. Results from implementing Manhattan Distance on datasets

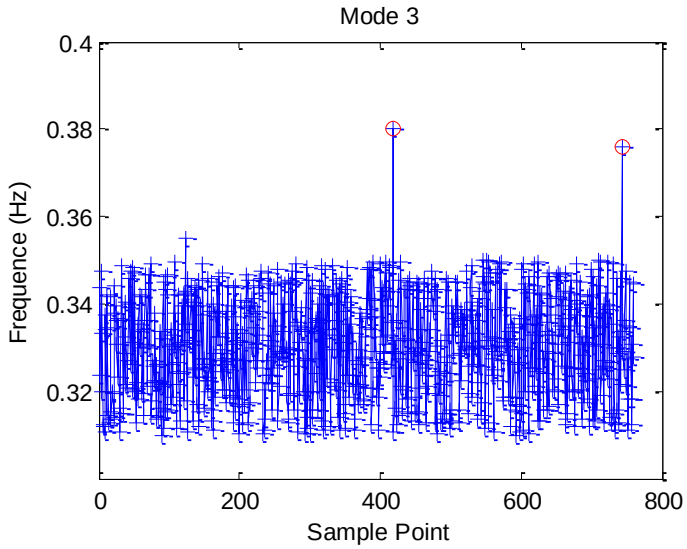
Sample Under Study	Datasets	Detection Rates (%)	False Alarm Rates (%)
Cable-Supported Bridge	M5 Frequency	60	0.00
	M4 Frequency	60	0.00
	M3 Frequency	66.67	0.00
	M2 Frequency	100	0.00
	M1 Frequency	100	0.00


Mode 5
Frequence (Hz)
0.405
0.4
0.395
0.39
0.385
0.38
0.375
0.37
0
200
400
600
800
Sample Point

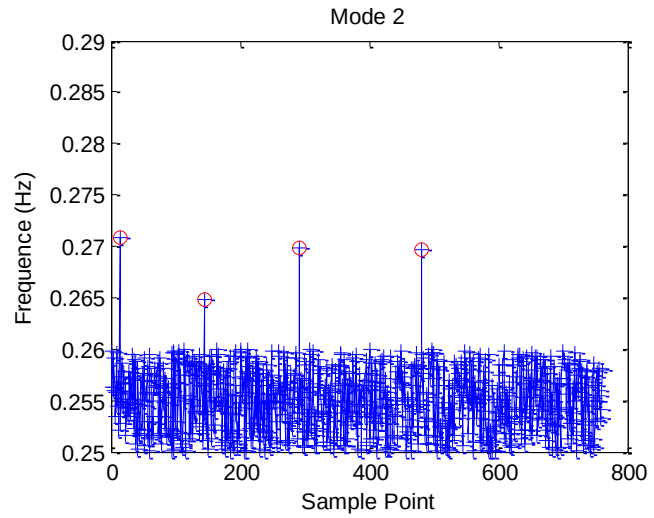
(a)


Mode 4
Frequence (Hz)
0.38
0.375
0.37
0.365
0.36
0.355
0.35
0
200
400
600
800
Sample Point

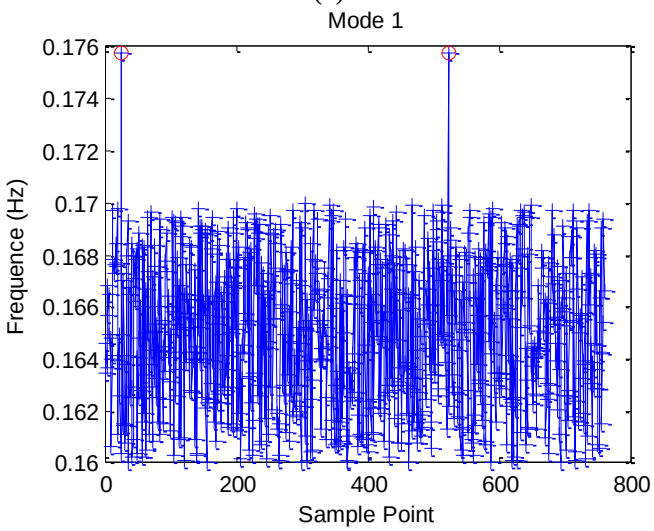
(b)


Mode 3
Frequence (Hz)
0.4
0.38
0.36
0.34
0.32
0
200
400
600
800
Sample Point

(c)



(d)



(e)

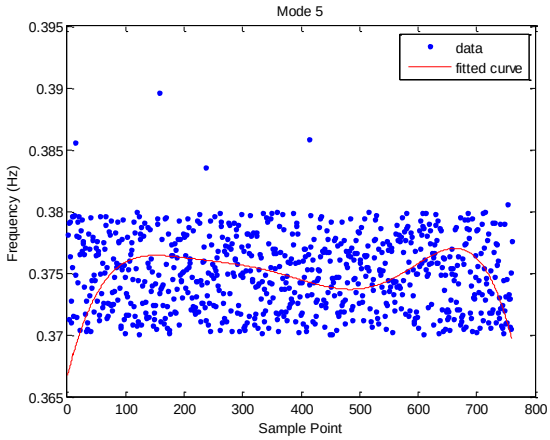
Fig. 16. Data detected as candidates for occurrence of damage in cable-supported bridge data using Manhattan Distance related to: a) M5 (Frequency for mode 5), b) M4 (Frequency for mode 4), c) M3 (Frequency for mode 3), d) M2 (Frequency for mode 2), e) M1 (Frequency for mode 1)

Table 7. Coefficients obtained for a_i, b_i and c_i in implementing Sum-of-Sines Curve Fitting with 4 (n = 4) series terms for five cable-supported bridge modes

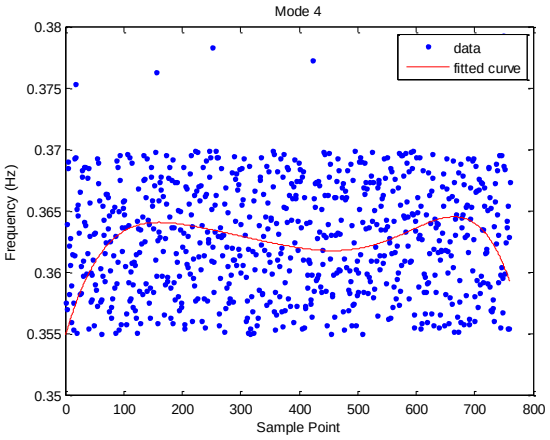
Samples Under Study	Coefficients	Frequency Modes				
		M_5	M_4	M_3	M_2	M_1
Cable-Supported Bridge	a_1	0.4113	0.6014	0.4971	0.3263	0.2045
	a_2	0.04924	0.3126	0.2156	0.07234	0.04045
	a_3	0.02554	0.1638	0.09155	0.002361	0.0009836
	a_4	0.01248	0.08935	0.04182	0.001462	0.0004495
	b_1	0.001725	0.003638	0.003272	0.001612	0.001847
	b_2	0.007666	0.007194	0.007023	0.003689	0.004566
	b_3	0.01334	0.0123	0.01187	0.01459	0.01476
	b_4	0.01508	0.01354	0.01306	0.0175	0.02263
	c_1	0.9411	0.2044	0.3335	1.056	0.873
	c_2	1.922	2.019	2.067	3.532	2.988
	c_3	3.031	3.31	3.455	3.241	2.34
	c_4	5.589	6.037	6.227	5.565	2.661

Table 8. Results from implementing Sum-of-Sines curve fitting on the datasets

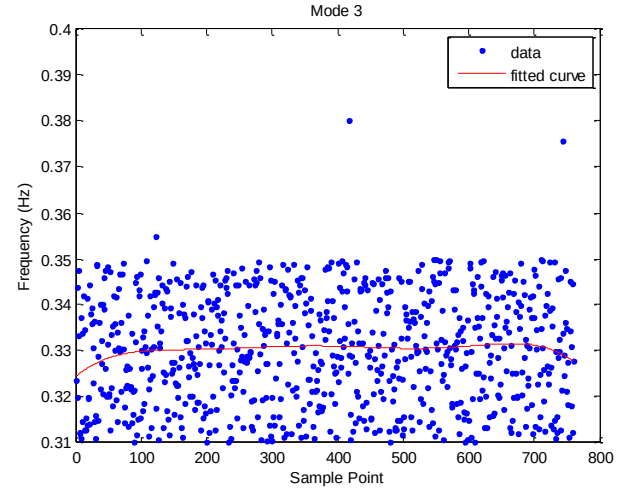
Sample Under Study	Datasets	Number of Series Terms	Detection Rates (%)	False Alarm Rates (%)
Cable-Supported Bridge	M5 Frequency	3	60	0.795%
		4	80	0.00
		5	80	0.00
		6	60	0.397
	M4 Frequency	3	80	0.00
		4	60	0.00
		5	100	0.00
		6	100	0.265
	M3 Frequency	3	66.67	0.00
		4	66.67	0.00
		5	66.67	0.00
		6	66.67	0.00
	M2 Frequency	3	75	0.132
		4	100	0.00
		5	75	0.00
		6	75	0.132
	M1 Frequency	3	100	0.00
		4	100	0.00
		5	100	0.00
		6	100	0.00



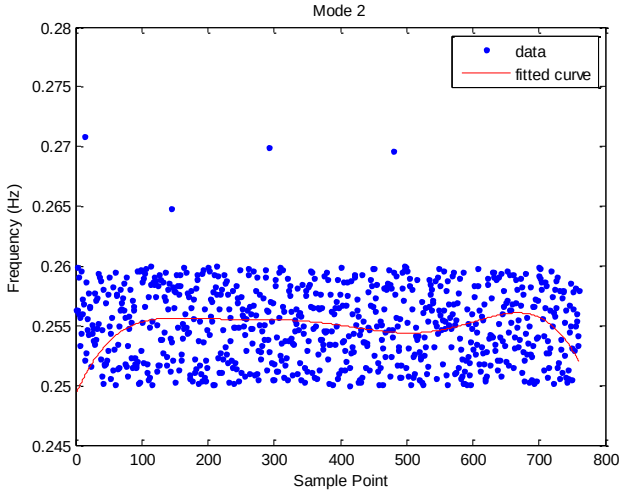
(a)



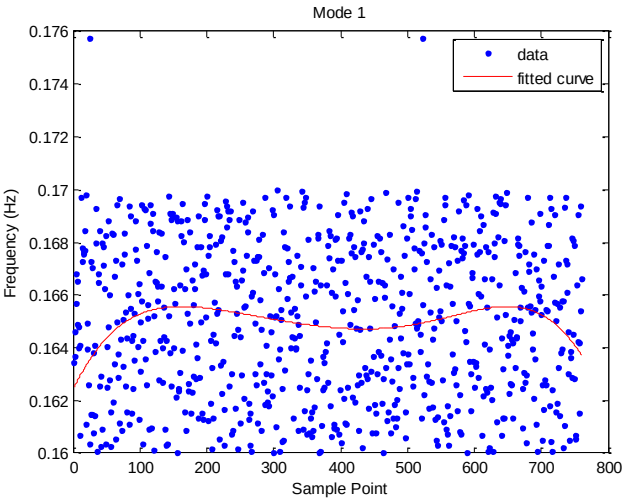
(b)



(c)

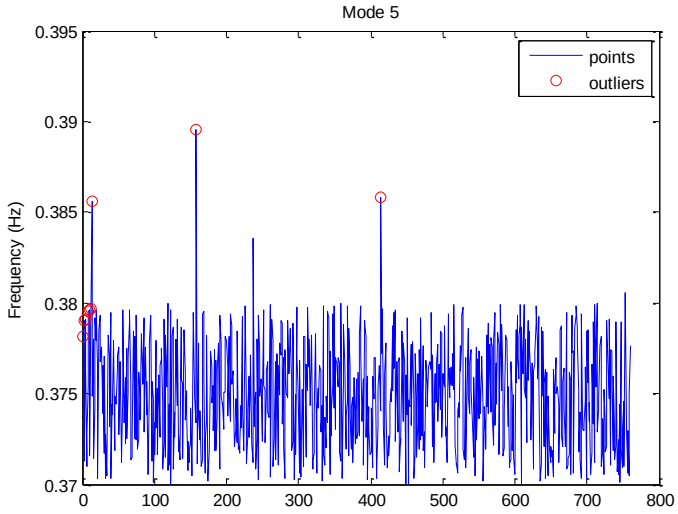


(d)

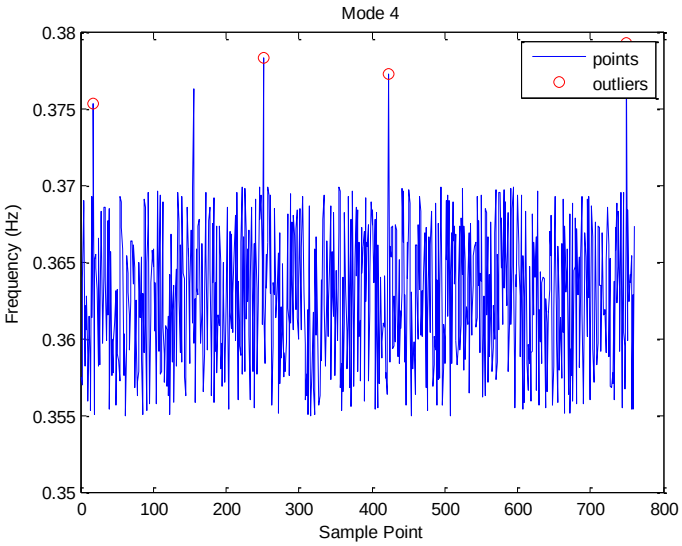


(e)

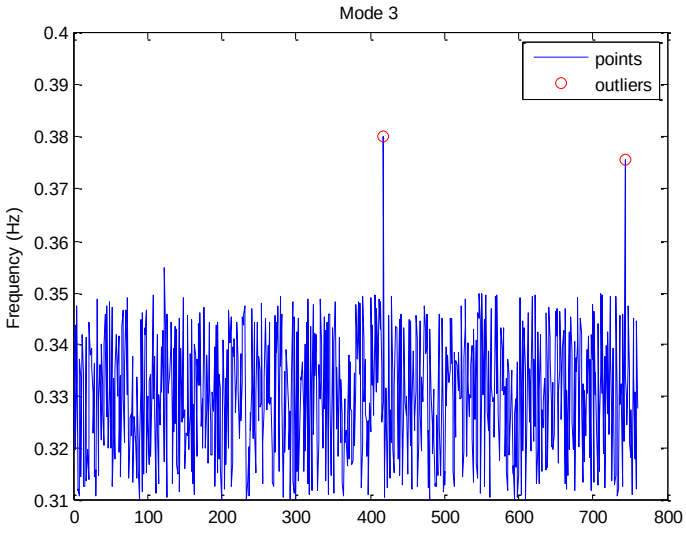
Fig. 17. Curve fitting for datasets related to cable-supported bridge having n = 3 series terms: a) M5 (Frequency for mode 5), b) M4 (Frequency for mode 4), c) M3 (Frequency for mode 3), d) M2 (Frequency for mode 2), e) M1 (Frequency for mode 1)


Mode 5
points
outliers
Frequency (Hz)
0.395
0.39
0.385
0.38
0.375
0.37
0
100
200
300
400
500
600
700
800

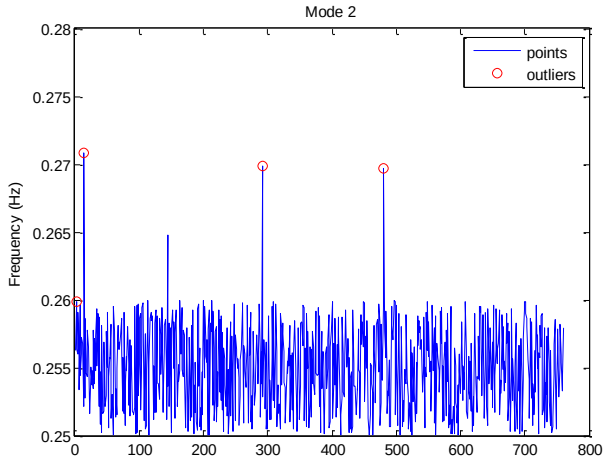
(a)


Mode 4
points
outliers
Frequency (Hz)
0.38
0.375
0.37
0.365
0.36
0.355
0.35
0
100
200
300
400
500
600
700
800
Sample Point

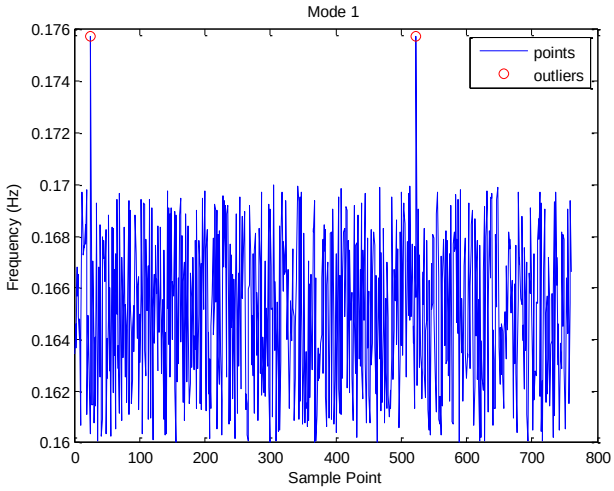
(b)


Mode 3
points
outliers
Frequency (Hz)
0.4
0.39
0.38
0.37
0.36
0.35
0.34
0.33
0.32
0.31
0
100
200
300
400
500
600
700
800

(c)



(d)



(e)

Fig. 18. Data detected as candidates for occurrence of damage in the cable-supported bridge dataset using Sum-of-Sines Curve Fitting having n = 3 series terms related to: a) M5 (Frequency for mode 5), b) M4 (Frequency for mode 4), c) M3 (Frequency for mode 3), d) M2 (Frequency for mode 2), e) M1 (Frequency for mode 1)

Table 9. Mean detection rates and false alarm rates for data with varying numbers of series terms

Sample Under Study	Datasets	Number of Series Terms	Detection Rates (%)	False Alarm Rates (%)
Cable-Supported Bridge	Frequencies for M_1 through M_5 frequencies	3	76.334	0.185
		4	81.334	0.00
		5	84.334	0.00
		6	80.334	0.159

Results from the Box Plot

Table 10 outlines the results emanating from the box plot method on the datasets while Figure 19 shows their respective box plots. In each plot, the central line in the box represents the median, the edges show the 25^{th} and the 75^{th} quartiles, and the end edges represent the limit for normal data. Data residing outside these limits represent the outliers. These data are denoted in the figure by the + sign. For instance, Figure 20 shows the detection of damage data candidates detected for 5 modes of the cable-supported bridge are encircled.

Table 10. Results from implementation of the box plot method on the datasets

Sample Under Study	Datasets	Detection Rates (%)	False Alarm Rates (%)
Cable-Supported Bridge	M5 Frequency	60	0.00
	M4 Frequency	40	0.00
	M3 Frequency	66.67	0.00
	M2 Frequency	75	0.00
	M1 Frequency	100	0.00

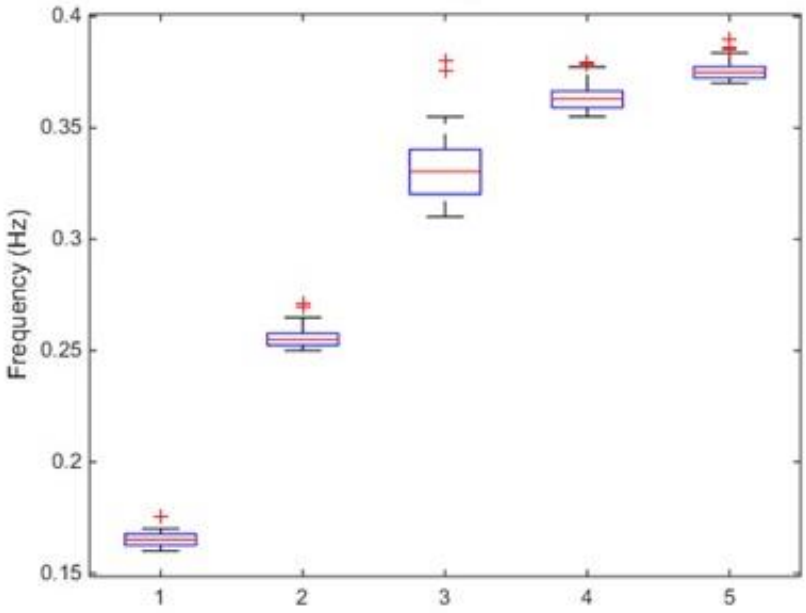
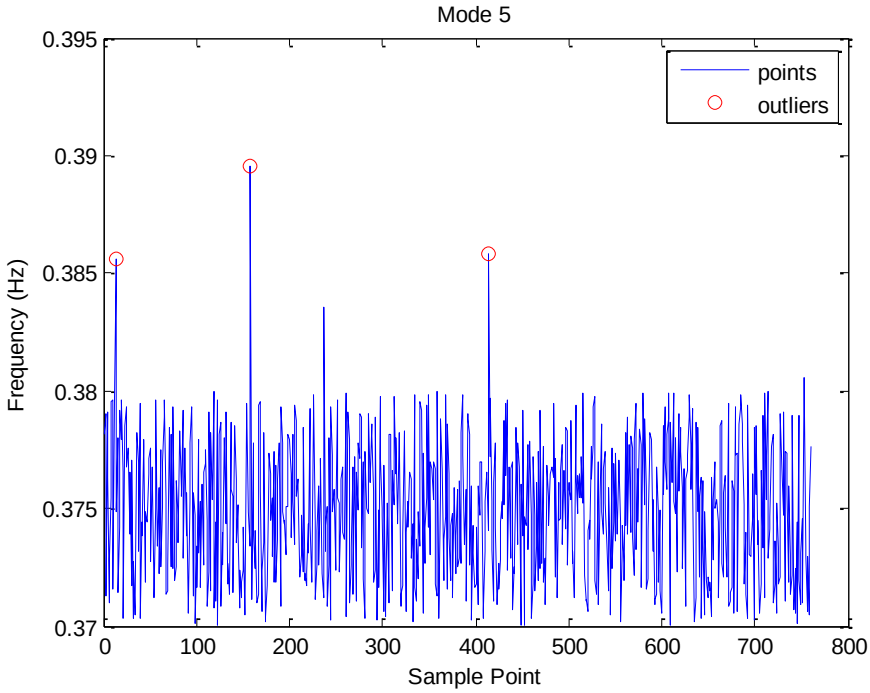
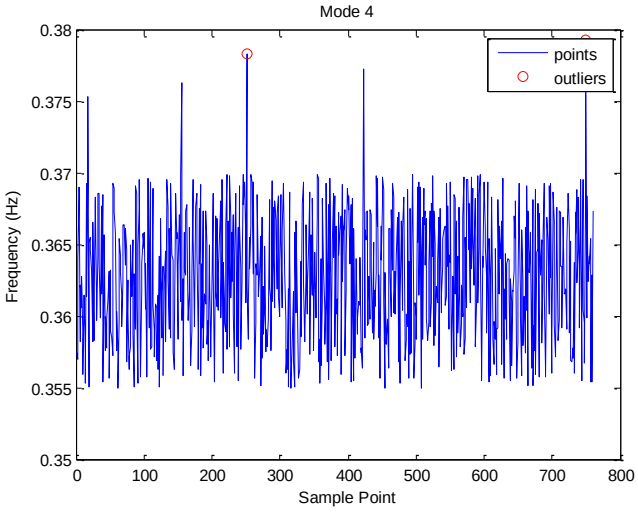


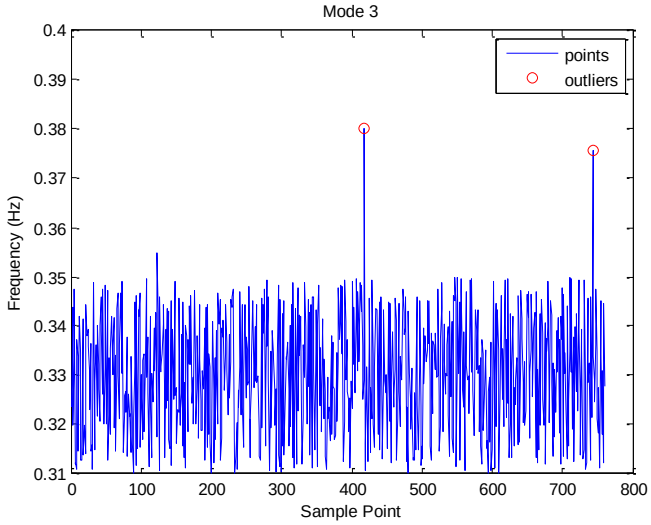
Fig. 19. Box plot of the five modes of the cable-supported bridge



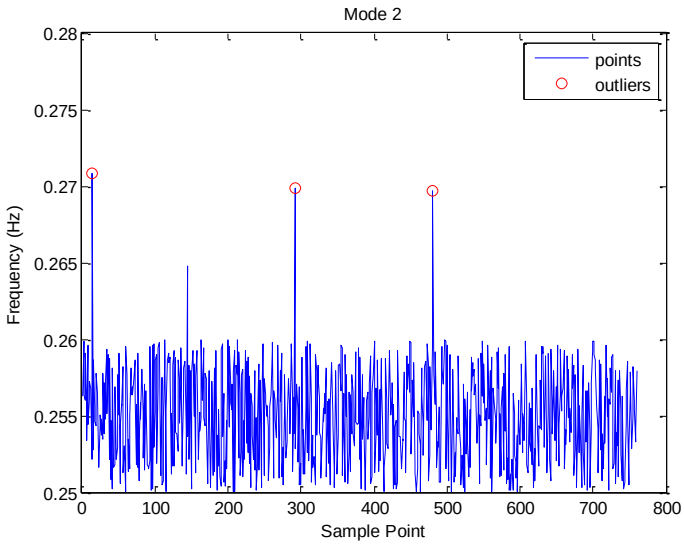
(a)


Mode 4
points
outliers
Frequency (Hz)
Sample Point
0.38
0.375
0.37
0.365
0.36
0.355
0.35
0
100
200
300
400
500
600
700
800

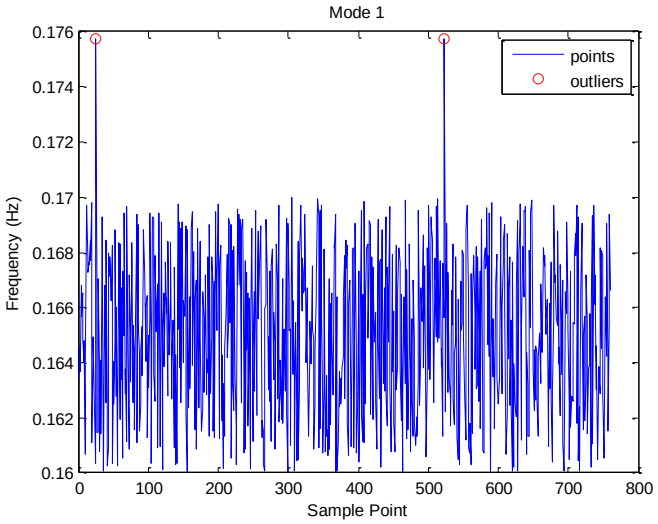
(b)


Mode 3
points
outliers
Frequency (Hz)
Sample Point
0.4
0.39
0.38
0.37
0.36
0.35
0.34
0.33
0.32
0.31
0
100
200
300
400
500
600
700
800

(c)


Mode 2
points
outliers
Frequency (Hz)
Sample Point
0.28
0.275
0.27
0.265
0.26
0.255
0.25
0
100
200
300
400
500
600
700
800

(d)



(e)

Fig. 20. Data detected as candidates for occurrence of damage in datasets for the cable-supported bridge using the box plot related to: a) M5 (Frequency for mode 5), b) M4 (Frequency for mode 4), c) M3 (Frequency for mode 3), d) M2 (Frequency for mode 2), e) M1 (Frequency for mode 1)

Comparison of Performance in the Methods

In this section, the performance and efficiency of methods in the datasets are compared. The ultimate performance of each method is calculated using the following relation: Performance of the Method=Mean of Detection Rate - Mean of False Alarm Rate

As can be seen from the results outlined in Table 11, the highest performance is related to the ANN with the least performance obtained for the DBSCAN method. The reason for the desirable performance of the ANN in detecting damage can be attributed to the suitability of the training phase. If the network is not trained appropriately it may not be able to detect damage correctly. Considering the low sensitivity of the DBSCAN to outlier data it detects as outlier data which are substantially distant from others in the dataset. This method was not capable of appropriately modelling damage data and normal data. Thus, it can be inferred that the performance of the method is lower than the other methods.

CONCLUSIONS

The present research studied the performance of anomaly detection methods such as Feed Forward ANN, DBSCAN clustering algorithm, Manhattan Distance, Sum-of-Sines curve fitting, and the Box Plot in detecting cable-supported bridge structural damage.

Table 11. Comparison of performance of the methods under study in the datasets

Methods	Mean of Detection Rate	Mean of False Alarm Rate	Performance/Efficiency
Artificial Neural Networks	100	4.528	95.472
Sum-of-Sines Curve Fitting	84.334	0.000	84.334
Manhattan Distance	77.334	0.000	77.334
Box Plot	68.334	0.00	68.334
DBSCAN Clustering Algorithm	---	---	---

Considering the tests performed, the performance of all methods except for the DBSCAN has been suitable. From among the methods under study, Artificial Neural Networks proved to be the method with the highest performance detecting damage data with the highest accuracy. The condition for desirable performance for the methods in detecting damage, is the optimized selection of the input parameters. The correct selection of parameters depends on the type of input data.

The feed forward artificial neural network utilizes a monitored learning method to detect damage. Thus, to employ the method there should be training data. In cases where there are no training data or their preparation costs excessively this method cannot be utilized. One of the other disadvantages of this method is that its performance may be weak in the course of confronting new samples whose pattern may be different from patterns of training phase patterns. As a result, under these conditions, it may not be possible to detect structural damage. Nevertheless, this method can optimally detect damage data whose patterns exist in the training stages. Therefore, this method can be used as an efficient method.

The DBSCAN clustering method, Manhattan Distance, Sum-of-Sines Curve Fitting and the Box Plot method utilize unsupervised learning to detect damage. Unsupervised methods do not require training data. Therefore, in cases where the preparation of training data is difficult and impossible, the implementation of these methods is cost-effective. Another advantage of these methods is that these methods are able to identify serious damage whose patterns are not previously recorded in the system. One of the disadvantages of these methods is that their correct performance depends on the input parameters of the algorithm and in case the values for these parameters are not correctly selected their performance is decreased.

Considered the afore-mentioned points, the most optimal damage detection method should be selected on the basis of data obtained for conditions of the structure, facilities available, types of damages sustained, the degree of importance and safety of the structure.

REFERENCES

Alguliyev, R.M., Aliguliyev, R.M., Imamverdiyev, Y.N. and Sukhostat, L.V. (2017). "An anomaly detection based on optimization", *International Journal of Intelligent Systems and Applications*, 9(12), 87-96.

Alguliyev, R., Aliguliyev, R. and Sukhostat, L. (2017). "Anomaly detection in Big data based on clustering", *Statistics, Optimization and Information Computing*, 5(4), 325-340.

Beliakov, G., Kelarev, A. and Yearwood, J. (2011). "Robust Artificial Neural Networks and Outlier Detection", *Journal of Mathematical Programming and Operations Research*, 61(12), 1467-1490 , Deakin University, Australia.

Benjamini, Y. (1988). "Opening the Box of a Boxplot", *The American Statistician*, 42(4), 257-262.

Bai, M., Wang, X., Xin, J. and Wang, G. (2016). "An efficient algorithm for distributed density-based outlier detection on big data", *Neurocomputing*, 181(C), 139-146.

Bai, L., Liang, J. and Dang, C. (2011). "An initialization method to simultaneously find initial cluster centers and the number of clusters for clustering categorical data", *Knowledge-Based Systems*, 24(6), 785-795.

Ester, M., Kriegel, H-P., Sander, J. and Xu, X. (1996). "A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise", *KDD'96 Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, Institute for Computers Science, University of Munich, Germany, 226-231.

Frigge, M., Hoaglin, D.C. and Iglewicz, B. (1989). "Some Implementations of the Boxplot", *The American Statistician*, 43(1), 50-54.

Gaffney, J. and Ulvila, J. (2001). "Evaluation of intrusion detectors: A decision theory approach", *In Proceedings of IEEE Symposium on Security and Privacy*, Oakland, CA, USA, 50-61.

Gagolewski, M., Bartoszuk, M. and Cena A. (2016). "Genie: A new, fast, and outlier-resistant hierarchical clustering algorithm", *Information Sciences*, 363, 8-23.

Hand, D., Mannila, H. and Smyth, P. (2001). "*rinciples of data mining*, The MIT Press.

Huang, J., Zhu, Q., Yang, L. and Feng, J. (2016). "A non-parameter outlier detection algorithm based on Natural Neighbor", *Knowledge-Based Systems*, 92, 71-77.

Johnson, R. and Wichern, D. (1992)."*Applied multivariate statistical analysis*, Prentice Hall.

Jiang, F., Liu, G., Du, J. and Sui, Y. (2016). "Initialization of *K*-modes clustering using outlier detection techniques", *Information Sciences*, 332, 167-183.

Karim, A.N.M., Nordin, A.N. and Begum, S. (2014), "Technical and Economic Feasibility of Sensor Technology for Health/Environmental Condition Monitoring", *Comprehensive Materials Processing*, 13, 499-514.

Latecki, L. J., Lazarevic, A. and Pokrajac, D. (2007). "Outlier Detection with Kernel Density Functions", *5th International Conference on Machine Learning and Data Mining in Pattern Recognition (MLDM)*, Leipzig, Germany, pp. 61-75.

Loureiro, A., Torgo, L. and Soares, C. (2004). "Outlier detection using clustering methods: A data cleaning application", *In proceedings of the data mining for business workshop*, University of Porto, Porto, Portugal.

Massart, D.L., Smeyers-Verbeke, A., Capron, X. and Schlesier, K. (2005). "Practical data handling visual presentation of data by means of box plots", *Journal of Vrije Universiteit Brussel*, 18(4), 215-218.

Montgomery, D.C., Peck, E.A. and Vining, G.G. (2012). *Introduction to Linear Regression Analysis*, 3rd Edition, John Wiley & Sons, New York, USA.

Motulsky, H. and Brown, R. (2006). "Detecting outliers when fitting data with nonlinear regression: A new method based on robust nonlinear regression and the false discovery rate", *BMC Bioinformatics*, 7(123), 1471-2105.

Ni, Y.Q. (2014). "Structural health monitoring of cable-supported bridges based on vibration measurements", *Proceedings of the 9th International Conference on Structural Dynamics*, EURODYN 2014, Porto, Portugal, pp. 65-72.

Sinwar, D. and Kaushik, R. (2014). "Study of Euclidean and Manhattan Distance Metrics using simple K-means clustering", *International Journal for Research in Applied Science and Engineering Technology*, 2, 270-274.

Tang, B. and He, H. (2017), "A local density-based approach for outlier detection", *Neurocomputing*, 241, 171-180.

Zhuang, W., Zhang, Y. and Grassle, J.F. (2004). "Identifying erroneous data using outlier detection techniques", *Proceedings Ocean Biodiversity Informatics*, International Conference on Marine Biodiversity Data Management, Hamburg, Germany, 37, 187-192.

Zhu, S. and Xu, L. (2018)."Many-objective fuzzy centroids clustering algorithm for categorical data", *Expert Systems with Applications*, 96, 230-248.