# [EN-A-005] Interactive prediction of airport noise monitoring data based on time-series similarity

(EIWAC 2017)

+Liu Chen-hui*, Chen Hai-yan*,***, Xie Hua**, Yuan Li-gang**

* College of Computer Science and Technology
Nanjing University of Aeronautics and Astronautics
Nanjing, China
gslzlch@126.com
chenhaiyan@nuaa.edu.cn
** College of Civil Aviation
Nanjing University of Aeronautics and Astronautics
Nanjing, China
marcuse@163.com
kelfen_yuan@126.com
*** Collaborative Innovation Center of Novel Software Technology and Industrialization
Nanjing, China

**Abstract:** Noise monitoring systems have been set up around some airports to capture the noise data, which can give a strong support to certain departments to make an effective strategy to control the impacts of the noise on surrounding environment. However, when the failure of monitoring nodes occurs, noise data of some areas cannot be collected in time. In this paper, an interactive prediction model based on the time-series similarity of monitoring nodes is proposed to repair the lost noise data. Firstly, a measurement of the time-series similarity based on trend segmentations (TSR_DIST) is introduced to calculate the similarity between the failure node and other nodes. And then, the nodes with high similarity are selected to form the feature set which is used to train the prediction model. Finally, the interactive prediction model based on feature-weighted support vector regression machine (FWSVR) is trained on the historical noise data of the Capital International Airport. The experiments results show that the proposed model can repair the lost data with high accuracy and has a good generalization performance.

**Keywords:** Airport noise monitoring, Interactive prediction, Time-series similarity measure, Feature weighted support vector machine

## 1. INTRODUCTION

The construction of large-scale airport has made the airport noise problem increasingly serious. In order to reduce the influence of the noise to the surrounding environment, it is necessary to monitor and collect the noise data around the airport. Currently, with the help of the Internet of Things technology, the noise monitoring nodes have been set up to collect the noise data in the affecting range around the airport. Researches and experiments are conducted on the noise data to support the decision of the noise control. Due to the impacts of bad weather (such as typhoon, blizzard, etc.), the service life and other factors, the monitoring nodes are easy to be broken and aging, resulting in the noise data collection failure . Subject to the complexity of the geographical distribution and equipments, hardware maintenance is usually difficult to complete in time[1]. Therefore, how to predict the noise value by software when the nodes cannot work is worth to be researched[2].

The analysis on the historical data of the noise time-series shows that there is a correlation between the monitoring nodes, for example, the noise sequence of adjacent monitoring nodes have similar value and trend. This suggests that, we can build an interactive prediction model to estimate the noise value of the failure monitored area by the correlation of the monitoring nodes.

To get the prediction model, the following two problems need to be addressed: the first one is the measurement of the similarity between the nodes; the second one is the establishment of the prediction model. For the first problem, this paper presents a new time-series trend segmentation representation (IEP-TSR), based on which, we design a time-series similarity measurement (TSR_DIST) to calculate the similarity between the monitoring nodes according to the historical noise data. For the second problem, we choose the feature-weighted support vector regression machine to train the prediction model, in which, the weight of each feature is calculated according to the similarity

This paper is organized as follows, the second section describes the state of the art of the related technologies; the third section introduces the selection of the similar monitoring nodes based on TSR_DIST; the fourth section introduces the interactive prediction of the airport noise monitoring nodes by the feature weighted support vector regression; the fifth and sixth sections summarize the experimental results and give the conclusion of the paper.

## 2. RELATED WORK

### 2.1 Interactive prediction

Interactive prediction is a paradigm for system state prediction using the relationship between the components of complex dynamic systems. The principle of this paradigm was first proposed by Takahara in 1965 to solve the dynamic optimization problem of complex systems[3]. It was then widely used in the field of the automatic control to predict and control the production process of large enterprises[4]. At present, the idea of interactive prediction has been further used in more areas. For the issue of intrusion detection, a single isolated alarm cannot reflect the real security situation of the network. After researching the relationship between the alarms caused by the intrusion, Morin and Benjamin[5] designed an interactive rules data model M2D2 to correlate the IDS alarms and proposed an alarm interactive predicting technology based on this data model. M2D2 improves the processing speed and quality of alarms, and has the alarm capability for the attack with the complex steps. For the prediction of buildings deformation, Tasci, Levent, and Erkan Kose[7] summarized the shortcomings of the previous research which only focused on the time-series of single monitoring node, and denoted that each deformation monitoring node is not isolated but correlated. Based on this, they proposed a gray correlational concept and a gray interactive prediction model which can take into account the correlational information between different monitoring nodes. The gray interactive prediction model has a good effect in the short-term prediction of the building deformation. In order to solve the monitoring time-series interruptions and data errors caused by the damage or leak of the monitoring nodes in the initial building stage, Ke Shanliang et.al.[8] proposed a repairing method which predicts the damaged data according to the correlation of similar monitoring nodes time-series. The prediction accuracy of this method is higher on the data of buildings deformation which has lower randomness, but it is not suitable for the airport noise data with higher randomness. Thus, in this paper, we researched on the interactive prediction method which is competent for airport noise prediction.

### 2.2 Time-series similarity measurement

The similarity measurement between the monitoring nodes is the measurement of time-series similarity which is a high dimension space distance between objects. The classical methods of time-series similarity measurement are generally divided into two categories: Lock-step Measures and Elastic Measures[9]. Lock-step Measures is an "one-to-one" measurement of the time-series and it is a universally preferred similarity measurement[10]. The most typical distance it uses is Euclidean distance, which is simple, efficient, easy to be understood and can be applied to different types of data.. However, Euclidean distance has a significant limitation that it requires the two measured series have equal length [11]. Different from Lock-step Measures, Elastic Measures is the "one-to-many" or Its common instances are Dynamic Time Warping (DTW) and Edit Distance[12][13][14]. DTW allows time-series to be matched in equal length after self-replication, which can overcome the limitation of Euclidean distance[15]. However, DTW is very sensitive to noise. The spatial complexity and time complexity of the calculation will be very high if the matching process is not optimized[16]. Edit Distance is the minimum number of editing operations required to convert between two strings, including character substitution, insertion and deletion. It has the ability to reduce the influence from noise, but cannot effectively deal with the amplitude excursion and other deformations of time-series.

Except for the above time-series similarity measurements, researchers also give some new interpretations of time-series similarity in terms of symbolization and trend, and put forward the corresponding similarity measuring method. Lin et al.[16] proposed a symbolic representation of time-series, Symbolic Aggregate Approximation (SAX), which normalizes and discretizes the time-series, reduces the dimension by PPA and converts time-series from numbers into strings. Then the similarity between the two series is the SAX distance. The traditional SAX is simple and efficient, but it does not take into account the trend (or direction) of a time-series, which causes that different time-series segments with same average may be mapped to same symbols. Besides, a Similarity Measure of Variation-Trends (SMVT) is proposed in [18] to intuitively and effectively measure the similarity of the trend of time-series. It reduces the dimension of time-series by segmented aggregate approximation, and symbolizes the time-series, then measures the trend of symbolized series.

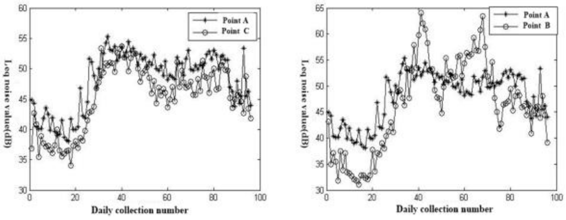
In summary, DTW is suitable for the features of airport noise monitoring data like real-time, highly dynamic and diverse monitoring frequency except the mass data. To address it, we combine SAX with DTW to propose a new time-series similarity measurement based on the trend segmentation representation. We also take into account the feature of the explicit trend of airport noise time-series and ensure that the new measurement can accurately measure the similarity between monitoring nodes of the airport noise.

## 3. TIME-SERIES SIMILARITY MEASURE BASED ON TREND SEGMENTATION

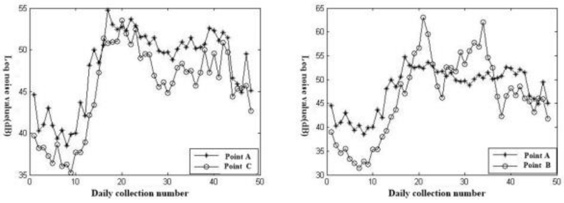
### 3.1 Time-series correlation between monitoring nodes

To establish the final interaction model and improve its prediction accuracy, we need to select the monitoring nodes with high similarity to the failure node from all the monitoring nodes, and remove the nodes with low similarity. Through the analysis of a large number of historical data, we found different similarities between the noise time-series from different monitoring nodes. Figure 1 shows the time-series variation similarity between the three monitoring nodes(A, B, C) at three time intervals, where the graphs on the left show the high similarity of noise time-series between node A and

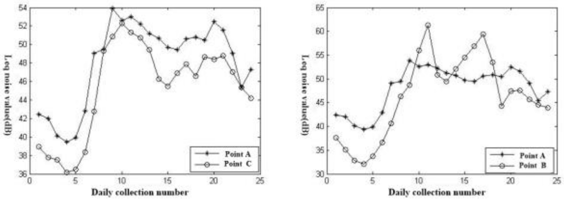
C, and the graphs on the right show the low similarity of noise time-series between node A and B.




(a) 15-minute interval of the similarity of the timing diagram




(b) 30-minute interval of the similarity of the timing diagram




(c) 60-minute interval of the similarity of the timing diagram

Figure 1 Schematic diagram of the similarity between time points of monitoring nodes

The subgraphs (a), (b), and (c) are the similarity comparison with three different intervals. It shows that the time-series trends and values of the node A and C are both similar. If node A fails to work, node C is more reliable to be the correlated monitoring node rather than node B.

### 3.2 IEP-TSR

The existing airport noise data are massive, the calculated amount will be innumerable if directly measuring the similarity between them. Thus, similar to dealing with the general time-series, we first need to re-represent the original time-series to reduce the dimension. As mentioned in Section 3.1, trend is an important feature of a airport noise time-series, which can visually reflect the dynamic changes of the noise. If the node failure occurs, the trend of its data will be significantly different from the normal one. Therefore, we need to focus on the trend measurement to find the changing status. In this paper we present an adaptive time-series trend segmentation representation method based on the Iterative End Point algorithm (IEP-TSR) and its basic steps are:

1) Normalize the original time-series $S =< p_1, p_2,..., p_n >$ to a standard sequence $\bar{S} =< \bar{p}_1, \bar{p}_2,..., \bar{p}_n >$ range from [0, 1];

2) Based on IEPA[19], $\bar{S}$ is divided into different segments with different trends and its dimension are reduced at the same time.

IEPA is a well-known approximation method to judge whether two curves are similar by calculating the maximum distance (i.e., Euclidean distance) between the original curve and the approximate curve. The algorithm is:

•*Input*:
Time series data with length of $n$: $S[p1,...,pn]$;
Distance threshold: $\varepsilon$; Max distance: $dmax$;
Distance between line segment and farthest point:
$dist^{pk}_{l[pi,pj]}$, $where\ i, j, k = 1,...n, and\ i < k < j$;
the farthest point: $p_{farthest}$;
Point-line distance metric algorithm: *prependicularDistance*;
$max = 0$;
*ResultList[]* is the segmented result.
•*Initialization process*:
1) Compute the distance between line segment $l[pi, pj]$ $(i, j = 1,..., n, and\ i < j)$ and each point $pk$ within $it$:
$dist = \text{prependicularDistance}(l[pi, pj], pk)$, $where\ k \in (1, n)$;
2) Find the farthest point $p_{farthest}$ to $l[pi, pj]$, $dmax = dist^{p_{farthest}}_{l[pi,pj]}$;
3) If $dmax > \varepsilon$
4) keep $p_{farthest}$
recursive on segment $S[pi, p_{farthest}]$ and $S[p_{farthest}, pj]$:
$recResults1[] = IEPA(S[pi, p_{farthest}], \varepsilon)$;
$recResults2[] = IEPA(S[p_{farthest}, pj], \varepsilon)$;
5) else
$ResultList[] = \{S[p1], S[pn]\}$;
6) End if
7) Get the final segmentation result:
$ResultList[] =$
$\{recResults1[1...length(recResults1)-1,$
$recResults2[1...length(recResults2)]\}$;

3) Symbolize the trend segments:

IEP-TSR uses a triplet to represent each trend segment of the dimension-reduced time-series, including the segment start point, trend variable, and segmentation mean. The trend variables $S_{si}$ and mean values $\bar{\mu}_i$ for each segment are calculated as follows:

$$\begin{cases} S_{si} = \dfrac{S_{p_{i+1}} - S_{p_i}}{p_{i+1} - p_i} \\ \bar{\mu}_i = \dfrac{\sum_{j=p_i}^{p_{i+1}-1} S_j}{p_{i+1} - p_i} \end{cases} \quad (1)$$

where $p_i$ is the starting point of the $i$ th trend segment. And the mean value is discretized to obtain the symbolic representation.

Finally, the original time-series $S =< p_1, p_2,..., p_n >$ is re-represented as a series of triples $S =< \tilde{S}_1, \tilde{S}_2,..., \tilde{S}_n >$, where $\tilde{S}_i =< p_i, S_{si}, \bar{\mu}_i >$. Figure 2 shows a time-series after segmenting, the gray dash line is the original time-series, and the red polyline is the segment after IEP-TSR process.

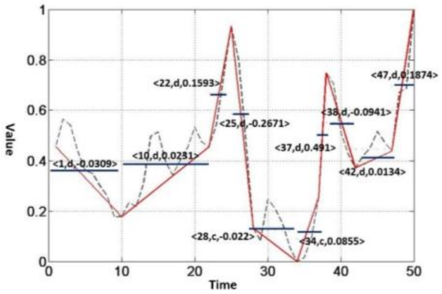



Figure 2 Time-series representation based on adaptive time-series trend segmentation representation

### 3.3 TSR-DIST

In order to measure the similarity between the monitoring nodes after the trend segmentation representation, we propose a time-series similarity measurement TSR_DIST, which is defined as follows.

**Definition 1 (TSR_DIST):** Given time-series $X$ and $Y$ are represented by IEP-TSR, and the similarity measurement of $X$ and $Y$ is defined as:

$$TSR-DIST(X,Y)=\begin{cases}0 & ,\ i=j=0\\ \infty & ,\ i=0\ or\ j=0\\ \sqrt{\left(mdist(\hat{x}_1,\hat{y}_1)\right)^2+\left(sdist(x_1,y_1)\right)^2} & \\ +\min\begin{cases}TSR-DIST(\mathrm{Re}\,st(X),\mathrm{Re}\,st(Y))\\ TSR-DIST(\mathrm{Re}\,st(X),Y)\\ TSR-DIST(X,\mathrm{Re}\,st(Y))\end{cases} & ,otherwise\end{cases}$$

(2)

where $\hat{x}_1$ and $\hat{y}_1$ represent the mean of the first trend segments of $X$ and $Y$ respectively; $x_1$ is the mean of the first trend segment of $X$ . $\mathrm{Re}\,st(X)$ represents the remaining series after removing the first variable from series $X$ . $sdist(x_1,y_1)$ and $mdist(\hat{x}_1,\hat{y}_1)$ are the distance between the trend variable and the trend mean Their formulas are:

$$sdist(x_1,y_1)=\sqrt{(S_{x_1}-S_{y_1})^2}$$

(3)

$$mdist(\hat{x}_i,\hat{y}_i)=\begin{cases}0 & if\ \left|\hat{x}_i-\hat{y}_i\right|\le 1\\ \sqrt{\left(\beta_{\max(\hat{x}_i,\hat{y}_i)\text{-}1}-\beta_{\min(\hat{x}_i,\hat{y}_i)}\right)^2} & otherwise\end{cases}$$

(4)

In equation (2), TSR_DIST measure both the trend mean and trend features between series, which can improve the measuring precision Besides, combining with DTW enables it to flexibly measure the distance between time-series with different lengths.

### 3.4 TSR_DIST performance verification

We compared TSR_DIST with Euclidean distance, DTW distance and SAX distance, to verify its superiority. Table 1 lists the classification results for 1-NN, where the lowest classification root mean square error (RMSE) are highlighted.

In Table 1, the RMSE of TSR_DIST in major datasets is the lowest (6/11), followed by SAX (2/11) and DTW (2/11), which shows that TSR_DIST produces smaller RMSE and achieves a higher reduction rate than Euclidean distance, DTW distance and SAX distance. Therefore, we decide to use the TSR_DIST in the interactive predictions of support vector regression.

## 4. INTERACTIVE PREDICTION OF AIRPORT NOISE MONITORING NODES BASED ON FEATURE WEIGHTED SUPPORT VECTOR REGRESSION

The training steps of the interactive prediction model of airport noise monitoring nodes based on feature-weighted support vector regression are:

**Step 1: Select the correlated monitoring nodes**

After obtaining the historical noise data from the failed node and all other normal nodes, we first use the TSR_DIST to calculate the similarity between normal nodes and the failed node.

Assume that, the failed node $Y$ has a data series $Y=(y_1,y_2,...,y_m)$ in one day, where $m$ is the observations moments in that day. Then, the noise data in the past $n$ consecutive days can be expressed as a matrix:

Table 1 1-NN classification error rates of EU (Euclidean distance); 1-NN best classification error rates, w lengths, and dimensionality reduction ratios of the SAX and DTW on 11 data sets. The lowest RMSE are highlighted in bold.

| Dataset No. | Data set Name | EU RMSE | SAX RMSE | SAX win-size | SAX ratio | DTW RMSE | TSR_DIST RMSE | TSR_DIST $\varepsilon$ | TSR_DIST ratio |
|---|---|---|---|---|---|---|---|---|---|
| 1 | synthetic_control | 0.12 | 0.25 | 4 | 0.25 | **0.003** | 0.045 | 0.87 | **0.5** |
| 2 | Gun_Point | 0.13 | 0.18 | 15 | 0.1 | **0.08** | 0.213 | 0.094 | **0.3** |
| 3 | Beef | 0.33 | 0.6 | 47 | 0.1 | 0.4 | **0.327** | 0.44 | 0.05 |
| 4 | BeetleFly | 0.112 | 0.133 | 8 | 0.125 | 0.132 | **0.087** | 0.27 | **0.959** |
| 5 | BirdChicken | 0.45 | 0.4 | 16 | 0.03 | 0.3 | **0.15** | 0.63 | **0.03** |
| 6 | Car | 0.267 | **0.265** | 10 | 0.018 | 0.43 | 0.4167 | 0.71 | **0.029** |
| 7 | Coffee | 0.107 | **0.071** | 10 | 0.036 | 0.214 | 0.143 | 0.409 | 0.3 |
| 8 | Yoga | **0.173** | 0.195 | 128 | 0.3 | 0.202 | 0.235 | 0.047 | 0.25 |
| 9 | Worms | 0.674 | 0.641 | 10 | 0.1 | 0.58 | **0.481** | 0.52 | 0.04 |
| 10 | ECG200 | 0.165 | 0.23 | 12 | 0.125 | 0.17 | **0.16** | 0.435 | **0.33** |
| 11 | Meat | 0.21 | 0.33 | 56 | 0.02 | 0.23 | **0.2** | 0.067 | 0.125 |

$$Data_Y = \begin{bmatrix} Y^1 \\ Y^2 \\ \dots \\ Y^n \end{bmatrix} = \begin{bmatrix} y_1^1 & y_2^1 & \dots & y_m^1 \\ y_1^2 & y_2^2 & \dots & y_m^2 \\ \dots & \dots & \dots & \dots \\ y_1^n & y_2^n & \dots & y_m^n \end{bmatrix} \quad (5)$$

where each row represents a time-series of noise data for a day, and each column represents the noise data at the same moments of different days. For normal node $X$, its time-series noise data for the past $n$ days can also be built as:

$$Data_X = \begin{bmatrix} X^1 \\ X^2 \\ \dots \\ X^n \end{bmatrix} = \begin{bmatrix} x_1^1 & x_2^1 & \dots & x_m^1 \\ x_1^2 & x_2^2 & \dots & x_m^2 \\ \dots & \dots & \dots & \dots \\ x_1^n & x_2^n & \dots & x_m^n \end{bmatrix} \quad (6)$$

Then, for nodes $X$ and $Y$, their TSR_DIST distance of consecutive $n$ days can be calculated by:

$$dist_{tsr-dist}(X,Y) = \frac{\sum_{i=1}^{n} D_{tsr-dist}(X^i, Y^i)}{n} \quad (7)$$

After calculating the distance between the failure node $Y$ and all normal nodes $C_X = \{X_1, X_2, \dots, X_r\}$, the nodes meet the conditions $dist_{tsr-dist}(X,Y) < dist_{threshold}$ are selected to be the correlated nodes, where $dist_{threshold}$ is the distance threshold.

After selection, we get the correlated node set $C_{X'} = \{X_1, X_2, \dots, X_{r'} | r' \le r\}$. The weight of each node in $C_{X'}$ is indicated by its distance to node $Y$, which is calculated by the following formula:

$$w_{X_i} = \frac{1 / dist_{tsr-dist}(X_i, Y)}{\sum_{j=1}^{r'} 1 / dist_{tsr-dist}(X_j, Y)} \quad , \quad i \le r' \quad (8)$$

According to $C_{X'} = \{X_1, X_2, \dots, X_{r'} | r' \le r\}$, we can prepare the training data set $D = \{(\overrightarrow{x_k}, y_k) \mid k = 1 \dots n \times m\}$ from the historical data, where $\overrightarrow{x_k} = (x_k^1, x_k^1, \dots, x_k^{r'})$ is the noise data of the $r'$ correlated nodes.

**Step 2: Construct a feature- weighted support vector regression model**

After modifying the weight set $W = \{w_{X_1}, w_{X_2}, \dots, w_{X_{r'}}\}$ to the feature-weighted diagonal matrix $M$ as:

$$M_X = \begin{bmatrix} w_{X_1} & & & \\ & w_{X_2} & & \\ & & \dots & \\ & & & w_{X_{r'}} \end{bmatrix} \quad (9)$$

It is multiplied to the RBF(radial basis function) kernel of the FWSVR model, as follows:

:

$$\begin{aligned} K_{M_X}(x_i, x) &= \exp\left(-\frac{\left\| x^T M_X - x_i^T M_X \right\|^2}{2\sigma^2}\right) \\ &= \exp\left(-((x - x_i)^T M_X M_X{}^T (x - x_i)) / 2\sigma^2\right) \end{aligned} \quad (10)$$

In order to improve the ability of the model, we use a parameter optimization algorithm to find the best $(\varepsilon, C)$ when the prediction accuracy, fitting precision, and generalization ability are balanced. In this model, Genetic Algorithm (GA) is used and the feasible intervals of the two parameters are arranged in a two-dimensional grid using Grid Search. The performances (measured by MSE and determination coefficient) of models with different $(\varepsilon, C)$ are verified by the 10-fold cross validation it guarantee that the prediction accuracy of the model is high while avoiding over-fitting. When highest accuracy is obtained, the corresponding best $(\varepsilon, C)$ is found, the training of the model is completed.

## 5. EXPERIMENTS AND RESULTS ANALYSIS

### 5.1 Preparation of data sets

In the experiments, the training dataset is the time-series noise data of the 16 monitoring nodes in the Capital International Airport in June 2016, and the testing data set is the data in the first 7 days of July 2016. The dataset is divided into three parts, 15-minutes interval (2880 training samples, 672 test samples), 30-minutes interval (1440 training samples, 336 test samples) and 1-hour interval (720 training samples, 168 test samples) and the measurement of the noise level is LEQ. We will experiment with the three parts of time-series data set respectively.

### 5.2 Time-series similarity measure method comparison experiment

*5.2.1 The selection of correlated time-series monitoring nodes*

On the training dataset, we measured the similarity of time-series between the 15 normal nodes and one failure node. In order to compare the performance of the three time-series similarity measurements, we chose 1/3 as the ratio threshold for the selection. The correlated monitoring nodes and their weights are shown in Table 2. There are 5 correlated monitoring nodes, so the dimension of the input attribute of the training dataset is 5 and the dimension of the output attribute is 1.

Table 2 Timing correlated monitoring nodes and their weights

| | Node3 | Node4 | Node5 | Node6 | Node15 |
|---|---|---|---|---|---|
| **Euclid** | 0.2152 | 0.1926 | 0.1823 | 0.1868 | 0.2230 |
| **DTW** | 0.2476 | 0.1677 | 0.1662 | 0.2043 | 0.2143 |
| **TSR_DIST** | 0.6129 | 0.5852 | 0.5845 | 0.6049 | 0.6003 |

*5.2.2 Model training and testing*

We use three similarity measurements to construct three different regression prediction models, which are named as Feature-Weighted Support Vector Regression based on DTW (DTW-FWSVR), Feature-Weighted Vector Regression based on Euclidean distance (Euclid-FWSVR) and Feature-Weighted Support Vector Regression based on TSR_DIST (TSR_DIST-FWSVR), respectively. The three models are trained with 720 training samples and tested with 168 test samples. Table 3 shows the performance of the three models in the training phase and the testing phase, respectively. The evaluation indicators are MSE and the determination coefficient. The determination coefficient is the square of the correlation coefficient, which reflects the fitting goodness of the model and ranges from 0 to 1. The closer the value of the determination coefficient is to 1, the higher the confidence of the fitting prediction model; the closer to 0, the lower the confidence.

Table 3 Comparison of three model training and test results

| Models | Stage | Evaluation index | | Dimension |
|---|---|---|---|---|
| | | MSE | decision | reduction |
| TSR_DIST-FWSVR | Training | 1.79425 | 87.7942% | 69% |
| | Testing | 4.44622 | 73.5766% | 70% |
| DTW – FWSVR | Training | 1.80026 | 87.6563% | / |
| | Testing | 4.40009 | 73.4073% | / |
| Euclid – FWSVR | Training | 1.97622 | 86.4572% | / |
| | Testing | 4.57176 | 73.0132 % | / |

In Table 3, it seems that TSR_DIST-FWSVR has no significant advantage over the DTW-FWSVR in the training and testing phases. However, TSR_DIST-FWSVR greatly improves the efficiency of calculation through the IEP-TSR process on the original noise time-series data which significantly reduces the dimensionality. Thus, TSR_DIST-FWSVR is more suitable for mass monitoring data application environment.

### 5.3 Experiments on three time-series datasets

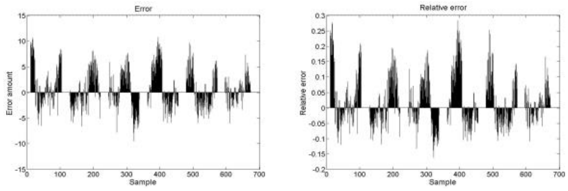
In this experiment, the TSR_DIST-FWSVR model was used to experiment on time-series datasets of three different time intervals. The results of the training and testing phase are shown in Table 4.

Table 4 Comparison of training and test results on three datasets

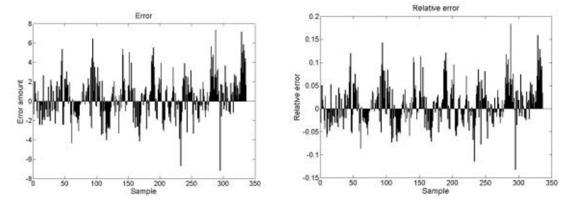
| Datasets | Stage | Evaluation index | | Dimension |
|---|---|---|---|---|
| | | MSE | decision | reduction |
| 15min | Training | 7.57572 | 74.7815% | 71% |
| | Testing | 11.0728 | 55.5476% | 70% |
| 30min | Training | 5.24408 | 79.5692% | 76% |
| | Testing | 7.80195 | 65.9379% | 73% |
| 1hour | Training | 1.79425 | 87.7942% | 69% |
| | Testing | 4.44622 | 73.5766% | 70% |

In Table 4, as the time interval widens, the prediction error is gradually decreasing, while the determination coefficient of the model is increasing. This result is caused by the randomness and uncertainty of the time-series. The smaller the time interval is, the more obvious impact is from this feature; on the contrary, the larger the time interval is, the less impact is from this feature.

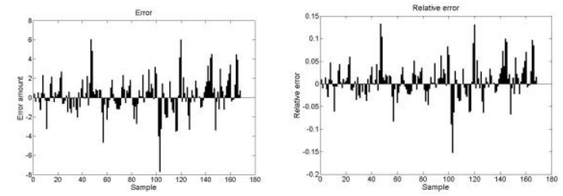
Figure 3 shows the error and relative error rate of the TSR_DIST-FWSVR model on the test dataset at three different timing intervals.




(a) 15-minute timing interval




(b) 30-minute timing interval




(c) 1-hour timing interval

Figure 3 Regression prediction error and relative error distribution

In subfigure (c), for 1-hour interval most of the predicted error rate is within $\pm 0.1$ and only a few of them are between $\pm 0.12$ and $\pm 0.15$, and these large error rates appear on the peak of the distribution. The errors basically fall between $-4dB$ and $+5dB$ with only a few values exceeding this range to $8dB$. With the timing interval decreasing, from the subfigure (a) and the subfigure (b), the error range and the relative error range are both expanding while still under control. For the 30-minute interval, the relative error is basically within $\pm 0.2$ and the error is mostly under $\pm 8dB$; for 15 minutes interval, the relative error is basically within $\pm 0.3$ and most of the error are under $\pm 10dB$. In summary, the overall performance of the model is relatively satisfactory. Although the error is a bit large, this result is acceptable because of the features of the time-series data itself and the uncertainty in the prediction of the failure monitoring node which increase the predicting difficulty.

## 6. CONCLUSION

In this paper, we focus on the error or loss of time-series noise data caused by airport noise monitoring node failure and try to find a solution from the software without using hardware. Based on the trend features of noise time-series data, a time-series representation based on trend segmentation (IEP-TSR) is proposed, and a time-series similarity measurement based on trend segmentation (TSR_DIST) is proposed. We started from studying and analyzing the time-series similarity between monitoring nodes and then compared the performance of the common time-series similarity measurements like Euclidean distance, DTW distance and the TSR_DIST distance in this model. We found that TSR_DIST is more superior to other measurements, so we use the TSR_DIST distance to select the monitoring nodes which with higher similarity to the failed node as the correlated nodes. TSR_DIST is used to calculate the distance between the time-series correlated monitoring node and the failed node in the historical data. Then the distance set is normalized to be the weights and multiplied to the corresponding kernel function to build the interactive prediction model based on the Feature-Weighted Support Vector Regression machine (TSR_DIST-FWSVR) for airport noise monitoring node. In the training phase, we use the cross validation to ensure that the trained model has better performance in terms of precision and generalization. In the testing phase, we use the TSR_DIST-FWSVR model to predict the noise data of the China Capital International Airport. The results show that TSR_DIST-FWSVR model has better predictive performance, and the similarity measurement TSR_DIST is better than the other measurements.

## 7. ACKNOWLEDGEMENTS

We would like to express our gratitude to both the editor and the anonymous reviews for their constructive comments that improved the quality of our manuscript to a large extent. This work is supported by the **National Natural Science Foundation of China** (Nos. **61501229**), the Fundamental Research Funds for the Central Universities (Nos. NS2015091, NJ20160013).

## 8. REFERENCES

[1] Ishii, Hirokazu, et al. "An aircraft noise prediction model considering meteorological effects on noise propagation developed for air traffic optimization." Journal of the Acoustical Society of America. vol.140, October, 2016. p, 3257-3257.

[2] Baraldi, A. N., and C. K. Enders. "An introduction to modern missing data analyses." Journal of School Psychology. vol. 48. February, 2010. p, 5-37.

[3] Mesarovic, M. D., J. D. Pearson, and Y. Takahara. "A multilevel structure for a class of linear dynamic optimization problems." Joint Automatic Control Conference. vol. 3. 1965. p, 93-99.

[4] Soldatenko, Sergei, and D. Chichkine. "Correlation and Spectral Properties of a Coupled Nonlinear Dynamical System in the Context of Numerical Weather Prediction and Climate Modeling." Discrete Dynamics in Nature and Society, vol. 2014, October, 2014. p,1-16.

[5] Morin, Benjamin. "M2D2: a formal data model for IDS alert correlation." Proceeding of the International Conference on Recent Advances in Intrusion Detection, Berlin, Heidelberg, (Germany). vol. 2516, October, 2002. p,115-137.

[6] Li, Peng, et al. "Application of gray relational analysis on the prediction of battery capacity." Proceedings of the 2014 IEEE 9th Conference on Industrial Electronics and Applications (ICIEA), Hangzhou, (China), June 2014. IEEE, October 2014. p, 2158-2297.

[7] Tasci, Levent, and Erkan Kose. "Deformation Forecasting Based on Multi Variable Grey Prediction Models." Journal of Grey System.vol.28, 2016.

[8] Ke Shanliang, Qiu Weining, and Hua Xianghong "Based on the monitoring node Deformation Monitoring similarity forecast model and data recovery." Journal of the Investigation Science and Technology, vol.5,2012. p, 9-11.

[9] Berthold, Michael R., and Frank Höppner. "On clustering time-series using euclidean distance and pearson correlation." Journal of the Computer Science, 2016.

[10] Berndt, Donald J., and James Clifford. "Using dynamic time warping to find patterns in time-series." Proceedings of the 3rd International Conference on Knowledge Discovery and Data Mining, Seattle, WA, (USA). Vol. 10. August, 1994.

[11] Lee, Hwangrae. "The Euclidean distance degree of Fermat hypersurfaces." Journal of Symbolic Computation. vol.8, 2016.

[12] Lines, Jason, and Anthony Bagnall. "Time-series classification with ensembles of elastic distance measures." Journal of the Data Mining and Knowledge Discovery. vol. 29, 2015. p, 565-592.

[13] Droppo, Jasha, and Alex Acero. "Context dependent phonetic string edit distance for automatic speech recognition." Proceeding of the 2010 IEEE International Conference on Acoustics Speech and Signal Processing (ICASSP), Dallas, TX, (USA). March, 2010.

[14] Chen, Lei, and Raymond Ng. "On the marriage of lp-norms and edit distance." Proceedings of the Thirtieth international conference on Very large data bases. Toronto, (Canada). vol.30.September, 2004.

[15] Kate, Rohit J. "Using dynamic time warping distances as features for improved time-series classification." Journal of the Data Mining & Knowledge Discovery. vol.30, 2016. p,283-312.

[16] Forestier, Germain, et al. "Faster and more accurate classification of time-series by exploiting a novel dynamic time warping averaging algorithm."Journal of the Knowledge & Information Systems. vol. 47, 2016. p, 1-26.

[17] Lin, Jessica, et al. "Experiencing SAX: a novel symbolic representation of time-series." Proceeding of the Data Mining and Knowledge discovery, Hingham, MA, (USA). vol.15, October 2007. p, 107-144.

[18] Zhang, Haitao, et al. "New similarity measure method on time-series." Journal of the Computer Engineering & Design. vol. 4, 2014.

[19] Lin, Jessica, et al. "A symbolic representation of time-series, with implications for streaming algorithms." Proceedings of the 8th ACM SIGMOD workshop on Research issues in data mining and knowledge discovery, San Diego, California, (USA). vol. 13, 2003.p, 2-11.

## 9. COPYRIGHT

**Copyright Statement**