Anomaly Location Method for QAR Data Based on Principal Component Analysis Hierarchical Clustering

Xiaoxia Gao1,*, Zhen Cheng2 and Weigang Huo1
1School of Computer Science and Technology, Civil Aviation University of China, Tianjin 300300, China;
2Chengdu CAAC Southwest Communication Network Ltd, Chengdu 610000, China

*Corresponding author

Abstract. The existing QAR (Quick Access Recorder) data anomaly detection algorithm can detect abnormal flights, but it cannot effectively locate the abnormal QAR data of abnormal flights. In order to address the issue, QAR data of abnormal flights are smoothed and pre-processed based on k-medoids algorithm. Pre-processed QAR data are transformed into a one-dimensional time series represented by angle cosine, and a subsequence feature matrix is generated from the angle cosine sequence by sliding window mechanism. Then, the dimension of the matrix is reduced based on principal component analysis, and the top-down hierarchical clustering is performed on row vectors of the reduced matrix according to the amount of information within each column. The anomaly nodes are detected according to the number of vectors contained in the clustering tree nodes. The abnormal data segments of QAR data of abnormal flights are located by the positions of vectors generated in the cosine angle sequence in the abnormal nodes. The experimental results on real flight data sets show that the proposed method can locate not only the known exceedance events, but also the abnormal fragments besides the monitoring items.

Keywords: Anomaly location; Principal component analysis; Hierarchical clustering; Flight operational quality assurance.

1. Introduction
With the development of aviation safety assurance technology, the incidence of aviation accidents is decreasing year by year, but at the same time, unsafe incidents that may lead to accidents happen frequently (Sun Ruishan, 2015). Quick Access Recorder (QAR) records a series of parameters during aircraft flight. QAR data is a typical multidimensional time series (MTS) data. Flight Operational Quality Assurance (FOQA) is a method of event detection and trend analysis for aircrew control and engine by analyzing QAR data. It is an important part of civil aviation safety management. At present, FOQA work is based on the principle of Exceedance Detection (ED). The rules of exceeding limit are formulated by experts, which are easy-to-use and understandable. However, ED technology can only detect predefined anomalies. There are still unknown data anomalies in QAR data. They cannot be recognized by existing rules, which are potential safety hazard.

Researchers have done a lot of work in anomaly detection of QAR data. Orca(Stephen and Schwabacher, 2003) is an anomaly detection algorithm for QAR data based on k-Nearest Neighbor (KNN). It can deal with large-scale QAR data sets by using pruning algorithm in the process of distance calculation. Bhaduri et al. (2011) proposes iOrca algorithm, which introduces a new index strategy and early termination strategy to further improve the efficiency of Orca algorithm. Multiple Kernel Anomaly Detection (Das et al., 2010) is an anomaly detection algorithm based on support
vector machine, which assumes that discrete variables have causal relationship with continuous variables. Symbolic Aggregate Approximation (Lin et al., 2007) is used to symbolize continuous variables. It can handle both continuous variables and discrete variables simultaneously. Das et al. (2013) introduces dynamic symbolic dynamic filtering (SDF) to extract statistical information in time series, and uses iOrca to detect anomaly on QAR data. Yang et al. (2017) proposed an improved wavelet clustering algorithm to solve the problem of uneven density of wavelet clustering algorithm. The validity of the method is verified by detecting the phenomena of bump and parking in the air appearing in the QAR data. Li et al. (2015) proposes a Cluster-based anomaly detection algorithm to detect abnormal flights in order to overcome the problem that ED technology cannot detect unknown anomalies. It transforms each flight data matrix into a vector. The dimensionality of large-scale matrix formed by the sample set is reduced by Principal Component Analysis (PCA). DBSCAN algorithm is used to cluster the row vector of the matrix. Li et al. (2016) proposes Cluster AD-Data Sample algorithm to detect anomaly in the QAR data. The algorithm uses the Gaussian Mixture Model (GMM) to model the sample set and generate cluster index numbers.

Although the above QAR anomaly detection algorithm can detect abnormal flights from the QAR data set, it cannot better explain the reasons for the abnormal occurrence. Abnormal flights need to be further analyzed by domain experts. In order to reduce the time cost of analysis of abnormal flights, the paper proposes a method of abnormal location of QAR data based on Hierarchical Clustering Based Principal Component Analysis (HC-PCA). The method transforms QAR data of abnormal flights into one-dimensional time represented by angle cosine sequence. Feature subsequence matrix is generated from the transformed sequence according to sliding window mechanism. PCA is used to reduce the dimension of the matrix. The row vectors of the matrix are clustered hierarchically from top to bottom according to the information quantity of each column of the reduced matrix. The abnormal fragments of QAR data are determined by the position of each vector included in the abnormal node of the clustering tree in the angle cosine sequence. Experiments on real flight QAR data show the effectiveness of the proposed location method.

2. Some Notations

A set of multidimensional time series (MTS) containing $N$ multidimensional time series samples can be represented as $S = \{S^1, S^2, \ldots, S^N\}$. If each sample $S^i$ contains $m$ variables and the length of each dimension time series is $n_i$, then $S^i = \{S^i_1, S^i_2, \ldots, S^i_m\}$, where the $k$-th dimension time series is $S^i_k = \{S^i_{k,1}, S^i_{k,2}, \ldots, S^i_{k,n_i}\}$.

A derivative sample set generated by sample set $S$ is expressed as $S' = \{S'^1, S'^2, \ldots, S'^N\}$. Then $S'$ is represented as $\{S'^1, S'^2, \ldots, S'^m\}$, and $S'^i$ is denoted as $\{S'^i_1, S'^i_2, \ldots, S'^i_{n_i-1}\}$. $S'^i_{k,j}$ is derived from $S'^i_{k,j+1} - S'^i_{k,j}$. The length difference between $S'$ and $S'$ is 1.

An angle cosine sequence dataset generated by $S'$ can be expressed as $T = \{T^1, T^2, \ldots, T^N\}$. Where $T'$ can be expressed as $T' = \{T'_1, T'_2, \ldots, T'_{n_i-2}\}$. The two arbitrary row vector in $S'$ are expressed as follows:

$Array_1 = [S'^i_{1,j}, S'^i_{2,j}, \ldots, S'^i_{n_i,j}]$,

$Array_2 = [S'^i_{1,j+1}, S'^i_{2,j+1}, \ldots, S'^i_{n_i,j+1}]$

The value of $T'_j$ in $T'$ is the angle cosine between $Array_1$ and $Array_2$. So the length of $T'$ is $n_i - 2$.

3. The QAR Data Smoothing Algorithm Based on k-medoids

Because of the high sampling rate of QAR data and the unstable flight environment, there are many subtle numerical changes in QAR data. These noises affect the anomaly detection algorithm, so it is necessary to smoothly preprocess QAR data. This paper designs a data smoothing algorithm based on k-medoids for QAR data sets. For any QAR parameter column, the following operations are
performed: The value of a parameter column is saved as an array \( A \) and sort it. The average interval size and interval variance between adjacent different values are calculated according to sorted \( A \). The interval threshold is calculated by using the two statistics, and the number of adjacent data intervals larger than the threshold is counted by traversing array \( A \). It is denoted as \( num \). The array \( A \) is divided into \( num \) sub-intervals and the median of each interval is used as the initial cluster center of k-medoids algorithm. The array \( A \) clustered by k-medoids according to the cluster center. The clustering results are used to smooth the data of the QAR parameter column. The smoothed result is the central point value of the cluster in which the data point belongs to. The smoothing algorithm based on k-medoids is described as follows.

Input: MTS sample set \( S = \{S^1, S^2, \ldots, S^n\} \) \( S = \{S^1, S^2, \ldots, S^m\} \), \( S_k = \{S^1_k, S^2_k, \ldots, S^n_k\} \)

Output: Smoothed MTS sample \( S' = \{S'^1, S'^2, \ldots, S'^n\} \)

For \( i = 1 : N \)

For \( j = 1 : m \)

\( A = S^j \);

Sort and traverse \( A \) to calculate the average interval \( \mu \) and the standard deviation of the interval \( \sigma \);

Set up array \( HKM[n] \), \( HKM[k](1 \leq k \leq n) \) records the cluster label of \( S^j \);

Set up threshold \( T = \mu + \sigma \);

Traverse \( A \) again and get the number of intervals greater than \( T \), which is denoted as \( num \);

The array \( A \) is divided into \( num \) interval with equal width. The median of each interval is regarded as the initial cluster center, which is denoted as \( HKcore_1 = [HKcore_1, HKcore_2, \ldots, HKcore_num] \), \( HKcore_k \) represents \( k \)-th cluster center.

Set up temporary array \( HKcore_1[num] \) and \( HNum[num] \). The initial value of element in the two array is 0;

For each \( A[row](1 \leq row \leq n) \) in array \( A \)

Calculate the distance between \( A[row] \) and every cluster centers in \( HKcore_1 \), and assign \( A[row] \) to the nearest cluster center. Assume that the cluster sign is \( \theta \), and \( 1 \leq \theta \leq num \); \( HKM[row] = \theta \);

\( HNum[\theta]++ \);

\( HKcore_1[\theta]+ = A[row] \);

End

Use the array \( HKcore_1 \) and \( HNum \) to calculate the new cluster centers;

Assuming that the number of non-zero elements in \( HNum \) is \( num_2 \), Set up the array \( HNum_2[num_2] \), the initial value of \( HNum_2 \) is 0;

Set up the array \( HKcore_2[num_2] \), and the non-zero element of the array \( HKcore_1 \) is assigned to \( HKcore_2 \) in order;

Set up the array \( HKcore_2[num_2] \), it is initialized to 0;

Do

For each \( A[row](1 \leq row \leq n) \) in array \( A \)

Calculate the distance between \( A[row] \) and every cluster centers in \( HKcore_2 \), and assign \( A[row] \) to the nearest cluster center. Assume that the cluster sign is \( \theta \), and \( 1 \leq \theta \leq num_2 \); \( HKM[row] = \theta \);
According to the principle of k-medoids cluster algorithm, cluster centers are updated by the array \( HNum_2 \) and \( HKcore_2 \), and the results are saved to \( HNum_2 \) and \( HKcore_2 \). Until the value of array \( HKM \) have no change. For each \( i \), calculate the distance between \( S_{j, row} \) and the cluster center in \( HKcore_2 \). If the nearest center sign is \( \theta \) then 

\[
S_{j, row}^* = HKcore_2[\theta]
\]

Output smoothed MTS sample set \( S^* \).

### 4. The QAR Data Anomaly Location Method Based on HC-PCA

MUTSCA <LRCE> algorithm (Huo et al., 2017) and full join rule hierarchical agglomeration clustering algorithm (Keogh et al., 2004) are used to detect the abnormal flights from QAR data set. Then, QAR data of abnormal flights are smoothed by the algorithm in Section 2 of this paper. According to the definition of derivative sample set in Section 1, the smoothed QAR data of abnormal flights are transformed into derivative sample. And the derivative sample is transformed into an angle cosine sequence based on the measurement method of vector angle cosine value, which represents the change of numerical trend of abnormal QAR sample. Suppose \( T^i \) is an angle cosine sequence \( \{T^i_1, T^i_2, \ldots, T^i_{n-1}\} \). Subsequence \( T^i_{sub} = \{T^i_{r-1}, T^i_{r}, \ldots, T^i_{r+length-1}\} \) of \( T^i \) is generated by a sliding window of length \( length \). And the mean, variance, fluctuation ratio of \( T^i_{sub} \), the position of \( T^i_{sub} \) on \( T^i \) are appended to the subsequence \( T^i_{sub} \). The vector \( C \) is generated by the appended subsequence \( T^i_{sub} \). The length of vector \( C \) is \( length+4 \). The fluctuation ratio of \( T^i_{sub} \) are calculated as follows.

Initialization the variable \( w = 0 \), if \( (T^i_{r-1} - T^i_{r}) \times (T^i_{k-1} - T^i_{k}) > 0 (0 \leq k \leq length-1) \), then the variable \( w \) is added one. The value of fluctuation ratio of \( T^i_{sub} \) is \( w / (length-2) \).

All the vector \( C \) generated by the angle cosine sequence \( T^i \) form the matrix \( CMatrix \). The form of \( CMatrix \) is represented as follows.

\[
\begin{bmatrix}
C_{1,1} & \cdots & C_{1,\text{length+4}} \\
\vdots & \ddots & \vdots \\
C_{n,\text{length+1}} & \cdots & C_{n,\text{length+1, length+4}}
\end{bmatrix}
\]

The algorithm of generating matrix \( CMatrix \) is described as follows:

Input: The angle cosine sequence \( T^i \), the sliding window length \( length \).
Output: The matrix \( CMatrix \), which is \( (n_i - length - 1) \times (\text{length}+4) \). The last column of \( CMatrix \) represents the position of each subsequence \( T^i_{sub} \) in \( T^i \).

For \( r_i = 1 \) to \( n_i - length - 1 \)

Create an empty array \( AC[length+4] \);

\[
AC[0:length-1] = [T^i_{r_i}, T^i_{r_i+1}, \ldots, T^i_{r_i+length-1}]
\]
Calculate the mean, variance, fluctuation ratio of \(T_i, T_{i+1}, ..., T_{i+\text{length}}\). The resulted values are denoted as \(\text{mean}, \text{var}, \text{w}\) respectively.

\[
\begin{align*}
AC[\text{length}] &= \text{mean}; \\
AC[\text{length} + 1] &= \text{var}; \\
AC[\text{length} + 2] &= \text{w}; \\
AC[\text{length} + 3] &= r_i;
\end{align*}
\]

The array \(Ac\) is assigned to \(\text{CMatrix}[r_i]\); 

End

Output matrix \(\text{CMatrix}\). PCA algorithm is used to reduce the dimension of the first \(\text{length} + 3\) column of \(\text{CMatrix}\). If the number of column is \(x-1\) after dimension reduction, the matrix of \((n_{\text{-}}\text{length} - 1)\) rows and \(x\) columns is generated, which is denoted as \(\text{CMatrixPCA}\). The form of \(\text{CMatrixPCA}\) is as follow:

\[
\begin{bmatrix}
  c_{1,1} & \cdots & c_{1,x} \\
  \vdots & \ddots & \vdots \\
  c_{n_i-\text{length}+1,1} & \cdots & c_{n_i-\text{length}+1,x}
\end{bmatrix}
\]

The values of column \(x\) of \(\text{CMatrixPCA}\) are the same as the values of column \(\text{length} + 4\) of \(\text{CMatrix}\). In the process of generating clustering tree, the first column of \(\text{CMatrixPCA}\) is used to generate root node, and the second column of \(\text{CMatrixPCA}\) is used to generate child nodes in the second layer of clustering tree, and so on. The K-medoids algorithm described in section 2 is used to cluster column data of \(\text{CMatrixPCA}\). The node of cluster tree are defined as follows.

Class \(\text{HNode}\)

\[
\begin{align*}
\text{ClustersNum} &; //\text{Records the number of clusters of the Node.} \\
\text{ClustersPhase} &; //\text{Records the number of layer of the clustering tree.} \\
\text{RecordsNum} &; //\text{Records the number of vectors of the Node.} \\
\text{double Records[[]]} &; //\text{Records the vectors of the Node.} \\
\text{ArrayList<\text{HNode}> ALHN} &; //\text{Records all the child nodes.}
\end{align*}
\]

The process of generating clustering tree is described as follows.

Input: \(\text{CMatrixPCA}\), The threshold number of vectors in the Node \(\text{CNum}\).

Output: The root node \(\text{HNode}\).

1. Creates root node \(\text{HNode}\), and initializes \(\text{ClustersPhase}\) to 0. All the row vectors of \(\text{CMatrixPCA}\) are saved to \(\text{HNode. Records}\). The pointer of the node \(\text{HNode}\) is saved to temporary pointer \(\text{HNodeT}\);

2. If \(\text{HNodeT.RecordsNum} > \text{CNum}\) Then

K-medoids algorithm is used to cluster the \(\text{ClustersPhase} + 1\) column of \(\text{HNodeT.Records}\). It generates \(K_{\text{temp}}\) clusters, and \(K_{\text{temp}}\) is assigned to \(\text{HNodeT. ClustersNum}\);

End if;

3. For \(\text{row}=1\) to \(K_{\text{temp}}\)

Create new node \(\text{HNodeT2}\), all the vectors in the \(\text{row}\)-th cluster are saved to \(\text{HNodeT2. Records}\). \(\text{HNodeT2. ClustersPhase+1} = \text{HNodeT2. Records} + 1\);

The node \(\text{HNodeT2}\) is added to \(\text{HNodeT.ALN}\); 

End for;

4. For \(\text{row}=1\) to \(\text{HNodeT.ALN}\).size

\(\text{HNodeT} = \text{HNodeT.ALN}.\text{get(row)}; //\text{Return the \(\text{row}\)-th child of the node \(\text{HNodeT}\). Go back to step 2;}

End
5. Return the root node $HNode$.

After generating the clustering tree of QAR samples of abnormal flights, the position of abnormal fragments is derived by traversing the cluster tree and the predefined threshold. There are two thresholds. One is $thresholdFixed$, which indicates the ratio between the number of vectors contained in the child and parent node. The other is $thresholdRelated$, which indicates the ratio between the number of vectors contained in the node and the average number of vectors contained by all the sibling nodes.

The process of locating abnormal position is as follows:

1. The pointer of $HNode$ is assigned to temporary pointer $HNodeT$;
2. Calculate the values of temporary variable $T1$ and $T2$.
   
   $T1= HNodeT. RecordsNum * thresholdFixed$;
   
   $T2= average * thresholdRelated$. Where the variable $average$ is the mean value of the number of vectors in all child nodes of the node $HNodeT$.
3. For $row=1$ to $HNodeT.ALNH.size$
   
   If $HNodeT.ALNH.get (row). RecordsNum$ less than $T1$ and $T2$ then
      
      The last column of $HNodeT.ALNH.get (row). Records$ is inserted into $ADset$.
   
   Else
      
      $HNodeT= HNodeT.ALNH.get (row)$; //Return the $row$-th child of the node $HNodeT$.
      
      Go back to step 2;
   
   End if;
4. Output $ADset$;

5. Experimental Results and Analysis

5.1. Experimental Data and Parameters

In this paper, the QAR data of 153 flights are selected, which is shown in Table 1. The QAR data is generated by two aircrafts. According to expert experience, 21 continuous parameters are selected from QAR data. Table 2 shows the names and meaning of some parameters used in the experimental results.

| Parameter name          | Parameter significance                                      |
|-------------------------|------------------------------------------------------------|
| Radio Altitude          | Real altitude difference between aircraft antenna and ground|
| Roll Angle              | The inclination angle of aircraft on roller shaft           |
| Attack Angle            | The angle between wing chord and air velocity               |
| Pitch                   | The angle between fuselage axis and horizontal plane        |
| Track Angle             | The heading of the fuselage roller relative to the ground   |
| Drift Angle             | The angle between the course and track of aircraft           |
| Vertical Acceleration   | Vehicle acceleration on the vertical                        |
| Vertical Speed          | Velocity of aircraft on vertical line                       |
| Airspeed                | Air Speed based on the relation between velocity and dynamic pressure at standard atmospheric pressure |

Table 1. Experimental data sets.

| Airplane Tail number | Sample size |
|----------------------|-------------|
| B-1                  | 44          |
| B-2                  | 109         |

Table 2. Description of parameters.
5.2. The Experiment of QAR Data Anomaly Location

In this section, we compared our HC-PCA algorithm with ClusterAD-DataSample which is a GMM-based flight data anomaly detection algorithm (Li L et al., 2016). 153 flights are taken as the detection objects, and the target is to locate the exceedance events in each flight sample. The parameter settings of HC-PCA algorithm in this paper are shown in Table 3.

Table 3. The parameter of HC-PCA algorithm

| Parameter name                             | value       |
|--------------------------------------------|-------------|
| Subsequence Length                         | 10          |
| The Retention Matrix Information Ratio of PCA | 0.98        |
| The Maximum Number of Vector in Cluster    | 100         |
| ThresholdFixed                             | 0.01        |
| ThresholdRelated                           | 0.1~2.0     |

The quality of ClusterAD-DataSample algorithm depends on the size of training set. So the number of samples contained in QAR data set is considered. Table 4 gives the comparison results of anomaly location detection. In the last two rows of the table 4 show the mean values of experiment result on 44 samples and 153 samples respectively. Exceedance events often have time continuity. But exceedance data point are recorded in our algorithm experiment, which leads to low accuracy. The calculation methods of precision $P$ and recall $R$ are shown in following formulas (1) and (2).

$$P = \frac{Ar}{Dn}$$

$$R = \frac{Ar}{An}$$

$Ar$ is the number of exceedance points contained in the experiment results, $Dn$ is the number of data points contained in the results, and $An$ is the total number of exceedance points contained in the sample set. Table 4 shows that the two algorithms are effective in locating QAR data exceedance events. There is little difference in precision and recall between the two algorithms. However, when the number of samples is insufficient, it is difficult for GMM algorithm to train a better GMM model, which leads to the decline of detecting ability of the ClusterAD-DataSample. However, HC-PCA algorithm does not need train the mode, and can directly locate and analyze the exceedance events of QAR data of a single flight.

Table 4. The result of detecting exceedance event.

| Sample Size | HC-PCA | ClusteralAD-DataSample |
|-------------|--------|------------------------|
|             | Precision | Recall  | Precision | Recall  |
| 1           | 1.9%    | 75.0%     | 0         | 0       |
| 44          | 1.4%    | 81.5%     | 1.5%      | 31.8%   |
| 153         | 2.0%    | 81.0%     | 0.9%      | 82.4%   |

5.3. QAR Abnormal Data Fragment Analysis

In this section, we first find out the outlier flights from 153 QAR sample sets according to the algorithm proposed by Huo et al. (2017) and Keogh et al. (2004). The upper limit of cluster number is 10, and the threshold of anomaly detection is 5%. That is to say, the outlier flight samples with 5% of the total sample set are output. Then HC-PCA algorithm is used to locate the abnormal position of outlier flight samples. The parameter setting of HC-PCA algorithm is the same as section 4.2. Because the value of length parameter of HC-PCA is 10, each abnormal position actually represents a subsequence with length of 10, which start with the detected abnormal position. In the process of integrating abnormal fragments, if the absolute difference between two abnormal positions is less than 10, the two abnormal fragment are merged. The final output abnormal fragment is shown in Table 5.
Table 5 shows some abnormal fragments of a flight. Sections 637-647 of the table 5 represents the right engine overload anomaly, and the grade of the anomaly is severely exceeded. Sections 7441-7464 represents several concurrent anomalies, which are the vertical acceleration medium level exceedance event, the landing thrust low level exceedance event and some severe warning indicators.

Table 5. Abnormal Fragments of a Flight.

| Starting position | End position | Starting position | End position |
|-------------------|--------------|-------------------|--------------|
| 9                 | 24           | 6064              | 6076         |
| 111               | 126          | 6555              | 6569         |
| 637               | 647          | 7349              | 7365         |
| 830               | 841          | 7441              | 7452         |
| 2369              | 2378         | 7455              | 7464         |

Tables 6 and 7 show the real data fragments of sections 7441-7452 and 6555-6569. Table 6 contains the data of exceedance events detected by existing FOQA technology, which are vertical acceleration exceedance and landing thrust exceedance. The discriminant rule of vertical acceleration exceedance is as follow: when the value of vertical acceleration is less than 0.3 or more than 1.8, it is a serious event. When the value belongs to the interval [1.6, 1.8], it is a medium event. The landing thrust exceedance event is mainly affected by the power of low-pressure compressor. The power unit is the percentage of the maximum power of the engine. When the aircraft enters the landing stage from the final approach stage, the landing thrust overrun event is determined if the power of the low-pressure compressor exceeds 40%. In order to facilitate readers to understand the above two events, Fig. 1 shows the line chart of vertical acceleration in section 7441-7465, and Fig. 2 shows the line chart of engine power. The two parameters are directly related to the above two exceedance events. As shown in Figure 1, the value of vertical acceleration curve is between 1.6 and 1.8 at abscissa point 12, so it belongs to medium exceedance event. In Figure 2, the engine power exceeds 40% before landing, so it’s a landing thrust exceedance event. Figure 3 shows the curve of angle of attack and roll, and Figure 4 shows the line chart of airspeed. The experimental results show that the method based on HC-PCA not only can effectively detect some monitoring items in FOQA, but also can detect the potential numerical anomaly fragments of non-exceedance events. The potential abnormal data fragments that do not reflect exceedance events will be submitted to domain experts for further analysis.

Figure 1. Vertical acceleration exceedance.

Figure 2. Landing thrust exceedance.
Figure 3. Potential abnormal angle of attack and roll sequence.

Figure 4. Potential abnormal airspeed Sequence.

6. Conclusion
In this paper, an anomaly location method for QAR data based on principal component hierarchical clustering is proposed. The method transforms QAR data of abnormal flights into one-dimensional time series represented by angle cosine. Then the sliding window is used to generate a set of feature row vectors of the subsequence on the one-dimensional time series, and PCA is used to reduce the dimension of these vectors. Finally, the reduced vectors are clustered to generate clustering tree, and the abnormal data fragments are output by traversing the clustering tree. The experimental results on real flight data show that the proposed method can effectively locate known exceedance event fragment and potential abnormal data fragments out of the monitoring items.

Reference
[1] SUN Rui-shan, YANG Yi-xuan, WANG Lei. Study on flight safety evaluation based on QAR[J]. China Safety Science Journal, 2015, 25(7): 87-92.
[2] STEPHEN D B, SCHWABACHER M. Mining Distance-Based Outliers in Near Linear Time with Randomization and a Simple Pruning Rule[C]. Acm Sigkdd International Conference on Knowledge discovery and data mining, 2003: 29-38.
[3] BHADURI K, MATTHEWS B L, GIANNELLA C R. Algorithms for speeding up distance-based outlier detection[C]. Acm Sigkdd International Conference on Knowledge discovery and data mining, 2011: 859-867.
[4] DAS S, MATTHEWS BL, SRIVASTAVA AN, et al. Multiple Kernel Learning for Heterogeneous Anomaly Detection: Algorithm and Aviation Safety Case Study[C]. Acm Sigkdd International Conference on Knowledge discovery and data mining, 2010: 47-56.
[5] LIN J, Keogh E, Li W, et al. Experiencing SAX: a novel symbolic representation of time series[J]. Data Mining & Knowledge Discovery, 2007, 15 (2): 107-144.
[6] DAS S, SARKAR S, Ray A, et al. Anomaly Detection in Flight Recorder Data: A Dynamic Data-driven Approach[C]. American Control Conference, 2013, 46(3): 2668-2673.
[7] YANG Hui, LI Zhen, HUO Weigang. Application of Improved Wavelet Clustering Algorithm in QAR Data[J]. Computer Engineering, 2017, 43(9): 29-33.
[8] LI L S, Das S, HANSMAN RJ, et al. Analysis of Flight Data Using Clustering Techniques for Detecting Abnormal Operations[J]. Journal of Aerospace Information Systems, 2015, 12(9): 1-12.
[9] LI L S, HANSMAN RJ, PALACIOS R, et al. Anomaly detection via a Gaussian Mixture Model for flight operation and safety monitoring[J]. Transportation Research Part C Emerging Technologies, 2016, 64: 45-57.

[10] HUO Weigang, CHENG Zhen, CHENG Wenli. Improved clustering algorithm for multivariate time series with unequal length[J]. Journal of Computer Applications, 2017, 37(12): 3477-3481.

[11] KEOGH E, LONARDI S, RATANAMAHAHANATA C A. Towards Parameter-Free Data Mining[C]. Tenth Acm Sigkdd International Conference on Knowledge Discovery and Data Mining, 2004: 206-215

[12] LI L S, HANSMAN RJ, PALACIOS R, et al. Anomaly detection via a Gaussian Mixture Model for flight operation and safety monitoring[J]. Transportation Research Part C Emerging Technologies, 2016, 64: 45-57.