To cite this article: WANG R H, CHEN H, GUAN C. Condition monitoring method for marine engine room equipment based on machine learning [J/OL]. Chinese Journal of Ship Research, 2020, 16(1). http://www.ship-research.com/EN/Y2020/V16/I1/158.

DOI: 10.19693/j.issn.1673-3185.02150

Condition monitoring method for marine engine room equipment based on machine learning

WANG Ruihan[1,2], *CHEN Hui*[*1,2], *GUAN Cong*[1,2]

1 School of Energy and Power Engineering, Wuhan University of Technology, Wuhan 430063, China
2 Key Laboratory of High Performance Ship Technology of Ministry of Education, Wuhan University of Technology, Wuhan 430063, China

Abstract: [**Objectives**] In order to realize the intelligent condition monitoring of marine engine room equipment, we introduced machine learning algorithms and propose a condition monitoring method based on manifold learning and isolation forest. [**Methods**] As condition-monitoring data is multi-dimensional, the proposed method extracted useful features through manifold learning, thereby reducing the dimensions and complexity of the raw data. An isolation forest algorithm was introduced to utilize the normal condition data to train and construct multiple forest detectors, realizing the fault monitoring of the target equipment. To validate the proposed scheme, we developed a two-stroke marine diesel engine in Matlab/Simulink to simulate reliable normal and fault condition datasets. [**Results**] Comparisons of the simulated datasets of the different fault monitoring schemes demonstrate that the proposed method has the highest fault detection rate of 98.5% and the lowest false alarm rate of 3%. [**Conclusions**] The method proposed in this study improves the fault monitoring performance of marine equipment.

Key words: marine diesel engine; fault monitoring; manifold learning; isolation forest

CLC number: U676.4^{+}2

0 Introduction

The engine room is the heart of a ship, and the working condition of the equipment in the engine room is crucial for safe and efficient ship operation. In ship design and manufacturing, ship automation is largely reflected in engine room automation. With the development of computer and communication technologies, marine engine room equipment and system have been gradually automated and intelligentized[1-3]. Conventional fault monitoring methods for engine room equipment are based on physical or mathematical models, empirical models (expert system and fault tree analysis), and reliability models (Bayesian methods and reliability assessment). Fault diagnosis relies mainly on the experience of operators and requires expert knowledge to model complex equipment objects. The enormous and rapid changes in the information of modern ships under different working conditions make it impossible to build accurate physical or mathematical models of marine engine room equipment. Traditional fault monitoring methods are unable to make an accurate diagnosis of the condition of equipment in the marine engine room. Therefore, these methods are unsuitable for intelligent engine rooms, in which equipment sensors and sensing technologies are used to monitor the condition of

Received: 2020 − 10 − 20 **Accepted:** 2020 − 11 − 16

Supported by: Project of the Ministry of Industry and Information Technology of China; National Key R&D Program of China (2019YFE0104600); National Natural Science Foundation of China (51909200)

Authors: WANG Ruihan, male, born in 1994, doctoral candidate. Research interests: health management and intelligent operation and maintenance of marine equipment. E-mail: rhan_wang@163.com

CHEN Hui, male, born in 1962, Ph.D., professor. Research interests: modeling and control of ship power systems and intelligent technology of ships. E-mail: hchen@whut.edu.cn

GUAN Cong, male, born in 1987, Ph.D., associate professor. Research interests: modeling, simulation, and control technologies of ship power systems. E-mail: guancong2008@126.com

***Corresponding author:** CHEN Hui

engine room equipment. The intelligent fault monitoring methods based on machine learning use condition monitoring data and various artificial intelligence algorithms to extract valid information and detect potential failures of mechanical equipment in a timely manner, without requiring prior knowledge about the object system. Compared with traditional fault monitoring methods, intelligent fault monitoring approaches based on machine learning effectively, accurately, and quickly process a large amount of condition monitoring information collected in the marine engine room, and provide more reliable condition diagnosis, which greatly reduces the involvement of anthropogenic factors in fault monitoring.

With the development of machine learning algorithms, fault monitoring methods based on machine learning have become a research hotspot. Machine learning algorithms have been widely used in fault diagnosis of equipment in the marine engine rooms. Gong et al. [4-5] used data enhancement and Dropout technology to improve the convolutional neural network and the accuracy of fault diagnosis of ship bearing systems. Support vector machine (SVM) was used instead of softmax classifier to achieve 99.86% fault diagnosis accuracy with an improved convolutional neural network-support vector machine (CNN-SVM) algorithm. Shang et al. [6] fused principal component analysis (PCA), K-means clustering, and BP neural network to greatly reduce the complexity of original data and improve the performance of the BP neural network. This multi-information fusion technique has been used for fault diagnosis of marine diesel engines to greatly increase the fault detection rate. Gong et al. [7] designed a novel convolutional neural network-global average pooling (CNN-GAP) for fault diagnosis of the marine DC-DC converters. Zhong et al. [8] proposed a depth belief network based on a restricted Boltzmann machine for intelligent diagnosis of marine diesel engines and achieved a 98.61% fault detection rate with the test dataset. Liu et al. [9] combined rough set theory and optimized directed acyclic graph (DAG)-SVM for fault diagnosis of main marine engines, which improved classification accuracy and reduced detection time. Meanwhile, machine learning algorithms such as SVM, neural network (NN), and random forest are also used in a wide range of industrial sectors. Konar et al. [10] achieved higher accuracy and speed of motor bearing fault detection by combining continuous wavelet transform with SVM. Fault monitoring methods based on machine learning algorithms reduce the reliance on prior knowledge of the target equipment. The condition monitoring data are used to estimate the real-time status of the object system, which realizes the adaptive extraction and intelligent diagnosis of fault characteristics with industrial big data.

The above fault diagnosis methods require a large number of data in normal and fault conditions to train diagnostic models. However, fault condition causes great damage to the performance of engine room equipment, and it is impossible for the prolonged operation of equipment in fault condition for data collection. Therefore, there is a lack of fault data to train the fault diagnosis models. As a result, outlier detection is needed for the fault diagnosis of marine engine room equipment. According to the principle of outlier detection, only the sample data in normal conditions is needed to train the model. Fault monitoring can be performed by defining the fault condition data as outliers that deviate from the normal condition samples. Outlier detection techniques mainly include one-class SVM, local outlier factor (LOF), nearest neighbor, and robust covariance (RC). Bicego et al. [11] designed a novel weighted one-class SVM by a clustering algorithm to improve the robustness of detection. Diez-Olivan et al. [12] used local outlier factor combined with K-means algorithm and fuzzy modeling to greatly improve the performance of marine diesel engine fault monitoring. Zhang et al. [13] proposed an angle-based subspace approach to select useful features and to improve the accuracy of high-dimensional anomaly detection. In this paper, isolation forest (iforest) algorithm was proposed for fault monitoring of marine engine room equipment. As a typical anomaly monitoring technique, iforest uses an ensemble learning strategy to integrate multiple anomaly monitoring decision trees and fuse the monitoring results of multiple sub-learners, in order to monitor outliers and to effectively improve the stability and accuracy of fault monitoring [14].

Engine room equipment generates high-dimensional monitoring data that include a variety of thermal parameters. Such data in high-dimensional feature space can easily result in the curse of dimensionality in anomaly monitoring algorithms and invalid fault monitoring models, thus failing to achieve optimal monitoring results. Feature selection is a very effective technique for information extraction prior to fault monitoring, and manifold learning is one of the commonly used methods for

feature selection [15]. Manifold learning transforms the dimension of the original data by establishing a dimensionality reduction mapping relationship in the high-dimensional space and embedding the low-dimensional manifold into the high-dimensional space of the original data [15]. In this paper, we explored the manifold learning methods such as multi-dimensional scaling (MDS), locally linear embedding (LLE), and t-distributed stochastic neighbor embedding (TSNE). Valid information was extracted from the original data with fused features, and high-dimensional data were reduced to two-dimensional data. The fused two-dimensional features were input into the iforest model for efficient data processing. In this paper, a simulation model of a two-stroke marine diesel engine was built in Matlab/Simulink to generate the data of diesel engine conditions. The high-dimensional data were pre-processed by manifold learning to reduce data dimensionality and complexity, and the processed data were input into the iforest model to detect the faults of the marine diesel engine.

1 Description of the fault monitoring method

In this section, we specifically describe the principles of manifold learning and iforest algorithms and build machine learning-based fault monitoring models with Python. Packages and databases such as numpy, pandas, and scikit-learn were used to facilitate the construction of fault monitoring models based on manifold learning and iforest.

1.1 Manifold learning

In topology, manifold is a local topological space with Euclidean geometry or a collection of points in space. Manifold can be simply interpreted as the generalization of curves in two-dimensional space and surfaces in three-dimensional space to higher dimensional spaces. The main idea of manifold learning is to map high-dimensional data to low-dimensional data so that the low-dimensional data can reflect some essential structural characteristics of the original high-dimensional data. The assumption of manifold learning is that high-dimensional data can be obtained by embedding a low-dimensional manifold structure into a high-dimensional space. The purpose of manifold learning is to map the data back to the low-dimensional space, thus enabling the simplification and visualization of the original high-dimensional data. In fault monitoring, manifold learning is a key step in data preprocessing to reduce the dimension and extract the features of the original data.

1.1.1 MDS

MDS is an unsupervised linear dimensionality reduction method that, based on the similarity of sample pairs, constructs a low-dimensional space that maximizes the similarity in sample distance between low- and high-dimensional spaces. After dimensionality reduction, the distance between any two points in the low-dimensional space should be the same as the distance in the original high-dimensional space. Assume that n points in a p-dimensional space form the matrix $\boldsymbol{X} = \{\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_i, ..., \boldsymbol{x}_j, ..., \boldsymbol{x}_n\}$, and $\boldsymbol{x}_n \in R^p$ means that the dimension of $\boldsymbol{x}$ in space R is p. The Euclidean distance between two points satisfies the following equation:

$$d_{ij} = \sqrt{(\boldsymbol{x}_i - \boldsymbol{x}_j)(\boldsymbol{x}_i - \boldsymbol{x}_j)^{\mathrm{T}}} \tag{1}$$

$\boldsymbol{Z}$ is the matrix after dimensionality reduction, and $\boldsymbol{Z} = z_1, z_2, ..., z_{n,} z_n \in R^{p'}$, where p' is the dimension of the matrix after reduction. Zero-mean normalization is performed for each component of $\boldsymbol{Z}$. $\boldsymbol{B}$ is the inner product matrix composed of b_{ij} and satisfies the following equation:

$$\boldsymbol{B} = \boldsymbol{Z}^{\mathrm{T}}\boldsymbol{Z} \tag{2}$$

$\boldsymbol{X}$ and $\boldsymbol{Z}$ have the same Euclidean distance. Therefore, b_{ij} and d_{ij} satisfy the following equations:

$$b_{ij} = \sum_{k=1}^{p} z_{ik} z_{jk} = z_i^{\mathrm{T}} z_j \tag{3}$$

$$d_{ij} = \sqrt{z_i^{\mathrm{T}} z_i + z_j^{\mathrm{T}} z_j - 2 z_i^{\mathrm{T}} z_j} = \sqrt{z_{ii} + z_{jj} - 2 z_{ij}} \tag{4}$$

$\boldsymbol{Z}$ is re-centralized to yield $\sum_{i,j=1}^{n} b_{ij} = 0$, and $\frac{1}{n^2}\sum_{i,j=1}^{n}(d_{ij})^2 = \frac{2}{n}\sum_{i=1}^{n} d_{ii}$, from which Equation (5) can be derived:

$$b_{ij} = -\frac{1}{2}\left(d_{ij}^2 - \frac{1}{n}\sum_{j=1}^{n} d_{ij}^2 - \frac{1}{n}\sum_{i=1}^{n} d_{ij}^2 + \frac{1}{n^2}\sum_{i,j=1}^{n} d_{ij}\right) \tag{5}$$

$\boldsymbol{Z}$ is an $n \times p'$ matrix (p' is a non-negative eigenvalue), and the rank of $\boldsymbol{B}$ satisfies the following equation:

$$rank(\boldsymbol{B}) = rank(\boldsymbol{Z}\boldsymbol{Z}^{\mathrm{T}}) = rank(\boldsymbol{Z}) = p' \tag{6}$$

$\boldsymbol{B}$ is a symmetric positive-semidefinite matrix with non-negative eigenvalue p' and zero eigenvalue $n - p'$. Therefore, $\boldsymbol{B}$ can be expressed as

$$\boldsymbol{B} = \Gamma \Lambda \Gamma \tag{7}$$

where $\Lambda = diag(\lambda_1, \lambda_2, \cdots, \lambda_p)$, and λ is the eigenvalue of $\boldsymbol{B}$; $\Gamma = (\gamma_1, \gamma_2, \cdots, \gamma_p)$, and γ is the eigenvector corresponding to the eigenvalue λ of $\boldsymbol{B}$. Therefore, $\boldsymbol{Z}$ satisfies the following equation:

$$\boldsymbol{Z} = \Gamma \Lambda^{\frac{1}{2}} \tag{8}$$

1.1.2 LLE

Unlike MDS, the core idea of LLE is that each sample point can be approximately reconstructed by a linear combination of adjacent points, which is equivalent to an approximation of complex geometries with segmented linear patches. This linear relationship in reconstruction should be maintained after the sample is projected to the low-dimensional space, and thus the reconstruction coefficients remain unchanged. The LLE algorithm can be implemented in three steps. First, k nearest neighbors of each sample point are identified. Then, the local reconstruction weight matrix of each sample point is calculated from its nearest neighbors. Finally, the output of the sample point is calculated by the local reconstruction weight matrix and its nearest neighbors. We assume that $\boldsymbol{x}_i$ can be expressed as $\boldsymbol{x}_j$, $\boldsymbol{x}_k$, and $\boldsymbol{x}_l$:

$$\boldsymbol{x}_i = w_{ij}\boldsymbol{x}_j + w_{ik}\boldsymbol{x}_k + w_{il}\boldsymbol{x}_l \tag{9}$$

where, w_{ij}, w_{ik}, and w_{il} are weight coefficients.

The initial point is reconstructed by a linear combination with weight coefficient w_{ij}. The reconstruction error is represented by the cost function $E(w)$, which satisfies the following equation:

$$E(w)=\sum_{i=1}^{n}\left|\boldsymbol{x}_i - \sum_{j=1}^{n} w_{ij}\boldsymbol{x}_j\right|^2 \tag{10}$$

$$\sum_{j=1}^{n} w_{ij} = 1 \tag{11}$$

$\boldsymbol{Z}$ is a dimension-reduced matrix and its data are in a d-dimensional space. The original points are in a D-dimensional space, and $D > d$. The LLE algorithm maintains the same weight coefficients w_{ij} of data in the d-dimensional space. x_j corresponds to z_j in the low-dimensional space, and satisfies the following equation:

$$E(\boldsymbol{Z}) = \min \sum_{i=1}^{n}\left|z_i - \sum_{j=1}^{n} w_{ij} z_j\right|^2 \tag{12}$$

This optimization is equivalent to solving the eigenvalue of a sparse matrix. If the sparse matrix is $\boldsymbol{M}$ and $\boldsymbol{Z} = (z_1, z_2, ..., z_m) \in R^{d\times m}$, $\boldsymbol{M}$ satisfies the following equations:

$$\boldsymbol{M}= (\boldsymbol{I}-\boldsymbol{W})^{\mathrm{T}}(\boldsymbol{I}-\boldsymbol{W}) \tag{13}$$

$$\boldsymbol{Z}\boldsymbol{Z}^{\mathrm{T}} = \boldsymbol{I} \tag{14}$$

where $\boldsymbol{Z}^{\mathrm{T}}$ consisting of eigenvectors can be obtained from the minimum eigenvalue d of $\boldsymbol{Z}$; $\boldsymbol{I}$ is the identity matrix, and $\boldsymbol{W}$ is the matrix of weight coefficients w_{ij}.

1.1.3 TSNE

TSNE is an algorithm for reducing the dimensionality of nonlinear data by mapping them to probability distribution through affine transformation. The data in the original space are represented by Gaussian joint probability, and the data in the embedding space are represented by t-distribution. First, the TSNE algorithm constructs a probability distribution of high-dimensional objects so that similar objects are more likely to be selected than dissimilar objects. Second, TSNE defines a similar probability distribution of points in the low-dimensional space, which maximizes the similarity of probability distributions in the high-and low-dimensional spaces.

Given N high-dimensional objects, TSNE first converts the Euclidean distance into probability p_{ij} to represent the similarity between x_i and x_j:

$$p_{j/i} = \frac{\exp(-|\boldsymbol{x}_i - \boldsymbol{x}_j|^2/2\sigma_i^2)}{\sum_{k\neq i}\exp(-|\boldsymbol{x}_i - \boldsymbol{x}_k|^2/2\sigma_i^2)} \tag{15}$$

$$p_{ij} = \frac{p_{j/i} + p_{i/j}}{2N} \tag{16}$$

where σ_i is the Gaussian variance of the data point $\boldsymbol{x}_i$; $p_{j/i}$ and $p_{i/j}$ are cost function parameters, and p_{ij} is the high-dimensional distribution probability between $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$.

TSNE aims to obtain a d-dimensional map y_1, y_2, ..., y_n ($y_n \in R^d$) that reflects p_{ij} as well as possible. With a similar method, the similarity q_{ij} is measured by low-dimensional data points y_i and y_j corresponding to the high-dimensional data points $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$. The q_{ij} is defined as

$$q_{ij} = \frac{(1+\|y_i - y_j\|^2)^{-1}}{\sum_{k=1}^{n}\sum_{k\neq l}(1+\|y_k - y_l\|^2)^{-1}} \tag{17}$$

If the dimensionality reduction is satisfactory and the data features are basically the same, $p_{ij} = q_{ij}$. The objective function C is expressed as the KL divergence between distribution Q and distribution P:

$$C = KL(P\|Q) = \sum_{i\neq j} p_{ij} \lg \frac{p_{ij}}{q_{ij}} \tag{18}$$

The gradient descent method is used to minimize the KL divergence:

$$\frac{\delta C}{\delta y_i} = 4\sum_{j=1}^{n}(p_{ij} - q_{ij})(y_i - y_j)(1 - \left\|y_i - y_j\right\|^2)^{-1} \tag{19}$$

The Gaussian distribution can be initialized to a smaller value σ. To speed up the optimization process and avoid local optimal solutions, we should use a larger momentum in the gradient:

$$Y^{(t)} = Y^{(t-1)} + \eta\frac{\delta C}{\delta Y} + \alpha(t)[Y^{(t-1)} - Y^{(t-2)}] \tag{20}$$

where $Y^{(t)}$ is the value after iterating for t times; η is

the learning efficiency, and $\alpha(t)$ is the momentum after iterating for t times.

1.2 iforest

The iforest algorithm uses a large number of binary trees, and it is an anomaly detection algorithm based on partition isolation and ensemble learning, without the need to model fault condition data. The core idea of iforest is to identify abnormal data by constructing binary trees because the abnormal data are relatively isolated from the normal data. Usually, only a few partitions are needed to separate the abnormal data and identify them as outliers. In the iforest model, binary trees are used for data partitioning, and the depth of the datapoint in binary trees reflects the "isolation" of the data. This algorithm introduces the concepts of isolation trees and average path length. The algorithm is roughly divided into two stages:

1) In the training stage, multiple subsamples are extracted from the training dataset to construct binary trees.

2) In the testing stage, the samples are passed by isolation trees to obtain the anomaly score of each test sample.

In the training stage, iforest algorithm is closely related to subspace anomaly detection. One data sample (n instances from d-variable distribution) is used to build one binary tree. A batch of samples is extracted from the whole data, and then a feature is randomly selected as the root node. A value is randomly selected between the maximum and minimum values of the feature. Sample data less than this value are assigned to the left branch and the remaining data to the right branch. After that, the above steps are repeated in the left and right data branches until the following conditions are met:

1) The data cannot be subdivided, namely that only one data point is present, or all data are the same;

2) The binary tree reaches a preset maximum depth.

In the testing stage, average path length and anomaly score are used to detect the anomalies. To calculate the anomaly score of data $\boldsymbol{x}$, we should first estimate its path length (depth) in each binary tree. Along a binary tree, the path starts from the root node and goes down according to the values of different features until reaching a leaf node. The path length of $\boldsymbol{x}$, $h(\boldsymbol{x})$, is measured by the number of edges, and the anomaly score can be expressed by s:

$$s(\boldsymbol{x},n)=2^{-\frac{E(h(\boldsymbol{x}))}{c(n)}} \tag{21}$$

$$c(n)=2H(n-1)-[2(n-1)/n] \tag{22}$$

$$H(k)=\ln(k)+\varepsilon \tag{23}$$

where ε=0.577 215 664 9, is the Euler's constant; $c(n)$ is the parameter used to normalize the path length $h(\boldsymbol{x})$ of sample data $\boldsymbol{x}$; $E(h(\boldsymbol{x}))$ is the average path length of $\boldsymbol{x}$ in binary tree ensemble; $s(\boldsymbol{x}, n)$ is the anomaly score of $\boldsymbol{x}$ obtained from the binary trees of n sample training data, and the range of $s(\boldsymbol{x}, n)$ is [0, 1]:

1) If $E(h(\boldsymbol{x}))$ is close to 0, and s is very close to 1, $\boldsymbol{x}$ is probably abnormal.

2) If $E(h(\boldsymbol{x}))$ is close to $n-1$, and s is very close to 0, $\boldsymbol{x}$ is probably normal.

3) If $E(h(\boldsymbol{x}))$ is close to $c(n)$ and all s values are around 0.5, there is no obvious outlier in the whole sample data.

2 Fault monitoring method based on manifold learning and iforest

In the engine room, equipment that requires condition monitoring includes the main propulsion diesel engine, power generation diesel engine, shaft system, propulsion control system, important auxiliary machinery, important pumps and motors, and anchor windlass. The main propulsion diesel engine is the heart of a ship, and its safe and reliable operation directly determines the safety of ship navigation. The main propulsion diesel engine is not only the most important mechanical equipment of a ship but also the equipment with the highest failure rate among all mechanical systems. The study on ship failure risk by the Swedish Club shows that main diesel engine failure accounts for 37.7% of total ship mechanical failure, and causes a total annual economic loss of about 202 million US dollars [16]. Therefore, it is important to reduce the failure rate of the main diesel engine system to ensure safe navigation. In this paper, the main diesel engine was used as the typical equipment in the marine engine room to study the intelligent fault monitoring method based on manifold learning and iforest.

It is not feasible to input raw data directly into the iforest model because the multidimensional raw data cause a curse of dimensionality and reduce the performance of fault monitoring if the model is trained directly. In the data pre-processing stage, suitable data features can be selected by human experience. We used manifold learning for dimensionality reduction of the original data, and there is no need for expert knowledge. By selecting and fusing

the features of the original data through manifold learning, we can reduce the complexity of the original data structure and construct new eigenvectors while retaining as much useful information as possible. Therefore, iforest model can be trained with less computational effort and only with the normal working condition dataset. The iforest model calculates the average path length for each normal condition point and sets the threshold by assuming that there are a small number of outliers in the normal condition data. Therefore, the normal and fault condition data of the test dataset can be classified. The iforest model is more suitable for fault monitoring in industrial design because it does not require fault condition data to train the fault monitoring model. The fault monitoring method based on manifold learning and iforest is shown in Fig. 1. Firstly, manifold learning is used in data preprocessing for feature selection and fusion. Secondly, the iforest model is trained with low-dimensional data for fault monitoring. Finally, model performance is evaluated using the fault detection rate (FDR) and false alarm rate (FAR).

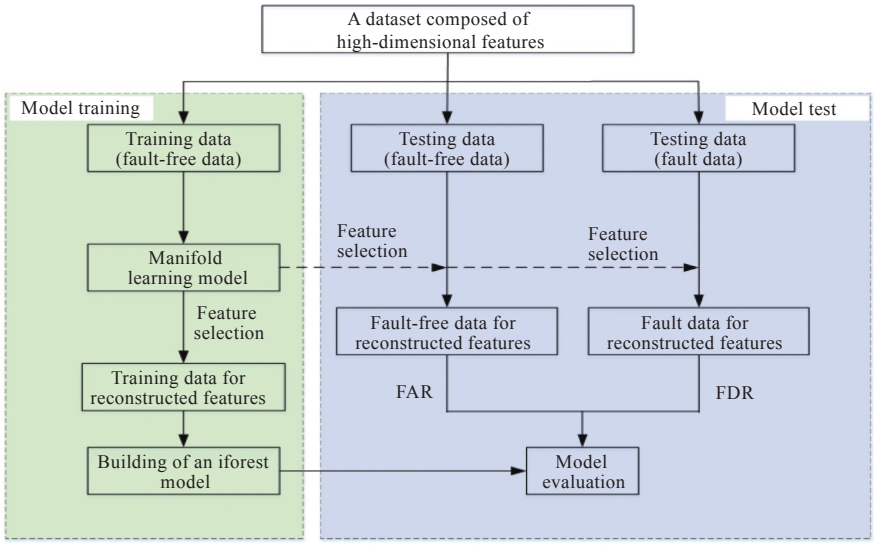


Fig. 1 Procedures of the manifold learning-iforest monitoring scheme

3 Faults monitoring based on simulation system of marine diesel engines

3.1 Simulation system of marine diesel engines

The fault monitoring method based on manifold learning and iforest requires diesel engine state data for model training and testing. Due to the lack of historical monitoring data of marine diesel engines and the serious damage caused by destructive tests to the diesel engines, the fault sample data of marine diesel engines are too scarce to test and evaluate the model. In this paper, the two-stroke marine diesel engine (7K98MC) was modeled and simulated in Matlab/Simulink, and its cylinder was simulated by a zero-dimensional model [17]. The simulation results of the diesel engine model were compared with the data from the shop test to determine the accuracy of the simulation model. The technical specifications of the 7K98MC marine two-stroke diesel engine are shown in Table 1, and the simulation model is shown in Fig. 2.

Table 1 Technical parameters of 7K98MC marine diesel engine

Technical specification	Value	Technical specification	Value
Cylinder diameter/mm	980	Maximum rated speed/(r·min^{-1})	94
Stroke/mm	2 660	Maximum mean indicated pressure/bar	18.2
Piston area/m^2	0.754 3	Maximum explosion pressure/bar	140.1
Machine weight/t	2 100	Turbocharger	3×TPL85-B11
Maximum power/kW	40 055	Firing order	1-7-2-5-4-3-6

To verify the accuracy of the 7K98MC simulation model, we compared the results of the simulation under different loads with the shop test data (Table 2).

As shown in Table 2, the model simulation results were consistent with the shop test, and the maximum error (4.53%) was found in the turbine speed of the diesel engine at 25% load. The errors between the simulated and the experimental data of other parameters were approximately 1%, which

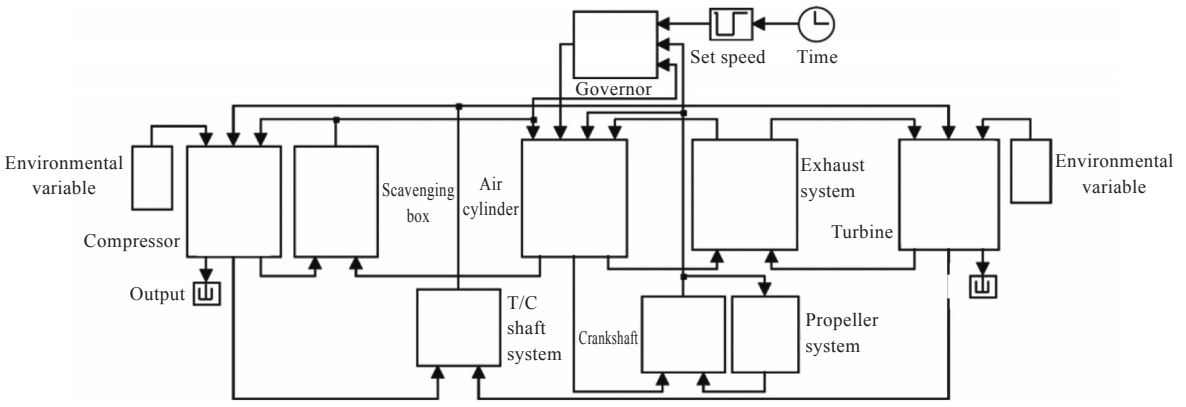


Fig. 2 Simulation model of 7K98MC marine diesel engine

Table 2 Comparison between simulation results and shop test data

Diesel engine load/%	Result	Power/kW	Fuel consumption/ ($g \cdot kW^{-1} \cdot h^{-1}$)	Maximum cylinder explosion pressure/bar	Cylinder compression pressure/bar	Turbine speed/ ($r \cdot min^{-1}$)	Scavenging box pressure/bar	Exhaust pipe temperature/K
25	Simulated value	10 105	186.04	74.04	47.95	4485	1.33	579.29
	Experimental value	10 014	186.37	73.60	47.40	4291	1.32	577.17
	Error/%	0.92	−0.18	0.59	1.16	4.53	0.54	0.37
50	Simulated value	20 226	179.60	99.43	72.62	7896	2.07	593.68
	Experimental value	20 028	179.46	98.00	72.00	7782	2.05	600.17
	Error/%	0.99	0.08	1.46	0.86	1.47	0.89	−1.08
75	Simulated value	30345	175.03	127.91	100.13	9 710	2.86	611.63
	Experimental value	30041	176.01	128.00	99.90	9 670	2.87	614.57
	Error/%	1.01	−0.55	−0.07	0.23	0.41	−0.40	−0.48
100	Simulated value	40462	177.69	138.40	125.67	10 943	3.58	660.83
	Experimental value	40055	177.98	139.40	126.70	10 946	3.63	663.90
	Error/%	1.02	−0.16	−0.71	−0.81	−0.03	−1.39	−0.46

verified the rationality and correctness of the simulation model of marine diesel engines.

3.2 Data description

The simulation model of 7K98MC was studied at 94 r/min and 100% load. The fault condition of the diesel engine was simulated by changing the compressor efficiency, cooler efficiency, and fuel injection time. Each working condition was described by 15 features, including effective power, effective fuel consumption rate, air-fuel ratio, maximum cylinder explosion pressure, maximum cylinder combustion temperature, compressor inlet pressure, compressor outlet temperature, intercooler outlet temperature, scavenging box pressure, scavenging box temperature, exhaust pipe pressure, exhaust pipe temperature, turbine outlet pressure, turbine outlet temperature, and turbine speed. The diesel-engine simulation model was running for about 30 min before stabilization, and the model was running for a total of nearly 200 h. A total of 700 samples under normal and fault conditions were collected. The data are described in Table 3.

Table 3 Simulation datasets

Category	Working condition	Number of features	Number of samples
1	Normal	15	400
2	Compressor failure	15	100
3	Air Cooler failure	15	100
4	Fuel injection timing error	15	100

In this study, the training dataset consisting of 200 normal condition samples was used to build a fault monitoring model based on manifold learning and iforest. The testing dataset consisted of 500 samples, including 200 samples under normal conditions and 300 samples under fault conditions (diesel engine compressor failure, cooler failure, and fuel injection timing error). The testing dataset was used to assess and compare the performance of dif-

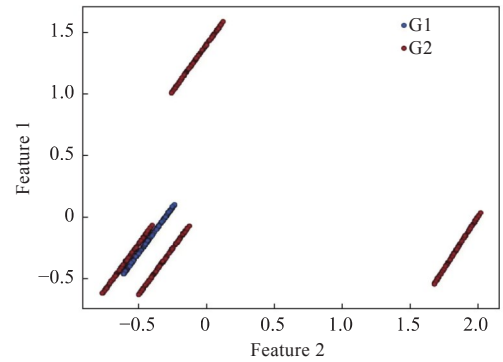
ferent fault monitoring methods.

3.3 Data dimensionality reduction based on manifold learning

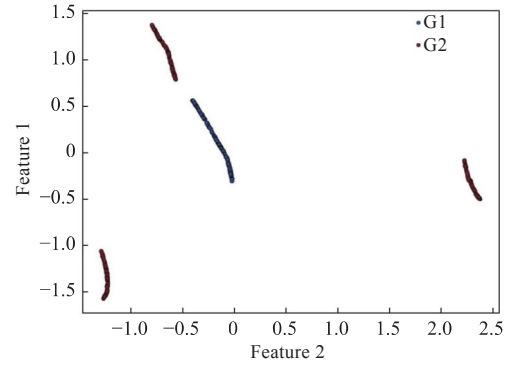
Since the working condition of a diesel engine can affect data features, the selection of features that accurately describe the normal and fault conditions of the diesel engine as well as the extraction and fusion of representative features are critical to diesel engine fault monitoring.

Dimensionality reduction based on manifold learning accurately describes the features of equipment in normal and abnormal working conditions through selection and fusion, thus reducing the number of features and the complexity of the original dataset for subsequent efficient data processing and classification. In this paper, a dataset consisting of 15 features was used to construct a matrix. By manifold learning, the 15-dimensional original data were reduced to 2-dimensional data, and the two fused features were used for fault monitoring. The effects of data-feature dimensionality reduction by manifold learning algorithms PCA, MDS, LLE, and TSNE were assessed by data visualization. Fig. 3 shows the performance of different manifold learning algorithms in dimensionality reduction of the same data. The dimensionality reduction of data with the same distribution by different algorithms was visualized. In Fig. 3, G1 represents the data under normal conditions, and G2 represents the data under fault conditions.

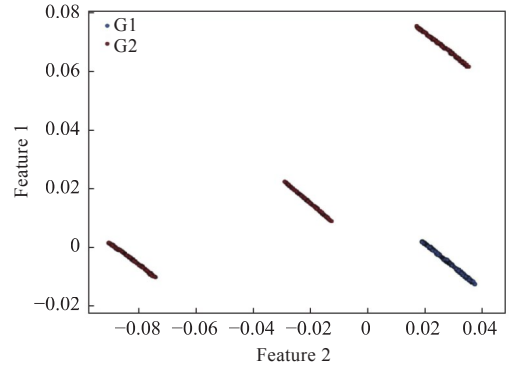
As shown in Fig. 3 (a), for the data with dimensionality reduction by PCA, G1 and G2 partially overlapped, indicating inefficient dimensionality reduction. From Fig. 3 (b) and (c), the MDS and LLE algorithms had better classification of data under different working conditions and thus better dimensionality reduction than PCA. Meanwhile, LLE had better feature extraction than MDS because the features extracted by LLE had wider spacing between datasets of different working conditions and could be more easily distinguished. As shown in Fig. 3 (d), the TSNE algorithm had the best dimensionality reduction, because the data from the same category clustered together, while the data of different categories were located far apart and could be easily distinguished. As a data visualization tool, manifold learning performed well in data dimensionality reduction and feature extraction, with the TSNE algorithm having the best performance in feature selection.



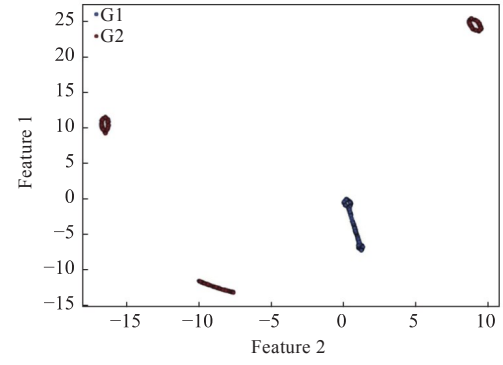
(a) Dimensionality reduction by PCA



(b) Dimensionality reduction by MDS



(c) Dimensionality reduction by LLE



(d) Dimensionality reduction by TSNE

Fig. 3 Dimensionality reduction effect of different manifold learning methods

3.4 Diesel engine fault monitoring based on iforest algorithm

Feature selection methods such as PCA, MDS, LLE, and TSNE were used to project the 15-dimensional data into a 2-dimensional space. The selected

and the fused features were input into the iforest model to monitor the state of the diesel engine. To verify the performance of iforest algorithm, we also studied outlier monitoring methods RC and one-class SVM.

The simulation data were used to study the fault monitoring method based on manifold learning and iforest. The training of the fault monitoring model involved only the data under normal working conditions (200 samples). The trained fault monitoring model was used to identify new normal and fault samples. The performance of the outlier monitoring model based on manifold learning was verified by calculating FAR and FDR. FAR is the ratio of the number of normal samples misclassified as abnormal samples to the total number of normal samples, and FDR is the ratio of the number of correctly classified abnormal samples to the total number of abnormal samples. Therefore, larger FDR and smaller FAR were associated with better performance of the fault monitoring method.

The combination of different manifold learning algorithms and outlier monitoring algorithms leads to fault monitoring methods with different performances. In this paper, the performance of different fault monitoring approaches was compared in boxplot (Fig. 4). The boxplot presents the minimum, lower quartile, median, upper quartile, maximum, and outliers, and evaluates the performance of fault monitoring methods from multiple perspectives. Table 4 shows the mean FDR and FAR for different fault monitoring methods.

As shown in Fig. 4, the method based on TSNE and iforest had the highest FDR, the lowest FAR, and narrow box width, indicating high stability of the monitoring method. Meanwhile, Table 4 shows that the fault monitoring method based on TSNE had the best performance with the same outlier monitoring algorithms, which further illustrates that TSNE had higher quality and lower loss in the dimensionality reduction of condition monitoring data of marine diesel engines.

The training of iforest model only requires the sample dataset under normal conditions. The iforest model calculates the average path length of each normal condition sample and defines a threshold for the classification of normal and abnormal data. Fig. 5 shows the threshold $T1$ obtained by calculating the average path length of normal samples with the fault monitoring methods combining different manifold learning algorithms and iforest. As shown in Fig. 5, the $T1$ obtained by TSNE-iforest was the best. Only a small number of normal condition samples were misclassified as abnormal condition samples, and all abnormal condition samples were correctly classified.

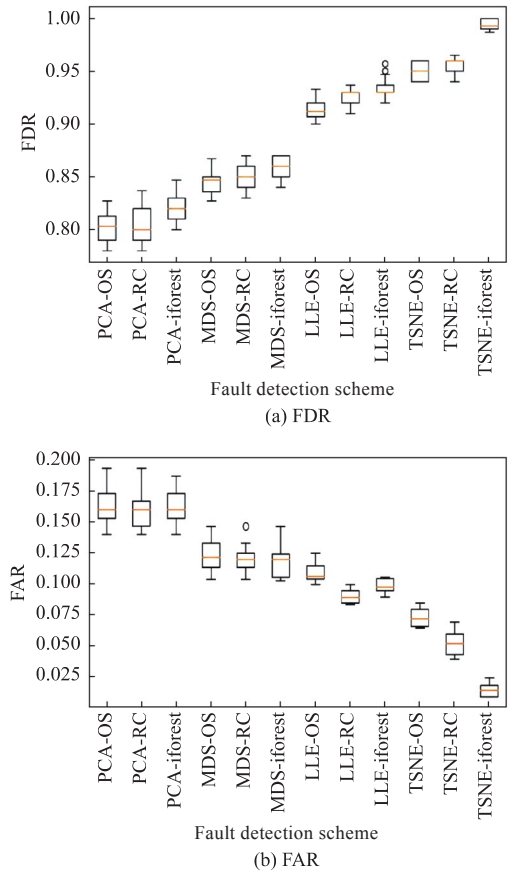
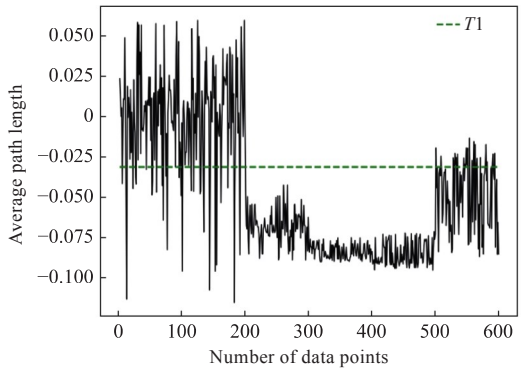


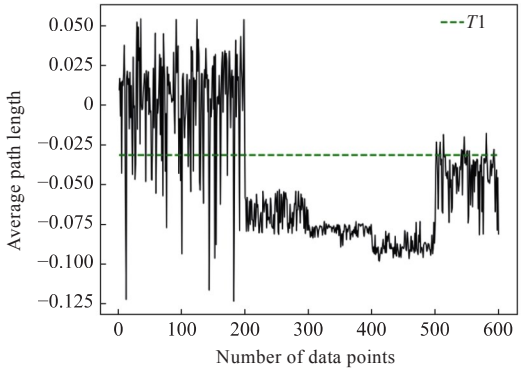
Fig. 4 Comparison of FDR and FAR under different hybrid fault monitoring schemes

Table 4 Mean FDR and FAR under different hybrid fault monitoring schemes

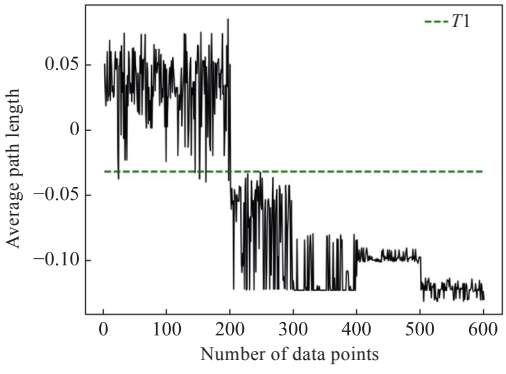
Method	FDR/%	FAR/%
PCA-OS	81.2	1.63
PCA-RC	81.1	1.62
PCA-iforest	81.3	1.6
MDS-OS	85	1.24
MDS-RC	85.5	1.23
MDS-iforest	87.1	1.2
LLE-OS	92.1	1.1
LLE-RC	93.1	8.5
LLE-iforest	93.4	9
TSNE-OS	94.9	7.5
TSNE-RC	96.1	6
TSNE-iforest	98.5	3



(a) Threshold of MDS-iforest



(b) Threshold of LLE-iforest



(c) Threshold of TSNE-iforest

Fig. 5 Thresholds of different fault monitoring schemes

4 Conclusions

To address the actual needs of marine engine room equipment, in this paper we proposed a diesel engine fault monitoring method based on the combination of manifold learning and outlier monitoring. The performance of this fault monitoring method was validated by condition data generated from a simulation model of marine diesel engines. The results are as follows:

1) Compared with PCA, manifold learning algorithms such as MDS, LLE, and TSNE effectively reduced the original 15-dimensional data to 2-dimensional data. In the data preprocessing stage, manifold learning greatly reduced the complexity of the original data and improved the performance of the fault monitoring model. TSNE algorithm had the best data dimensionality reduction effect.

2) Compared with fault monitoring algorithms such as RC and one-class SVM, iforest had higher FDR and lower FAR, and only required data from normal working conditions for training models and monitoring marine diesel engine faults.

3) Fault monitoring method based on TSNE and iforest yielded a suitable threshold value that accurately classified normal condition data and fault condition data.

The method based on TSNE and iforest effectively improved the accuracy and reliability of marine diesel engine fault monitoring. This method uses only normal condition samples for fault monitoring and is more suitable for the actual working conditions of marine engine room equipment. Moreover, the method has high diagnostic stability and thereby possesses both theoretical and applied reference implications.

References

[1] ELAMIN F, FAN Y B, GU F S, et al. Diesel engine valve clearance detection using acoustic emission [J]. Advances in Mechanical Engineering, 2010, 2: 495741.

[2] KOWALSKI J, KRAWCZYK B, WOŹNIAK M. Fault diagnosis of marine 4-stroke diesel engines using a onevs-one extreme learning ensemble [J]. Engineering Applications of Artificial Intelligence, 2017, 57: 134−141.

[3] CHEN H, ZHANG Z H, GUAN C, et al. Optimization of sizing and frequency control in battery/supercapacitor hybrid energy storage system for fuel cell ship [J]. Energy, 2020, 197: 117285.

[4] GONG W F, CHEN H, ZHANG Z H, et al. Intelligent fault diagnosis for rolling bearing based on improved convolutional neural network [J]. Journal of Vibration Engineering, 2020, 33 (2): 400−413 (in Chinese).

[5] GONG W F, CHEN H, ZHANG M L, et al. Intelligent diagnosis method for incipient fault of motor bearing based on deep learning [J]. Chinese Journal of Scientific Instrument, 2020, 41 (1): 195−205 (in Chinese).

[6] SHANG Q M, WANG R H, CHEN H, et al. Application of multi-information fusion technology for fault diagnosis in marine diesel engine [J]. Navigation of China, 2018, 41 (3): 26−31 (in Chinese).

[7] GONG W F, CHEN H, ZHANG Z H, et al. A data-driven-based fault diagnosis approach for electrical power DC-DC inverter by using modified convolutional neural network with global average pooling and 2-D feature image [J]. IEEE Access, 2020 (8): 73677−73697.

[8] ZHONG G Q, JIA B Z, XIAO F, et al. Intelligent fault diagnosis of marine diesel engine based on deep belief network [J]. Chinese Journal of Ship Research, 2020,

15 (3): 136−142, 184 (in Chinese).

[9] LIU G Q, LIN Y J, ZHANG Z Z, et al. Main marine engine fault diagnosis method based on rough set theory and optimized DAG-SVM [J]. Chinese Journal of Ship Research, 2020, 15 (1): 68−73 (in Chinese).

[10] KONAR P, CHATTOPADHYAY P. Bearing fault detection of induction motor using wavelet and support vector machines (SVMs) [J]. Applied Soft Computing, 2011, 11 (6): 4203−4211.

[11] BICEGO M, FIGUEIREDO M A T. Soft clustering using weighted one-class support vector machines [J]. Pattern Recognition, 2009, 42 (1): 27−32.

[12] DIEZ-OLIVAN A, PAGAN J A, SANZ R, et al. Data-driven prognostics using a combination of constrained K-means clustering, fuzzy modeling and LOF-based score [J]. Neurocomputing, 2017, 241: 97−107.

[13] ZHANG L W, LIN J, KARIM R. An angle-based subspace anomaly detection approach to high-dimensional data: with an application to industrial fault detection [J]. Reliability Engineering & System Safety, 2015, 142: 482−497.

[14] LI X P, GAO X, YAN B, et al. An approach of data anomaly detection in power dispatching streaming data based on isolation forest algorithm [J]. Power System Technology, 2019, 43 (4): 1447−1456 (in Chinese).

[15] ZHANG J, WANG Y, LI K H, et al. Multi-source sensor body area network data fusion model based on manifold learning [J]. Computer Science, 2020, 47 (8): 323−328 (in Chinese).

[16] MALM L A, ENSTRM J, HULTMAN A. Main engine damage study [EB/OL]. [2020-10-16] . http: //www.swedishclub.com.

[17] GUAN C, THEOTOKATOS G, ZHOU P L, et al. Computational investigation of a large containership propulsion engine operation at slow steaming conditions [J]. Applied Energy, 2014, 130: 370−383.

基于机器学习的船舶机舱设备状态监测方法

王瑞涵[1,2], 陈辉*[1,2], 管聪[1,2]

1 武汉理工大学 能源与动力工程学院,湖北 武汉 430063

2 武汉理工大学 高性能舰船技术教育部重点实验室,湖北 武汉 430063

摘　要: [**目的**] 为实现船舶机舱设备的智能状态监测,引入机器学习算法,提出一种结合流形学习和孤立森林的船舶机舱设备状态监测方法。[**方法**] 由于船舶机舱设备的状态监测数据是多维度数据,基于该监测系统,通过流形学习来提取有效的数据特征,实现对原始数据的降维,减少数据复杂度。基于孤立森林算法,在仅利用正常工况数据集的情况下,训练并构建多个子森林检测器,用于实现对目标设备的故障监测。在Matlab/Simulink环境下建立大型船舶二冲程柴油机模型,对其正常工况和故障工况下的数据进行仿真,以验证该方案的有效性。[**结果**] 通过状态仿真数据对不同故障监测方案性能的比较,验证了所提故障监测方案具有98.5%的故障检测率和3%的故障虚警率。[**结论**] 所提方法能显著提高船舶机舱设备的故障监测性能。

关键词: 船舶柴油机; 故障监测; 流形学习; 孤立森林