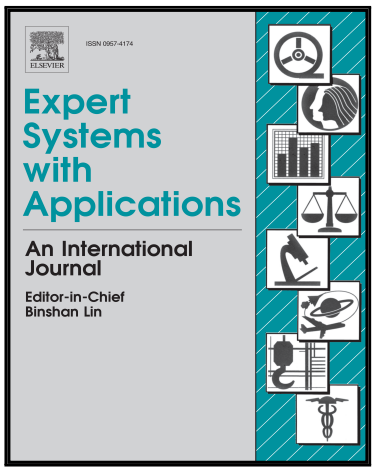

A scalable fuzzy support vector machine for fault detection in transportation systems

Jie Liu , Enrico Zio

PII: S0957-4174(18)30095-2
DOI: 10.1016/j.eswa.2018.02.017
Reference: ESWA 11821

To appear in: *Expert Systems With Applications*

Received date: 11 October 2017
Revised date: 7 February 2018
Accepted date: 8 February 2018

Please cite this article as: Jie Liu , Enrico Zio , A scalable fuzzy support vector machine for fault detection in transportation systems , *Expert Systems With Applications* (2018), doi: 10.1016/j.eswa.2018.02.017

Highlights

- Fuzzy support vector machine is used for fault detection in high speed train.
- A new KNN-based method is proposed for calculating fuzzy membership values.
- The new method is scalable for imbalanced big data.
- The computation burden is largely reduced in the experiment on a real dataset.

A scalable fuzzy support vector machine for fault detection in transportation systems

Jie Liu[1] and Enrico Zio[2, 3, 4 *]

[1]School of Reliability and Systems Engineering, Beihang University, 37 Xueyuan Road, Haidian, Beijing, China (e-mail: liujie805@buaa.edu.cn)

[2]Chair on system science and energetic challenges, EDF Foundation, Laboratoire Genie Industriel, CentraleSupélec, Université Paris-Saclay, 3 rue Joliot Curie, 91190 Gif-sur-Yvette, France (e-mail: enrico.zio@centralesupelec.fr)

[3]Energy Department, Politecnico di Milano, Piazza Leonardo da Vinci, 32, 20133 Milano, Italy (e-mail: enrico.zio@polimi.it)

[4]Sino-French Risk Science and Engineering Lab, Beihang University, 37 Xueyuan Road, Haidian, Beijing, China

[*]Corresponding author

Abstract

Prognostics and health management can improve the reliability and safety of transportation systems. Data collected from diverse sources provide a chance and at the same time a challenge for data-driven PHM methods and models. The data often exhibit challenging characteristics like imbalanced data on normal and faulty conditions, noise and outliers, data points of different importance for the data-driven model, etc. In this paper, a *k* nearest neighbors-based fuzzy support vector machine is proposed for reducing the computational burden and tackling the issue of imbalance and outlier data, in fault detection. Fault detection is mathematically a classification problem. In this paper, the reverse nearest neighbors technique is adopted for detecting outliers and the *k* nearest neighbors technique is used to identify the borderline points for defining the classification hyperplane in support vector machines. Considering the position of each data point and the distribution of its nearest neighbors, a new method is proposed for calculating their estimation error costs. The proposed method is verified by comparison with several benchmark methods on five public datasets. Then, a real case study concerning fault detection in a braking system of a high-speed train is considered.

Keywords: prognostics and health management; high-speed train; fuzzy SVM; fuzzy membership calculation; imbalanced data

1. Introduction

Prognostics and Health Management (PHM) is considered as an important and efficient way for increasing the safety, benefit, reliability and efficiency of transportation systems, relying on past and current information on environmental, operational and usage records to detect (Liu et al., 2017) and diagnose (Guzinski et al., 2010) degradation, to predict future conditions (Kecman et al, 2015) and schedule proper maintenance interventions (Yan et al., 2016; Zio, 2012). By the recording of data on the system conditions, data-driven methods have been widely integrated for analyzing and managing faults and accidents in transportation systems (Liu, 2017; Zilko et al., 2016; Li and He, 2015). One

advantage of data-driven methods is the limited demand on the physical understanding of failure mechanism in a specific system to build the model.

In this paper, we focus on fault detection of transportation systems with data-driven methods. Fault detection is mathematically a classification problem. The objective is to find the classification hyperplane separating the healthy and faulty scenarios. Considering the availability of data on faulty scenarios, the data-driven methods can be generally categorized into supervised learning and unsupervised learning. For unsupervised learning, one popular and efficient way is to calculate the deviation of the monitored data from the values predicted by a model trained on healthy scenarios (Jiang and Wei, 2017). One drawback is that the classification border (or plane) can be conservative when the collected healthy data are not compact and, then, the fault detection rate can be very low especially in the case of partial overlapping between the healthy and faulty scenarios (López et al., 2013). Faulty scenarios can help better estimating the classification hyperplane. In this paper, considering the nature of the data available, supervised learning is considered. In practice, the faulty scenarios are rare for safety-critical and highly reliable systems. On the other hand, the faults in a system can be of diverse type (Liu et al., 2017). For example, in the braking system of a high-speed train, the frequently occurring faults include communication link fault, repeated dual inhibit fault, contactor fault-VCB, pneumatically brake stop cock closed on bogie. These faults do not influence the principal function of the braking system. The data on the non-critical faults in the braking system can be collected with a low cost, and are not critical as they can improve the performance of data-driven methods of faulty detection. Still, compared to the data on healthy scenarios, the collected fault data are still very limited. Thus, the between-class imbalance emerges as an important challenge.

For safety critical systems as nuclear power plants, high-speed trains and aerospace shuttles, large amounts of data have been collected and stored through integrated sensors, video inspections, hand-held field tables and other sources (Attoh-Okine, 2014; Zarembski, 2014). From such data, proper data-driven approaches can extract complex relations among variables for accurate fault detection, as shown in Tanaka (2015), Li et al. (2013), Kim (2016) and Hu et al. (2017). The previous works are more like for offline processing of the faults, as the demand on the hardware, e.g. storage space and calculation speed, is too high and the calculation can only be carried out on some computation servers. However, as indicated in Attoh-Okine (2014), timely processing the data and providing instantly the fault detection results are critical for decision making by operators in case of failures of critical infrastructures. Some interesting work have been proposed in recently published papers. Typicality and Eccentricity Data Analytics is adopted in Bezerra et al. (2016) for fault detection in an industrial process in an unsupervised manner. An incremental clustering procedure is proposed in Lemos et al. (2013) for adaptively training fuzzy classifiers for diagnosis. Costa et al. (2016) propose an unsupervised and online learning fuzzy rules framework, which can detect and adapt to concept drift and concept evolution. The previous works show satisfactory results for online fault detection/diagnosis with limited data, but they may not be as suitable for large amount of data in case of between-class imbalance, as considered in this work. Reducing the data size is one feasible way for on-board fault detection, as the computation time of data-driven models may decrease dramatically as the data size reduces (Fine and Katya, 2001). The challenge is how to select the reduced dataset from all the available data.

Besides the high computation time and high demand on storage space, the following characteristics of the large amount of data also challenge data-driven methods for fault detection.

1) As the system, e.g. the High-Speed Train (HST), is highly reliable (Liu et al., 2017), the majority of the collected data are of normal operation conditions (majority class), while only a small part concern faulty condition (minority class). Fault detection methods should, then, be able to

perform well with (highly) imbalanced data where the data on the situation of interest, i.e. faulty conditions of a transportation system, occupies only a small fraction of the collected data, because the performance of the conventional data-driven methods is normally not satisfying with imbalanced data.

2) The collected data contain complicated information about the transportation system, but for a specific objective, e.g. fault detection of a specific equipment, these data may not all be informative. Then, selecting useful feature variables and data points for building an efficient and effective data-driven model is very important (Li et al., 2013).

3) The collected data may contain noise and outliers, which can contaminate the data-driven models (Jagadish et al., 2014). These noise and outliers may come from the sensors, human errors, incompleteness of information, etc. (Thaduri et al., 2015). They should be eliminated in the preprocessing part or the data driven methods must be noise and outliers tolerant.

4) The data points for training a model may exhibit different importances for the model's performance (Zhang et al., 2014); thus, they also need to be differentialized in the model.

Considering the previous challenges, in this work, we propose a novel methodology to reduce the influence of noise and outliers, decrease the data size, handle imbalanced data and reflect the different importances of the data points. A real case study is considered concerning a braking system in a HST. Although numerous data-driven methods are available in the literature, e.g. support vector machines (You et al., 2014), artificial neural networks (Tisan and Chin, 2016), fuzzy rule-based systems (Antonelli et al., 2017), they can not be integrated directly with large amount of data to fulfill the diverse objectives. A *k* Nearest Neighbors-based Fuzzy Support Vector Machine (KNN-FSVM) is proposed in this paper. The reverse nearest neighbor method is used to detect and eliminate the noise and outliers (Radovanović et al., 2015). Data points having less opportunities to be among the *k* nearest neighbors of other data points from the same class are judged as outliers and noise. The importances of different data points are considered in a Fuzzy Support Vector Machine (FSVM) model by assigning different fuzzy membership values (costs for the estimation error) in the objective function: the higher the importance of a data point, the larger its assigned cost for the estimation error of this data point. The objective for the training of the FSVM model is to minimize the total cost of the estimation error on the whole training dataset. By assigning smaller cost to the outliers and noisy data points, the method itself has also the capability to reduce their influence on the classification hyperplane. Some FSVM approaches have been proposed for tackling imbalanced data (Batuwita and Palade, 2010; Fan et al., 2017; Lin and Wang, 2004). The strategy is to assign relatively larger costs to estimation errors on data points in minority class than on those in the majority class. Previous methods proposed to do so are not scalable with large amount of data, as the assigned costs are always definitely positive. The proposed KNN-FSVM can effectively reduce the data size by selecting the borderline data points and assigning zero cost to the other data points, as in FSVM, the borderline data points are more important than the others in defining the classification hyperplane. A strategy is also proposed to calculate the fuzzy membership values of the selected borderline data points considering their positions and local distributions.

To verify the effectiveness of the proposed method, comparisons with popular benchmark methods on five public imbalanced datasets are carried out. And a real case study concerns fault detection with real data collected from a braking system in a HST. The data contains 43 features related to the target problem. But not all of them are informative and outliers exist. The experiment results show the effectiveness and efficiency of the proposed methodology.

The rest of the paper is structured as follows. Section 2 presents the concept of FSVM and, then, details of the proposed method are given. The proposed method is, then, verified in Section 3 by application to several public datasets and a real case study of fault detection of the braking system in a HST. Some conclusions are drawn in Section 4.

2. A KNN-based FSVM approach

Multiple challenges need to be solved for fault detection with large amount of data, including the computational burden, imbalanced data, noise and outliers, etc. SVM is an appropriate choice as its solution depends only on a small number of training data points, i.e. the so-called support vectors. In combination with fuzzy theory, we can assign small costs to estimation error on less important data points and null costs to those on noise and outliers. In addition, if the potential support vectors can be justified before training the model, the computational burden of the method can be further reduced.

In this Section, we first review the FSVM method and, then, give details of the proposed methodology for calculating costs of estimation errors (i.e. fuzzy membership values in FSVM) on different data points.

2.1 Fuzzy support vector machine

Considering a binary classification problem represented by a dataset $\{(\boldsymbol{x}_i, y_i)|i = 1, 2, \dots, N\}$ where $\boldsymbol{x}_i \in \Re^n$ being an n-dimensional input vector and $y_i \in \{-1, +1\}$ being the class label, FSVM aims at finding the classification hyperplane in the feature space characterized by $\varphi(\boldsymbol{x})$ that separates the data points into two classes. The classification hyperplane is represented by

$$\langle \boldsymbol{\omega}, \boldsymbol{x} \rangle + b = 0, \tag{1}$$

and the best classification hyperplane is that with maximal margin and is found by solving the following maximal-margin optimization problem:

$$\begin{aligned} &\min \tfrac{1}{2}\|\boldsymbol{\omega}\|^2 \\ &s.t.\, y_i * (\langle \boldsymbol{\omega}, \boldsymbol{x}_i \rangle + b) \geq 1, with\ i = 1, 2, \dots, N \end{aligned}. \tag{2}$$

However, the datasets can rarely be linearly separable, even if they are mapped into a high-dimensional feature space by $\varphi(\boldsymbol{x})$. Slack variables are introduced into the optimization problem, which becomes

$$\begin{aligned} &\min \tfrac{1}{2}\|\boldsymbol{\omega}\|^2 + C\sum_{i=1}^{N} m_i \varepsilon_i \\ &s.t.\, y_i * (\langle \boldsymbol{\omega}, \boldsymbol{x}_i \rangle + b) \geq 1 - \varepsilon_i \\ &\varepsilon_i \geq 0, with\ i = 1, 2, \dots, N \end{aligned}, \tag{3}$$

with ε_i being the slack variable representing the estimation error, C being the tradeoff between the maximal margin and the minimal total cost for estimation errors on all the training data points and m_i being the fuzzy membership values reflecting their importances for their own class. The higher the fuzzy membership values, the more important the corresponding data points. The smaller the fuzzy membership values, the smaller the effect of the corresponding data points on the optimal classification hyperplane.

From another viewpoint, if we consider C as the cost assigned for an estimation error, then, the misclassification cost for data point $(\boldsymbol{x}_i, y_i)$ is $m_i C$. Therefore, FSVM can find a more robust hyperplane by maximizing the margin and allowing some misclassification of less important data points such as outliers and noise.

To solve the optimization problem of FSVM in Equation (3), its dual Lagrange problem is constructed as follows:

$$\begin{aligned} &\max \sum_{i=1}^{N} \alpha_i - \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} \alpha_i \alpha_j y_i y_j k(\boldsymbol{x}_i, \boldsymbol{x}_j) \\ &\quad s.t. \sum_{i=1}^{N} \alpha_i y_i = 0 \; and \; 0 \le \alpha_i \le m_i C \end{aligned}, \tag{4}$$

with α_i being the Lagrange multiplier and $k(\boldsymbol{x}_i, \boldsymbol{x}_j)$ being the inner product of the feature vectors of $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ in the feature space, i.e. $\langle \varphi(\boldsymbol{x}_i), \varphi(\boldsymbol{x}_j) \rangle$.

By solving Equation (4) with its Karush-Kuhn-Tucker (KTT) conditions, the classification hyperplane is represented as

$$\sum_{i=1}^{N} \alpha_i y_i k(\boldsymbol{x}_i, \boldsymbol{x}) + b = 0. \tag{5}$$

Finally, the decision function of FSVM is given by

$$f(\boldsymbol{x}) = sign(\sum_{i=1}^{N} \alpha_i y_i k(\boldsymbol{x}_i, \boldsymbol{x}) + b). \tag{6}$$

2.2 KNN-based fuzzy membership value calculation

The most important part of FSVM is the calculation of the fuzzy membership values m_i. The state-of-art methods are not scalable with large amount of data. Thus, a KNN-based method is proposed in this paper for fuzzy membership values calculation. The key idea is to assign null membership values to the data points far from the borderline and assign positive values only to the borderline data points, since the classification hyperplane for a SVM model is highly dependent on the borderline data points. Furthermore, the borderline data points are given different values reflecting their corresponding importance.

2.2.1 Borderline data points detection

SVM solves nonlinear classification problems by mapping the original data into a high-dimensional feature space where the problem becomes linear (Huang et al., 2015). As shown in Figure 1, The optimal hyperplane position is constrained largely by the support vectors (Li et al., 2016). The potential support vectors are very probably the data points lying on the border of different classes, i.e. the borderline data points. The computation time of a SVM increases exponentially with the number of data points. Thus, by selecting only the borderline data points to train a SVM model, one can reduce the computational burden and the demand on storage space. FSVM, as a type of SVM, inherits such capabilities.

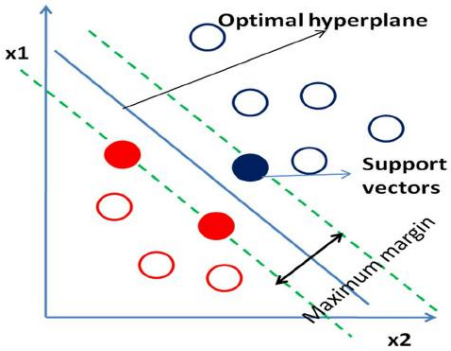

Fig. 1. Illustration of SVM in a bi-dimensional space (Chauhan et al., 2014).

Reverse KNN is used to detect and eliminate the outliers. Data points that are not among the k nearest neighbors of any data point from its own class are judged as outliers and, then, eliminated. The most common and simple way to determine borderline data points is based on KNN (Choi and

Rokett, 2002). The idea is that the data points having their *k* nearest neighbors from different classes are the borderline data points. And the other data points having their *k* nearest neighbors from only one class (not necessarily its own class) are neglected and deleted from the training data set.

2.2.2 Fuzzy membership value calculation

Only the borderline data points selected as in Section 2.2.1 are used for training a FSVM model. The method for calculating their fuzzy membership values (the costs for their estimation errors) is detailed in this Section.

Since the data set is imbalanced and normally the minority class is of great interest for the practitioners, the fuzzy membership values of the data points from the minority class should be larger than those of the data points from the majority class (Elkan, 2001). Let m_i^+ be the fuzzy membership value of a data point $\boldsymbol{x}_i^+$ from the minority class, while m_i^- being that of a data points $\boldsymbol{x}_i^-$ from the majority class: a fuzzy membership function is defined, as in Batuwita and Palade (2010), as

$$\begin{aligned} m_i^+ &= r^+ g(\boldsymbol{x}_i^+) \\ m_i^- &= r^- g(\boldsymbol{x}_i^-)' \end{aligned} \tag{1}$$

with $g(\boldsymbol{x}_i)$ being a function generating a value between 0 and 1 reflecting the importance of $\boldsymbol{x}_i$ in its own classes. The values of r^+ and r^- reflect the class imbalance and $r^+ > r^-$. If $r^+ = 1$, then, $r^- = r$, with $r < 1$. The function $g(\boldsymbol{x}_i)$ is expressed as a monotone decaying function of a distance measure

$$g(\boldsymbol{x}_i) = \frac{2}{1+\exp(\beta d_i)}, \tag{2}$$

with β being the steepness of the decay and d_i being a function of two parts, d_i^1 and d_i^2:

$$d_i = h(d_i^1, d_i^2). \tag{3}$$

The smaller the distance d_i is, the more important the data point is and the larger its fuzzy membership value is.

The data point having more nearest neighbors from its own class is more important and its fuzzy membership value should be relatively larger (Chawla et al., 2003). Thus, the first part d_i^1 of Equation (3) reflects the closeness of a data point to the other data points from its own class. Suppose N_i^+ data points out of the k nearest neighbors of a data point $\boldsymbol{x}_i$, are from the minority class and the rest $N_i^- = k - N_i^+$ nearest neighbors from the majority class, the value of d_i^1 is calculated as follows,

$$d_i^1 = \begin{cases} \frac{eN_i^+}{N_i^-}, \ for\ \boldsymbol{x}_i\ from\ majority\ class \\ \frac{N_i^-}{eN_i^+}, \ for\ \boldsymbol{x}_i\ from\ minority\ class \end{cases}, \tag{4}$$

where e is a balancing factor equal or larger than 1. For example, consider a data point $\boldsymbol{x}_1$ from the minority class and a data point $\boldsymbol{x}_2$ from the majority class have both only half of the k nearest neighbor from the minority class. The distance measure in Equation (4) shows that $d_1^1 \geq d_2^1$, because the same situation in KNN occurs more frequently to be real for a data point from the majority class than for a data point from the minority class with imbalanced class distribution. And the correct classification of $\boldsymbol{x}_1$ is more important than $\boldsymbol{x}_2$.

Even if two data points from the same class have the same value from Equation (4), their importance for the estimation accuracy can still be different. As shown in Figure 2, the 6 nearest neighbors of

these two data points $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ from the minority class contain both 3 data points from the minority class. Equation (2) gives the same value for these two data points, while their local distributions of the nearest neighbors show that it is more important to classify correctly $\boldsymbol{x}_1$ than $\boldsymbol{x}_2$, as the nearest neighbors of $\boldsymbol{x}_1$ from the two classes are located on opposite sides of $\boldsymbol{x}_1$.

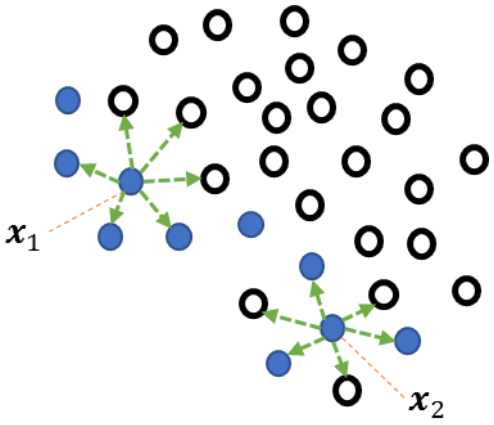

Fig. 2. Illustration of different local distributions of nearest neighbors. (empty and solid circles represent respectively data points from two classes)

Thus, the second part d_i^2 of Equation (3) depends on the local distribution of the nearest neighbors of one data point from different classes, including their separability and alignment. In order to express this difference, normal vectors are introduced. Suppose the k nearest neighbors of a data point $\boldsymbol{x}_i$ are noted as $\boldsymbol{x}_{i,j}, j = 1,2,\dots,k$, and the first k_1 neighbors are from the minority class and the rest are from the majority class: the normal vectors of the neighbors from the two classes are calculated as

$$\begin{aligned} \boldsymbol{v}_i^+ &= \frac{1}{k_1}\sum_{j=1}^{k_1}\frac{\boldsymbol{x}_{i,j}-\boldsymbol{x}_i}{\|\boldsymbol{x}_{i,j}-\boldsymbol{x}_i\|} \\ \boldsymbol{v}_i^- &= \frac{1}{k-k_1}\sum_{j=k_1+1}^{k}\frac{\boldsymbol{x}_{i,j}-\boldsymbol{x}_i}{\|\boldsymbol{x}_{i,j}-\boldsymbol{x}_i\|}, \end{aligned} \tag{5}$$

where $\|\blacksquare\|$ represents the norm of a vector. Then, the value d_i^2 is calculated as

$$d_i^2 = \frac{\langle\frac{\boldsymbol{v}_i^+}{\|\boldsymbol{v}_i^+\|},\frac{\boldsymbol{v}_i^-}{\|\boldsymbol{v}_i^-\|}\rangle+1}{\|\boldsymbol{v}_i^+\|_*\|\boldsymbol{v}_i^-\|+\varepsilon}, \tag{6}$$

with $\langle\blacksquare,\blacksquare\rangle$ being the inner product, ε being a small positive value. Thus, d_i^2 is always a positive value. The inner product of two normal vectors is the cosine of the angle between the two normal vectors. With the same norm of the two normal vectors, i.e. $\|\boldsymbol{v}_i^+\| = \|\boldsymbol{v}_i^-\|$, the more separable are the data points from the two classes, the smaller is the value d_i^2. With the same angle between these two normal vectors, the closer the nearest neighbors from the same class, the larger the norm of their normal vector is and the smaller the value d_i^2 is.

Finally, the distance d_i in Equation (2) is expressed as

$$d_i = \gamma d_{i,n}^1 + (1-\gamma)d_{i,n}^2, \tag{7}$$

with γ being a parameter between 0 and 1 regularizing the importance of the two parts and $d_{i,n}^1$, $d_{i,n}^2$ being the normalized value of d_i^1, d_i^2 given by Equations (4) and (6).

The unknown parameters of the proposed method include the regularization term C in FSVM, the parameters of the selected kernel method, the steepness parameter β in Equation (2), the balancing factor e in Equation (4) and the parameter γ in Equation (7).

3. Fault detection of the braking system in a HST

Experimental results are reported in this Section with comparisons to several benchmark methods including FSVM based on distance to actual hyperplane (FSVM-CIL) in Batuwita and Palade (2010), FSVM based on class certainty of samples (entropy-FSVM) in Fan et al. (2017), SVM with Synthetic Oversampling Technique (SVM-SMOTE) in Chawla et al. (2002) and SVM with Random UnderSampling (SVM-RUS) in Batista et al. (2004). These benchmark methods are popular for imbalanced data and more details of these methods can be found in the corresponding references. Five public datasets from KEEL datasets (Alcala-Fdez et al., 2009) are used to test the generalization ability of the proposed method and, then, the real data collected from a braking system of a HST is considered for fault detection. The objective is to show the effectiveness of the proposed method, i.e. KNN-FSVM and to justify its efficiency. The experiments are carried out on a server with 4 Intel Xeon CPU E5-4667 V4 @ 2.20 GHz (Broadwell) and 512 Gb of RAM.

The experiment process is shown in Figure 3. The difference between the training process of the benchmark methods and that of KNN-FSVM is that, for training a KNN-FSVM model, the reverse nearest neighbors method is used as the first step to detect and eliminate the outlier data points and, then, the borderline data points selected by KNN from the cleaned training dataset, form the final training dataset. Data points that have all the k nearest neighbors from its own class are deleted from the training dataset as they are not part of the borderline dataset. These two original steps, i.e. outlier elimination and borderline data points selection are not integrated in the benchmark methods.

The experiment follows an inner-outer-loop five-fold cross validation. The whole dataset is divided into five subsets. Each time one subset is selected as test dataset and the rest form the training dataset. The experiment is repeated five times and the reported results represent the average accuracies. Each time, the five-fold cross validation method is adopted to tune the hyperparameters of all the methods.

Initialization: The original dataset $\boldsymbol{S} = [\boldsymbol{x}_i, y_i], i = 1, 2, \dots, N$ is randomly divided into five subsets $\boldsymbol{S}_j, j = 1, 2, \dots, 5$;
$\boldsymbol{S_{Tr}}$ represents the training dataset and $\boldsymbol{S_{Te}}$ represents the test dataset;
There are totally k models for comparison;
$AUC(m, j), m = 1, 2, \dots, k \; and \; j = 1, 2, \dots, 5$ is the m-th model's accuracy on the j-th test dataset.

Begin:

For j = 1:5

$\boldsymbol{S_{Te}} = \boldsymbol{S}_j$ and $\boldsymbol{S_{Tr}} = \boldsymbol{S} - \boldsymbol{S_{Te}}$;
Train the models on $\boldsymbol{S_{Tr}}$ using 5-fold cross validation;
Test the trained models with $\boldsymbol{S_{Te}}$;
Calculate $AUC(m, j)$ of the test results.

End

The result of the m-th model is defined as $mean(AUC(m, :))$.

End

Fig. 3. Illustration of the experiment process.

Considering the imbalanced data, the Area Under the ROC (Receiver Operating Characteristics), or simply AUC, is taken as the accuracy metric in this paper (Huang and Ling, 2005). AUC is an informative single metric for accuracy and it is used in many works on imbalanced data (Gao et al., 2014; Barua et al., 2014). Readers can refer to the corresponding references for more details.

3.1 Validation of the proposed method

The characteristics of the five public datasets are shown in Table 1. The datasets are characterized by different imbalance ratios. The number of the borderline data points selected by KNN-FSVM, as shown in the table is much smaller than the original data size and, thus, the computation burden is largely reduced compared to that of the benchmark methods using the original training datasets.

Table 1. Characteristics of the public datasets and number of selected borderline data points.

Dataset	Original data size	Number of input variables	Imbalance ratio	Number of selected borderline points
ecoli3	336	7	8.6	84
ecoli-0-6-7_vs_5	220	6	10	48
ecoli4	336	7	15.8	27
ecoli-0-1_vs_2-3-5	244	7	9.17	48
ecoli-0-1_vs_5	240	6	11	24

The results given by different methods in terms of AUC are reported in Table 2 and the best results are marked bolded. The values in the parentheses after the AUC values are the ranks among all methods. The mean ranks of all methods are shown in the last column. We can see that KNN-FSVM gives the second-best results for the considered public datasets. The accuracy differences between KNN-FSVM and FSVM-hyp-lin are not significant with respect to the AUC values. Wilcoxon signed rank test (Fehr and Gächter, 2002) which is a famous and efficient pairwise comparison method considering both the rank and performance difference, shows that KNN-FSVM and FSVM-hyp-lin which obtains the highest mean rank give comparable results. KNN-FSVM obtains a better rank than SVM-SMOTE, SVM-RUS and entropy-SVM in the experiment. This fact proves that the borderline data points are more important than other data points for defining the classification hyperplane of the SVM models. SVM-RUS gives the worst results in the experiment because of the information loss during the random undersampling of the majority class.

Thus, from Tables 1 and 2, we can conclude that the proposed method gives satisfactory accuracy for the selected public datasets with a much lower computational burden.

Table 2. Accuracy on public datasets with respect to AUC.

	ecoli3	ecoli-0-6-7_vs_5	ecoli4	ecoli-0-1_vs_2-3-5	ecoli-0-1_vs_5	Average rank
FSVM-hyp-lin	**0.9267 (1)**	**0.9612 (1)**	0.9619 (3)	**0.9166 (1)**	0.9556 (2)	1.6
entropy-FSVM	0.9116 (3)	0.9125 (3)	**0.9810 (1)**	0.9145 (2)	0.9341 (3)	2.4
SVM-SMOTE	0.8974 (4)	0.8925 (4)	0.9560 (4)	0.8850 (4)	0.9136 (5)	4.2
SVM-RUS	0.8902 (5)	0.8750 (5)	0.9544 (5)	0.8759 (5)	0.9159 (4)	4.8
KNN-FSVM	0.9224 (2)	0.9337 (2)	0.9722 (2)	0.9070 (3)	**0.9659 (1)**	2

3.2 The case study of the braking system of a HST

The braking system, controlling the deceleration efficiency in case of emergency and normal stops, is a safety-critical system in HST. Various sensors are integrated for monitoring its condition. These variables can be divided into internal variables and external variables. Internal variables are the ones monitoring the condition of a braking system, such as internal temperature, brake effort achieved, suspension pressure, odometer, stop cock closed, etc. External variables are the ones related to the whole train, such as GPS position, traction, speed, line current, line voltage, vehicle driving hours, etc. For one sensor, even if the data are monitored and recorded every hour, there are more than 50 000 values within one year. The amount of data can reach several terabytes and petabytes if the monitoring frequency increases. These data may be used to train data-driven models, which can serve as auxiliary on-board alarming systems in a HST. However, the computer in a HST can not handle such huge data timely and it is necessary to reduce the data size while keeping a comparable accuracy.

In this experiment, the data collected from a braking system of a HST within one year are considered. The objective is fault detection and different types of fault are not distinguished in this experiment. Thus, the output of the collected data is -1 (faulty) and +1 (healthy). There are 43 feature variables in total, related to the system and only 149 out of the available 28996 data points correspond to faulty conditions. The imbalanced ratio reaches 184.6. For confidential concerns, the variables names can not be explicitly given in the manuscript. The monitored variables can be numeric or nominal. As FSVM can only handle numeric data, the nominal data are converted into numeric data.

Between-class separability is used as the criterion for selecting useful feature variables. The between-class separability of the healthy and faulty data with respect to the j-th feature variable is defined as follows:

$$\lambda_j = \frac{(m_{j,+} - m_{j,-})^2}{\delta_{j,+}^2 + \delta_{j,-}^2}, \tag{8}$$

where $m_{j,+}$ and $\delta_{j,+}^2$ are the mean and variance of the values of the j-th feature variable of the faulty data, and $m_{j,-}$ and $\delta_{j,-}^2$ are the corresponding mean and variance of the healthy data. The features giving a between-class separability lower than a predefined threshold Th_{fs} are eliminated.

Only the variables with a separability value larger than $Th_{fs} = 0.1$ are selected. Then, the noise and outliers detected by the reverse 22 nearest-neighbors method are deleted from the training dataset. Totally, 601 data points are selected as the borderline data points of which 140 are on the faulty condition. Nearly all the data points on the faulty condition have been kept in the borderline dataset and a large part of the data points on normal condition have been deleted as they are far from the borderline.

Table 4 shows the detailed experiment results including the average AUC values, numbers of negative and positive data points used in the final training datasets, the computation times and the storage requests for calculation. One can observe that the proposed method KNN-FSVM gives slightly worse results than FSVM-hyp-lin and entropy-FSVM, but it uses much less computation time and storage: 0.88s and 0.003 Gb for KNN-FSVM vs +1200s and 5.1 Gb for FSVM-hyp-lin and entropy-SVM. SVM-RUS can also reduce dramatically the data size, but it gives relatively worse results because the randomly selected data points on normal condition may not represent the original data distribution. The demand on computation power and storage is the largest for SVM-SMOTE. And SVM-SMOTE

does not give satisfactory results in the case study, as it may oversample also the noise and outliers in the minority class.

Table 4. Results on the real dataset from the braking system of a HST.

	AUC value	# of negative training data points	# of positive training data points	Computation time (s)	Storage request (Gb)
FSVM-hyp-lin	0.9468	25953	143	1211.11	5.1
entropy-FSVM	0.9346	25953	143	1246.52	5.1
SVM-SMOTE	0.9063	25953	25953	3925.09	21.4
SVM-RUS	0.8348	143	143	0.19	0.001
KNN-FSVM	0.9244	461	140	0.88	0.009

The performance of KNN-FSVM can be highly influenced by the k value in the reverse KNN for outlier elimination. For different k values, Figure 4 shows the AUC values of KNN-FSVM for the case of $\boldsymbol{S_{Te}} = \boldsymbol{S_1}$ and $\boldsymbol{S_{Tr}} = \boldsymbol{S} - \boldsymbol{S_{Te}}$ as in Figure 3. It is observed that the AUC values change with the k value. For a small k value, many data points can be judged as outliers and the selected borderline data points become sparse, so that the classification hyperplane is no longer precise. For a large k value, some outliers may be retained and treated as borderline data points. The estimated separation hyperplane is biased to the outliers. Thus, small or a large k values can both influence the performance of KNN-FSVM, which means that during the training, the k value needs to be carefully tuned. From Figure 4, a small k value is more dangerous than a large k value, in this case study: this can be explained by the fact that the outliers are not numerous in the collected data.

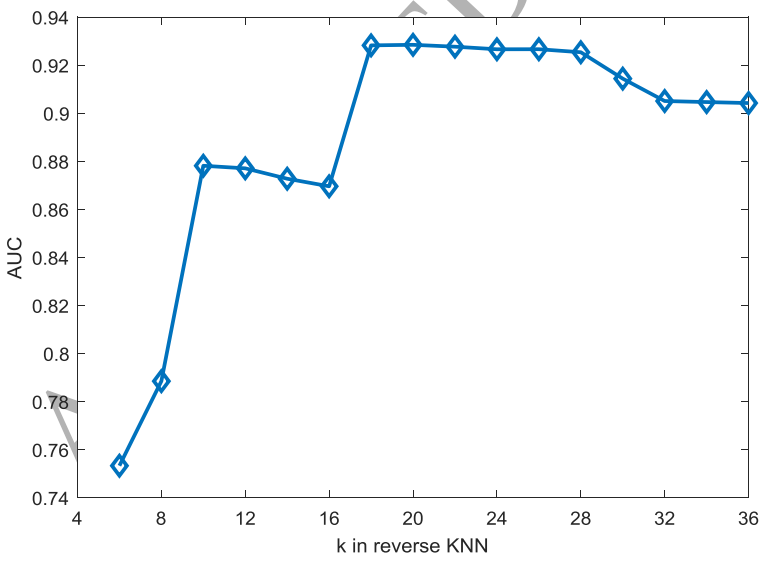

Fig. 4 AUC of KNN-FSVM with different k values in reverse KNN for outlier detection.

4. Conclusion

The large amount of data paradigm provides opportunities but also poses challenges for fault detection in transportation systems by data-driven methods. In this paper, we focus especially on

imbalanced data size, noise and outliers, different importances of data points in the fault detection model and large computational burden. A KNN-based FSVM (KNN-FSVM) method is proposed for reducing data size and reflecting the importance of different training data points. The reverse nearest neighbor method is used to detect outliers in the training dataset, to be deleted. Considering the SVM property that the classification hyperplane lies more likely in the overlapping (borderline) region, we use KNN to justify the borderline points which are, then, the only ones kept in the training dataset, with significant computational burden reduction. A new strategy is proposed for calculating the fuzzy membership values in FSVM, considering the local distribution of the selected borderline data points. The proposed method is verified successfully on several public imbalanced datasets. The case study concerning a braking system of a HST shows that the proposed method can reduce greatly the computational burden and storage demand for calculation, while keeping a satisfactory classification accuracy. Note that one challenge of the proposed framework is to tune the parameters, especially the k value in KNN. More data points are judged as borderline data points with a larger k value. Thus, a larger k may make the model more robust, but also more time-consuming in processing the data. A good trade-off between robustness and efficiency should be achieved, i.e. a proper stop criterion is necessary for tuning the k value. One possible way is to increase monotonically the k value during the parameters tuning process and to stop this process when the accuracy improvement is no longer significant to the fault detection performance.

The classification problem may be more difficult if the faulty data are distributed in several blocks and the least one can do for this situation is to train multiple one-against-all classifiers. In this paper, we propose a FSVM framework for binary classification to train a supervised SVM model with borderline data points selected from both healthy and faulty data. Theoretically, by selecting proper SV candidates from different faulty blocks and the healthy data, one FSVM is capable of handling the case where the faulty data are located in disjunct blocks. It is important to note that the method is influenced by the faulty data size. The proposed method may not fit the case when there exists one block with only one or two data points as they may be treated as noise and outliers. It would be interesting to test the proposed method for the situation.

References

Alcala-Fdez, J., Sanchez, L., Garcia, S., Del Jesus, M.J., Ventura, S., Garrell, J.M., Otero, J., Romero, C., Bacardit, J., Rivas, V.M. and Fernandez, J.C., 2009. KEEL: a software tool to assess evolutionary algorithms for data mining problems. Soft Computing, 13(3), pp.307-318.

Antonelli, Michela, Dario Bernardo, Hani Hagras, and Francesco Marcelloni. "Multiobjective Evolutionary Optimization of Type-2 Fuzzy Rule-Based Systems for Financial Data Classification." IEEE Transactions on Fuzzy Systems 25, no. 2 (2017): 249-264.

Attoh-Okine, Nii. "Big data challenges in railway engineering." In *Big Data (Big Data), 2014 IEEE International Conference on*, pp. 7-9. IEEE, 2014.

Barua, Sukarna, Md Monirul Islam, Xin Yao, and Kazuyuki Murase. "MWMOTE--majority weighted minority oversampling technique for imbalanced data set learning." *IEEE Transactions on Knowledge and Data Engineering* 26, no. 2 (2014): 405-425.

Batista, Gustavo EAPA, Ronaldo C. Prati, and Maria Carolina Monard. "A study of the behavior of several methods for balancing machine learning training data." *ACM Sigkdd Explorations Newsletter* 6, no. 1 (2004): 20-29.

Batuwita, Rukshan, and Vasile Palade. "FSVM-CIL: fuzzy support vector machines for class imbalance learning." *IEEE Transactions on Fuzzy Systems* 18, no. 3 (2010): 558-571.

Bezerra, Clauber Gomes, et al. "An evolving approach to unsupervised and Real-Time fault detection in industrial processes." *Expert Systems with Applications* 63(2016):134-144.

Chawla, Nitesh V., Kevin W. Bowyer, Lawrence O. Hall, and W. Philip Kegelmeyer. "SMOTE: synthetic minority over-sampling technique." *Journal of artificial intelligence research* 16 (2002): 321-357.

Chauhan, Arun, Devesh Chauhan, and Chittaranjan Rout. "Role of gist and phog features in computer-aided diagnosis of tuberculosis without segmentation." *PloS one* 9, no. 11 (2014): e112980.

Chawla, Nitesh V., Aleksandar Lazarevic, Lawrence O. Hall, and Kevin W. Bowyer. "SMOTEBoost: Improving prediction of the minority class in boosting." In *European Conference on Principles of Data Mining and Knowledge Discovery*, pp. 107-119. Springer Berlin Heidelberg, 2003.

Choi, Se-Ho, and Peter Rockett. "The training of neural classifiers with condensed datasets." *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* 32, no. 2 (2002): 202-206.

Costa, Bruno Sielly Jales, et al. "Unsupervised classification of data streams based on Typicality and Eccentricity Data Analytics." *IEEE International Conference on Fuzzy Systems* IEEE, 2016.

Elkan, Charles. "The foundations of cost-sensitive learning." In *International joint conference on artificial intelligence*, vol. 17, no. 1, pp. 973-978. Lawrence Erlbaum Associates Ltd, 2001.

Fan, Qi, Zhe Wang, Dongdong Li, Daqi Gao, and Hongyuan Zha. "Entropy-based fuzzy support vector machine for imbalanced datasets." *Knowledge-Based Systems* 115 (2017): 87-99.

Fehr, Ernst, and Simon Gächter. "Altruistic punishment in humans." Nature 415, no. 6868 (2002): 137-140.

Fine, Shai, and Katya Scheinberg. "Efficient SVM training using low-rank kernel representations." *Journal of Machine Learning Research* 2, no. Dec (2001): 243-264.

Gao, Ming, Xia Hong, and Chris J. Harris. "Construction of neurofuzzy models for imbalanced data classification." *IEEE Transactions on Fuzzy Systems* 22, no. 6 (2014): 1472-1488.

Guzinski, Jaroslaw, Haitham Abu-Rub, Marc Diguet, Zbigniew Krzeminski, and Arkadiusz Lewicki. "Speed and load torque observer application in high-speed train electric drive." *IEEE Transactions on Industrial Electronics* 57, no. 2 (2010): 565-574.

Hu, Hexuan, Tang, Bo, Xue-jiao Gong, Wei Wei, and Huihui Wang. "Intelligent fault diagnosis of the high-speed train with big data based on deep neural networks." *IEEE Transactions on Industrial Informatics* (2017).

Huang, Jin, and Charles X. Ling. "Using AUC and accuracy in evaluating learning algorithms." *IEEE Transactions on knowledge and Data Engineering* 17, no. 3 (2005): 299-310.

Huang, Su-Dan, Guang-Zhong Cao, Zheng-You He, J. F. Pan, Ji-An Duan, and Qing-Quan Qian. "Nonlinear modeling of the inverse force function for the planar switched reluctance motor using sparse least squares support vector machines." *IEEE Transactions on Industrial Informatics* 11, no. 3 (2015): 591-600.

Jagadish, H. V., Johannes Gehrke, Alexandros Labrinidis, Yannis Papakonstantinou, Jignesh M. Patel, Raghu Ramakrishnan, and Cyrus Shahabi. "Big data and its technical challenges." *Communications of the ACM* 57, no. 7 (2014): 86-94.

Jiang, Dong Nian, and L. I. Wei. "Fault detection method based on data-driven residual evaluation strategy." Control & Decision 32.7(2017):1181-1188.

Kecman, Pavle, and Rob MP Goverde. "Online data-driven adaptive prediction of train event times." *IEEE Transactions on Intelligent Transportation Systems* 16, no. 1 (2015): 465-474.

Kim, Seongdo. "Forecasting short-term air passenger demand using big data from search engine queries." *Automation in Construction* 70 (2016): 98-108.

Lemos, Andre, W. Caminhas, and F. Gomide. "Adaptive fault detection and diagnosis using an evolving fuzzy classifier." *Information Sciences* 220.1(2013):64-85.

Li, Hongfei, Buyue Qian, Dhaivat Parikh, and Arun Hampapur. "Alarm prediction in large-scale sensor networks—A case study in railroad." In *Big Data, 2013 IEEE International Conference on*, pp. 7-14. IEEE, 2013.

Li, Xiaojie, Jiancheng Lv, and Zhang Yi. "An Efficient Representation-Based Method for Boundary Point and Outlier Detection." *IEEE Transactions on Neural Networks and Learning Systems* (2016).

Li, Zhiguo, and Qing He. "Prediction of Railcar Remaining Useful Life by Multiple Data Source Fusion." *IEEE Transactions on Intelligent Transportation Systems* 16, no. 4 (2015): 2226-2235.

Lin, Chun-fu, and Wang, Sheng-de. "Training algorithms for fuzzy support vector machines with noisy data." *Pattern recognition letters* 25, no. 14 (2004): 1647-1656.

Liu, Jie, Yan-Fu Li, and Enrico Zio. "A SVM framework for fault detection of the braking system in a high speed train." *Mechanical Systems and Signal Processing* 87 (2017): 401-409.

Liu, Xiang. "Statistical Causal Analysis of Freight-Train Derailments in the United States." *Journal of Transportation Engineering, Part A: Systems* (2016): 04016007.

López, Victoria, et al. "An insight into classification with imbalanced data: Empirical results and current trends on using data intrinsic characteristics." *Information Sciences* 250.11(2013):113-141.

Radovanović, Miloš, Alexandros Nanopoulos, and Mirjana Ivanović. "Reverse nearest neighbors in unsupervised distance-based outlier detection." *IEEE transactions on knowledge and data engineering* 27, no. 5 (2015): 1369-1382.

TANAKA, Mikio. "Prospective study on the potential of big data." *Quarterly Report of RTRI* 56, no. 1 (2015): 5-9.

Thaduri, Adithya, Diego Galar, and Uday Kumar. "Railway assets: a potential domain for big data analytics." *Procedia Computer Science* 53 (2015): 457-467.

Tisan, Alin, and Jeannette Chin. "An End-User Platform for FPGA-Based Design and Rapid Prototyping of Feedforward Artificial Neural Networks With On-Chip Backpropagation Learning." IEEE Transactions on Industrial Informatics 12, no. 3 (2016): 1124-1133.

Yan, Xihui, Baigen Cai, Bin Ning, and Wei ShangGuan. "Online distributed cooperative model predictive control of energy-saving trajectory planning for multiple high-speed train movements." *Transportation Research Part C: Emerging Technologies* 69 (2016): 60-78.

You, Deyong, Xiangdong Gao, and Seiji Katayama. "Multisensor fusion system for monitoring high-power disk laser welding using support vector machine." *IEEE Transactions on Industrial Informatics* 10, no. 2 (2014): 1285-1295.

Zarembski, Allan M. "Some examples of big data in railroad engineering." In *Big Data (Big Data), 2014 IEEE International Conference on*, pp. 96-102. IEEE, 2014.

Zhang, Xiao, Enrique Onieva, Asier Perallos, Eneko Osaba, and Victor CS Lee. "Hierarchical fuzzy rule-based system optimized with genetic algorithms for short term traffic congestion prediction." *Transportation Research Part C: Emerging Technologies* 43 (2014): 127-142.

Zilko, Aurelius A., Dorota Kurowicka, and Rob MP Goverde. "Modeling railway disruption lengths with Copula Bayesian Networks." *Transportation Research Part C: Emerging Technologies* 68 (2016): 350-368.

Zio, Enrico. "Prognostics and health management of industrial equipment." *Diagnostics and prognostics of engineering systems: methods and techniques* (2012): 333-356.