


## Article

# Thyristor Aging-State-Evaluation Method Based on State Information and Tensor Domain Theory

Zhaoyu Lei <sup>1,2</sup>, Jianyi Guo <sup>2,\*</sup>, Yingfu Tian <sup>1</sup>, Jiemin Yang <sup>1</sup>, Yinwu Xiong <sup>1</sup>, Jie Zhang <sup>1</sup>, Ben Shang <sup>2</sup> and Youping Fan <sup>2,\*</sup> 

<sup>1</sup> Tianshengqiao Bureau, CSG EHV Power Transmission Company, Xingyi 210094, China; leizhaoyu@ehv.csg.cn (Z.L.); tianyingfu@ehv.csg.cn (Y.T.); xrlz@126.com (J.Y.); xiongyinwu@ehv.csg.cn (Y.X.); zhangjiequli@126.com (J.Z.)

<sup>2</sup> School of Electrical Engineering and Automation, Wuhan University, Wuhan 430072, China; 2015302540235@whu.edu.cn

\* Correspondence: 2020202070007@whu.edu.cn (J.G.); ypfan@whu.edu.cn (Y.F.); Tel.: +86-138-0497-3133 (J.G.); +86-133-9716-9986 (Y.F.)

**Abstract:** The thyristor is the key device for the converter of the ultra-high-voltage DC (UHVDC) project to realize AC–DC conversion. The reliability of thyristors is directly related to the safe operation of the UHVDC transmission system. Due to the complex operating environment of the thyristor, there are many interrelated parameters that may affect the aging state of thyristors. To extract useful information from the massive high-dimensional data and further obtain the aging state of thyristors, a supervised tensor domain classification (STDC) method based on the adaptive synthetic sampling method, the gradient-boosting decision tree, and tensor domain theory is proposed in this paper. Firstly, the algorithm applies the continuous medium theory to analogize the aging state points of the thyristor to the mass points in the continuous medium. Then, the algorithm applies the concept of the tensor domain to identify the aging state of the thyristor and to transform the original state-identification problem into the state classification surface determination of the tensor domain. Secondly, a temporal fuzzy clustering algorithm is applied to realize automatic positioning of the classification surface of each tensor sub-domain. Furthermore, to solve the problem of unbalanced sample size between aging class data and normal class data in the state-identification domain, the improved adaptive synthetic sampling algorithm is applied to preprocess the data. The gradient-boosting decision tree algorithm is applied to solve the multi-classification problem of the thyristor. Finally, the comparison between the algorithm proposed and the conventional algorithm is performed through the field-test data provided by the CSG EHV Power Transmission Company of China's Southern Power Grid. It is verified that the evaluation method proposed has higher recognition accuracy and can effectively classify the thyristor states.

**Keywords:** UHVDC project; thyristor; aging state evaluation; tensor domain theory; data analysis



**Citation:** Lei, Z.; Guo, J.; Tian, Y.; Yang, J.; Xiong, Y.; Zhang, J.; Shang, B.; Fan, Y. Thyristor Aging-State-Evaluation Method Based on State Information and Tensor Domain Theory. *Electronics* **2021**, *10*, 2700. <https://doi.org/10.3390/electronics10212700>

Academic Editor: Flavio Canavero

Received: 3 October 2021

Accepted: 4 November 2021

Published: 5 November 2021

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

The operating environment of thyristors in UHVDC transmission converters is complex, and there are a large number of interrelated parameters that can affect the aging state of the thyristors. Therefore, a large amount of high-dimensional state monitoring data is sampled for the state evaluation of the thyristors in UHVDC transmission converters [1,2]. To extract the information that mainly affects the aging state of the thyristor operation and to understand the actual aging state of the thyristor, the mathematical tools that can effectively process the high-dimensional massive data and extract relevant features need to be introduced.

In recent years, tensor theory, as a generalized form of higher-dimensional vectors and matrices [3], is nowadays gradually applied to various fields to solve engineering problems

such as massive and high-dimensional data [4], image recognition [5], industrial IoT data processing [6], and data-clustering processing [7].

The application of tensor theory in the field of state evaluation focuses on delineating the tensor domain classification surface between normal samples and samples with different degrees of aging. There are two main kinds of approaches including model-driven approaches and data-driven approaches [8–10]. The model-driven approaches are difficult to realize because the complexity of the operating characteristics, the operating environment, and the internal structure of the thyristor makes it difficult to model the thyristors accurately. On the other hand, data-driven approaches are widely applied in the field of thyristor state evaluation coupled with the rapid development of sensors. The data-driven approaches aim to obtain state information from state-monitoring data through machine-learning methods. In the learning process, it is usually assumed that the differences in the amount of sample data for all categories in the dataset are small. However, according to the thyristor operation data obtained in the field test, the dataset shows typical inter-class imbalance characteristics, i.e., the amount of aging data is extremely small compared to the monitoring data under normal states. The problem of inter-class data imbalance has a great impact on the accuracy of data-driven thyristor state evaluation and tends to lead to inefficient algorithms [11]. Therefore, the inter-class imbalance needs to be solved to obtain the interclass tensor domain classification surfaces by training effectively.

So far, the problem of inter-class imbalance in the dataset for thyristor state evaluation has not been fully developed yet. Existing algorithms in other research fields that can be applied to solve the inter-class imbalance problem are introduced and discussed as follows. The down-sampling technique balances data by reducing the amount of data in most classes [12]. However, important information may be lost, which will affect the classification accuracy. The cost-sensitive algorithm increases the loss of misclassified samples by using different cost matrices, which is to some extent superior to the conventional sampling algorithm [13]. However, the performance of its algorithm is highly influenced by its dataset and is less controllable. The oversampling technique increases the amount of data in the few classes of samples according to the sample-distribution characteristics of the dataset. The synthetic minority oversampling technique (SMOTE) algorithm focuses on synthesizing a new minority class of samples based on the interpolation between the samples and their  $k$  nearest neighbors [14]. However, the SMOTE algorithm generates the category overlap problem because the selected  $k$  nearest neighbors are not considered to be in the same category of the current sample points when a new sample is synthesized. The adaptive synthetic sampling algorithm overcomes these problems and is more efficient [15]. The state evaluation of thyristors is a multi-category problem with more than two categories of data imbalance. The improved adaptive synthetic sampling algorithm is proposed in this paper to solve the multi-category data imbalance problem and can be better applied to thyristor state evaluation.

The gradient-boosting decision tree algorithm is one of the commonly used algorithms to obtain the classification surface [16]. The algorithm is based on the integrated machine-learning technique of decision trees, which builds the model in a phased manner and achieves convergence through error correction during the training process. The classifier of the gradient-boosting decision tree learns from the partial features one at a time, rather than learning all features recursively and repeatedly. At each step, a new base model is included to correct the errors made by the previous base model. Thus, the gradient-augmentation method has the potential to provide a more accurate classifier. However, the algorithm works well for systematic recognition only in the case of inter-class balanced data and has difficulties in dealing with the inter-class imbalance in the dataset [17]. To deal with the imbalance case, the STDC algorithm is further proposed in this paper. The algorithm firstly preprocesses the data by the improved adaptive synthetic sampling algorithm to realize the data inter-class equilibrium state. Then, the gradient-boosting decision tree algorithm is applied to determine the tensor sub-domain classification surface to realize the state evaluation of thyristors.

In summary, the main contribution of this paper is as follows. (1) Based on the related mathematical knowledge of tensor theory and the continuous medium theory, the aging state point of the thyristor is analogous to the mass point in the continuous medium. Then, the aging state of the thyristor is identified through the concept of the tensor domain, and the original state-identification problem is transformed into the determination of the state of the tensor domain category classification surface. (2) Applying the temporal fuzzy clustering algorithm, the classification surface of each tensor subdomain is automatically located. (3) On the basis of tensor theory, the STDC algorithm is proposed. Firstly, the adaptive synthetic sampling algorithm is improved so that it can be applied in the multi-classification problem. Then, the data in (2) are preprocessed to balance the data samples of each category, and the gradient-boosting decision tree algorithm is applied to obtain the tensor sub-domain classification surfaces of the multi-classification problem. (4) The state classification accuracy of this paper is compared and analyzed with the  $k$  nearest neighbor (KNN) algorithm, support vector machine (SVM) algorithm, and Gaussian Naive Bayesian (GNB) algorithm. The results show that the improved STDC algorithm proposed achieves the best classification effect, and the recognition accuracy of each state reaches more than 93%, which can meet the practical engineering requirements.

## 2. A Machine-Learning Method for State Evaluation Based on Tensor Theory

### 2.1. Tensor Theory Introduction

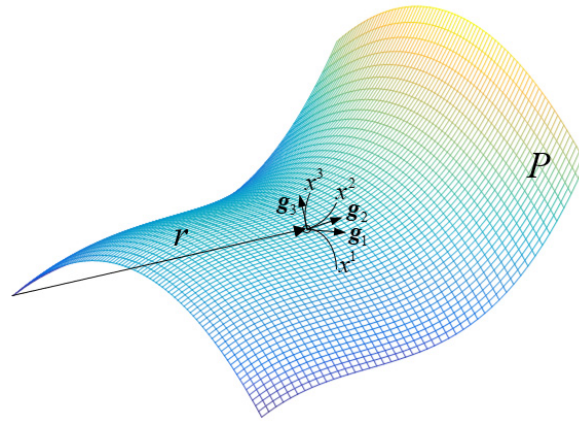
The aging state of the thyristor performance is now considered as a material point, and the thyristor aging and failure process satisfies the continuity requirement of the continuous medium concept. In addition, the aging and failure process of the thyristor shows a strong directionality according to the difference of environmental parameters and the length of service time. Therefore, it is proposed that all the state points corresponding to the whole failure process of the thyristor can be considered as a continuous medium for the following research. For a power electronic or mechanical component, the space formed by all the monitored parameters corresponding to the period from the initial perfect state to the complete failure state is called the tensor domain.

The tensor domain state-evaluation theory is based on multi-source heterogeneous monitoring information and spatial classification. The operational aging state of a component or a system can be characterized by inputting real-time information. Furthermore, the aging process and velocity change of any aging state of a component or a system can also be characterized by the velocity of the material point (state point).

In the continuous medium, different material points occupy different spatial positions at the same moment, which describes the spatial positions of the material points. At different moments, the same material point can also occupy different spatial positions, which describes the law of the motion of the material points. Additionally, the law of the motion of all the material points constitutes the law of the motion of the continuous medium, which is usually described by Eulerian coordinates as  $x^i$ . It can describe the station background of the motion of the continuous medium. The spatial position of an arbitrary material point can be represented by the vector path coordinate  $r(x^1, x^2, x^3)$ .  $g_i$  is the covariant basis vector, which is the position function of the material point in space and can be expressed as  $g_i = \partial(x^1, x^2, x^3)/\partial x^i$ .

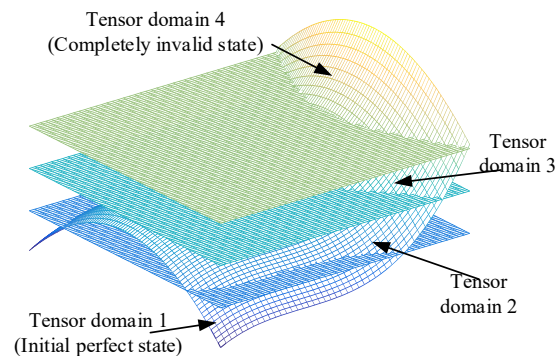
The state aging of a component or a system is generally described by a continuous state surface in the three-dimensional space, as shown in Figure 1. According to the continuous medium theory, the state aging process of a component or a system is continuous. The aging of a component or a system gradually transforms from the blue area to the yellow area. The blue area corresponds to the initial perfect state (i.e., new, unused ideal state), while the yellow area corresponds to the complete failure state (i.e., unserviceable state). For a certain aging state point, the aging speed will have different possible directions due to the difference of environmental parameters and service time, which is reflected in the tensor domain as the possible change direction of a material point not being unique. For

the aging state points under different service times, it is reflected in the tensor domain as the state material points in different spatial locations.



**Figure 1.** Schematic diagram of the component state-evaluation method based on tensor theory.

For the practical engineering application, the aging state function in the tensor domain needs to be discretized to describe the operation state of a component or a system. The aging state function is divided by the classification surfaces, and the tensor domain is divided into several sub-regions, each corresponding to an aging state. An example of the aging state classified into four cases is shown in Figure 2.



**Figure 2.** Schematic diagram of the aging state classification.

The coordinates of the material point of the aging state of a component or a system are assumed to be expressed in terms of the vector path as  $\mathbf{r} = \mathbf{r}(x^i)$ ,  $i = 1, 2, 3$ . The coordinate values here are the parameters related to the thyristor operation or the characteristic quantities extracted from the parameters.

Applying the tensor theory to evaluate the aging state of a component or a system means to treat the aging state as a material point in the tensor domain. First, the correspondence between the aging state and the tensor needs to be determined, and then the classification surface to classify the state mass point tensor domain into several tensor subdomains is determined. The aging state material point in each tensor subdomain corresponds to a class of the aging state. The aging state identification and evaluation of a component or a system can be realized by such classification.

Here, we define the aging state classification surface decision function as:

$$f(\mathbf{r}) = \text{sign}[S_{\text{classification}}(\mathbf{r})] \quad (1)$$

This decision function classifies the different vector paths  $\mathbf{r}$  into different tensor state subdomains by the classification surface function  $S_{\text{classification}}$ .

In order to accurately identify the aging state of the thyristor, it is necessary to clarify the classification decision function, through which each aging state prime is divided into the state tensor subdomain to which it belongs. The aging state identification results are expressed as:

$$Sub\_domain(\mathbf{r}) = multisign[S_{classification}(\mathbf{r})] = \{1, 2, \dots, m\} \quad (2)$$

$K=1,2,\dots,k$

where  $Sub\_domain(\mathbf{r})$ —aging state judgment function,  $multisign(\cdot)$ —multi-valued symbolic function,  $K = 1, 2, \dots, k$ —the classification of the state tensor domain into  $m$  state subdomains requires  $k$  classification surfaces. The value domain of  $multisign(\cdot)$  can be any integer between 1 and  $m$ . Each integer represents a class of aging states, and the maximum value  $m$  means that the tensor domain state space is classified into a total of  $m$  state subdomains.

For different types of power electronic components, due to the difference in the production process, there are differences in their aging process laws, i.e., their corresponding state space tensor domains are different. Furthermore, the aging law of a component or a system is closely related to environmental conditions. The time-varying characteristics of the environmental conditions lead to the time-varying characteristics of the aging state variations indirectly. In other words, the tensor domain corresponding to the aging state material point has a deformation to time, and this real-time deformation can be expressed by the partial derivative of the material point vector path to time.

According to the previous discussion, the vector path  $\mathbf{r} = \mathbf{r}(x^i(\xi^j, t))$  of a state material point in a continuous medium is related to both the spatial coordinates of the state material point and time. The velocity of the state material point can be expressed in terms of the time partial derivative:

$$v = \left( \frac{\partial \mathbf{r}}{\partial t} \right)_{\xi^j} = \frac{\partial \mathbf{r}}{\partial x^i} \left( \frac{\partial x^i}{\partial t} \right)_{\xi^j} = \mathbf{g}_i v^i \quad (3)$$

As is can be seen from Equation (3),  $v^i$  is related to both time  $t$  and the material point  $\xi^j$ .

Assuming that the time  $t$  remains constant, the partial derivative of the state material point vector path with respect to its coordinate position  $\xi^j$  represents the covariant basis vector  $\hat{\mathbf{g}}_i(\xi^j, t)$  at the moment. The physical meaning of  $\hat{\mathbf{g}}_i(\xi^j, t)$  is the deformation of a state material point in the tensor domain at time  $t$ . This covariant basis vector  $\hat{\mathbf{g}}_i(\xi^j, t)$  is then derived with respect to time to obtain the deformation rate of the continuous medium in the tensor domain corresponding to the aging state as:

$$\frac{d\hat{\mathbf{g}}_i}{dt} = \left[ \frac{\partial}{\partial t} \left( \frac{\partial \mathbf{r}}{\partial \xi^j} \right) \right]_{t, \xi^j} \quad (4)$$

## 2.2. Classification of Tensor Subdomain

In this section, the thyristor degradation model based on the tensor domain estimation theory will be applied to automatically locate the classification surfaces of each tensor subdomain using a temporal fuzzy clustering algorithm [18].

Here, we assume the sample series is as  $X = \{z_i = [t_i, \mathbf{x}_i^T]^T | 1 \leq i \leq N\}$ , where  $N$  is the data length of the time series  $X$ . If  $\mathbf{x}_i = \{x_{ij} | j = 1, 2, \dots, M\}$  is a multivariate sample, and  $M$  is the number of dimensions of the sample.

A segment of  $X$  can represent a time-continuous series of sample points  $z_a, z_{a+1}, \dots, z_b$ , denoted as  $S(a, b) = \{a \leq i \leq b\}$ . We assume that the sample series  $X$  can be divided into  $c$  non-overlapping segments, denoted as:

$$S(a, b) = \{S_k(a_k, b_k) | 1 \leq k \leq c\} \quad (5)$$

where

$$a_1 = 1, b_c = m, a_k = b_{k-1} + 1 \quad (6)$$

The sample series segmentation objective function is defined as  $cost(S_X^c)$ , which is generally defined as the distance between the true data points of the sample series and the data points of the sample series fitting function (generally using a linear or a polynomial function).

$$cost(S_X^c) = \sum_{k=1}^c cost(S_k) = \sum_{k=1}^c cost(S(a_k, b_k)) \quad (7)$$

The region boundary  $a_k, b_k, k = 1, 2, \dots, c$  is calculated by Equation (7) to obtain the optimal segment location. Currently, dynamic programming and various heuristic algorithms shown as Equation (8) are commonly used to minimize the objective function.

$$cost(S_X^c) = \sum_{k=1}^c \sum_{i=s_{k-1}+1}^{s_k} \|x_i - v_k^x\|^2 = \sum_{k=1}^c \sum_{i=1}^N \beta_k(t_i) D^2(x_i, v_k^x) \quad (8)$$

where  $v_k^x$ —clustering center of the  $k$ th tensor subdomain,  $D^2(x_i, v_k^x)$ —distance from  $x_i$  to the cluster center, and  $\beta_k(t_i)$ —membership function of  $x_i$  belonging to the  $k$ th tensor subdomain.

In general,  $\beta_k(t_i)$  is a (0,1) binary function, i.e., a crisp subordination degree is applied. However, in practical situations, there are fuzzy boundaries between the tensor subdomains, which are not suitable for the use of the crisp membership function. Therefore, a multivariate mixed Gaussian function shown as Equation (9) is applied as the sequence-fitting function of the clustering prototype, and the optimal region division is obtained by minimizing the sum of squared weighted distances between the data points and the center of the clustering prototype.

$$cost(S_X^c) = \sum_{k=1}^c \sum_{i=1}^N (\mu_{k,i})^m D^2(z_i, \eta_k) \quad (9)$$

where  $\mu_{k,i}$ —membership of the sample data point  $z_i = [t_i, x_i^T]^T$  in the  $k$ th tensor subdomain,  $m$ —fuzzy clustering weighting index whose value is 2 generally,  $D^2(z_i, \eta_k)$ —distance between the sample and the clustering prototype, and  $\eta_k$ —clustering prototype function for the  $k$ th tensor subdomain, which is the multivariate mixed Gaussian function.

We assume that the sample sequence obeys a multivariate Gaussian distribution with the expectation  $v_k$  and the covariance matrix  $F_k$ .  $p(z_i|\eta)$  shown as Equation (10) denotes the probability density function of the sample data points belonging to the  $c$  tensor subdomains.

$$p(z_i|\eta) = \sum_{k=1}^c p(z_i|\eta_k) p(\eta_k) \quad (10)$$

where

$$p(z|\eta_k) = \frac{1}{(2\pi)^{(n+1)/2} \sqrt{\det(F_k)}} \exp\left(-\frac{1}{2}(z - v_k)^T (F_k)^{-1} (z - v_k)\right) \quad (11)$$

$\eta_k$  is the clustering prototype function  $\eta_k = \{p(\eta_k), v_k, F_k | k = 1, 2, \dots, c\}$  for the  $k$ th tensor subdomain. The distance function  $D^2(z_i, \eta_k)$  is inversely proportional with respect to the membership  $p(z|\eta_k)$  of  $z_i$  in the  $k$ th tensor subdomain, and the time variable  $t_i$  and the feature variable  $x_i$  in the sample data are independent of each other. Then, we can obtain:

$$p(z_i|\eta_k) = \frac{1}{D^2(z_i, \eta_k)} = \alpha_{k(\eta_k)} D_t \times D_x \quad (12)$$



where

$$\begin{aligned} D_t &= \frac{1}{\sqrt{2\pi\sigma_{k,t}^2}} \exp\left(-\frac{1}{2} \frac{(t_i - v_k^t)^2}{\sigma_{k,t}^2}\right) \\ D_x &= \frac{1}{(2\pi)^{r/2} \sqrt{\det(A_k)}} \exp\left(-\frac{1}{2} \frac{(x_i - v_k^x)^T}{(A_k)} (x_i - v_k^x)'\right) \end{aligned} \quad (13)$$

$\alpha_{k(\eta_k)}$ —initial probability of clustering,  $D_t$ —distance between the time variable of the  $i$ th sample data point from the center of the clustered time variable,  $D_x$ —distance between the feature variable of the  $i$ th sample data point and the center of the clustering feature variable,  $v_k^x$ —clustering center of the  $k$ th tensor subdomain in the feature space, and  $r$  is the rank of the characteristic variable distance  $A_k$ .  $A_k$  can be estimated by the fuzzy covariance matrix of multivariate Gaussian distribution  $F_k$ , which can be calculated as:

$$F_k = \frac{\sum_{i=1}^N (\mu_{k,i})^m \left( (x_i - v_k^x) (x_i - v_k^x)^T \right)}{\sum_{i=1}^N (\mu_{k,i})^m} \quad (14)$$

To facilitate the inversion of the covariance matrix  $F_k$ , the strong correlation between variables must be eliminated. Principal component analysis (PCA) can perform a series of operations and transformations on high-dimensional data to eliminate correlations between high-dimensional data so as to achieve dimensionality reduction while retaining the information of the original variables as much as possible [19]. We assume the covariance matrix  $F_k$  has  $q$  non-zero eigenvalues (in descending order)  $\lambda_{k,l}$  and their corresponding eigenvectors  $u_{k,l}$ , where  $l = 1, 2, \dots, q$ . Then, we can obtain:

$$F_k = U_k \Lambda_k U_k^T \quad (15)$$

where

$$\begin{aligned} F_k &= U_k \Lambda_k U_k^T \Lambda_k = \begin{vmatrix} \lambda_{k,1} & & & \\ & \lambda_{k,2} & & \\ & & \ddots & \\ & & & \lambda_{k,q} \end{vmatrix} \\ U_k &= [u_{k,1}, u_{k,2}, \dots, u_{k,q}] \end{aligned} \quad (16)$$

For the feature variable  $x_i$  of the sample data, the PCA algorithm is applied to reduce to  $q$  dimensions. Then, we can obtain  $y_{k,i} = W_k^{-1}(x_i) = W_k^T(x_i)$ , where  $W_k = U_{k,q} \Lambda_{k,q}^{1/2}$ . The value of  $W_k$  can also be calculated by the probabilistic principal component analysis (PPCA) algorithm through  $W_k = U_{k,q} (\Lambda_{k,q} - \sigma_{k,x}^2 I)^{1/2} R_k$  [20], where  $R_k$  is a  $q \times q$  orthogonal transformation matrix and  $\sigma_{k,x}^2$  can be calculated as:

$$\sigma_{k,x}^2 = \frac{1}{M - q} \sum_{j=q+1}^M \lambda_{k,j} \quad (17)$$

So far, the automatic fuzzy tensor subdomain classification algorithm has been converted into an optimization problem. The objective function is:

$$\text{cost}(S_X^c) = \sum_{k=1}^c \sum_{i=1}^N (\mu_{k,i})^m D^2(z_i, \eta_k) \quad (18)$$

The constraint condition includes:

$$\begin{cases} U = [\mu_{k,i}]_{cN}, \mu_{k,i} \in [0, 1], \forall k, i \\ \sum_{k=1}^c \mu_{k,i} = 1, \forall i \\ 0 < \sum_{i=1}^N \mu_{k,i} < N, \forall k \end{cases} \quad (19)$$

The optimization problem can be solved by the alternating optimization (AO) algorithm [21], and the basic steps are as follows.

Step 1: Initialization

The number of segments of the sample sequence  $X$  and the dimensionality of the feature vector space retained by the PCA algorithm are given. The suitable termination condition  $\varepsilon$ ,  $\varepsilon > 0$  is chosen and  $W_k$ ,  $v_k^x$ ,  $\sigma_{k,x}^2$ ,  $\mu_{k,i}$  are initialized.

Step 2: Loop Computation

First, the parameters of the clustering prototype are calculated as Equations (20)–(27).

The initial probability of clustering can be calculated as:

$$\alpha_k = \frac{1}{N} \sum_{i=1}^N \mu_{k,i} \quad (20)$$

The clustering center is:

$$\mathbf{v}_k^x = \frac{\sum_{i=1}^N (\mu_{k,i})^m (\mathbf{x}_i - W_k \langle y_{k,i} \rangle)}{\sum_{i=1}^N (\mu_{k,i})^m} \quad (21)$$

where

$$\begin{aligned} \langle y_{k,i} \rangle &= M_k^{-1} W_k^T (\mathbf{x}_i - \mathbf{v}_k^x) \\ M_k &= \sigma_{k,x}^2 I + W_k^T W_k \end{aligned} \quad (22)$$

The weights can be updated as:

$$\tilde{W}_k = F_k W_k \left( \sigma_{k,x}^2 I + M_k^{-1} W_k^T F_k W_k \right)^{-1} \quad (23)$$

The variance can be updated as:

$$\sigma_{k,x}^2 = \frac{1}{q} \text{tr} \left( F_k - F_k W_k M_k^{-1} \tilde{W}_k^T \right) \quad (24)$$

The characteristic variable distance can be calculated as:

$$A_k = \sigma_{k,x}^2 I + \tilde{W}_k \tilde{W}_k^T \quad (25)$$

The clustering prototype center for the sample series time is:

$$v_k^t = \frac{\sum_{i=1}^N (\mu_{k,i})^m t_i}{\sum_{i=1}^N (\mu_{k,i})^m} \quad (26)$$



The variance for the sample series time is:

$$\sigma_{k,t}^2 = \frac{\sum_{i=1}^N (\mu_{k,i})^m (t_i - v_k^t)^2}{\sum_{i=1}^N (\mu_{k,i})^m} \quad (27)$$

Second, the clusters are merged as Equations (28)–(31). For the two adjacent tensor subdomains  $S_o$  and  $S_p$ , the similarity of the two subdomains is compared to determine whether they need to be merged. Since the PCA algorithm is used as mentioned before, a PCA-based similarity factor calculated as Equation (28) is applied to perform as one of the merging criteria.

$$S_{PCA}^{o,p} = \frac{1}{q} \sum_{o=1}^q \sum_{p=1}^q \cos^2 \theta_{o,p} = \frac{1}{q} \text{tr} \left( U_{o,q}^T U_{p,q} U_{p,q}^T U_{o,q} \right) \quad (28)$$

where  $U_{o,q}$  and  $U_{p,q}$  are the first  $q$ th principal components of the feature vectors of the tensor subdomain  $S_o$  and  $S_p$ , respectively.

Another merging criterion is the distance between the clustering centers of the feature vectors of the tensor subdomains  $S_o$  and  $S_p$ , which is calculated as:

$$D(v_o^x, v_p^x) = v_o^x - v_p^x \quad (29)$$

Since the clustering process is performed within the sample sequence, the fuzzy decision algorithm is applied to measure the clustering similarity of each tensor subdomain in the whole. The overall similarity matrix of the decision process is:

$$H = \{h_{o,p} | 1 \leq o \leq c, 1 \leq p \leq c\} \quad (30)$$

where

$$h_{o,p} = \sqrt{\frac{m_1 + m_2}{2}} \quad (31)$$

$$m_1 = \left( \frac{1}{c(c-1)} \sum_{o=1}^c \sum_{\substack{p=1 \\ p \neq o}}^c S_{PCA}^{o,p} \right)^2$$

$$m_2 = \left( \frac{1}{c(c-1)} \sum_{o=1}^c \sum_{\substack{p=1 \\ p \neq o}}^c D(v_o^x, v_p^x) \right)^2$$

When  $h_{o,p}$  is greater than the set threshold, the tensor subdomains  $S_o$  and  $S_p$  are merged. The loop computation ends when the algorithm termination condition  $\varepsilon$  is reached.

### 2.3. Data-Processing Algorithms

#### 2.3.1. Improved Adaptive Synthetic Sampling Algorithm

The main principle of the adaptive synthetic sampling algorithm is to increase the size of the few data classes adaptively according to the data density distribution so as to enhance the sensitivity of the classifier to the few data classes, especially the sensitivity of those samples that are difficult to achieve the learning function. The main shortcoming of the adaptive synthetic sampling algorithm is that it can only be applied to binary systems generally, which can be undesirable when there are abnormal data samples. In this section, an improved adaptive synthetic-sampling algorithm that is capable of handling multiple classification problems is proposed. The specific algorithm steps are introduced below [22].

### Step 1: Sample Sorting

The majority of the category sample is found by the label of the data sample, which is shown in Equation (32).

$$\begin{cases} S_{\max} = \max\{S_0, S_1, \dots, S_N\} \\ \text{sort}\{S_0, S_1, \dots, S_N\} = \{CS_0, CS_1, \dots, CS_N\} \end{cases} \quad (32)$$

where  $S_{\max} = CS_N$ .

### Step 2: Expanding of the First Category

The imbalance degree between the first and the second type of data is defined as:

$$k_{N-1} = CS_{N-1} / S_{\max} \quad (33)$$

To minimize the generation of samples on subsequent decision boundaries, the following individual categories are arranged according to the size of the data samples. An acceptable inter-category imbalance threshold  $k_{th}$  is set. If  $k_{N-1} \geq k_{th}$ , the data volume of category  $CS_{N-1}$  is considered to be satisfied and no new samples need to be synthesized for it. On the other hand, if  $k_{N-1} \leq k_{th}$ , it is necessary to synthesize new samples for the category  $CS_{N-1}$  to expand its data volume. There are three steps to realize the synthesis.

First, calculate the amount of data to be synthesized for the few category expansions as:

$$N_{N-1} = (S_{\max} - CS_{N-1}) \times \beta, \quad \beta \in [0, 1] \quad (34)$$

where  $\beta$  indicates the level of data balance that will be achieved after data expansion for the few categories. When  $\beta$  is 1, a fully balanced amount of category data can be achieved. When  $\beta$  is 0, no new samples will be synthesized. We assume the sample satisfies  $x_i \in CS_{N-1}$ , and the  $k$  nearest neighbor samples of  $x_i$  can be obtained based on Euclidean distance. The probability that these nearest neighbor samples belong to the category  $CS_N$  can be calculated as:

$$E_{n,N-1} = \Delta_{n,N-1} / K \quad (35)$$

where  $\Delta_{n,N-1}$  is the number of these nearest neighbor samples in the category  $S_{\max}$ , and  $E_{n,N-1} \in [0, 1]$ .

Second,  $E_{n,N-1}$  is normalized to obtain the ratio distribution as:

$$\hat{E}_{n,N-1} = E_{n,N-1} / \sum_{n=1}^{CS_{N-1}} E_{n,N-1} \quad (36)$$

Third, the required number of samples to be synthesized can be calculated as:

$$n_{N-1} = \hat{E}_{n,N-1} \times N_{N-1} \quad (37)$$

For every  $x_i \in CS_{N-1}$ ,  $n_{N-1}$  data samples are required to be added, and the expanded sample number of  $CS_{N-1}$  can be calculated as:

$$\text{new}CS_{N-1} = CS_{N-1} + \sum n_{N-1} \quad (38)$$

### Step 3: Expanding of the Second Category

Similar to Step 2 mentioned above, the imbalance degrees  $r_{N-2,1}$  and  $r_{N-2,2}$  of the data volume of the third category with respect to the first category and the second category, respectively, after the expansion of the sample are calculated as Equation (39).

$$\begin{aligned} r_{N-2,1} &= CS_{N-2} / S_{\max} \\ r_{N-2,2} &= CS_{N-2} / \text{new}CS_{N-1} \end{aligned} \quad (39)$$

The imbalance degrees are compared with the set threshold of the inter-category imbalance to determine the number of new samples needed,  $N_{N-2,1}$  and  $N_{N-2,2}$ .

$$\begin{aligned} N_{N-2,1} &= (S_{\max} - CS_{N-2}) \times \beta \times \left( \frac{r_{N-2,1}}{r_{N-2,1} + r_{N-2,2}} \right) \\ N_{N-2,2} &= (newSC_{N-1} - CS_{N-2}) \times \beta \times \left( \frac{r_{N-2,2}}{r_{N-2,1} + r_{N-2,2}} \right) \end{aligned} \quad (40)$$

The ratio of the nearest neighbor samples to all samples  $E_{n,N-2,1}$  and  $E_{n,N-2,2}$  are calculated as Equation (41).

$$\begin{aligned} E_{n,N-2,1} &= \Delta_{n,N-2,1} / K \\ E_{n,N-2,2} &= \Delta_{n,N-2,2} / K \end{aligned} \quad (41)$$

The ratio distribution is then calculated as Equation (42).

$$\begin{aligned} \hat{E}_{n,N-2,1} &= E_{n,N-2,1} / \sum_{n=1}^{CS_{N-2}} E_{n,N-2} \\ \hat{E}_{n,N-2,2} &= E_{n,N-2,2} / \sum_{n=1}^{CS_{N-2}} E_{n,N-2} \end{aligned} \quad (42)$$

The amount of data needed to expand the new sample can be calculated as:

$$\begin{aligned} n_{N-2,1} &= \hat{E}_{n,N-2,1} \times N_{N-2,1} \\ n_{N-2,2} &= \hat{E}_{n,N-2,2} \times N_{N-2,2} \end{aligned} \quad (43)$$

Then, the second category sample can be expanded, and the expanded sample number of  $CS_{N-2}$  can be calculated as:

$$newCS_{N-2} = CS_{N-2} + \sum g_{N-2,1} + \sum g_{N-2,2} \quad (44)$$

#### Step 4: Expanding of Other Categories

The process of Step 3 is repeated to expand the sample data of other categories including  $CS_{N-3}$ ,  $CS_{N-4}$ ,  $CS_{N-5}$ , ...,  $CS_1$ . To eliminate the influence of abnormal data points, if  $E_n = 1$ ,  $n = 1, 2, 3, \dots, CS_m$  appear during the calculation, which means that the sample data point is completely surrounded by the other categories of data, it can be considered as an abnormal sample point for elimination. The flow chart of the improved adaptive synthetic sampling algorithm is shown in Figure 3.

#### 2.3.2. Gradient-Boosting Decision Tree Algorithm

In this section, the gradient-boosting decision tree algorithm is applied as a classifier for the state thyristor state evaluation to obtain the tensor subdomain classification surfaces. The algorithm is derived from the decision tree. The errors can be corrected by a differentiable loss function during the training process, which can realize further convergence. The training process of the gradient-boosting decision tree algorithm is shown in Figure 4. The algorithm obtains one weak classifier in each iteration, and then all the resulting classifiers are trained based on the residuals of the previous classifier. During the training process, the bias-reduction approach is applied to increase the classifier accuracy so as to obtain the final classifier with the best accuracy. Generally, the classification decision trees are chosen as the base classifier of the algorithm, and the depth of these classification decision trees is not deep due to the requirement of the base classifier. Finally, the total classification model is obtained by weighting and summing the weak classifiers obtained in each training round.

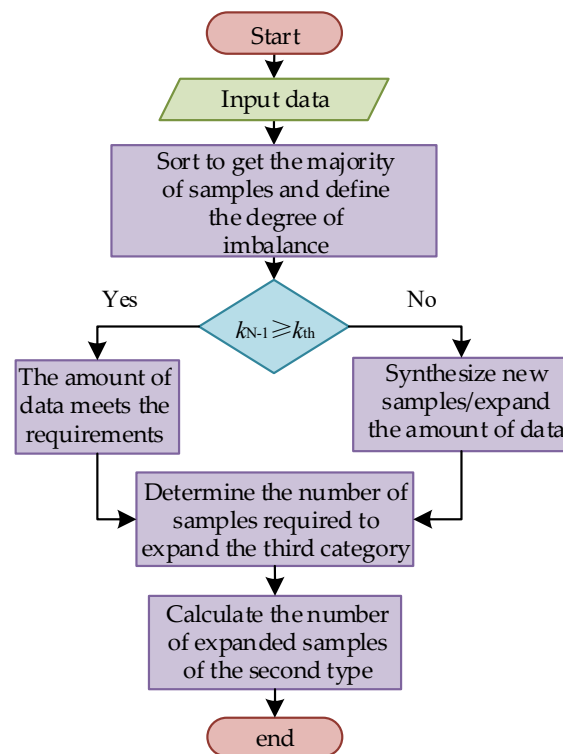


Figure 3. Flow chart of the improved adaptive synthetic sampling algorithm.

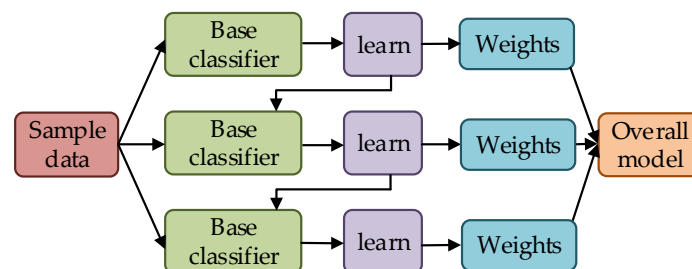


Figure 4. Training process of the gradient-boosting decision tree algorithm.

The total classification model can be expressed as:

$$F_m(x) = \sum_{m=1}^M \alpha_m t_m(x; \theta_m) = \sum_{m=1}^M T_m(x; \theta_m) \quad (45)$$

where  $x$ —input sample,  $t$ —classification tree,  $\theta_m$ —parameter of the weak classifier, and  $\alpha_m$ —weight of each tree. The model is trained for a total of  $M$  rounds, and weak classifiers are generated in each training round. The corresponding weak classifier loss function is:

$$\hat{\theta}_m = \operatorname{argmin}_{\theta_m} \sum_{i=1}^N L(y_i, F_{m-1}(x_i) + T(x_i; \theta_m)) \quad (46)$$

$F_{m-1}(x_i)$  is the current classifier model. Generally, the best parameters for the next weak classifier are obtained by the empirical risk-minimization method when applying the gradient-boosting decision tree algorithm. Specifically, it is the selection of the loss function, which mainly includes the square loss function, logarithmic loss function, and 0–1 loss function. The difference in the squared loss function is consistent with the concept of residuals. The fastest direction of descent can be obtained by making the loss function fall along the gradient direction, which is the reason for the application of the gradient in the

algorithm. The algorithm applies the negative gradient of the loss function corresponding to each round of the weak classifier to fit the decision tree. The negative gradient direction of the loss function of the current weak classifier is fitted at each iteration so as to reduce the loss function as soon as possible during the training process and converge to the optimal local solution or the optimal global solution.

The steps of the gradient-boosting decision tree algorithm are as follows.

**Step 1: Extraction of Relevant Features**

The dataset to characterize the thyristor state is established based on the environmental conditions related to the operation of the converter valve in the UHVDC transmission project (including the temperature and humidity of the valve hall and the electromagnetic interference situation), the thyristor operating voltage and current, and the field test results.

**Step 2: Comparison of the Imbalance Degree between Categories**

The inter-category imbalance threshold is set to 0.5 in the evaluation. The inter-category imbalance degree of the dataset is compared with the threshold. If the imbalance degree of the dataset is higher than the threshold, the improved adaptive synthetic sampling algorithm introduced above is applied to balance the data in the dataset. On the contrary, if the imbalance degree of the dataset is lower than the threshold, no data balancing is performed.

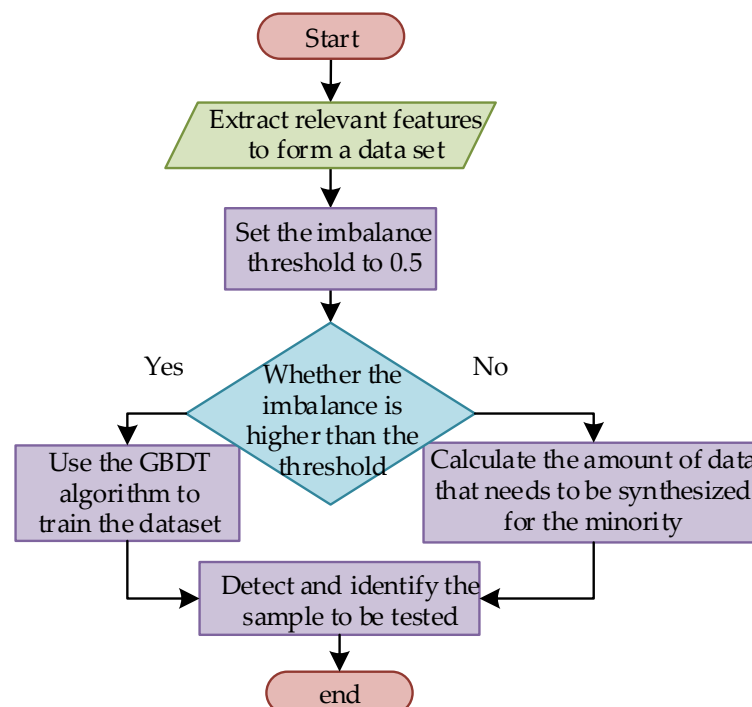
**Step 3: Training**

The gradient-boosting decision tree algorithm is applied to train the data in the dataset, and the tensor domain state classification surface is obtained by the training.

**Step 4: Verification of the Algorithm and Identification of the Test Samples**

The reliability of the algorithm is verified by the existing data. The algorithm can also be further applied to identify the state of the test samples.

In summary, the flowchart of the gradient-boosting decision tree algorithm is shown in Figure 5.



**Figure 5.** Flowchart of the gradient-boosting decision tree algorithm.

### 3. Thyristor Aging State-Evaluation Method

#### 3.1. Thyristor Degradation Mechanism

According to the actual engineering statistics, the degradation mechanism of the general thyristor can be divided into three parts: early failure period, occasional failure period, and depletion failure period. The early failure period generally occurs at the early stage of the product into use. Failures of the early failure period are mainly caused by defects in the production chain, which can be strictly controlled through the factory test. The failure rate is low and stable during the occasional failure period. The depletion failure period refers to the stage when the number of components failures obviously increases after a certain number of years of use. This stage is mainly caused by abrasion, aging, and other reasons that lead to the degradation of the product performance and failure.

Specific degradation phenomena can be classified into different failure modes according to the different causes of failure, which are discussed as follows.

##### Mode 1: Overvoltage-Failure-Based Mode

Thyristor overvoltage failure means that the applied voltage to the thyristor exceeds the rated voltage of the thyristor. Under severe operating conditions, the local leakage current at the PN junction terminal may be too high. Due to the presence of positive feedback, the local current will continue to increase, eventually leading to thyristor failure.

##### Mode 2: Overcurrent-Failure-Based Mode

The thyristor is subjected to a large surge current under special conditions, and the duration of the surge current is about ten milliseconds. At the moment when the current reaches its maximum value, the thyristor has not yet expanded to full area conduction, at which time a large temperature difference will be generated between different areas (conduction and non-conduction areas). This temperature difference causes uneven stress on the various parts of the silicon, and the more stressed part will be damaged.

##### Mode 3: High-Current-Rise-Rate-Based Mode

In the thyristor-opening process, if the anode current change rate exceeds the critical value, the component in the closing area may age quickly due to high local temperature, and in serious cases, may directly be destructed by one impulse.

##### Mode 4: High-Voltage-Rise-Rate-Based Mode

If the forward blocking voltage rises too fast, the thyristor may not open correctly without reaching the turnaround voltage, which may even lead to device damage.

##### Mode 5: Temperature-Characteristics-Based Mode

The electrical characteristics of power devices are closely related to temperature, and the internal temperature rise of the device is a key factor affecting the device's aging state. Different temperatures have a large impact on the parameters of thyristors. For example, when the temperature rises, the control pole trigger current becomes smaller, and the turn-off time becomes longer.

#### 3.2. The Complete Process of the State-Evaluation Method

As shown in Figure 6, the dataset in this article includes a basic index, statistics of failures over the years, and experimental data. In Figure 6, the factory test of thyristor refers to the test conducted by the manufacturer before the thyristor is put into the market, including the test of forward blocking leakage current, gate voltage, gate current, etc. The infrared temperature measurement results are the external temperature of the thyristor recorded during the factory test and the other tests. Based on the above data, this article firstly uses the temporal fuzzy clustering algorithm based on the related mathematical knowledge of tensor theory to realize the automatic positioning of the classification surface of tensor subdomains. Secondly, the improved adaptive synthetic sampling algorithm is improved to balance the data of various categories. Finally, the gradient-boosting decision tree algorithm is used to train and test the equalized data to obtain the state-evaluation result of the thyristor.



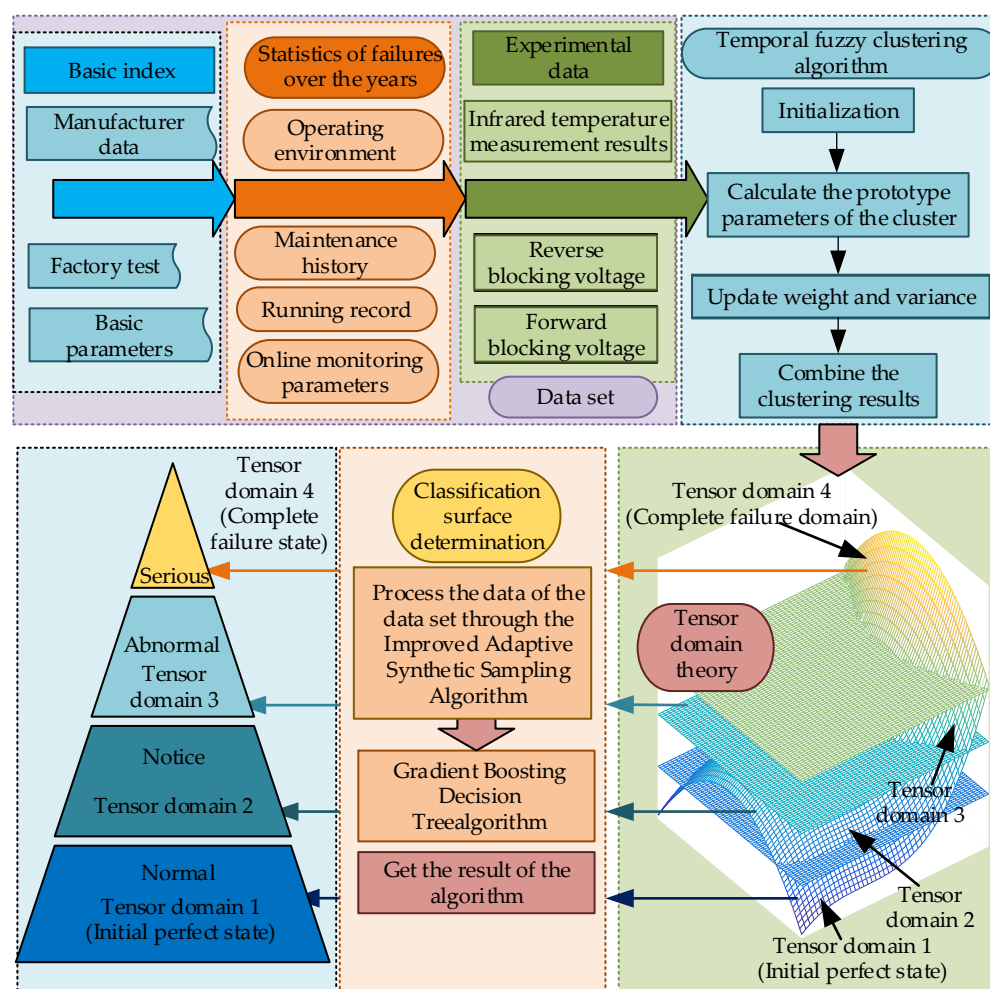


Figure 6. Flow chart of the STDC thyristor evaluation method.

## 4. Result and Analysis

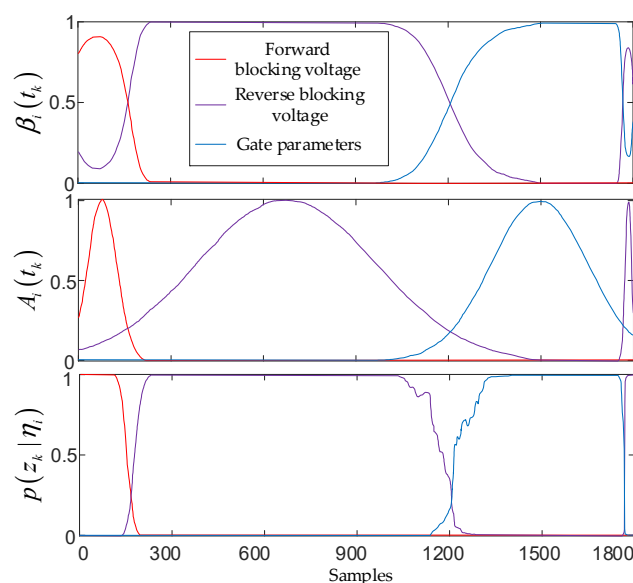
### 4.1. Dataset

The data of the samples used in this article mainly come from the relevant thyristors of the Tian-Guang and Gao-Zhao HVDC transmission lines. They mainly include the following two parts: the first part is the relevant test data of the thyristors in the Zhaoqing converter station of China Southern Power Grid conducted by Xi'an Peri Power Semiconductor Converter Technology Co., Ltd. Specifically, these data include the electrical performance test, thermal stress and mechanical stress test of the thyristor, as well as part of the maintenance records. The second part uses the relevant sample data of the Tianshengqiao converter station provided by CSG EHV Power Transmission Company of China Southern Power Grid, which specifically include various inspection information, maintenance records, and operating environment monitoring. There are 11 thyristors applied to form the dataset, which is shown in Table 1. The numbers of ls1\_1, ls1\_2, ..., ls4\_2 represent each single thyristors, respectively.

Table 1. Dataset.

Dataset Type	Operating Conditions			
Training set	115 °C	139.7 °C	167.7 °C	200 °C
	ls1_1	ls2_1	ls3_1	ls4_1
	ls1_2	ls2_2		
	ls1_3			
Validation set	ts1_4	ts2_3	ts3_2	ts4_2

The data of the features of ls1\_1 in the dataset are automatically segmented, and the result is shown in Figure 7.



**Figure 7.** The segmentation result graph of the dataset.

Therefore, the segmentation result of the dataset ls1\_1 is [1,180], [181,1205], [1206,1770], [1771,1800]; therefore, the aging process of the thyristor can be divided into four stages: normal stage, attention stage, abnormal stage, and invalidation stage, which are consistent with actual use. In this paper, the invalidation state is regarded as the tensor domain 4. The normal stage, the attention stage, and the abnormal stage are denoted as the tensor domain 1, domain 2, and domain 3. The remaining datasets can also be divided into states, and the results are shown in Table 2.

**Table 2.** Results of data classification.

Thyristor Number	Normal (Tensor Domain 1)	Attention (Tensor Domain 2)	Abnormal (Tensor Domain 3)	Invalidation (Tensor Domain 4)
ls1_1	(1,180)	(181,1205)	(1206,1770)	(1771,1800)
ls1_2	(1,144)	(145,181)	(182,422)	(423,485)
ls1_3	(1,304)	(305,511)	(512,1208)	(1209,1286)
ts1_4	(1,322)	(323,405)	(406,453)	(454,496)
ls2_1	(1,203)	(204,423)	(424,468)	(469,522)
ls2_2	(1,260)	-	(261,456)	(457,493)
ts2_3	(1,122)	(123,304)	-	(304,342)
ls3_1	(1,155)	(156,233)	-	(234,241)
ts3_2	(1,198)	(199,366)	(367,433)	(434,502)
ls4_1	(1,89)	(90,266)	-	(267,344)
ts4_4	(1,166)	(167,229)	(230,298)	(299,346)

It is worth noting that not all the thyristors under test go through these four degradation stages. In the test sample, Thyristor ls2\_2 skips the notice stage. Thyristor ls2\_3, Thyristor ls3\_1, and Thyristor ls4\_1 skip the abnormal stage. The phenomenon that thyristors may skip some degradation states is because although the type of these thyristors under test are the same, the individual differences of these thyristors still exist. It is a normal phenomenon in engineering applications.

#### 4.2. Algorithm Evaluation Indicators

The following indicators are mainly used for algorithm evaluation, which are the detection rate, false acceptance rate, total correct rate, and Fleiss Kappa [22]. The definition of the evaluation index is shown in Table 3.

**Table 3.** Algorithm performance evaluation indicators.

Evaluation Index	Symbol	Definition
The detection rate	$DR_i$	The proportion of samples that are correctly classified into that category in a certain category of samples to the total number of samples of that category
False acceptance rate	$FAR_i$	The proportion of the number of samples that are misclassified into a certain category to the total number of samples that are not in the category
Correct rate	$CR$	Accuracy of classification of samples of all categories
Fleiss Kappa	/	An indicator that indicates the consistency between the output classification of the classifier and the real situation, and the calculation method is shown in [23]

According to the explanation in Table 3, the closer the values of and are to 1, the higher the classification accuracy of the algorithm. The closer the value is to 0, the lower the misclassification rate of the algorithm.

To verify the effectiveness and accuracy of the thyristor condition evaluation method proposed in this paper, the datasets of  $ls1\_1$ ,  $ls1\_2$ ,  $ls2\_1$ ,  $ls2\_2$ ,  $ls3\_1$ , and  $ls4\_1$  are used for training, and other datasets are used for testing.

To verify the effectiveness of the STDC integration algorithm, the results are compared with the k-nearest neighbor algorithm (KNN), Support Vector Machine (SVM), and Gaussian NB (Gaussian naive Bayesian).

In the experiment, the number of nearest neighbors of the KNN classifier is 10, all points in the neighborhood are weighted by mean, and Euclidean distance is used as the decision function. The SVM algorithm uses RBF as the kernel function; the error penalty parameter is set to 1.0, and the termination tolerance of the algorithm is  $1 \times 10^{-3}$ . The Gaussian NB algorithm uses the default parameters of scikit-learn, the prior probability of each category is adjusted according to the input data, and the single sample weight is 1. The sub-sample ratio of the gradient-boosting decision tree algorithm is 0.8. The deviation loss function with probability output is selected as the classification decision function. The learning rate is set to 0.1, and other parameters are default.

#### 4.3. Results and Analysis

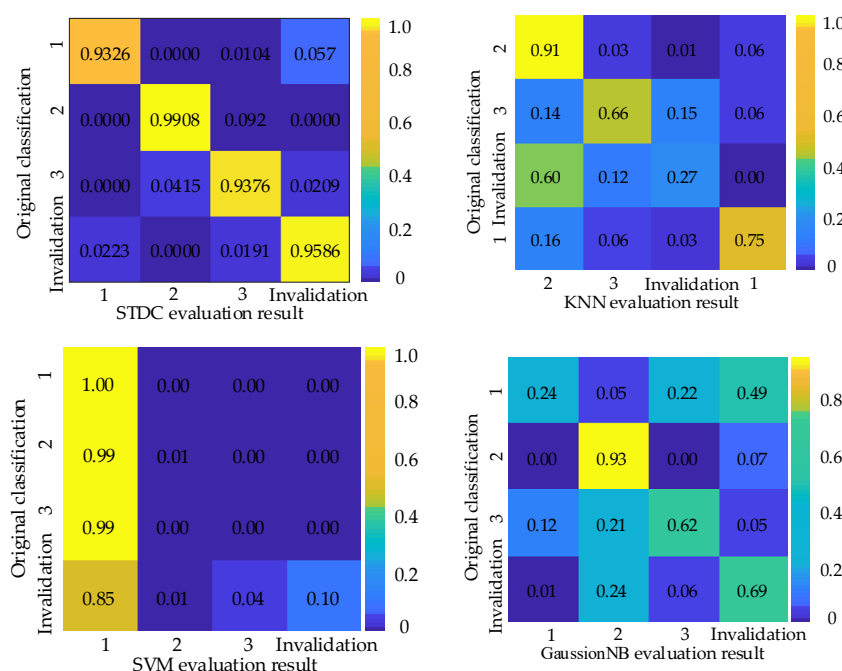
To verify the effectiveness and accuracy of thyristor state classification and identification proposed, the data of Thyristor  $lsl\_1$ ,  $lsl\_2$ ,  $lsl\_1$ ,  $lsl\_2$ ,  $lsl\_1$ ,  $lsl\_2$ ,  $lsl\_3$ ,  $lsl\_4$ ,  $lsl\_3$ ,  $lsl\_3$ ,  $lsl\_2$ , and  $lsl\_4\_2$  are applied for training and the data of Thyristor  $lsl\_3$ ,  $lsl\_4$ ,  $lsl\_3$ ,  $lsl\_2$ , and  $lsl\_4\_2$  are applied for testing. The results of algorithm evaluation indicators are shown in Table 4.

**Table 4.** The result of the evaluation index of the algorithm.

Evaluation Index	Result	Evaluation Index	Result
$DR_1$	0.9326	$FAR_1$	0.0674
$DR_2$	0.9908	$FAR_2$	0.0092
$DR_3$	0.9376	$FAR_3$	0.0624
$DR_4$	0.9586	$FAR_4$	0.0414
$CR$	0.9549	$FK$	0.9023

It can be seen from Table 2 that the DR of each state category is higher than 0.93, the CR is higher than 0.95, and the FK value is greater than 0.9. The results show that the method in this paper can effectively evaluate the state of the thyristor.

The confusion matrix of the classification results of the four algorithms is shown in Figure 8. It can be seen from Figure 8 that different algorithms have different sensitivity to different state categories. KNN and SVM algorithms are more vulnerable to unbalanced datasets. The minimum DR of the KNN algorithm is only 0.27, which is much lower than the value in this paper. SVM algorithm has an over-fitting phenomenon in the experiment, which is caused by the difference of penalty coefficients between samples of different categories. Although the maximum DR of the Gaussian NB algorithm reaches 0.93, the minimum DR value is only 0.24. The algorithm in this paper has achieved a better classification effect, and the minimum value of DR has reached 0.93.



**Figure 8.** Comparison of the confusion matrix of the four algorithms.

Each algorithm is tested 100 times and calculated the average running time. The results are shown in Table 5.

**Table 5.** The average execution time.

Algorithm	Running Time
KNN	1.0138 s
SVM	2.5685 s
Gaussian NB	1.1344 s
STDC	1.4389 s

It can be seen from Table 5 that the running speed of the SVM algorithm is the slowest. The running speed of the algorithm in this paper is close to that of the Gaussian NB algorithm. Although the KNN algorithm has the fastest running speed, its classification accuracy is far lower than the method in this article. Therefore, to sum up, the method in this paper has the highest recognition rate while maintaining a fast operation speed.

Furthermore, it is worth mentioning that although the aging-state-evaluation method proposed in this paper has been validated only by the dataset of thyristors, the algorithm can be essentially applied for the degradation analysis of other different semiconductor devices.

## 5. Conclusions

In order to extract useful information from the massive high-dimensional data and obtain the aging state of thyristors, the supervised tensor domain classification (STDC) method based on the adaptive synthetic sampling method, the gradient-boosting decision tree, and the tensor domain theory is proposed in this paper. Firstly, it is proposed to use the concept of the tensor domain to identify the aging state of the thyristor. Then, the original state-identification problem is transformed into the determination of the state of the tensor domain category classification surface. Secondly, applying the temporal fuzzy clustering algorithm, the classification surface of each tensor subdomain is automatically located. Thirdly, for the data imbalance of the different state categories of the thyristor, an improved adaptive synthetic sampling algorithm is applied to preprocess the data. Furthermore, the gradient-boosting decision tree algorithm is applied to solve the multi-classification problem. Finally, the method proposed is verified through historical operating data of thyristors and related experimental data. The results show that the STDC method proposed achieves the best classification effect, and the recognition accuracy of each state reaches more than 93%, which can meet the practical engineering requirements. The corresponding maintenance strategies for thyristors in different states are mentioned, which can provide the reference for practical engineering applications.

**Author Contributions:** Conceptualization, Z.L.; methodology, Y.T.; software, J.Y.; validation, Y.X.; investigation, J.Z.; resources, Z.L.; data curation, Y.T.; writing—original draft preparation, J.G.; supervision, Y.F. and B.S.; funding acquisition, Z.L. and Y.F. All authors have read and agreed to the published version of the manuscript.

**Funding:** This publication was financed in part by the National Key R&D Program of China under Grant 2016YFB0900903 and in part by the Science and Technology Project of the China Southern Power Co., Ltd. under Grant CGYKJXM20180229.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Zhu, B.; Lu, J.; Zhang, X.; Dai, Y.; Wang, X. Study on the High Frequency Voltage Oscillation of Thyristor During the Initial Stage of Pulse Capacitor Discharge. *J. Trans. China Electrotech. Soc.* **2020**, *35*, 1272–1278.
2. Jiang, Y.; Liu, Z.; Wang, W.; Zhou, W.; Xu, H. Research on multi-state monitoring system of substation equipment based on edge-cloud collaboration. *J. Power Syst. Prot. Control* **2021**, *49*, 138–144.
3. Wang, X.; Yang, L.; Kuang, L.; Liu, X.; Zhang, Q.; Deen, M. A tensor-based big-data-driven routing recommendation approach for heterogeneous networks. *J. IEEE Netw.* **2019**, *33*, 64–69. [[CrossRef](#)]
4. Kuang, L.; Hao, F.; Yang, L.; Lin, M.; Luo, C.; Min, G. A tensor-based approach for big data representation and dimensionality reduction. *J. IEEE Trans.* **2014**, *2*, 280–291. [[CrossRef](#)]
5. Duchenne, O.; Bach, F.; Kweon, I.; Ponce, J. A tensor-based algorithm for high-order graph matching. *J. IEEE Trans.* **2011**, *33*, 2383–2395. [[CrossRef](#)]
6. Li, X.; Niu, J.; Bhuiyan, M.; Wu, F.; Karuppiyah, M.; Kumari, S. A robust ECC-based provable secure authentication protocol with privacy preserving for industrial Internet of Things. *J. IEEE Trans.* **2017**, *14*, 3599–3699. [[CrossRef](#)]
7. Cichocki, A.; Mandic, D.; De Lathauwer, L.; Zhou, G.; Zhao, Q.; Caiafa, C.; Phan, H. Tensor decompositions for signal processing applications: From two-way to multiway component analysis. *J. Signal Process.* **2015**, *32*, 145–163. [[CrossRef](#)]
8. Tang, Y.; Xu, Z.; Yue, B.; Li, X. Modeling of Reverse Recovery Process of Series Thyristor Based on Single Thyristor Level Circuit Topology and Study of Commutation Over-voltage. *J. High Volt. Eng.* **2020**, *46*, 4435–4447.
9. Liu, Y.; Wang, S.; Dai, L.; Dong, M.; Yao, W.; Liu, C. Analysis and Modeling-simulation on Turn-off Characteristics of HVDC Thyristor. *J. High Volt. Eng.* **2020**, *46*, 4447–4454.
10. Ye, M.; Li, G.; Chen, X.; Zhou, C.; Pang, L.; Zhang, Q. Protection Characteristics of Thyristor Control Unit and Simulation Model of Thyristor-level for HVDC Converter Valve. *J. Power Syst. Technol.* **2020**, *44*, 1445–1452.
11. Hu, Z.; Wang, L.; Qi, L.; Li, Y.; Yang, W. A Novel Wireless Network Intrusion Detection Method Based on Adaptive Synthetic Sampling and an Improved Convolutional Neural Network. *IEEE Access* **2020**, *8*, 195741–195751. [[CrossRef](#)]

12. Feng, L.; Wang, H.; Jin, B.; Li, H.; Xue, M.; Wang, L. Learning a Distance Metric by Balancing KL-Divergence for Imbalanced Data sets. *IEEE Trans. Syst.* **2019**, *49*, 2384–2395.
13. Khan, S.; Hayat, M.; Bennamoun, M.; Sohel, F.; Togneri, R. Cost-Sensitive Learning of Deep Feature Representations From Imbalanced Data. *IEEE Trans. Neural Netw. Learn. Syst.* **2018**, *29*, 3573–3587.
14. Feng, W.; Huang, W.; Bao, W. Imbalanced Hyperspectral Image Classification With an Adaptive Ensemble Method Based on SMOTE and Rotation Forest With Differentiated Sampling Rates. *IEEE Geosci. Remote Sens. Lett.* **2019**, *16*, 1879–1883. [[CrossRef](#)]
15. Cao, Y.; Zhao, X.; Zhou, Z.; Chen, Y.; Liu, X.; Lang, Y. MIAC: Mutual-Information Classifier with ADASYN for Imbalanced Classification. In Proceedings of the 2018 International Conference on Security, Pattern Analysis, and Cybernetics (SPAC), Jinan, China, 14–17 December 2018; pp. 494–498.
16. Xie, C.; Wang, J.; Xie, X.; Liu, Z.; Bai, J. PV Output Prediction Based on Gradient Boosting Decision Tree Model With Bayesian Optimization Algorithm and Fine-grained Features. *J. Power Syst. Technol.* **2020**, *44*, 689–696.
17. Yu, X.; Zhao, Z.; Ma, Y.; Zheng, R.; Xi, Z.; Ma, C. Multi-label Text Classification for Power ICT Custom Service System Based on Binary Relevance and Gradient Boosting Decision Tree. *J. Autom. Electr. Power Syst.* **2021**, *45*, 144–151.
18. Yu, C.; Wu, Z.; Tan, L.; Tu, X.; Yang, Y.; Wang, L. Multivariate time series fuzzy clustering segmentation mining algorithm. *J. Chin. J. Eng.* **2014**, *36*, 260–265.
19. Li, Y.; Yang, D. Local component based principal component analysis model for multimode process monitoring. *J. Chin. J. Chem. Eng.* **2021**, *34*, 116–124. [[CrossRef](#)]
20. Zheng, D.; Zhou, L.; Song, Z. Kernel Generalization of Multi-Rate Probabilistic Principal Component Analysis for Fault Detection in Nonlinear Process. *J. IEEE/CAA J. Autom. Sin.* **2021**, *8*, 1465–1476. [[CrossRef](#)]
21. Li, K.; Zhou, B.; Jia, J.; Du, Y.; Lu, Y.; Cheng, H.; Wang, S.; Wang, H. Alternating Iterative Optimization Algorithm Based on Multispectral Absorption. *J. Eng. Thermophys.* **2018**, *39*, 1598–1608.
22. Kou, L. Service Status Identification and System Reliability Evaluation Method Based on Tensor Domain Theory for Rail Train with Monitoring Data. Ph.D. Thesis, Beijing Jiaotong University, Beijing, China, 2019.
23. Liang, N.; Zhang, Z.; Liu, C.; Wang, F.; Zhuo, F. Failure Mechanism and Lifetime Model Analysis on Thyristor in HVDC Converter Valve Under EMP. *J. Power Capacit. React. Power Compens.* **2019**, *40*, 142–146.