UDC 621.452.3:681.518.54:004.032.26 **doi: 10.32620/aktt.2021.4sup2.09**

A. TAMARGAZIN, L. PRYIMAK, V. SHOSTAK

National Aviation University, Kyiv

METHODS OF CLUSTERING PARAMETERS IN THE CREATION OF NEURAL NETWORK MULTI-MODE DYNAMIC MODELS OF AIRCRAFT ENGINES

The presence on modern aviation gas-turbine engines of dozens and even hundreds of sensors for continuous registration of various parameters of their operation makes it possible to collect and process large amounts of information. This stimulates the development of monitoring and diagnostic systems. At the same time the presence of great volumes of information is not always a sufficient condition for making adequate managerial decisions, especially in the case of evaluation of the technical condition of aviation engines. Thus it is necessary to consider, that aviation engines it is objects which concern to individualized, i.e. to such which are in the sort unique. Therefore, the theory of creating systems to assess the technical state of aircraft engines is formed on the background of the development of modern neural network technology and requires the formation of specific methodological apparatus. From these positions in the article the methods which are used at carrying out clustering of the initial information received at work of modern systems of an estimation and forecasting of a technical condition of aviation gas-turbine engines are considered. This task is particularly relevant for creating neural network multimode models of aircraft engines used in technical state estimation systems for identification of possible failures and damages. Metric, optimization and recurrent methods of input data clustering are considered in the article. The main attention is given to comparison of clustering methods in order to choose the most effective of them for the aircraft engine condition evaluation systems and suitable for implementation of systems with meta-learning. The implementation of clustering methods of initial data allows us to breakdown diagnostic images of objects not by one parameter, but by a whole set of features. In addition, cluster analysis, unlike most mathematical-statistical methods do not impose any restrictions on the type of objects under consideration, and allows us to consider a set of raw data of almost arbitrary nature, which is very important when assessing the technical condition of aircraft engines. At the same time cluster analysis allows one to consider a sufficiently large volume of information and sharply reduce, compress large arrays of parametrical information, make them compact and visual.

Keywords*: aircraft engine; diagnostics; neural network.*

Introduction

In the last few years, various methods of creating multi-mode dynamic models of aircraft engines for evaluation and prediction of their technical condition have been rapidly developed. Among them are deep convolutional neural networks, recurrent neural networks, logistic regression, vector reference method, k-means method, etc. One of the key distinguishing features of the application of these methods is the ability to master the solution of complex problems from scratch, using quite a lot of time and data for training [1, 2].

But at creation of systems of an estimation of technical condition of modern aviation gas-turbine engines there is a problem of creation of such methods which would allow to acquire new skills and to adapt to various conditions of operation of engines without possibility to spend training of diagnostic system in each situation from zero. Instead of this, the Condition Analysis system needs to be able to learn new tasks from previous experiences rather than looking at each new task in isolation. In this case it is necessary to use meta-learning [3]. Systems for evaluating the technical condition of aircraft engines using meta-learning methods should be able to learn with high quality from just a few examples per class, as well as to adapt to new classes that were not presented in the training sampling.

1. The problem statement

The methods used in modern aviation engines condition estimation systems use neural network methods, as with supervised learning, as with unsupervised learning.

The supervised learning algorithm uses data for which there is a labeling: each set of engine performance parameters has a label of the class to which it belongs. These labels are then used in training to provide signals about how to change the parameters of the trained model to improve the quality of the classification. In the standard setting of the classification problem, the classes in supervised learning coincide with the classes in the test. In this case, each data element corresponds to a real number that is used to change the pa-

ISSN 1727-7337 (print)
ISSN 2663-2217 (online)

rameters of the taught model so that to maximally approximate the number predicted by the model to the true one, at least in a root-mean-square sense.

Self-learning algorithms work with data without markup. The goal of methods in this category is to determine some useful structure in the data. In particular, such a structure may be an estimate of the probability distribution that generated the input data, and partitioning the data into groups of similar examples in the case of clustering problem.

Thus, in the case of a self-learning problem for an input random vector **x** it is necessary to estimate the probability distribution P**(x)** or some properties of this distribution that are of interest. In supervised learning, a number **y** (class label) is available for each **x,** and the algorithm is trained to predict **y** for **x** by constructing an estimate P(**y**|**x**). A separate category is reinforcement learning, which teaches a sequence of actions in some environment to achieve a goal, using the environment's response to the actions [4].

Supervised learning algorithms when used with neural networks are able to work qualitatively in various tasks of classification, regression and pattern recognition within the framework of a complex system of evaluation of the technical condition of aircraft engines. However, they require large amounts of marked-up data for training. Without this, the accuracy of supervised learning algorithms decreases significantly. Preparing such data for each task is an extremely labor-intensive operation requiring manual labor and much time. For self-learning algorithms, on the contrary, such data markup is not required. Standard classification methods are not able to quickly and qualitatively adapt for new classes of data, which were not in the training samples. This property leads to the consideration of algorithms of information classification by a small number of examples (few-shot learning).

Self-learning methods are used for obtaining some useful information about data partitioning in contrast to supervised learning methods where such indication is the markup. An example of such useful information is dividing data into groups of similar examples, as in the case of the clustering problem. In the classical setting, the goal of self-learning algorithms is to find the "best" representation of data that preserves as much information about the input data as possible while being simpler than the original [5]. That is why the task of clustering is to identify the groups in the input data that will give this "best" representation.

2. The problem solution

If (Ω,F,P) is a probability space, T is a σ-algebra of subsets of Ω, P(X) is a probability distribution defined on the set of input data

$$X = \{x^1, x^2, ...\},$$

which is a subset of the Euclidean pro-space R^d ($d > 1$). We denote by 1...k the set of indices {1, 2, ..., k} and suppose that input dataset X is divided into k unknown subsets:

$$\left\{X_1^*, X_2^*, ..., X_k^*\right\}: X = \bigcup_{i\in\overline{1,k}} X_i^* .$$

The probability distribution P(X) can be represented by a mixture of distributions:

$$P(X) = \sum_{i=1}^{k} p_i P(X_i^*) ,$$

where p_i ($p_i > 0$) and $P(X_i^*)$, $i = \overline{1,k}$, are the corresponding probabilities and distributions.

Thus, the clustering problem is to find the optimal partition χ of the input dataset X into k non-empty clusters:

$$\chi(X) = \{X_1, X_2, \ldots, X_k\} : X = \bigcup_{i=1}^{k} \mathbf{X}_i$$

and

$$\mathbf{X}_i \cap \mathbf{X}_j = \varnothing, i \neq j .$$

The partitioning of χ is defined by the function $\gamma_\chi : X \to \overline{1,k}$, which assigns to each point X the cluster index:

$$X_i = \{x \in X \mid \gamma_\chi(x) = i\} ,$$

and the clustering problem can be written as:

$$\chi^* = \left\{X_1^*, X_2^*, \ldots, X_k^*\right\}.$$

In order to solve the clustering problem, we introduce a penalty function q_i, which determines the "proximity" to cluster i, $i = \overline{1,k}$. In order to obtain the optimal clustering, it is necessary to minimize the mean risk functional:

$$F(\chi) = E_\chi f(\chi, x) \to \min_{\chi} , \qquad (1)$$

where E_χ is the conditional expectation given a fixed set χ and $f(\chi, x) = q_{\gamma_\chi(x)}(\chi, x)$. We will interpret vectors θ_i, $i = \overline{1,k}$, as cluster centers, and matrices G_i, $i = \overline{1,k}$, as covariance matrices. Then the clustering quality (1) has the form:

$$F(\chi) = \sum_{i=1}^{k} \int_{X_i} q_i(\theta_i, \Gamma_i, x) P(dx) \to \min_{\chi} . \qquad (2)$$

For $i = \overline{1,k}$ and fixed $x \in X$ of each function $q_i(x)$ depends only on θ_i and G_i. Then we can choose a clustering rule as

$$X_i(\Theta, \Gamma) = (x \in X) : q_i(\theta_i, \Gamma_i, x) \le q_j(\theta_j, \Gamma_J, x),$$

$$j = \overline{1, i-1};$$

$$q_i(\theta_i, \Gamma_i, x) \le q_j(\theta_j, \Gamma_J, x),$$

$$j = \overline{i+1,k},\ i = \overline{i,k},$$

where $\Theta = (\theta_1, \theta_2, ..., \theta_k)$ is a (dk)-matrix;

G is a set consisting of k matrices $G_1, G_2, ..., G_k$, where

$$\Gamma_i \in R^{d\times d},\ i = \overline{1,k}.$$

A true partitioning of the input data into clusters, which minimizes (2), leads to the following rule, according to which **x** belongs to a particular cluster:

$$z = \arg\min_{i=\overline{1,k}} q_i(\theta_i, \Gamma_i, \mathbf{x}),$$

where $z = z(\theta, G, x)$ is the label of the cluster to which the data point x is assigned. Let $e_l \in R^k$ be a vector consisting of zeros and one at position z, then:

$$F(\Theta,\Gamma) = \int_X e_z^T q(\theta,\Gamma,x)P(dx) \to \min_{\Theta,\Gamma},$$

where $q(\theta,\Gamma,x) \in R^k$ is a vector of values

$$q_i(\theta_i,\Gamma_i,x),\ i = \overline{1,k}.$$

In important special case corresponds to a uniform distribution $P(\cdot)$ and a penalty function which is the square of the Mahalanobis distance

$$q_i(\theta_i,\Gamma_i,x) = (x-\theta_i)^T\Gamma_i^{-1}(x-\theta_i). \qquad (3)$$

One of the most common and widely used is the case of Gaussian Mixture Model (GMM) [6]:

$$f(\chi,x) = f(\Theta,\Gamma,x) = \sum_{i=1}^{k} p_i G(x\mid\theta_i,\Gamma_i),$$

where $G(x|_i,G_i)$ is the density of Gaussian distribution with mean $\theta_i \in R^d$ and covariance matrix G_i, $i = \overline{1,k}$.

Therefore, using the sequence of input data $\{x^1, x^2,...\}$ and a given value of k, we find the parameters $\theta_i \in R^d$ and G_i, $i = \overline{1,k}$, of the Gaussian distributions whose mixture has generated the sequence of input data.

The problem of finding the unknown parameters $\theta_i \in R^d$ and G_i, $i = \overline{1,k}$, is closely connected with the clustering problem, which is to find:

$$F(\chi) = \sum_{i=1}^{k}\int_{X_i}(x-\theta_i)^T\Gamma_i^{-1}(x-\theta_i)P(dx) \to \min_{\chi}.$$

For input data $\{x^1,x^2,...,x^n\}$ this functional takes the form:

$$F(\Theta,\Gamma) = \sum_{i=1}^{k}\sum_{x^j\in X_i}(x^j-\theta_i)^T\Gamma_i^{-1}(x^j-\theta_i) \to \min_{\Theta,\Gamma}, \qquad (4)$$

$$j = \overline{1,n}.$$

With the clustering problem statement introduced above, finding the optimal GMM parameters is equivalent to finding the optimal clustering parameters, since if the sequence $\{x^n\}$ is generated by a GMM with parameters Θ^* and Γ^*, then with these parameters the minimum of the following expression is achieved:

$$-\sum_{x\in X}\ln\left(\sum_{i=1}^{k} p_i G(x\mid\theta_i,\Gamma_i)\right),$$

which can be rewritten as:

$$\sum_{i=1}^{k}\sum_{x^j\in X}\left(-\ln p_i(2\pi)^{-\frac{d}{2}}|\Gamma_i|^{-1} + \right.$$

$$\left. + \frac{1}{2}(x^j-\theta_i)^T\Gamma_i^{-1}(x^j-\theta_i)\right),$$

$$i = \overline{1,n}.$$

Denote the first term in this expression by L and the second term by R, then

$$\arg\min_{\Theta,\Gamma} L = \arg\min_{\Theta,\Gamma} R = (\Theta^*,\Gamma^*) \text{ при } n\to\infty.$$

Then:

$$F(\Theta,\Gamma) = \sum_{i1=}^{k}|X_i|^{-1}\sum_{x^j\in X_i}(x^j-\theta_i)^T\Gamma_i^{-1}(x^j-\theta_i),$$

$$j = \overline{1,n}$$

and consequently

$$\arg\min_{\Theta,\Gamma} F(\Theta,\Gamma) = \arg\min_{\Theta,\Gamma} R = (\Theta^*,\Gamma^*) \text{ при } n\to\infty.$$

In [7] an interesting model of a mixture of Gaussian distributions with sparse parameters was suggested. This approach constructs a model with a large number of parameters, many of which are then shrunk to zero in the process of fitting to the data (hence the "sparseness" in the name). To model such properties it is necessary to use a priori distributions that have a large probability mass concentrated at zero and "long tails" to fit possible large parameters of the system. In order to achieve such behavior in the model from [7], the mean value is set as:

$$\theta_i \in R^d,\ \theta_{il} \sim N(0,\sigma_i^2),\ \sigma_i \sim C_+(0,1),\ l = \overline{1,d},$$

where $C_+(0,1)$ denotes the Cauchy distribution bounded on the positive axis with parameters 0 and 1. The diagonal covariance matrix can then be given as:

$$\Gamma_i = \mathrm{diag}\left(\sigma_1^2,\sigma_2^2,\ldots,\sigma_d^2\right),\ \sigma_j \sim C_+(0;0.5),\ j = \overline{1,d}.$$

This choice of distributions corresponds to the "horseshoe method" prior [8], the use of which allows us to satisfy the sparsity properties described above due to the fact that the Cauchy distribution has a large mass of probabilities centered around a mode. Such a distribution has tails that decay so that the parameters will decrease to zero, unless the data imply otherwise.

The weights $p_i \sim D(e_0, e_1,..., e_0)$, $i = \overline{1,k}$ according to [7] are supposed to belong to a Dirichlet distribution where the parameter $e_0 \sim G(\alpha_p, k\alpha_p)$ is obtained from a gamma distribution with mean k^{-1} and variance $(\alpha_p k^2)^{-1}$, $\alpha_p = 10$.

In spite of a large number of various approaches to

ISSN 1727-7337 (print)
ISSN 2663-2217 (online)

the clustering problem, the algorithms based on minimization of quantities determining dissimilarity of objects remain one of the most demanded. Among such approaches, the most popular is the k-means method that searches for a partition χthat minimizes the sum of squares of inter-cluster distances. Each cluster is characterized by a corresponding centroid θ_j, $i=\overline{1,k}$. In this case we will consider all matrices G to be unit matrices. Then (3) becomes a Euclidean distance. The k-means method optimizes the functional (4). This problem is NP-hard [9], and in the worst case the time complexity of the algorithm is exponential [10].

Based on the ideas of the k-means method, we can suggest a k-medoids algorithm. At the input of the algorithm X – data, k – number of clusters, maximum number of iterations. The output $\hat{\Theta}$ is an estimation of the medoid, χ – data partitioning.

Step 1: Randomly choose k initial medoids $\hat{\Theta}=\left(\hat{\theta}_1,\hat{\theta}_2,\ldots,\hat{\theta}_k\right)$ from elements of X.

Step 2: A non-medoid data point is randomly chosen.

Step 3: This point is made a medoid instead of the closest medoid.

Step 4: If the value of functional (4) with the new medoid has become smaller, then we leave that point as the medoid.

Steps 2, 3, and 4 are repeated until the medoids stop changing or the maximal number of iterations is reached.

This algorithm uses random initialization, which can lead to poor clustering quality. One of the common causes of this result is cluster collapse, where several clusters are too close together, while others begin to include "superfluous" elements.

Therefore the initial centroids are iteratively chosen so that they are as far away from each other as possible, and the algorithm then takes the following form:

Step 1. The first centroid is chosen randomly from the initial data.

Step 2. For each data element the Euclidean distance to the nearest centroid is calculated.

Step 3: The new centroid is selected among the elements of the original data with probability proportional to the square of the distance for this element, which was chosen at Step 2.

Steps 2 and 3 are repeated until there are k centroids.

One more algorithm which can be used at the decision of a problem of clustering of the input data at operation of system of an estimation of a technical condition of aviation engines is EM (expectation-maximization) algorithm [11]. It is based on likelihood maximization, when the model depends on hidden parameters. The EM algorithm can be used to estimate the parameters of a mixture of Gaussian distributions [2].

Let's consider a two-component GMM with diagonal covariance matrices and dispersions σ_1^2, σ_2^2:

$$f(\Theta,\Gamma,x)=(1-p)G\left(x\mid\theta_1,\sigma_1^2\right)+pG\left(x\mid\theta_2,\sigma_2^2\right),$$

and the likelihood function takes the form

$$\sum_{i=1}^{N}\ln\left[(1-p)G\left(x^i\mid\theta_1,\sigma_1^2\right)+pG\left(x^i\mid\theta_2,\sigma_2^2\right)\right].$$

We introduce a hidden parameter δ_i, on which depends to which component of GMM the data element x^i belongs: if $\delta_i = 0$, then $x^i \sim N\left(\theta_1,\sigma_1^2\right)$, and if $\delta_i = 1$, then $x^i \sim N\left(\theta_2,\sigma_2^2\right)$ for $i=\overline{1,N}$. Then the likelihood function is written as

$$\sum_{i=1}^{N}\left[(1-\delta_i)\ln G\left(x^i\mid\theta_1,\sigma_1^2\right)+\delta_i\ln G\left(x^i\mid\theta_2,\sigma_2^2\right)\right]+$$

$$+\sum_{i=1}^{N}\left[(1-\delta_i)\ln(1-p)+\delta_i\ln p\right].$$

Estimates θ_1, θ_2, σ_1, σ_2 are constructed from the corresponding samples. The estimate δ_i is constructed as:

$$\hat{\gamma}_i=P(\delta_i=1\mid\theta_1,\sigma_1,\theta_2,\sigma_2).$$

Then the algorithm can be represented as follows:

Step 1: $\hat{\theta}_1,\hat{\theta}_2$ are randomly selected from elements X, $\hat{\sigma}_1=1$, $\hat{\sigma}_2=1$, $p=0.5$.

Step 2: Classification:

$$\hat{\gamma}_i=\frac{\hat{p}G\left(x^i\mid\hat{\theta}_2,\hat{\sigma}_2^2\right)}{(1-\hat{p})G\left(x^i\mid\hat{\theta}_1,\hat{\sigma}_1^2\right)+\hat{p}G\left(x^i\mid\hat{\theta}_2,\hat{\sigma}_2^2\right)},\ i=\overline{1,N}.$$

Step 3: Maximization:

$$\hat{\theta}_1=\frac{\sum_{i=1}^{N}(1-\hat{\gamma}_i)x_i}{\sum_{i=1}^{N}(1-\hat{\gamma}_i)},$$

$$\hat{\theta}_2=\frac{\sum_{i=1}^{N}\hat{\gamma}_i x_i}{\sum_{i=1}^{N}\hat{\gamma}_i},$$

$$\hat{\sigma}_1=\frac{\sum_{i=1}^{N}(1-\hat{\gamma}_i)(x_i-\hat{\theta}_1)^2}{\sum_{i=1}^{N}(1-\hat{\gamma}_i)},$$

$$\hat{\sigma}_2=\frac{\sum_{i=1}^{N}\hat{\gamma}_i(x_i-\hat{\theta}_2)^2}{\sum_{i=1}^{N}\hat{\gamma}_i},$$

$$\hat{p} = \sum_{i=1}^{N} \frac{\hat{\gamma}_i}{N}.$$

Steps 2, 3 are repeated until convergence or until the maximum number of iterations is reached.

From the description of this algorithm, we see that the k-means method is its special case, in which each cluster is characterized only by corresponding centroid, and dispersion estimates are not constructed. The k-means method uses unambiguous binding of initial data elements to corresponding clusters, in contrast to EM-algorithm, which performs ambiguous binding due to estimation.

If the number of clusters is not known in advance, we can recommend using the proximity propagation method [12]. The basic idea of this method is that the input data are divided into groups based on how similar they are to each other.

So if X is given some similarity metric such that if $s(x^i,x^j) > s(x^i,x^k)$, then example x^i is more similar to example x^j than to x^k. An example of such a metric is $s(x^i,x^j) = -\|x^i - xj\|^2$. If the matrix $R = \{r_{i,k}\}$ describes how well the k-th example fits to be representative of the i-th example. Matrix $A = \{a_{i,k}\}$ describes how right it would be for the i-th example to choose the k-th as a representative. The output of the algorithm is cluster labels for each element of the original data $(c_1, c_2, ..., c_n)$.

Step 1: Matrices R and A are initialized with zero values.

Step 2:

$$\forall i,k : r_{i,k} = s(x^i x^k) - \max_{k' \neq k}\left(a_{i,k'} + s\left(x^i, x^{k'}\right)\right).$$

Step 3:

$$\forall i,k, i \neq k : a_{i,k} = \min\left(0, r_{k,k} + \sum_{i' \notin i,k} \max(0, r_{i',k})\right).$$

Step 4:

$$\forall k : a_{k,k} = \sum_{i' \neq k} \max(0, r_{i',k}).$$

Steps 2, 3, 4 are repeated until the maximum number of iterations is reached.

Step 5:

$$\forall i : c_i = \arg\max_k (a_{i,k} + r_{i,k}).$$

In many cases the input data of the aircraft engine condition estimation system has significant noise. Then it is better to use DBSCAN (Density-based spatial clustering of applications with noise) method for data clustering [13]. In this case clusters can be of any shape (in contrast to k-means method, where convex clusters are considered).

In describing this algorithm a number of notions are used:

– point p is called principal if at least t points are at a distance not exceeding ε;

– point q is directly reachable from p if the point q is at a distance at most ε from the point p. In this case, the point p must be the principal point;

– point q is called reachable from p if there exists a path $(p_1,p_2,...,p_n)$ where $p_1 = p$, $p_n = q$ and every point p_{i+1} is directly reachable from p_i;

– all points not reachable from the main points are defined as outliers;

– if p is a principal point, it is said to form a cluster together with all reachable points from it.

The DBSCAN algorithm is then written as follows:

Step 1: Among all points in X the main points are distinguished.

Step 2: For each principal point, the connectivity components in the neighborhood graph are defined, ignoring all non-main points.

Step 3: For each non-major point a cluster is found if it is ε-neighbor, otherwise the point is considered an outlier.

Another effective clustering method in the presence of significant noise is the variational Bayesian method for mixing Gaussian distributions. This method is an extension of the EM method that can determine the number of components in a mixture [14]. The variational method adds regularization by integrating information from a priori distributions. This makes the method more stable but requires more operations than EM algorithm.

Evaluating the quality of the constructed partitioning of the raw data into clusters is a difficult task. At present, there are a lot of different criteria for its solution [15], which can be divided into two groups in accordance with the availability of information on the partitioning of the original data into clusters: when the true partitioning is known, and when it is unknown.

The Adjusted Rand Index (ARI) is chosen as the first criterion [16]. This metric allows to evaluate the quality of clustering when the partitioning of data into clusters is known in advance. Let T be the true partitioning, C be the clustering result, then: a is the number of pairs of elements that at both T and C lie in the same cluster; b is the number of pairs of elements that lie in different clusters at T, and in different clusters at C.

The Rand Index (RI) is given as

$$RI = \frac{a+b}{C_2^n},$$

then

$$ARI = \frac{RI - E(RI)}{\max(RI) - E(RI)}.$$

Thus, this criterion allows us to show by how much the constructed partition of the data

$$\chi = \{X_1, X_2, \ldots, X_k\}.$$

is close to the benchmark partition

ISSN 1727-7337 (print)
ISSN 2663-2217 (online)

$$\chi^* = \{\mathbf{X}_1^*, \mathbf{X}_2^*, \ldots, \mathbf{X}_k^*\}.$$

The ARI criterion takes on a value from -1 to 1: the closer to 1, the higher the quality of clustering.

The second criterion used is Normalized Mutual Information (NMI) [17]. This metric measures the consistency between the true partitioning of the data and the partition obtained using the clustering algorithm.

Let

$$U = \{U_1, U_2, ..., U^k\}$$

and

$$V = \{V_1, V_2, ..., V_k\}$$

are two partitions of data of volume N into k clusters. Then the mutual information (MI) between U and V is given as:

$$MI(\mathbf{U}, \mathbf{V}) = \sum_{i=1}^{|\mathbf{U}|} \sum_{j=1}^{|\mathbf{V}|} \frac{\left|U_i \cap V_j\right|}{N} \log\left(\frac{N\left|U_i \cap V_j\right|}{\left|U_i\right|\left|V_j\right|}\right).$$

NMI is defined as:

$$NMI(\mathbf{U}, \mathbf{V}) = 2\frac{MI(\mathbf{U}, \mathbf{V})}{H(\mathbf{U}) + H(\mathbf{V})},$$

where

$$H(\mathbf{U}) = -\sum_{i=1}^{|\mathbf{U}|} \frac{\left|U_i\right|}{N} \log\left(\frac{\left|U_i\right|}{N}\right)$$

and

$$H(\mathbf{V}) = -\sum_{j=1}^{|\mathbf{V}|} \frac{\left|V_j\right|}{N} \log\left(\frac{\left|V_j\right|}{N}\right).$$

NMI takes a value between 0 and 1: the closer the value is to 1, the higher the clustering quality is.

Another effective criterion for evaluating the quality of clustering is the Fowlkes-Mallows Index (FMI) [18]:

$$FMI = \frac{TP}{\sqrt{(TP + FP)(TP + FN)}}.$$

Where TP is the number of element pairs that belong to the same cluster in the true and in the obtained partitioning; FP is the number of element pairs that belong to the same cluster in the true partitioning, but different in the obtained partitioning; FN is the number of element pairs that belong to the same cluster in the obtained partitioning, but different in the true partition. FMI criterion takes a value between 0 and 1: the closer to 1, the higher the quality of clustering.

Conclusion

The considered methods of clustering make it possible to process a fairly large volume of information coming from dozens of sensors installed in various systems of gas turbine engines and sharply reduce and compress large arrays of this parametric information, making it compact and visual. A mathematically justified general randomized algorithm of stochastic approximation for clustering in the mixture model of Gaussian distributions, which can work qualitatively under unknown but limited disturbances together with the methods considered in [19]. At the same time, these methods allow us to implement meta-learning on a small amount of initial information, which is important for updating the diagnostic engine model after each flight.

References (GOST 7.1:2006)

1. The elements of statistical learning: data mining, inference and prediction [Text] / T. Hastie, R. Tibshirani, J. Friedman [et al.] // The Mathematical Intelligencer. – 2005. – Vol. 27, no. 2. – P. 83-85.

2. Bishop, C. M. Pattern Recognition and Machine Learning [Text] / C. M. Bishop. – Springer, 2006. – 734 p.

3. Learning long-term dependencies with gradient descent is difficult [Text] / Y. Bengio, P. Simard, P. Frasconi [et al.] // IEEE Transactions on Neural Networks. – 1994. – Vol. 5, no. 2. – P. 157-166.

4. Sutton, R. S. Reinforcement Learning: an Introduction [Text] / R. S. Sutton, A. G. Barto. – MIT press, 2018. – 552 p.

5. Barlow, H. B. Unsupervised learning [Text] / H. B. Barlow // Neural Computation. – 1989. – Vol. 1, no. 3. – P. 295-311.

6. McLachlan, G. J. Mixture Models: Inference and Applications to Clustering [Text] / G. J. McLachlan, K. E. Basford // Journal of the Royal Statistical Society. Series C (Applied Statistics). – M. Dekker New, York, 1989. – Vol. 38, Iss. 2. – P. 384-385. DOI: 10.2307/2348072.

7. Dahlin, J. Sparse Bayesian ARX models with flexible noise distributions [Text] / J. Dahlin, A. Wills, B. Ninness // IFAC-Papers On Line. – 2018. – Vol. 51, no. 15. – P. 25-30.

8. Carvalho, C. M. The horseshoe estimator for sparse signals [Text] / C. M. Carvalho, N. G. Polson, J. G. Scott // Biometrika. – 2010. – Vol. 97, no. 2. – P. 465-480.

9. Megiddo, N. On the complexity of some common geometric location problems [Text] / N. Megiddo, K. J. Supowit // SIAM Journal on Computing. – 1984. – Vol. 13, no. 1. – P. 182-196.

10. Shindler, M. Fast and accurate k-means for large datasets [Text] / M. Shindler, A. Wong, A. W. Meyerson // Advances in Neural Information Processing Systems. – 2011. – P. 2375-2383.

11. Dempster, A. P. Maximum likelihood from incomplete data via the EM algorithm [Text] / A. P. Dempster, N. M. Laird, D. B. Rubin // Journal of the Royal Statistical Society: Series B (Methodological). – 1977. – Vol. 39, no. 1. – P. 1-22.

12. Frey, B. J. Clustering by passing messages between data points [Text] / B. J. Frey, D. Dueck // Science. – 2007. – Vol. 315, no. 5814. – P. 972-976.

13. Density-based algorithm for dis-covering clusters in large spatial databases with noise. [Text] / M. Ester, H.-P. Kriegel, J. Sander, X. Xu [et al.] //

KDD. – 1996. – Vol. 96. – P. 226-231.

14. Attias, H. Inferring parameters and structure of latent variable models by variational Bayes [Text] / H. Attias // Proceedings of the 15th conference on Uncertainty in Artificial Intelligence. – Morgan Kaufmann Publishers Inc. 1999. – P. 21-30.

15. Pfitzner, D. Characterization and evaluation of similarity measures for pairs of clusterings [Text] / D. Pfitzner, R. Leibbrandt, D. Powers // Knowledge and Information Systems. – 2009. – Vol. 19, no. 3. – P. 361.

16. Hubert, L. Comparing partitions [Text] / L. Hubert, P. Arabie // Journal of Classification. – 1985. – Vol. 2, no. 1. – P. 193-218.

17. Strehl, A. Cluster ensembles – a knowledge reuse framework for combining multiple partitions [Text] / A. Strehl, J. Ghosh // Journal of Machine Learning Research. – 2002. – Vol. 3. – P. 583-617.

18. Fowlkes, E. B. A method for comparing two hierarchical clusterings [Text] / E. B. Fowlkes, C. L. Mallows // Journal of the American Statistical Association. – 1983. – Vol. 78, no. 383. – P. 553-569.

19. Tamargazin, A. A. Neural net-work interpolation parameters of a multi-mode dynamic model of the aircraft engine [Text] / A. A. Tamargazin, L. B. Pryimak // Авіаційно-космічна техніка і технологія. – 2020. – № 7(167). – С. 98-104. DOI: 10.32620/aktt.2020.7.14.

References (BSI)

1. Hastie, T., Tibshirani, R., Friedman, J., Franklin, J. The elements of statistical learning: data mining, inference and prediction, *The Mathematical Intelligencer*, 2005, vol. 27, no. 2, pp. 83-85.

2. Bishop, C. M. *Pattern Recognition and Machine Learning*. Springer Publ., 2006. 734 p.

3. Bengio, Y. Learning long-term dependencies with gradient descent is difficult, *IEEE Transactions on Neural Networks*, 1994, vol. 5, no. 2, pp. 157-166.

4. Sutton, R. S., Barto, A. G. *Reinforcement Learning: an Introduction*. MIT press Publ., 2018. 552 p.

5. Barlow, H. B. Unsupervised learning, *Neural Computation*, 1989, Vol. 1, no. 3, pp. 295-311.

6. McLachlan, G. J., Basford K. E. Mixture Models: Inference and Applications to Clustering. *Journal of the Royal Statistical Society. Series C (Applied Statistics),* M. Dekker New York, 1989, vol. 38, iss. 2, pp. 384-385. DOI: 10.2307/2348072.

7. Dahlin, J., Wills, A., Ninness, B. Sparse Bayesian ARX models with flexible noise distributions. *IFAC-Papers On Line*, 2018, vol. 51, no. 15, pp. 25-30.

8. Carvalho, C. M., Polson, N. G., Scott, J. G. The horseshoe estimator for sparse signals. *Biometrika*, 2010, vol. 97, no. 2, pp. 465-480.

9. Megiddo, N., Supowit, K. J. On the complexity of some common geometric location problems. *SIAM Journal on Computing*, 1984, vol. 13, no. 1, pp. 182-196.

10. Shindler, M., Wong, A., Meyerson, A. W. Fast and accurate k-means for large datasets. *Advances in Neural Information Processing Systems*, 2011, pp. 2375-2383.

11. Dempster, A. P., Laird, N. M., Rubin, D. B. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society: Series B (Methodological)*, 1977, vol. 39, no. 1, pp. 1-22.

12. Frey, B. J., Dueck D. Clustering by passing messages between data points. *Science*, 2007, vol. 315, no. 5814. pp. 972-976.

13. Ester, M. A, Kriegel, H.-P., Sander. J., Xu, X. Density-based algorithm for dis-covering clusters in large spatial databases with nois. *KDD*, 1996, vol. 96, pp. 226-231.

14. Attias, H. Inferring parameters and structure of latent variable models by variational Bayes. *Proceedings of the 15th conference on Uncertainty in Artificial Intelligence*, Morgan Kaufmann Publishers Inc. 1999, pp. 21-30.

15. Pfitzner, D., Leibbrandt, R., Powers, D. Characterization and evaluation of similarity measures for pairs of clusterings. *Knowledge and Information Systems*, 2009, vol. 19, no. 3, pp. 361.

16. Hubert, L., Arabie, P. Comparing partitions. *Journal of Classification*, 1985, vol. 2, no. 1, pp. 193-218.

17. Strehl, A., Ghosh, J. Cluster ensembles – a knowledge reuse framework for combining multiple partitions. *Journal of Machine Learning Research*, 2002, vol. 3, pp. 583-617.

18. Fowlkes, E. B., Mallows, C. L. A method for comparing two hierarchical clusterings. *Journal of the American Statistical Association*, 1983, vol. 78, no. 383, pp. 553-569.

19. Tamargazin, A. A., Pryimak, L. B. Neural network interpolation parameters of a multi-mode dynamic model of the aircraft engine. *Aviacijno-kosmicna tehnika i tehnologia – Aerospace technic and technology*, 2020, no. 7 (167), pp. 98-104. DOI: 10.32620/aktt.2020.7.14

Надійшла до редакції 16.06.2021, розглянута на редколегії 16.08.2021

МЕТОДИ КЛАСТЕРИЗАЦІЇ ПАРАМЕТРІВ ПРИ СТВОРЕННІ НЕЙРОМЕРЕЖЕВИХ БАГАТОРЕЖИМНИХ ДИНАМІЧНИХ МОДЕЛЕЙ АВІАЦІЙНИХ ДВИГУНІВ

О. А. Тамаргазін, Л. Б. Приймак, В. В. Шостак

Наявність на сучасних авіаційних газотурбінних двигунах десятків, а то й сотень датчиків для безперервної реєстрації різноманітних параметрів їх роботи дає можливість збирати й обробляти великі обсяги інформації. Це стимулює розвиток моніторингових і діагностичних систем. У той же час наявність великих обсягів інформації не завжди є достатньою умовою для вироблення адекватних управлінських рішень, особливо у випадку оцінки технічного стану авіаційних двигунів. При цьому треба враховувати, що авіаційні двигуни це об'єкти, які відносяться до індивідуалізованим, тобто до таких, які є у своєму роді унікальними. Тому теорія створення систем оцінки технічного стану авіаційних двигунів формується на тлі розвитку су-

часних нейромережевих технологій і потребує формування специфічного методологічного апарата. Із цих позицій у статті розглядаються методи, які використовуються при проведенні кластеризації вхідної інформації одержуваної при роботі сучасних систем оцінки й прогнозування технічного стану авіаційних газотурбінних двигунів. Особливо актуальна ця задача при створенні нейромережевих багаторежимних моделей авіаційних двигунів використовуваних у системах оцінки технічного стану для ідентифікації можливих відмов і пошкоджень. У статті розглядаються метричні, оптимізаційні та рекурентні методи кластеризації вхідних даних. При цьому основна увага приділена порівнянню методів кластеризації з метою вибору найбільш ефективних з них при роботі систем оцінки технічного стану авіаційних двигунів і придатних для реалізації систем з мета-навчанням. Реалізація методів кластеризації вхідних даних дозволяє робити розбивку діагностичних образів об'єктів не по одному параметру, а по цілому набору ознак. Крім того, кластерний аналіз на відміну від більшості математико-статистичних методів не накладає ніяких обмежень на вид розглядаємих об'єктів, і дозволяє розглядати множину вхідних даних практично довільної природи, що дуже важливо при оцінці технічного стану авіаційних двигунів. У той же час кластерний аналіз дозволяє розглядати досить великий обсяг інформації й різко скорочувати, стискати великі масиви параметричної інформації, робити їх компактними й наочними.

Ключові слова: авіаційний двигун; діагностика; нейрона мережа.

МЕТОДЫ КЛАСТЕРИЗАЦИИ ПАРАМЕТРОВ ПРИ СОЗДАНИИ НЕЙРОСЕТЕВЫХ МНОГОРЕЖИМНЫХ ДИНАМИЧЕСКИХ МОДЕЛЕЙ АВИАЦИОННЫХ ДВИГАТЕЛЕЙ

А. А. Тамаргазин, Л. Б. Приймак, В. В. Шостак

Наличие на современных авиационных газотурбинных двигателях десятков, а то и сотен датчиков для непрерывной регистрации разнообразных параметров их работы даёт возможность собирать и обрабатывать большие объёмы информации. Это стимулирует развитие мониторинговых и диагностических систем. В то же время наличие больших объёмов информации не всегда является достаточным условием для выработки адекватных управленческих решений, особенно в случае оценки технического состояния авиационных двигателей. При этом надо учитывать, что авиационные двигатели – это объекты, которые относятся к индивидуализированным, т. е. к таким которые являются в своём роде уникальными. Потому теория создания систем оценки технического состояния авиационных двигателей формируется на фоне развития современных нейросетевых технологий и нуждается в формировании специфического методологического аппарата. С этих позиций в статье рассматриваются методы, которые используются при проведении кластеризации исходной информации, получаемой при работе современных систем оценки и прогнозирования технического состояния авиационных газотурбинных двигателей. Особенно актуальна эта задача при создании нейросетевых многорежимных моделей авиационных двигателей, используемых в системах оценки технического состояния для идентификации возможных отказов и повреждений. В статье рассматриваются метрические, оптимизационные и рекуррентные методы кластеризации исходных данных. При этом основное внимание уделено сравнению методов кластеризации с целью выбора наиболее эффективных из них при работе систем оценки технического состояния авиационных двигателей и пригодных для реализации систем с метаобучением. Реализация методов кластеризации исходных данных позволяет производить разбиение диагностических образов объектов не по одному параметру, а по целому набору признаков. Кроме того, кластерный анализ в отличие от большинства математико-статистических методов не накладывает никаких ограничений на вид рассматриваемых объектов, и позволяет рассматривать множество исходных данных практически произвольной природы, что очень важно при оценке технического состояния авиационных двигателей. В тоже время кластерный анализ позволяет рассматривать достаточно большой объем информации и резко сокращать, сжимать большие массивы параметрической информации, делать их компактными и наглядными.

Ключевые слова: авиационный двигатель; диагностика; нейронная сеть.

Тамаргазин Александр Анатолиевич – д-р техн. наук, зав. каф. технологий аэропортов, Национальный авиационный университет, Киев, Украина.

Приймак Людмила Борисовна – канд. техн. наук, доц. каф. технологий аэропортов, Национальный авиационный университет, Киев, Украина.

Шостак Валерий Владиславович – аспирант кафедры технологий аэропортов, Национальный авиационный университет, Киев, Украина.

Aleksandr Tamargazin – Doctor of Technical Science, Head of Dept. of Airport Technologies, National Aviation University, Kiev, Ukraine,
e-mail: avia_icao@mail.ru, ORCID: 0000-0002-9941-3600.

Liudmyla Pryimak – Candidate of Technical Science, Assistant Professor of Dept. of Airport Technologies, National Aviation University, Kiev, Ukraine,
e-mail: ludmila-joy@ukr.net, ORCID: 0000-0002-3354-9820.

Valerii Shostak – PhD student of Dept. of Airport Technologies, National Aviation University, Kiev, Ukraine,
e-mail: Shostak1997.v@gmail.com, ORCID: 0000-0001-5508-9223.