# The classification of technical equipment states, using a neural nets approach

**Master Thesis**

Submitted to the
Department of Computer Science and Languages at
Anhalt University of Applied Sciences

in fulfillment of the requirements for the degree of
Master of Science

Obukhov Egor
(Matr. Nr.: 4063180)
supervisor:
Prof. Dr. G. Schwenzfeger

**Annotation**

The Maser thesis is devoted to developing a model to technical state of gas turbine engine estimation. The approaches to preparation data, especially to handle unbalanced data were presented in the thesis. In order to efficient estimation of model performance, the special metric was chosen.

Goal of the master thesis is analyzing of monitoring parameters data and developing a model of technical state of GTE estimation based on the data.

**Content**

## Introduction

The problem of gas turbine engine defects predetermination always has been actual. In traditional way, the diagnosis of engines goes by using models, created based on statistical processing data of monitoring and physical regularities. To get this data, there are special detectors measuring parameters of engine while it's working. The fail of data with monitoring parameters consist of such information as: temperature and pressure of air on input of engine, temperature and pressure of air on output of turbine, temperature of oil, operating time and so on. The number of monitoring parameters can reach the 100 and more.

The main purpose of engineer - diagnostician is to determine engine defects before its crashed or before maintenance inspection, by using data of monitoring parameters. So, as it said before to solve the task, the methods based on physical regularities are applied. Each defect has some deviation of parameters of working engine, so by analyzing the deviations it's possible to determine the reason of appearance the defect. It is a really difficult task, because of significant amount of information and difficulties of associates between defects and monitoring parameters [1].

The present project offers to solve the task by means approach based on neural nets modeling.

In the case neural net modeling means to create a model or ensemble of models to classify different state of engine. The state is characterized by values of monitoring parameters. So to determine defective state of an engine, it is necessary to determine the parameters which represent that defect. Thereby, will be created the training set to modeling the classifiers of estimation technical states of gas turbine engine.

As input data will be values of monitoring parameters with combination of class corresponding to them.

As internal data will be some weights of net's neurons, its structure and configuration parameters.

As output data will be used the assessment of technical state of an engine.

## 1. The technologies of Data Mining and using of it to estimation of gas turbine engine technical state

The methods of Data Mining allow to solve a lot of tasks, the main tasks are classification and clusterization. Let's consider them in more details:

Classification task is determination of object class using some characteristic of the object. The main think that the number of classes is known before classification.

Clusterization task is searching of independent groups (clusters) and characteristics of it. The task solving allows to better understand of the data structure.

There is another type of classification the tasks. It's based on type of solving the tasks: supervised learning and unsupervised learning. So, if make some association of it, then the task of classification fits to supervised learning and to unsupervised learning fit clusterization task. Let's consider the types of learning in more detail.

There are several steps of solving the supervised learning task. The first of all it is necessary to make a classifier model. Next step is learning of the classifier on a training data. That's mean checking the efficiency of quality the model and keeping to learn the model until it has the required quality, if it is possible on the training data.

So, unsupervised learning task is a technique for extracting information from input data. Obviously, if in the data is some structure or patterns, then the clusterization model must find it, and the task don't suggest any training of the clusterization model. The advantage of the task is a possibility of its solving without any initial information about input data.

Thereby, in the project of models developing of technical engine state estimation was solved both of the tasks. The models based on Neural Nets and Decision Trees were used as method of classification. The clusterization task was

solved to make some preparation of input data (monitoring parameters). The K-means method was used as clusterization algorithm.

### 1.1 Neural Net

Let's consider Neural Nets as a classification method allowing to estimate technical state of an engine. Neural Nets is a class of models, it's based on biological aspects of human's brain working. So after step of learning it is able to solve some different task of data mining.

The process of learning this type of models means iterative processing of input data, correction of model's weights in a way of the best prediction efficiency on training data. After training on a data, the network is ready to making predictions.

The one of the main advantages of the models is that the models can theoretically approximate any continuous function and that's way to a researcher is not necessary to make a hypothesis concerning the model. However it has such disadvantage that the resulting decision of modeling depends on initial configuration of network and it is difficult to practical interpret that in a traditional analytic terminus [2].

### 1.2 Decision Tree

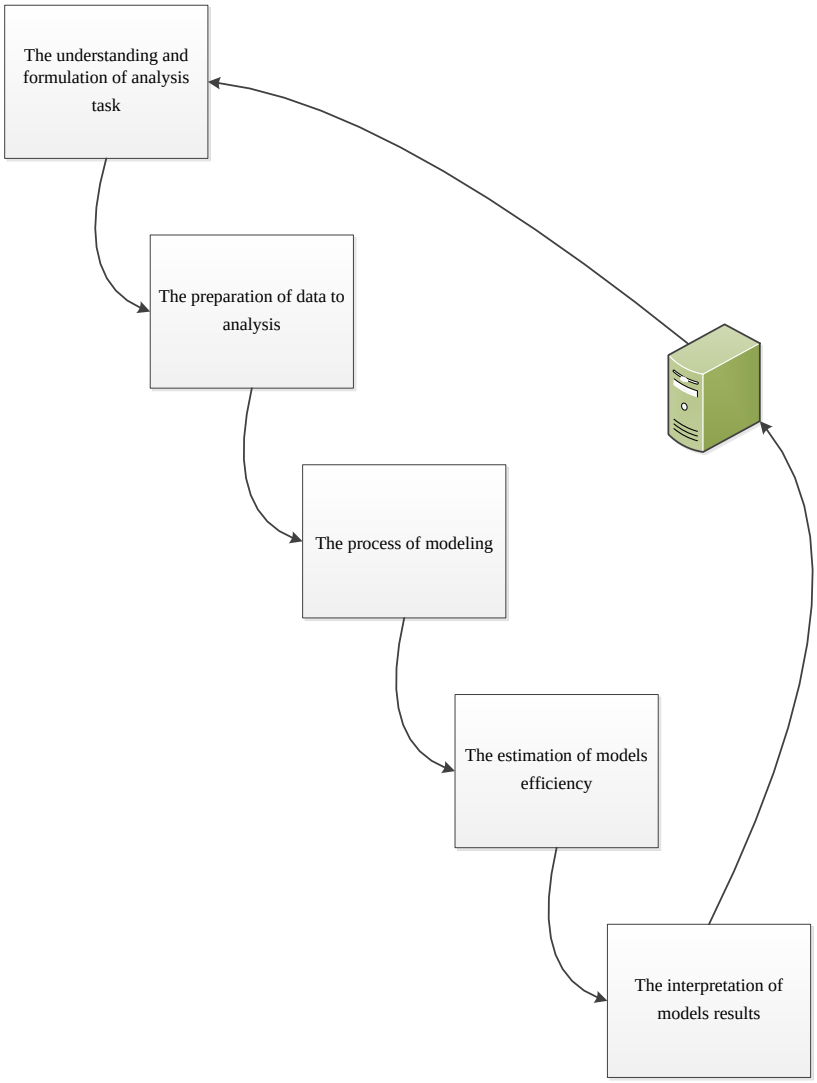
A decision tree is a tree-like graph or model. It is like inverted tree, in top of it has a root and it grows downstairs. The tree has nodes and as it said before, the main node is called root node. The nodes that do not have child nodes are called the leaf nodes. Each non-leaf node corresponds one of the input attribute. For example if we have such variable of input data as temperature of gas, then in the tree will be corresponding non-leaf node. As that variable is numerical, it outgoes with some specified ranges. Concerning to leaf node it assume possible values of target (label) attribute such as "true" or "false" [3,4].

In order to generate a decision tree algorithm of recursive partitioning is used. Recursive partitioning means repeatedly splitting on the values of attributes.

In every recursion the algorithm follows the following steps: choosing an attribute to splitting, processing of example set based on type of attributes, creating the resulted tree. There are several conditions to stop the recursion or splitting, for example no one of attributes doesn't reach a specific threshold. This can be adjusted using corresponding parameter.

### 1.3 Formalization of main steps of Data Mining

For effective conducting of data analysis, just applying of Data Mining methods is not enough, though the step of modeling is the main one [5]. The whole process consists of several steps. Let's consider them to determine that there are a lot of problem not only on modeling step and it is necessary to take attention on the whole process, picture 1.




Picture 1 The steps of Data Analysis process

On the first step it is important to formulate the goals and chose the methods to reach the goals. All of the decisions will have influence on the final efficiency.

The second step consists of preparation of data. The process will be considered in more details in fathers parts of the project.

Third step is process of modeling. It could be the applying of difficult combination of different methods the making simple and fast models to express analysis. So it is useful to make analysis of data from other points of view.

The next step is estimation of modeling efficiency. The really simple and often usable methods are based on splitting data set into two parts. One part is usually bigger then another one. On the bigger part, a model of Data Mining is trained, on the other part the model is checked. The efficiency of modeling can be estimated based on the difference of accuracy between testing and training groups.

The last step is interpretation of resulted models to use that in making decisions, making some rules and so on. Here is some integration with experts systems. As much efficient is the step will be the efficient of solving the current task.

## 2. Estimation of methods, which are used to preparation of data of monitoring parameters

### 2.1 Preparation of data

In order to apply any method of Data Mining to data, the data is necessary to prepare. The preparation of data means excluding not efficient variables, such as identical variables. The number of variables can be a really large, so including all of the variables will significantly increase the time of computing.

Thereby, as preprocessing of data it is necessary to take most important variables in context of current research. To make this preparation of data, statistical methods based on correlation analysis, linear regression are used. Those methods allow to estimate influence of one variable to other.

At this way, data cleaning of table columns (variables - attributes) is processed. At the similar way it can be useful to make preprocessing of data by cleaning table rows.

#### 2.1.1 Handling of unbalanced data sets

A data set used to train models of estimation technical state of gas turbine engine is unbalanced data set. It means there is a big bias the amount of instances of the classes. For example if we have two classes, the first one contains 100 instances and the second one 1000 instances, then the presented data set is unbalanced with 1 to 10 rate. In the case first class is minority class, the other class is majority class.

Most of machine learning algorithms work better, when both classes of the training set have an equal number of elements. If the number of instances of one class is far different from the other, then problems appear. This is the best illustrated with the following example:

If consider the current data set (parameter monitoring) and train a machine learning algorithm on the data, suppose there are two possible outputs as follows:

1) 10 instances of minority class and 20 of the majority class are misclassified.

2) 2 instances of minority class and 60 of the majority class are misclassified.

If we calculate the performance of the model by the amount of misclassified examples, then obviously that first case is better. However in terms of correct classification of minority class (especially when the class has priority), then second case is the best choice. Thereby it is important to have a correct metric for efficient estimation of the model performance [6].

Unbalanced data sets are a special case for classification problems. This type of sets supposes a new challenging problem for Data Mining, since standard classification algorithms usually consider a balanced training set. So, the question is how to handle with it.

There are different ways to handle unbalanced data. Let's consider several of those, which could be useful for the specific task:

1) In the first approach, it is necessary to divide the major class into L distinct clusters, then train L classifiers, where each classifier is trained on only one of the distinct clusters, but on all of the data from the minority class. To be clear, the data from the minority class are used in the training of all L classifiers. Finally, use ensemble of the L learned classifiers as a final classifier.

2) This is similar to number (1), but a little different. Let N be the number of samples in the minority class. Cluster the majority class into N clusters (agglomerative, K-means clustering etc.), and use the resulting cluster mediods as the training data for the majority class. To be clear, you throw out the original training data from the majority class, and use the mediods instead. Finally, the classes are balanced.

3) The third one is based on boosting. The algorithm trains the first learner, L1, on the original data set. The second learner, L2, is trained on a set on which L1 has around 50% chance to be correct. The third learner, L3, is trained on the cases on which L1 and L2 disagree. As output, return the majority of the classifiers. Why it improves, the classification can be found in [7,8].

In case of application this method on the unbalanced data. Let L1 return always true (majority class). L2, is being trained, where L1 inconvenient. L3 trained, when L1 and L2 disagree, that is L2 predicts false (minority class). Therefore, false (minority class) prediction can be only when both L2 and L3 predicts false.

Whatever method one can use will help in some ways, but hurt in others. To improve the efficiency, one can train separate models using all of the methods listed above, and then perform model averaging over all of them.

The methods are not applied yet, and on the stage of research, the task was to find out the ideas and approaches, which could be useful to the task. In future, it is necessary to apply the methods to prove the efficiency and suitability of it.

Thereby, the results of efficiency estimation of those approaches are presented in further parts of the thesis.

### 2.1.2 Analysis of quality problem and number of input variables

The data are recorded from detectors of control of monitoring parameters and have a large dimensionality around 40 attributes. It makes high requirements to number of elements of input data. In order to reduce the dimensionality of the data and find out the effective number of variables, we applied the following approaches:

- Correlation analysis [9].

The correlation analysis was applied to estimate the association between the output, target value and input variables. It was done to exclude less important input

variables. So the main idea of the analysis is to find some variables, which least of all correlated with output result.

To conduct the correlation analysis the following steps were performed:

1) Choosing the input and output variables
2) Configuration of correlation parameters;
3) Excluding insignificant variables;
4) Analysis of the result.

Let's consider applying of the methods on the data of monitoring parameters of gas turbine engine. To perform the analysis Deductor 5.3 was used. In the beginning we have data set with 40 attributes, one of them is target attribute or label attribute, the attribute is technical state of engine. Thereby, it is possible to make around 40 pairs: input attribute and label attribute. The correlation is a number between -1 and +1 that measures the degree of association between two attributes. In the case the correlation analysis provided the value of association between label attribute (technical state) and one, every time different input attribute. To estimate the degree of correlation of the attributes in table 1 presented interpretation of correlation values

Table 1 Interpretation of correlation values

| Negative value | Positive value | Interpretation |
|---|---|---|
| -0,2 <V< 0,2 | | Very low correlation |
| -0,5<V<-0,2 | 0,2<V<0,5 | Low correlation |
| -0,7<V<-0,5 | 0,5<V<0,7 | Middle value of correlation |
| -0,9<V<-0,7 | 0,7<V<0,9 | High correlation |
| V < -0,9 | V > 0,9 | Very high correlation |

On the next step method of computing correlation was chosen as Pirson's method and values of correlation between each of input variables and label variable were computed. The values of correlation are presented on picture 2 on crossing of rows with the input variables and column with the technical state.

| № | Input variables | Technical state | Value |
|---|---|---|---|
| 39 | I_KTT | | 0,175 |
| 30 | SKVv | | 0,184 |
| 1 | Nv | | -0,239 |
| 10 | Vk_rk | | 0,294 |
| 29 | POSv | | 0,299 |
| 36 | I_VBVR | | -0,353 |
| 37 | I_KBVR | | 0,357 |
| 38 | I_KPK | | 0,368 |
| 20 | tm_v | | -0,381 |
| 11 | Vk_zp | | 0,382 |
| 14 | Gt | | -0,424 |
| 27 | Gskv | | -0,436 |
| 3 | tv | | 0,452 |
| 24 | tk | | -0,479 |
| 5 | Tt | | -0,483 |
| 21 | tm_kvd | | -0,506 |
| 2 | Avna | | 0,513 |
| 31 | Nv_Popr | | -0,514 |
| 32 | Tt_prog | | -0,522 |
| 17 | Pt1k | | 0,538 |
| 8 | Vv_rk | | 0,547 |
| 35 | I_KPVD | | 0,561 |
| 4 | Pv_v | | 0,581 |
| 12 | H | | 0,603 |
| 28 | ZPV2_o | | 0,607 |
| 22 | Pv | | 0,634 |
| 23 | Pk | | 0,666 |
| 9 | Vv_zp | | 0,680 |
| 26 | PtPv | | 0,693 |
| 7 | tm_tnd | | 0,693 |
| 34 | I_VSUT | | 0,696 |
| 6 | tm_tvd | | 0,713 |
| 16 | Pt | | 0,717 |
| 19 | Psuf | | 0,736 |
| 25 | tm_gp | | 0,756 |
| 18 | Pm | | 0,761 |
| 15 | Tgg | | 0,771 |
| 33 | Nk_prog | | 0,777 |
| 13 | M | | 0,859 |

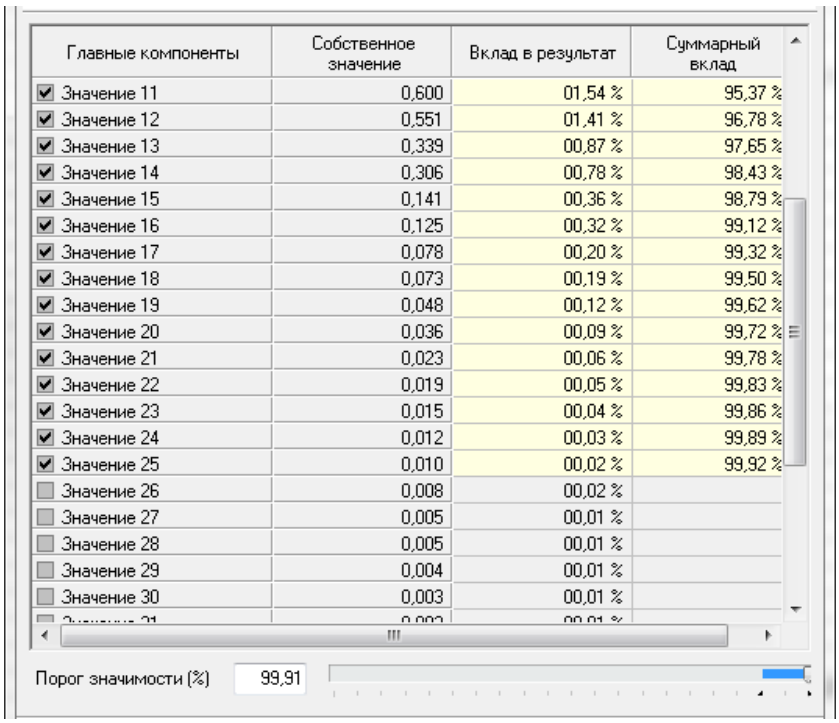
Picture 2 Matrix of correlation

According to the table 1 we decided to exclude some variables with value of correlation lower than 0.5. Thereby, 15 variables which correlated to label attribute with value lower than the middle rate were excluded.

- Compare to correlation analysis, the factor analysis [10] considers association not only between two variables. The goal of factor analysis is to reduce of factors dimensionality. The main idea of the analysis is to select new factors which more adequate represent the input data.

In the similar way let's estimate the efficiency of the methods on data set of monitoring parameters by using Deductor 5.3. After importing the input data to the analysis, method of factor's decision was chosen as Varimax method (the method is more usable in most cases, because it provides easier interpretation of factors). The parameter of choosing factors number is established as 25.

So, the number of variables marked as 25 and the principal components were computed. The results of it presented in table on picture 3.



| Главные компоненты | Собственное значение | Вклад в результат | Суммарный вклад |
|---|---|---|---|
| Значение 11 | 0,600 | 01,54 % | 95,37 % |
| Значение 12 | 0,551 | 01,41 % | 96,78 % |
| Значение 13 | 0,339 | 00,87 % | 97,65 % |
| Значение 14 | 0,306 | 00,78 % | 98,43 % |
| Значение 15 | 0,141 | 00,36 % | 98,79 % |
| Значение 16 | 0,125 | 00,32 % | 99,12 % |
| Значение 17 | 0,078 | 00,20 % | 99,32 % |
| Значение 18 | 0,073 | 00,19 % | 99,50 % |
| Значение 19 | 0,048 | 00,12 % | 99,62 % |
| Значение 20 | 0,036 | 00,09 % | 99,72 % |
| Значение 21 | 0,023 | 00,06 % | 99,78 % |
| Значение 22 | 0,019 | 00,05 % | 99,83 % |
| Значение 23 | 0,015 | 00,04 % | 99,86 % |
| Значение 24 | 0,012 | 00,03 % | 99,89 % |
| Значение 25 | 0,010 | 00,02 % | 99,92 % |
| Значение 26 | 0,008 | 00,02 % | |
| Значение 27 | 0,005 | 00,01 % | |
| Значение 28 | 0,005 | 00,01 % | |
| Значение 29 | 0,004 | 00,01 % | |
| Значение 30 | 0,003 | 00,01 % | |

Порог значимости (%) 99,91

Picture 3 Table with computed principal components

During the analysis of factors, 25 principal components were selected from input data set. How much efficient this approach can be on modeling will be checked on the next step of the research.

The similar task of dimensionality reducing can be performed by means nonlinear transformation on base of auto associative networks (reproduce the input data in output). The number of internal elements of such neural nets is small, because of that the reducing of data is processed. So typical structure of the neural net has 5 layers – middle one is for reducing data, others are for nonlinear transformation.

- Sensitivity analysis [11,12].

During modeling the neural nets in Statistica Neural Networks, it is possible to use different combination of input variables. It is possible to ignore some variables, so the resulted neural net will not use them as input variables. It is also possible to perform sensitivity analysis of input variables. The procedure allows to make conclusion concerning importance of input variables to a specific neural net and delete some inputs with low rate of sensitivity if it is necessary.

So, the sensitivity analysis was performed using the current data set of monitoring parameters. During the analysis, the input variables were excluded by rotation. In order to determine sensitivity of the variables, the neural net was trained on testing values (on whole data set, with all available variables) and error N1 of the neural net was computed. After that the neural net was trained on the same values, but in the case the excluding of some values was performed and the error N2 of the resulted net was computed as well.

As some information (one of input variables) was deleted, then it is logically that the error will change. The value of sensitivity – is ratio error N2 to error N1. The sensitively is the net to the input variable, the bigger the ratio of the errors. If the ratio less or equal to 1, than excluding of the variable doesn't affect on performance of the neural net. The results are presented in table 2.

Table 2 The results of sensitivity analysis

| | Error | Vv_zp | Tv | Pk | Pt | Tt | Tm_v | .. |
|---|---|---|---|---|---|---|---|---|
| Ratio 1 | 0,396 | 0,6785 | 0,6787 | 0,742 | 0,6792 | | 1,004 | .. |
| Rank 1 | | 19 | 16 | 13 | 15 | | 3 | .. |
| Ratio 2 | 0,273 | 0,989 | 0,961 | 1,0009 | 0,983 | 1,045 | 1,049 | .. |
| Rank 2 | | 25 | 33 | 22 | 27 | 11 | 10 | .. |
| Ratio 3 | 0,363 | 0,924 | 1,002 | 1,0074 | 1,0004 | | 0,9994 | .. |
| Rank 3 | | 25 | 13 | 11 | 14 | | 17 | .. |
| Ratio 4 | 0,144 | 0,944 | 1,013 | 1,0451 | 0,984 | 1,038 | 0,991 | .. |
| Rank 4 | | 37 | 20 | 14 | 33 | 17 | 27 | .. |

During the analysis 10 neural nets of MLP type were trained, but only 4 of them whit lower error were selected. To each variable ratio of errors was computed and rank of importance the variables was determined. As shown in the table above, the parameter Tt (temperature of gas) didn't take part in training the first and the third nets, it had a bad influence on error rate. Hence it is possible to make a conclusion about high importance of the variables. It was determined that the parameter VV-zp is not an important variable and it may be excluded. Thereby, data set of 27 variables was received.

### 2.2 Comparative analysis of approaches efficiency

AS result of preparation data, three data sets were received:

1) A set after correlation analysis

2) A set after factor analysis

3) A set after sensitivity analysis

In order to estimate the efficiency and determine most resultative of the approaches of preparation data it's necessary to perform validation on modeling classifiers. If compare the accuracies of the resulted models, computed by means confusion matrix, it is possible to make a conclusion about suitability and usability of a specific approach of preparation data.

At the first it is necessary to determine the accuracy of model on the initial data, that's mean on date without any manipulations with variables.

Table 3 The results of modeling on the initial data

| | True<br>Defective state | True<br>Normal state |
|---|---|---|
| Predicted<br>Defective state | 19 | 4 |
| Predicted<br>Normal state | 27 | 368 |
| Recall | 41,30% | 98,02% |
| The initial data. Classification accuracy – 92.58%. | | |

In table 3 shown defective states of engine are recognized with quite low accuracy, so it obviously is problem.

Table 4 Results of modeling on data after factor and correlation analyses

| Data after factor analysis. Classification accuracy 91,39% | | |
|---|---|---|
| | True<br>Defective state | True<br>Normal state |
| Predicted<br>Defective state | 25 | 15 |
| Predicted<br>Normal state | 21 | 357 |
| Recall | 54,35% | 95,97% |

Data after correlation analysis. Classification accuracy 91,21%

| | | |
|---|---|---|
| Predicted Defective state | 24 | 8 |
| Predicted Normal state | 22 | 364 |
| Recall | 52,17% | 97,85% |

If estimate the results of modeling efficiency after factor and correlation analyses, then it's seen recall of defective class is increased, but the rate is still not high enough to effective functioning of the model.

The accuracy of classification of model trained on data after sensitivity analysis is 77.04%, table 4. Excluding of some variables had positive influence on recognition of defective state of engine, but in the same time the accuracy of model in general is reduced, because of reducing of recognition of other class on 20 %.

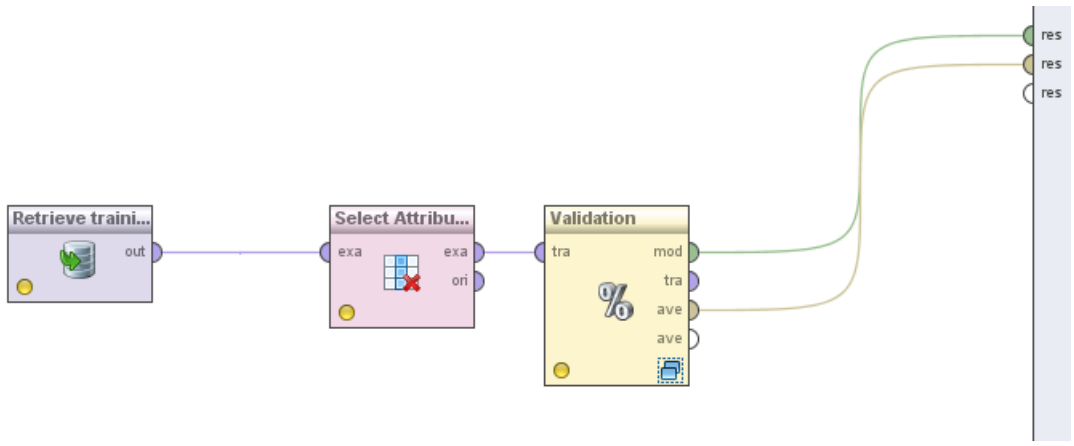
Table 5 Results of modeling on data after sensitivity analysis

| | True Defective state | True Normal state |
|---|---|---|
| Predicted Defective state | 10 | 51 |
| Predicted Normal state | 5 | 178 |
| Recall | 66,66% | 77,72% |

Thereby, the accuracy of recognition (recall) of defective states was improved on 20% by applying of the approaches of data preparation, but recall of class of normal states was reduced on the same 20 %.

### 3. Creating models of gas turbine engine technical state estimation

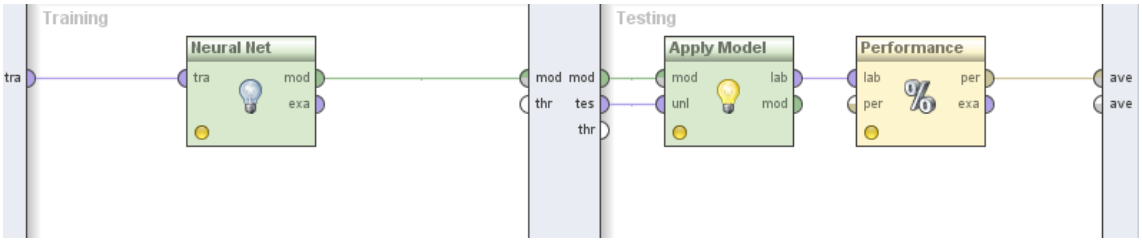
As software for modeling was used Rapid Miner studio. At the first stage it was necessary to build a basic structure of the classification model. It is mean to choose and connect all of the necessary blocks. The resulted structure is presented on picture 4.




Picture 4 Structure of classification model in Rapid Miner

Let's consider all of the elements of the model in more detail:

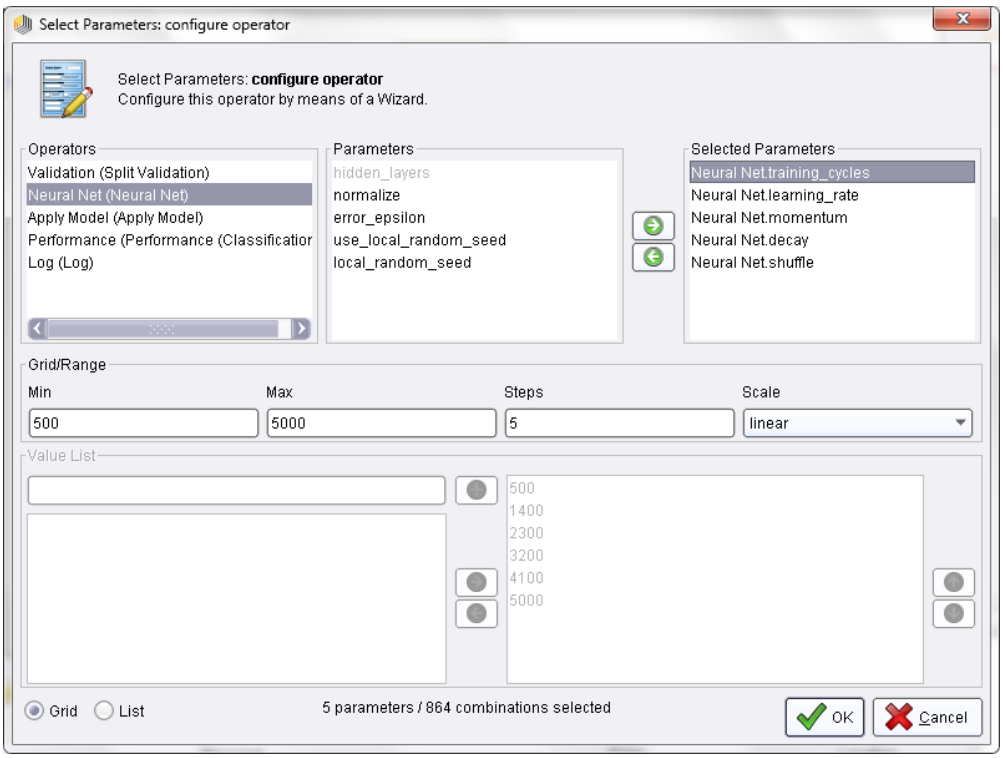
- *Retrieve block*, here is processed loading of the example set to the classification model.
- *Select attributes block* allows filtering attributes of the input data,
- *Validations block* splits up the input data set on training set and testing set. The block has a subprocess, picture 5.




Picture 5 The subprocess of Validation block

As one can see from the picture above, the subprocess of Validation block includes three blocks. In the case it is block of classification model – multilayer perceptron, block of applying the trained classification model on testing data set and the last block of performance estimation.

In order to optimize the parameters of a classification model it was used "optimize parameters" block. The block finds optimal values of the parameters in the specified range. It allows finding the best combination of the parameters as well.




Picture 6 Optimization parameters of Neural Net operator

For example, it is necessary to find the optimal parameters of Neural Net operator, picture 6. At the first step one have to choose that operator, then specify the parameters to optimization and the last step is to make some limits of the parameters. All of the values in the specified range cannot be checked, so there is a "Steps" field, which determine the number of values to be checked from the range.

It is a really convenient and fast way to tune a model, but using the optimization block it is possible to easily get the problem of overfitting.

### 3.1 Modeling of the multilayer perceptron

A multilayer perceptron (MLP) is a typical common kind of neural nets and a good fit to almost all types of tasks. It is feedforward neural net, so we have several successive connected layers and each current neuron gets and processes signals from neurons of the previous layer. The input layer is used just to transfer information into a hidden (computing) layer. The amount of neurons in the input and output layers is determined by the dataset parameters (number of attributes and

type of target attribute). More information about working of this type of neural nets available in [13].

After preparation the input data set by applying some methods of reducing attributes and finding most effective combination of them, it's improved the classification efficiency. In the stage, let's see what influence has the approaches to balanced data set on modeling efficiency.

1) The first data set, after the preparation and normalization of attributes. Its set includes 25 values of working state of GTE and 25 GTE with some failure.

Table 6 Results of modeling on the first data set

| | True Defective state | True Normal state |
|---|---|---|
| Predicted Defective state | 13 | 347 |
| Predicted Normal state | 13 | 1143 |
| Recall | 50% | 76,1% |
| Accuracy – 66.2% | | |

In order to create the first set was used 1-5 values of monitoring parameters, which are taken before stopping GTE working, so the values describe a GTE with a failure. As values which describe working GTE, all other available values were taken (around 250 values per each of 25 GTE). Thereby, that data set has a big disbalancing of the classes, table 6.

2) At the next step the values of class with working GTE were taken in a better way. The number of GTE and the proportion between the classes was the same like in the previous case.

Table 7 Results of modeling on the second data set

| | True Defective state | True Normal state |
|---|---|---|
| Pr. Defective state | 7 | 57 |
| Pr. Normal state | 3 | 177 |
| Recall | 70% | 75,6% |
| Accuracy – 75.4%. | | |

In that current case the values of class with working ГТД were taken in the following way: as 1-5 values before appearing a failure are values of the first class with not working ГТД, so values after repairing the engine are values of the second case with working engine. Thereby the second data set was prepared, each working engine in the case has around 10-15 values of monitoring parameters.

The model was trained on the data set and the result of recall the minority class (the first class with now working engine) was increased on 20 %, table 7.

3) In order to estimate the efficiency of balancing data approaches, a model was trained on the data balanced by approach based on clustering of majority class values. Thereby using the values of cluster centers, instead of majority class values, that current data set was balanced, table 8.

Table 8 Results of modeling on the third data set

| | True Defective state | True Normal state |
|---|---|---|
| Predicted Defective state | 21 | 7 |
| Predicted Normal state | 4 | 25 |
| Recall | 87,5% | 78,1% |
| Accuracy – 80.7%. | | |

That approach of balancing data was quite effective in improving the recall of the key class (minority class) and accuracy in general.

### 3.2 Modeling of the probabilistic neural net

A probabilistic neural net (PNN) is some kind of a neural net which similar to a radial basis function (RBF) network. The main difference is that a PNN has one neuron (in hidden layer) for each point in training set, that means the kind of net requires more memory space to store the model then MLP. It is not a good match to work with large training sets. PNN is quite accurate with small to medium size data sets and it has modifications, such as matrix of loss, which add an

additional layer with rates (price) of classification errors. The modification allows to make some accent on one class, the possibility fits well to the current task of determination state of an engine. More information about the structure of hidden layers and activation functions available in [14].

In the similar way, a PNN model was build and trained, but Rapid Miner doesn't have possibilities of modeling such type of neural nets. So, Statsoft Statistica was used instead of Rapid Miner to build the PNN model. Statsoft Statistica was used to make sensivity analysis in previous parts of the research.

The parameters of PNN model were configured as following:

1 Smoothing – 0.2,

2 Apriority probabilities 50/50 % (as the data set is balanced)

As input data balanced data set from the last step of modeling MLP was used. The results of modeling are presented in table 9.

Table 9 Results of PNN modeling

| | True Defective state | True Normal state |
|---|---|---|
| Predicted Defective state | 24 | 5 |
| Predicted Normal state | 6 | 25 |
| Recall | 80% | 83,3% |
| Accuracy – 81.6%. | | |

The result of modeling the PNN shows that the efficiency is the same as in the previous case of training MLP model on the same data set.

### 3.3 Decision Tree modeling

In order to estimate the efficiency of neural net approach in general, let's consider the model based on decision tree. The model was built in Rapid Miner using the same structure pictures 4-6, but instead of Neural Net operator, Decision Tree was used.

Decision Tree operator has several parameters. To configure the parameters, block of optimization parameters was used and the resulting parameters are:

The criterion of attribute splitting – Information Gain

Minimal size for split – 4

Minimal leaf size – 4

Minimal gain – 0.001

Maximal depth – 10

1) The result of modeling on unbalanced data is presented in table below.

Table 10 Results of Decision Tree modeling

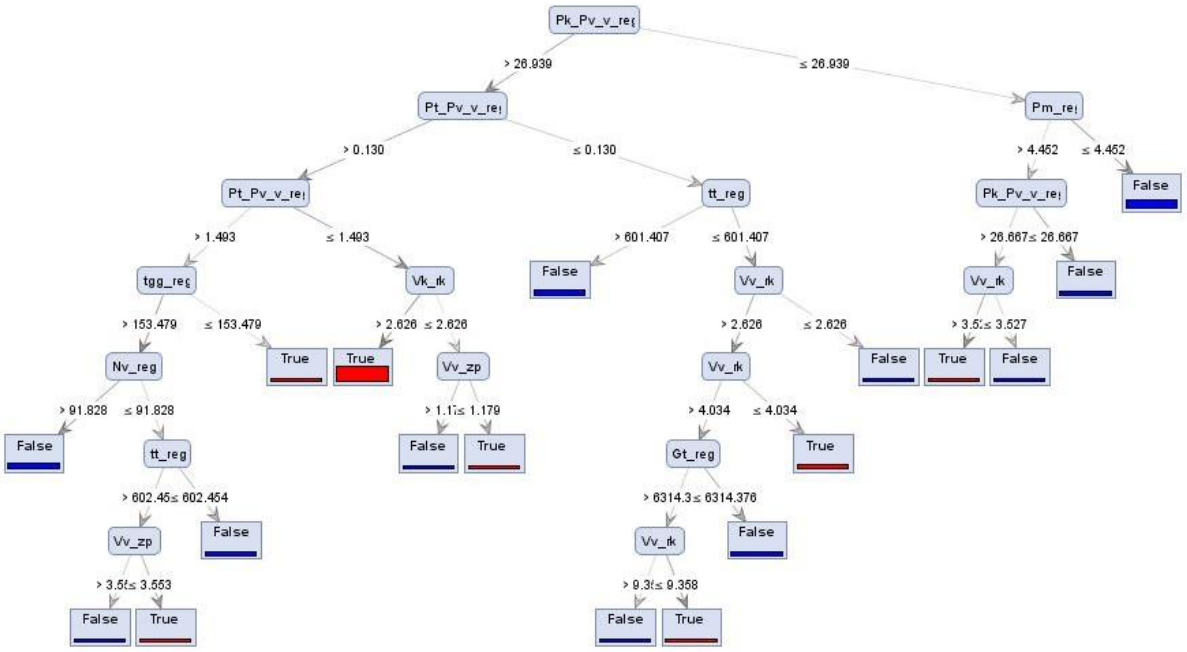
| | True Defective state | True Normal state |
|---|---|---|
| Predicted Defective state | 14 | 18 |
| Predicted Normal state | 32 | 2361 |
| Recall | 30,43% | 99,24% |
| Accuracy – 97.94%. | | |

2) If consider the balanced data set and train the decision tree model on the data set, we get the following results of classification, table.

Table 11 Results of Decision Tree modeling on balanced data set

| | True Defective state | True Normal state |
|---|---|---|
| Predicted Defective state | 41 | 11 |
| Predicted Normal state | 10 | 40 |
| Recall | 82% | 78,4% |
| Accuracy –79.41%. | | |

The tree built on the balanced data set has the following structure, presented on picture 7. This example clearly shows, how easy to interpret the data through decision trees.




Picture 7 The structure of the Decision Tree

The first node or the root node of the tree is parameter Pk_Pv – pressure rate. Splitting elements is performed based on created rules during training the model. A node that doesn't have child nodes are called the leaf nodes. Each leaf node represented a value of an output attribute (like a value of 1 class or second class).

## 4 Analysis and estimation of modeling results

### 4.1 Choosing a metric to estimation of classifiers efficiency

As was said before, the regular classification rate (classification accuracy) is not a good metric, because if one correctly classifies only the instances of the majority class (class with many instances), this metric still gives a high rate of accuracy like in the table 10, however the true rate of efficiency is different.

In the case, when the minority class is a class represents fault states of GTE, it is more important to accurately classify the vectors of this class, than the vectors of the other class. That's why the confusion matrix was chosen as a metric to determine the quality of the model, table 12. The matrix allows to estimate the recall of a specific class and get a clearer representation of the model efficiency.

Table 12 Confusion matrix

| Confusion matrix | | The real, true values | |
|---|---|---|---|
| | | True | False |
| Predicted values | True | tp | fp |
| | False | fn | tn |

In order to calculate the recall of both classes it is necessary to split the data set into 3 parts:

- Training set,
- Testing set,
- Validation set.

The first part is to train the classification model. The data should have not less than 50 % of whole data set. The second part is to make an intermediate testing of modeling efficiency while process of tuning classifier. The last part is to a final one-time validation. It allows to make more accurate estimation of the model, because testing on a validation set is some kind of training (changing parameters of a model and retraining the model).

So, let's compute recall of the minority class of the data set of engine monitoring parameters, formula (2).

$$\text{Accuracy} = \frac{tp+tn}{\text{tp+tn+fp+fn}} = \frac{17+275}{17+275+4+18} = 0{,}9299 \quad (1)$$

The quality of resulted model using such metric as accuracy is equal to 0.9299, formula (1), that value is characterized the model as high quality classifier, but it is not like that.

$$Recall = \frac{tp}{\text{tp+fn}} = \frac{17}{17+18} = 0{,}4857 \quad (2)$$

The computed value of minority class recall allows to determine that the model has low rate of recognition the key class. It is possible to make conclusion concerning usability of the model based on the rate.

### 4.2 Comparative analysis of models, based on chosen metric.

The following models: MLP, PNN, Decision tree were trained while modeling on received data sets. To estimate the efficiency of presented models and to make a conclusion of practicability the models, the rates of chosen metric were computed

Table 13 Comparative analysis of received models

| Type of model | Type of data | Accuracy / Recall of minority class |
|---|---|---|
| MLP | Balanced data set | 80.7% / 87.5% |
| PNN | Balanced data set | 81.6% / 80% |
| Decision Tree | Balanced data set | 79.4% / 82% |

Thereby, while estimation of modeling results, table 13, we identified that MLP model has the best quality of classification, especially considering the most important rates to the current task such as recognition of the minority class.

## Conclusion

In the work the classification of existing methods of Data Mining was performed. In addition, the ways of efficiency improving of those methods were considered. The ways were based on principals of data preparation to modeling.

On the basis of neural networks approach the model (MLP) of technical state estimation of gas turbine engines was developed. The processes of parameter optimization of the model and preparation of input data by means reducing of input attributes and balancing of number values between classes were performed.

The result of applying this model was received: the accuracy is 80.7%, it is quite high rate of classification quality, especially considering fact that the key class of defect state an engine was recognized by 87.5% accuracy.

During the research the approaches to preparation data of monitoring parameters and the model of technical state estimation were developed and all of it allows to improve efficiency of diagnostic work and improve results of technical state estimation in general,

**List of reference**


1. Konev S.V., Sichinava Z.I., Halliulin V.F., Jasnickij L.N. Vozmozhnosti primenenija nejrosetevyh tehnologij dlja prognozirovanija neispravnostej aviacionnyh dvigatelej. Ajerokosmicheskaja tehnika i vysokie tehnologii – 2005. Materialy 8 Vserossijskoj nauchno-tehnicheskoj konferencii – Perm': PGTU, 2005. – 174 p.
2. Hajkin, S. Nejronnye seti:polnyj kurs, 2-e izd., ispr.:Per. s angl. /S. Hajkin.– M.:OOO I.D. Vil'jams, 2006. – 1104 p.
3. Reference to "Rapid Miner" program. Available at: http://docs.rapidminer.com/studio/operators/ (accessed 20 May 2016)
4. Breiman, L., Friedman, J.H., Olshen, R.A., and Stone, C.J. Classification and Regression Trees, Wadsworth, Belmont, CA, 1984. Since 1993 this book has been published by Chapman & Hall, New York.
5. Gajdyshev I. Analiz i obrabotka dannyh: special'nyj spravochnik — SPb: Piters., 2001. — 752 p.
6. Egor Obukhov, Handling the problem of unbalanced data sets in the classification of technical equipment states (Material of conference of 4th International Conference on Applied Innovations in IT). Germany - 2016.
7. I. Witten and E. Frank, Data Mining: Practical Machine Learning Tools and Techniques. Morgan Kaufmann Publ., 2011. 629 p.
8. Robert E. Schapire, “The Strength of Weak Learnability” Machine Learning, 5(2):197–227, 1990
9. Barsegjan A.A., Kuprijanov M.S. Analiz dannyh i processov: uchebnoe posobie. 3 izdanie, pererabotannoe i dopolnennoe. SPb.: BHV-Peterburg, 2009. - 512 p.
10. Richard L. Factor Analysis, second edition, Hillsdale: Lawrence Erlbaum Associates. 1983. – 454 p.

11. *Spravochnyerukovodstva SPO "Statistica". (*Reference to "Statistica" program). Available at: http://www.statsoft.ru/resources/support/info.php (accessed 20 May 2016)

12. Halafjan, A.A. STATISTICA 6. Statisticheskij analiz dannyh. 3-e izd. ucheb. / A.A. Halafjan. – M.: Binom-Press, 2007. – 512 p.

13. S. Sathyanarayana, “A gentle introduction to backpropagation”, July 22, 2014

14. Specht, D. F. "Probabilistic neural networks". Neural Networks 3:1990. - 109–118 p.