CART-tool for the mathematical modeling of control system of aviation engine

S S Tovkach

Automation & Power Management Department, National Aviation University,
1 Liubomyra Huzara ave., Kyiv 03058, Ukraine

E-mail: ss.tovkach@gmail.com

Abstract. A mathematical tool of multidimensional data mining aviation engine control system has been considered. It can be effectively applied for finding in both detailed and aggregated data the main sequence of the mathematical modeling: input of initial data and calculation of thermodynamic parameters; design of the engine flow path; calculation of engine parameters in the design mode with power simulation of turbomachines; design of the flow path of blade machines; strength design and mass analysis. The analysis of data of aviation engine control systems using decision trees is considered, which consists in choosing the method of branching by the values of predicted variables used to predict, the belonging of the analyzed objects to certain classes of values of the dependent variable. In accordance with the hierarchical nature of decision trees, such branches occur sequentially, starting from the root vertex, passing to the vertices-descendants, until further branching stops and "unbranched" vertices-descendants are terminal

1. Inroduction

Modern requirements for the mathematical description of the engine and its properties at the initial design stage allow at the stage of choosing its technical appearance as optimization variables, along with the traditional parameters of the working process, the use of such indicators as the number of stages of turbomachines, the number of blades, the power circuit, the range of applied materials and other indicators characterizing the engine as a real design.

The software used to determine the geometric appearance and mass data of aviation gas turbine engines at the initial design stage, as a rule, is based on generalized dependencies that take into account the influence of the main process parameters and engine dimensions on its mass in the simplest engine mass models and on nodal mass models in detailed approaches. Mathematical models of the geometric shape and mass of the engine are being developed based on even more detailed information about the basic units and engine parts, up to the creation of a solid geometric model of the engine.

The engine and its components are represented as consisting of elementary objects, which are simplified representations of real engine parts. Compliance level of elementary objects, which are simplified representations of real engine parts. The level of correspondence of elementary objects to real ones increases with the development of program versions. The nomenclature of elementary objects can also be expanded as the engine model is detailed.

Basic geometric dimensions (diameters and lengths) of elementary objects are determined as a result of calculation and alignment of the flow path. For a more accurate determination of the shapes of a number of objects, taking into account the detailed strength analysis, external software components are used. In this case, the concept of multilevel multidisciplinary modeling is implemented. The object-oriented approach is extremely convenient for operations of grouping, editing, determining the number and changing the properties of both the most elementary objects and groups of objects with varying degrees of generalization [1-3].

The goal of this research is to define the multidimensional data mining tool that should find patterns in both detailed and aggregated data of automatic control system (ACS) of aviation engine with varying degrees of generalization.

2. Aviation Engine Modeling Interface

The interface script (QUEST and CART) includes the possibility of a two-level simulation of the flow path. In the case of designing a new engine (from a clean slate), a unit-by-unit thermodynamic calculation of the engine at the design mode and the linkage of the parameters of the flow path in relation to the number of stages, rotational speeds of rotors, diametric dimensions and other parameters of the units are carried out at the selected level of gas-dynamic and mechanical loading of the blade machines and supplied conditions of choice. The data obtained at the first level of modeling are the basis for modeling the flow path at the second level with a general description of the blades.

CART-tool. CART (Classification And Regression Trees) is a program that, when building a tree, performs a full enumeration of all possible variants of one-dimensional branching [2].

The main differences of the CART algorithm from other algorithms are [2,3]: binary representation of the decision tree; function for the quality estimation splitting, tree pruning mechanism; final tree selection; regression tree construction.

The QUEST and CART analysis options complement each other. In cases where there are many predictor variables with a large number of levels, the CART search can be quite lengthy. It also tends to branch out those predictor variables that have more levels. However, since a full enumeration of options is performed here, there is a guarantee that a branching option will be found that gives the best classification (in relation to the training set).

In the CART algorithm, each node of the decision tree has two children. At each step of building a tree, the rule formed at the node divides the given set of examples (training sample) into two parts: a part in which the rule is executed and a part in which the rule is not executed. To select the optimal rule [5,6], the function of estimation the quality of the partition is used, which is based on the intuitive idea reducing impurity (uncertainty) in the node. In the CART algorithm, the idea of "impurity" is formalized in an index $Gini$. If the dataset T contains n class data, then the index $Gini$ is defined as [6]:

$$Gini(T) = 1 - \sum_{i=1}^{n} p_i^2, \tag{1}$$

where p_i is the probability (relative frequency) of class i in T.

If the set T is split into two parts T_1 and T_2 with the number of examples in each N_1 and N_2, accordingly, the quality index of the split is [6]:

$$Gini_{split}(T) = \frac{N_1}{N} Gini(T_1) + \frac{N_2}{N} Gini(T_2). \tag{2}$$

The best partition is the one for which the minimum is $Gini_{split}(T)$.

Let denote N is the number of examples in the parent node, L, R is the number of examples in the left and right children, respectively, l_i and r_i is the number of instances of the ith class in the left and right children. Then the quality of the partition is estimated:

$$Gini_{split} \to \min, \tag{3}$$

where $Gini_{split} = \frac{L}{N}\left(1 - \sum_{i=1}^{n}\left(\frac{l_i}{L}\right)^2\right) + \frac{R}{N}\left(1 - \sum_{i=1}^{n}\left(\frac{r_i}{R}\right)^2\right).$

Since multiplication by a constant does not play a role in minimization, the following transformations of the selection criterion [6,7] are possible:

$$Gini_{split} = L - \frac{1}{L}\sum_{i=1}^{n} l_i^2 + R - \frac{1}{R}\sum_{i=1}^{n} r_i^2; \tag{4}$$

$$Gini_{split} = N - \left(\frac{1}{L}\sum_{i=1}^{n} l_i^2 + \frac{1}{R}\sum_{i=1}^{n} r_i^2 \right); \tag{5}$$

$$Gini_{split} \to \min; \tag{6}$$

$$\tilde{G}_{split} \to \max, \tag{7}$$

where $\tilde{G}_{split} = \frac{1}{L}\sum_{i=1}^{n} l_i^2 + \frac{1}{R}\sum_{i=1}^{n} r_i^2$

Thus, the best partition will be the one for which the value $\tilde{G}_{split}$ is maximum.

Less commonly, the CART algorithm uses other partitioning criteria Twoing, Symmetric Gini, etc. [8,9]. Options for end-to-end automated calculation are possible.

The **tree pruning mechanism** is the most serious difference of the CART-algorithm from other tree construction algorithms. CART considers pruning as a compromise between two issues: obtaining a tree of optimal size and obtaining an accurate estimate probability of erroneous classification [10].

The main problem with pruning is a large number of everyone possible cut off subtrees for one tree. More precisely, if a binary tree has $|T|$-sheets, then there is approximately $\left[1.5028369^{|T|}\right]$ pruning of subtrees [10-14]. The basic idea of the method is not consider all possible subtrees, limiting yourself only 'best representatives' as estimated below.

Let $|T|$ is the number of sheets of the tree, $R(T)$ is the classification error tree equal to the ratio of the number of incorrectly classified examples to the number of examples in the training sample. Define $C_\alpha(T)$ is the total estimate of the T tree as [14]:

$$C_\alpha(T) = R(T) + \alpha * |T|, \tag{8}$$

where T is the number of sheets (terminal nodes) of the tree, α is some parameter varying from 0 to $+\infty$. The total estimate of the tree consists of two components: errors of classification of a tree and a penalty for it complexity. If the tree classification error is unchanged, then with the increasing α the total estimate of the tree will increase.

Let's define $T_{\max}$ is the maximum in size tree which need to be pruning. If we fix the value of α, then it exists the smallest minimizable subtree α, that performs the following conditions:

$$C_\alpha(T(\alpha)) = \min_{T \le T_{\max}} C_\alpha(T)$$

$$\text{if } C_\alpha(T) = C_\alpha(T(\alpha)) \text{ then } T(\alpha) \le T$$

The first condition says that there is no such tree subtree $T_{\max}$, which would have a lower cost than $T(\alpha)$ for this value α. The second condition says, that, if there is more than one subtree, having a given full value, then we choose the lowest tree.

Although α has an infinite number of values, there is a finite number of tree subtrees $T_{\max}$. You can create a sequence decreasing subtrees of the tree $T_{\max}$: $T_1 > T_2 > T_3 > ... > \{t_1\}$, (where t_1 is root node of the tree) such that T_k is the least minimized subtree for $\alpha \in [\alpha_k, \alpha_{k+1})$. This means, that you can get the following tree in sequence by applying prunning to the current tree. It allows you to develop an efficient algorithm for finding the smallest of the subtree to be minimized for different values of α. First tree of this sequence is the smallest subtree of the tree $T_{\max}$ with the same classification error as $T_{\max}$, i.e. $T_1 = T(\alpha = 0)$.

For calculating T_1 in $T_{\max}$, it is necessary to find any pair of sheets with a common ancestor that can be combined, i.e. pruning in parent node without increasing classification error $R(t) = R(l) + R(r),$ where r and l are sheets of node t. The search should continue until there will be no more such pairs. As a result, we get a tree with the same estimate as $T_{\max}$ for $\alpha = 0$, but less branched than $T_{\max}$.

The next tree in sequence and the corresponding the value α is obtained as follows:

Let T_t denote a branch of the tree T with the root node t. Let define, for what values of α the tree $T - T_t$ will be better than T. If we pruning in node t, then its contribution to the total estimation of the tree $T - T_t$ becomes $C_\alpha(\{t\}) = R(t) + \alpha$, where $R(t) = r(t) * p(t)$, $r(t)$ is the classification error of node t and $p(t)$ is the proportion of cases that 'passed' through node t. Alternative option: $R(t) = \frac{m}{n},$ where m is the number of cases of classified nodes is incorrect, and n is the total number of classified nodes for the whole tree.

The contribution of T_t to the total estimation of the tree T is $C_\alpha(T_t) = R(T_t) + \alpha|T_t|,$ where $R(T_t) = \sum_{t' \in T_t} R(t').$

The tree $T - T_t$t will be better than T when $C_\alpha(\{t\}) = C_\alpha(T_t).$ In this way: $R(T_t) + \alpha * |T_t| = R(t) + \alpha.$ Solving this equation for α, we get [15]:

$$\alpha = \frac{R(t) - R(T_t)}{|T_t| - 1}. \tag{9}$$

Thus, for any node t in T_1, if we increase α, when $\alpha = \frac{R(t) - R(T_{1,t})}{|T_{1,t}| - 1}$ is the tree obtained by pruning at node t will be better than T_1.

The main idea is as follows: calculate this value α for of each node in the tree T_1, and then select “low links”, i.e. nodes for which the quantity $g(t) = \frac{R(t) - R(T_{1,t})}{|T_{1,t}| - 1}$ is the smallest. Pruning T_1 in these nodes to get T_2 is the next tree in the sequence. Then we continue this process for the resulting tree and so on until we get the root node [16].

Algorithm for calculating a sequence of trees.

```
$T_1 = T(\alpha = 0); \alpha_1 = 0, k = 1$
While $T_k > \{root\ node\}$ do begin
for all nonterminal nodes (sheets) in $t \in T_k$
$g_k(t) = \frac{R(t) - R(T_{k,t})}{|T_{k,t}| - 1}$
$\alpha_{k+1} = \min_t g_k(t)$
Get round from top to bottom all nodes and pruning those where $g_k(t) = \alpha_{k+1}$ so that get $T_{k+1}$
$k = k + 1$
end
```

Final tree selection.

So, we have a sequence of trees, and need to select the best tree out of it, the one that can be used in further. The most obvious and most effective is selection of the final tree by testing on a test set. Naturally, the quality of testing largely depends on the volume of the test sampling and "uniformity" of data that fell into the training and test sample. The tree, that gave the minimum classification error, will be the best.

Regression tree construction.

Creation a regression tree is a lot like a tree classification. Firstly, build a tree of maximum size, then pruning the tree to the optimal size.

The main advantage of trees compared to other regression methods is the ability to work with multidimensional problems and tasks, in which there is a dependence of the output variable on the variable or variables of categorical type.

The main idea is to divide the entire space into rectangles, optionally the same size in which the output variable is considered constant. There is a relationship between volume the training sample and the tree response error [16, 17].

The process of building a tree occurs sequentially. On the first step, get the regression estimate simply as a constant throughout space of examples. Consider the constant as the arithmetic mean the output variable in the training set.

So, if denote all values of the output variable as $Y_1, Y_2, ..., Y_n,$, then the regression estimate is obtained: $\hat{f}(x) = (\frac{1}{n}\sum_{i=1}^{n} Y_i) I_r(x),$ where R is the space of training examples, n is the number of examples, $I_r(x)$ is the indicator function of space is, in fact, a set of rules, describing the entry of the variable x into space. Space R viewed as a rectangle. In the second step, divide space in two parts. Some variable x_i is chosen and if variable of numeric type, then define:

$$R_1 = \{x \in R : x \le a\}, R_2 = \{x \in R : x > a\}$$

If x_i is of categorical type with possible values $A_1, A_2, ..., A_q$, then some subset $I \subset \{A_1, ..., A_n\}$ is chosen and define

$$R_1 = \{x \in R : x \in I\}, R_2 = \{x \in R : x \in \{A_1, A_2, ..., A_q\} \setminus I\}$$

The regression estimate takes the form [17]:

$$\hat{f}(x) = \left(\frac{1}{|I_1|}\sum_{I_1} Y_i\right) I_{R_1}(x) + \left(\frac{1}{|I_2|}\sum_{I_2} Y_i\right) I_{R_2}(x),$$

where $I_1 = \{i, x_i \in R_1\}$ and $|I_1|$ is the number of elements in I_1.

The sum of the squares of the differences is used as an estimate:

$$E = \sum_{i=1}^{n} \left(Y_i - \hat{f}(x_i)\right)^2$$

A partition with the minimum sum of the squares of the differences is chosen.

The partitioning continues until each subspace contains a small number of examples or the sum of the squares of the differences becomes less than a certain threshold.

Conclusion

The criteria for choosing the basic tool for creating an improved decision support system in the control processes of aviation engines has been developed and formulated according to the factors of complementarity of the potential of operational analytical processing and data mining. The expediency of using a complex combination analysis technology for the detailing operation of the ACS of the aviation engine was substantiated and the use of the CART software application for mathematical modeling of ACS of the aviation engine, which implements the functions of a binary representation of a decision tree, quality estimation splitting, tree pruning mechanism, final tree selection, regression tree construction has been considered.

CART-tool is defined as the choice of branching option, is a complete search of trees with one-dimensional branching by the CART (Classification and Regression Trees) method for categorized and ordinal predicted variables. In this method, all possible branching options for each predicted variable are sorted, and there is the one that gives the greatest growth for the agreement criterion.

References

[1] Allan Seabridge and Ian Moir 2020 *Design and Development of Aircraft Systems* (Wiley; 3rd Edition) p 400 **ISBN-13:** 978-1119611509

[2] Wei-Yin Loh 2011 *Classification and Regression Trees (Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* vol 1, issue 1) pp 14 **ISBN-13:** 978-1498754071

[3] Tovkach S S 2019 *Organization of distributed information systems the aviation gas turbine engine (Electronics and Control Systems* vol 61 issue 3) pp 29 **DOI:** 10.18372/1990-5548.61.14210

[4] Gbadebo Ayoade and Amir el Chamry 2018 *Secure data processing for IoT middleware systems (The Journal of Supercomputing* vol 75 issue 7) pp 1 **DOI:** 10.1007/s11227-018-26868

[5] Bagai I E, Roedig U and Hollick M 2015 *Using channel state information for tamper detection in the internet of things* (*Proceedings of the 31st Annual Computer Security Applications Conference*, ACSAC 2015) pp 131. **DOI:** 10.1145/2818000.2818028

[6] Daniya T, Geetha M. and Suresh Kumar K. 2020 *Classification and regression trees with Geni Index (Advances in Mathematics Scientific Journal* vol 9 issue 10) pp 1857 **DOI**: 10.37418/amsi.9.10.53

[7] Zugai Marcin 2017 *Reconfiguration of Unmanned Aircraft Control Systems* (*Transactions on Aerospace Research* vol 2017 issue 2) pp 80 **DOI:** 10.2478/tar-2017-0017

[8] Yuliana Wirma, Anita Maharani and Zainur Hidayah 2020 *Indonesia's Civil Servants' Performance at Aviation Engineering: Exploration Study (Integrated Journal of usiness and Economics* vol 4 issue 2) pp 160 **DOI:** 10.33019/ijbe.v4i2.279

[9] Naser Er F Ab and Bhardwaj A K *2014 Important Pitot Static System in Aircraft Control System (American Journal of Engineering Research* vol 03 isuue 10) pp 138 **e-ISSN:** 2320-0847

[10] Lukai Mao and Xu Xianlian 2019 *Design on four-axis aircraft control system based on Somatosensory interaction (IOP Conference Series Materials Science and Engineering)* pp. 042052. **DOI:** 10.1088/1757-899x/569/4/042052

[11] Vinnichenko A V, Nazarevich S A and Karpova I R 2020 *Automated reverse engineering control system model (Issues of radio electronics)*. **DOI:** 10.21778/2218-5453-2020-4-39-49

[12] Gryadunov K I, Kozlov A N, Nemchikov M L and Melnikova I S 2019 *Aviation engines diagnistics by estimating the metal contamination in oils (Civil Aviation High Technologies* vol 22 issue 3) pp 35 **DOI:** 10.26467/2079-0619-2019-22-3-35-44

[13] Terentev Mikhail, Karpenko Sergey and Zylin Eugene 2018 *Nonparametric method for failures diagnosis in the actuating subsytem of aircraft control system (IOP Conference Series Materails Science and Engineering* vol 312 issue 1) p 012025 **DOI:** 10.1088/1757-899X/312/1/012025

[14] Boonamnuay S, Kerdrasop Nittaya and Kerdrasp K 2018 *Classificaion and regression tree with resampling for classifying imbalanced data (International Journal of Machine Learning and Computing* vol 8, issue 4) pp 336 **DOI:** 10.18178/ijmlc.2018.8.4.708

[15] Nytrebych Zinovii and Pukach Petro 2019 *Analysis of measurement systems mathematical models by using the comparison of functions (Mathematical modeling and computting* vol 6 issue 2) pp 268 **DOI:** 10.23939/mmc2019.02.268

[16] Rubtsova I D and Ovsyannikov D A 2017 *Intense quasiperiodic beam dynamics in accelerating system: mathematical model and optimization method (Journal of Physics Conference Series* vol 941 issue 1) pp 012092 **DOI:** 10.1088/1742-6596/941/1/012092

[17] Yurkov N K and Shtykov R A 2020 The imorovement of control system of furnace equipment by using the new mathematical model of gas mixing (Unversity proceeding. Volga region vol 1 issue 9) pp 90 **DOI:** 10.21685/2072-3059-2020-1-9