Combinatorial GMDH algorithm with successive selection of arguments

Samoilenko O.A., Stepashko V.S.

International Research and Training Center of Information Technologies and Systems of NAS of Ukraine, prospect Akademika Glushkova,40,Kyiv,03680,Ukraine

soa_pga@mail.ru; astrid@irtc.org.ua

Abstract. *The paper considers the problem of solving tasks with large number of arguments by combinatorial GMDH algorithm. To solve this problem, successive selection of the most informative arguments is suggested. Use of this algorithm enables to essentially accelerate the retrieval for the best subset of regressors and to solve tasks with considerably larger number of regressors compared with ordinary combinatorial GMDH algorithm of exhaustive search of arguments.*

Keywords

Inductive modelling, GMDH, combinatorial algorithm, argument, successive selection.

1 Introduction

Inductive GMDH algorithms suppose examination of all possible variants of task solving and selection of the best variant (model). We include as more arguments as possible to the sample to build the most exact model. Howewer if number of arguments is very great then examination of all variants takes much time and is often impossible. Therefore, there is the problem of acceleration of the combinatorial algorithm namely by means of the most informative arguments selection. The problem has been investigated in [1], [2] where it was suggested to estimate the level of arguments informativeness regarding to the module of the argument correlation coefficient with the output variable (MCC). In our investigation, the level of argument informativeness is estimated using an algorithm suggested in [3].

2 Problem statement

1. Suppose we are given a data sample W=(X y), dim W=n×(m+1).
2. The relationship between an output y and $s_0<m$ relevant inputs holds:

$$y = X_{s0}\theta^o + \xi = \overset{o}{y} + \xi, \qquad (1)$$

$\overset{o}{y}$ is an exact or true output of an object (signal),

θ^o is a vector of true parameters,

ξ is a vector of stochastic disturbances (noise),

X_{s_0} is a submatrix of the matrix X with s_0 vectors influencing the output value y in that the number s_0 and composition of the vectors X_{s_0} is unknown.

3. It is necessary to search for the optimal model in the form:

$$y = X_s \theta(s), \tag{2}$$

$\theta(s) = [\theta_1 \theta_2 ... \theta_s]^T$ is a vector of unknown parameters being estimated.

Vector of estimated parameters $\theta(s)$ determines a model of the complexity s for the sample W.

The quality of a model is determined as the regularity criterion $AR(s)$ supposing division of the sample X into 2 subsamples X_A and X_B. We estimate model parameters on the training subsample X_A and calculate the error on the testing subsample X_B.

$$AR(s) = \| y_{Bs} - X_{Bs} \theta_{As} \|^2, \tag{3}$$

where θ_{As} is the vector of parameters estimated on the subsample X_A.

The model of optimum quality:

$$s^* = \arg\min_{s=1,m} AR(s). \tag{4}$$

When using the combinatorial algorithm, a retrieval of all possible models with selection of the best model by the criterion (2) is carrying out. When number of arguments is not very large the exhaustive search can be carried out. In such a way, the total amount P_m of all possible models containing 1...m arguments is calculated by the formula:

$$P_m = \sum_{j=1}^{m} C_m^j = 2^m - 1 \tag{5}$$

When the arguments number is greater than 20, the exhaustive search for the acceptable limit of time is often impossible.

3 Solving the problem

Let us start with an example: m=20, n=50, s_0=10, and analyze the dependence of the criterion AR on the model complexity s. This dependence is illustrated in the Figure 1.

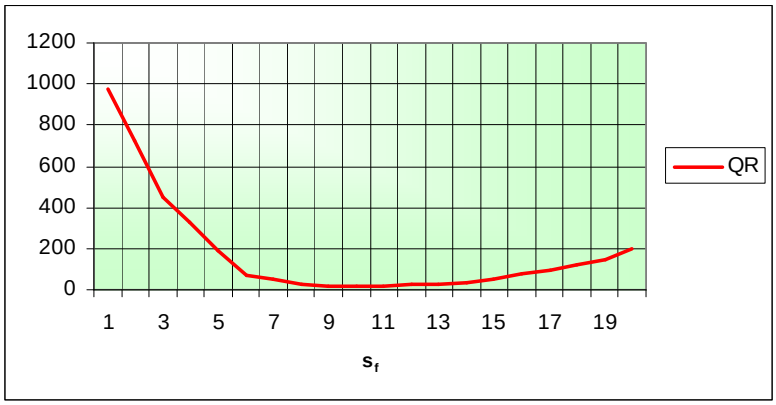

Fig. 1. Dependence of the criterion AR on the model complexity s

As the figure shows, the model quality for the complexity greater than s_0 becomes lower and it has no sense to analyze such models. Hence it is better to use an algorithm which does not consider all models and sorts models of the complexity 1, then 2 and so forth, until the criterion for the next complexity becomes to increase.

Thus the retrieval algorithm with successive complication is as follows (Alg. 1):

step 1: The structures of complexity s are generated and matrices X_A, y are built;

step 2: Model coefficients of any structure are estimated using Gauss method on the training subsample X_A;

step 3: The quality criterion *AR(s)* for a model is calculated on the testing subsample X_B and the best model of complexity s is selected;

step 4: If the model quality for s is better than that for s-1 then the complexity of models is increased and we turn to step 1 else the cycle is finished.

The Figure 2 illustrates the computational effectiveness of this algorithm for different s_0 as comparing to the exhaustive search algorithm.

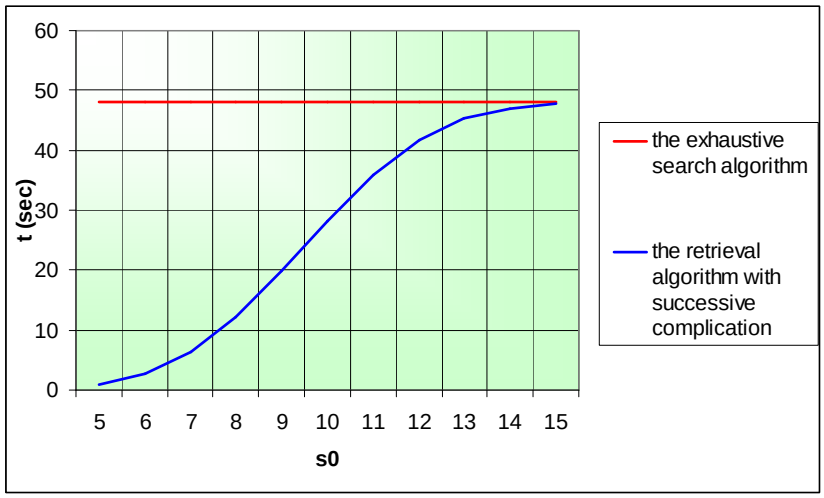

Fig. 2. Comparison of the algorithms effectiveness as depending on s_0

As the figure shows, if $s_0 = 10$ than such algorithm spends almost half of the time needed for the exhaustive search algorithm.

As it is evident from (5), removing of an argument from the set arguments halves the time of searching. Consequently, for the acceleration of finding of the best model it is needed to find such arguments which will not substantially influence on the model and to remove them from the set, leaving most informing. Such approach is offered in [1], where it is suggested to estimate the level of arguments informativeness regarding to the module of the argument correlation coefficient with the output variable. In [3] it is shown that the level of arguments informativeness can be estimated considering how many of the best models contain this argument.

On the basis of these results, let us consider the algorithm with successive selection of arguments. Models are selected by this algorithm using the algorithm of successive complication with increasing of it until the calculation time is permissible. The best models and arguments included to these models are then examined only. A new set which consists only of those arguments taking active part in forming the best models is thus formed. Further models are built on this new set and the sequence of such operations are again repeated until the set will contain so many arguments that it would be possible to perform an exhaustive search.

We describe this algorithm as the following steps (Alg. 2):

step 1: build the models of the complexities allowing to fit the given time limit using the Alg. 1;

step 2: select a subset of F the best models by an external criterion;

step 3: rank all the arguments being contained in this F models by the coefficient q_i, $=1\ldots m$, specifying the frequency of an i-th argument occurrence in the best models;

step 4: form a new sample by removing the arguments with the least values of q_i;

step 5: perform the exhaustive search of models if the amount of arguments in the new sample is acceptable or return to the step 1 otherwise.

Let us investigate the effectivenees of this algorithm at first theoretically.

When the task with m=20, s_0=10 is solved by Alg. 1, then the amount P_m of all possible models containing no more than s_0 arguments is calculated with the use of formula (5).

$$P_m = \sum_{j=1}^{10} C_{20}^{j} = 616665 \qquad (6)$$

Let us consider the same task for m = 200 using the Alg. 2 and calculate how many models this algorithm may build for the task solving.

Tab. 1. Theoretical amount of models on each stage of the algorithm with successive selection of arguments, m=200

Stages	**m**	**s_{max}**	***P_m***
Stage 1	200	2	20 100
Stage 2	100	3	166 750
Stage 3	50	4	251 175
Stage 4	25	5	68 405
Exhaustive search	10	10	1 023
Total amount of models, Alg. 2			507 453

As the Table 1 shows, the amount of models being built by the algorithm of successive selection of arguments Alg. 2 for 200 arguments is 507 453. It is less than the total amount of models sorted by the algorithm of successive complication Alg. 1 for 20 arguments. Theoretically speaking, one may say the second algorithm can solve a task with 200 arguments faster than the first one with 20 arguments for the same time.

The practical experiment was carried out for 20 arguments to makes it possible to compare results of sorting by use of the exhaustive search and the successive selection of arguments. The results of this experiment are reflected in the Table 2.

Tab. 2. Amount of models on each stage of the algorithm with successive selection of arguments, m=20

Stages	**m**	**s_{max}**	***P_m***
Stage 1	20	6	60459
Stage 2 (exhaustive search)	14	14	16383
Total amount of models, Alg. 2			76842

The models amount built by the Alg. 2 to get the result of the exhaustive search with 20 arguments is equal to 76842 that is considerably less than that by the Alg. 1 (616665, see (6)). As for the computing time, the figures are as follows: 3 sec for Alg. 2 and 24 sec for Alg.1. The combinatorial algorithm with the exhaustive search finds the same model in 48 sec.

However there are new problems here. When we remove the less informative arguments, are we sure if all the remaining arguments are relevant? What criteria are better to use for the effective successive selection of arguments? These problems are to be considered further.

4 Conclusion

An algorithm is offered with the successive selection of arguments accelerating considerably the process of search for the best model as well as extending the range of application of the combinatorial type GMDH algorithms.

References

[1] Ivakhnenko A.G., Ivakhnenko G.A., Savchenko E.A., and Wunsch D. Problems of Further Development of GMDH Algorithms: Part 2 // *Pattern Recognition and Image Analysis , Vol. 12,* № 1, 2002, p.6-18.

[2] Ivahnenko A.G., Ivahnenko G.A., Savchenko E.A. Conception of the successive algorithmic approaching (lowering) to the exact decision of interpolation tasks of artificial intelligence // *Cybernetics and computing engineering,* № 127, 2000, p.47-58. (In Russian)

[3] Stepashko V.S., Koppa Y.V. The Experience of application of the ASTRID system for the design of economic processes from statistical data // Cybernetics and computing engineering. - 1998. - V.117. – p. 24-31. (In Russian)