ONE APPROACH FOR THE OPTIMIZATION OF ESTIMATES CALCULATING ALGORITHMS

A.A. Dokukin

Abstract: In this article the new approach for optimization of estimations calculating algorithms is suggested. It can be used for finding the correct algorithm of minimal complexity in the context of algebraic approach for pattern recognition

Keywords: Pattern recognition, estimates calculating algorithms.

Introduction

This work is made in the context of algebraic approach [1] (in what follows, we use the notation and definitions from [1,2]) for pattern recognition. The task of recognition is considered. We have a set *M* of possible objects. It is presumed that $M = M_1 \times ... \times M_n$, there M_i are sets of possible values of *i*-th feature, and some semi-metrics are defined on each of them. The set *M* is divided into *l* classes $K_1,...,K_l$. The task of recognition is defined by the conventional learning information $I_0 = \{S_1,...,S_m,\alpha(S_1),...,\alpha(S_m)\}$, and the finite sample, $\beta(S^i) = (\beta_{i1},...,\beta_{il})$ of test objects. Here $S_1,...,S_m$ are descriptions of training sequence objects $S_i = (a_{i1}, a_{i2},...,a_{in}), a_{ij} \in M_j, \quad i = \overline{1,m}, j = \overline{1,n}$, and $\alpha(S_i) = (\alpha_{i1},...,\alpha_{il})$ are information vectors of objects S_i, with respect to the properties $P_j(S) \equiv \{S \in K_j\}, j = \overline{1,l}$. Correspondently $\beta(S^j) = (\beta_{i1},...,\beta_{il})$ are information vectors of S^j.

The task is to find algorithm in the algebraic closure of some set of recognition operators that calculates information vector for each $S^i \in \tilde{S}^q$. As such system the defined below class of ECA (estimates calculating algorithm) is considered.

Yu.I. Zhuravlev have proved [1] that there exixts a correct polynomial in the algebraic closure of ECA, i.e. polynomial that provides no errors on the control information $\tilde{S}^q, \{\beta(S^1),...,\beta(S^q)\}$.

Estimates calculating algorithm A is defined as $A = B \cdot C$, where $B(I_0, \tilde{S}^q) = \|\Gamma_{ij}\|_{q\times l} = \|\Gamma_j(S^i)\|_{q\times l}$ is recognition operator, $C(\|\Gamma_{ij}\|_{q\times l}) = \|\beta_{ij}\|_{q\times l}$ is solving rule.

$$\Gamma_j(S^i) = x_1\Gamma_j^1(S^i) + x_0\Gamma_j^0(S^i). \tag{1}$$

$$\Gamma_j^1(S^i) = \frac{1}{Q_1}\sum_{S\in\tilde{K}_j}\sum_{\omega\in\Omega_A}\gamma(S^i)p(\omega)B(\omega S^i, \omega S) \tag{2}$$

$$\Gamma_j^0(S^i) = \frac{1}{Q_0}\sum_{S\in C\tilde{K}_j}\sum_{\omega\in\Omega_A}\gamma(S^i)p(\omega)\overline{B(\omega S^i, \omega S)}. \tag{3}$$

Following notation is used:

- The *j*-th class and its addition are denoted as $\tilde{K}_j = K_j \cap \{S_1,...,S_m\}$ and $C\tilde{K}_j = \{S_1,...,S_m\} \setminus \tilde{K}_j$.

- Let $\{\Omega\}$ is the set of all subsets of $\{1,...,n\}$. Some subset Ω_A of Ω is attributed to an algorithm. Its elements $\omega_t = \{i_1,...,i_{k_t}\} \in \{\Omega_A\}$ are called support sets and $p(\omega_t) = p_{i1} + ... + p_{ik_t}$ are their weights, $p(\omega_t) \geq 0$.
- $\gamma(S^i) \geq 0$ are weights of training objects.
- $B(\omega S^i, \omega S)$ is proximity function. We use proximity functions only of the following type. Let $\varepsilon_1,...,\varepsilon_n$ are non-negative numbers, let also $\omega S = \{a_{i1},...,a_{ik}\}$, $\omega S' = \{b_{i1},...,b_{ik}\}$ then $B(\omega S, \omega S') = \begin{cases} 1, & \rho_{i1}(a_{i1}, b_{i1}) \leq \varepsilon_{i1},...., \rho_{ik}(a_{ik}, b_{ik}) \leq \varepsilon_{i\varepsilon} \\ 0, & otherwise \end{cases}$.
  $\overline{B(\omega S^i, \omega S)} = 1 - B(\omega S^i, \omega S)$.

Denote a set of recognition operators by $\{\tilde{B}\}$. Let $B', B'' \in \{\tilde{B}\}$, $B'(I_0, \tilde{S}^q) = \left\| \Gamma_{ij}' \right\|_{q \times l}$, $B''(I_0, \tilde{S}^q) = \left\| \Gamma_{ij}'' \right\|_{q \times l}$, b is a scalar. Following operations bB', $B' + B''$, $B' \cdot B''$ can be defined on this set as shown below.

$$(bB')(I_0, \tilde{S}^q) = \left\| b\Gamma_{ij}' \right\|_{q \times l} \tag{4}$$

$$(B' + B'')(I_0, \tilde{S}^q) = \left\| \Gamma_{ij}' + \Gamma_{ij}'' \right\|_{q \times l} \tag{5}$$

$$(B' \cdot B'')(I_0, \tilde{S}^q) = \left\| \Gamma_{ij}' \cdot \Gamma_{ij}'' \right\|_{q \times l} \tag{6}$$

The closure $M(\{\tilde{B}\})$ with respect to operations (4)-(6) is associative algebra with commutative multiplication. Operators from $M(\{\tilde{B}\})$ can be presented as polynomials of operators from $\{\tilde{B}\}$. If $B \in M(\{\tilde{B}\})$ then $B = \sum B_{i_1} \cdot B_{i_2} \cdot ... \cdot B_{i_t}$. The maximum number of multipliers in its items is called the degree of recognition operator.
The family $M(\{A\})$ of algorithms $A = B \cdot C$ such that $B \in M(\{\tilde{B}\})$ is called algebraic closure of $\{A\}$.
Finally we will need some more terms from [3] to continue the statement. The informational matrix $\left\| \beta_{i,j} \right\|_{q \times l}$ is considered. Suppose $M = \{(i,j)\}$, $i = 1,...,q$, $j = 1,...,l$, $M_\alpha = \{(i,j) \mid \beta_{i,j} = \alpha\}$, $\alpha \in \{0,1\}$.
Operator $B \in M(\{\tilde{B}\})$ is called admissible if there exists at least one pair $(i,j) \in M_1$ such that for all pairs $(u,v) \in M_0$ $\Gamma_j(S^i) > \Gamma_v(S^u)$. This pair is called marked. It is proved also [3] that the greater value $d(i,j,B) = \min\limits_{(u,v) \in M_0} (\Gamma_j(S^i) - \Gamma_v(S^u))$ is the smaller degree of item will be needed to construct the correct polynomial.
Thus in order to construct a correct algorithm of minimal complexity or to make inductive procedure of constructing it (like for example one in [4]), we need to find the algorithm of maximum $d(i,j,B)$ in some family of algorithms. This article is devoted to solving of maximization task in two particular subsets of ECA.

γ-optimization

First, denote by $\{B\}_\gamma$ the subset of ECA with the following parameters:

- $x_0 = 0, x_1 = 1$,
- Ω_A consists of all support sets of equal fixed power k. $p_i = 1/k, i = 1,...,n$,
- $\tilde{\gamma} \in [0,1]^m$,
- $\varepsilon_1,...,\varepsilon_n$ are fixed.

Let we have $(i, j) \in M_1$. The task is to find $\tilde{\gamma}^* \in [0,1]^m$ such that

$$\max_{B\in\{B\}_\gamma} \min_{(u,v)\in M_0} (\Gamma_j(S^i) - \Gamma_v(S^u)) = \min_{(u,v)\in M_0} (\Gamma_j(S^i) - \Gamma_v(S^u))|_{\tilde{\gamma}=\tilde{\gamma}^*} . \quad (7)$$

As shown in [1], in case of this special format of support vectors, the estimations (1)-(3) can be transformed into simple view:

$$\Gamma_j(S^i) = x_1\Gamma_j^1(S^i) + x_0\Gamma_j^0(S^i) = \Gamma_j^1(S^i),$$

$$\Gamma_j^1(S^i) = \frac{1}{Q_1}\sum_{S\in\tilde{K}_j}\sum_{\omega\in\Omega_A}\gamma(S^i)p(\omega)B(\omega S^i, \omega S) =$$

$$= \frac{1}{Q_1}\sum_{S\in\tilde{K}_j}\gamma(S)((\delta(S,S^i)\cdot p(\omega))V^1 + (\tilde{\delta}(S,S^i)\cdot p(\omega))V^0)$$

Here $\delta(S,S^i) \in \{0,1\}^n$ is the characteristic vector $\delta_u((a_1,...,a_n),(b_1,...,b_n)) = \begin{cases} 1, \rho_u(a_u,b_u) \le \varepsilon_u \\ 0, \rho_u(a_u,b_u) > \varepsilon_u \end{cases}$,

$\tilde{\delta}(S,S^i) \in \{0,1\}^n$ is its denial : $\tilde{\delta}_u = 1 - \delta_u$, $q(S,S^i) = \sum_{u=0}^{n}\delta_u$.

$$V^1(S,S^i) = \sum_{u=0}^{\varepsilon} C^u_{n-q(S,S^i)} C^{k-u-1}_{q(S,S^i)-1}, \quad V^0(S,S^i) = \sum_{u=1}^{\varepsilon} C^{u-1}_{n-q(S,S^i)-1} C^{k-u}_{q(S,S^i)} .$$

So in the $\{B\}_\gamma$ family of ECA, the estimation $\Gamma_j(S^i) - \Gamma_v(S^u)$ is linear function on $\tilde{\gamma} \in [0,1]^m$, that is $\Gamma_j(S^i) - \Gamma_v(S^u) = L_{i,j,u,v}(\tilde{\gamma})$. So the task transforms into another one, i.e. to find $\arg\max_{\tilde{\gamma}\in\tilde{\gamma}^*} \min_{(u,v)\in M_0} L_{i,j,u,v}(\tilde{\gamma})$, there $L_{i,j,u,v}(\tilde{\gamma}) = \sum_{s=1}^{m} l_{s,i,j,u,v}\gamma_s$. This task in turn can be transformed into *t* tasks of linear programming, there $t = |M_0|$ (we enumerate all those linear combinations as $L_1,...,L_t$ in any order):

$$\tilde{\gamma}^* = \arg\max_{\tilde{\gamma}\in\{\tilde{\gamma}_1^*,...,\tilde{\gamma}_t^*\}} L_i(\tilde{\gamma}_i^*)$$

$$\begin{cases} \tilde{\gamma}_i^* = \arg\max_{\tilde{\gamma}} L_i(\tilde{\gamma}) \\ L_i(\tilde{\gamma}) \le L_1(\tilde{\gamma}) \\ \ldots \\ L_i(\tilde{\gamma}) \le L_t(\tilde{\gamma}) \\ \tilde{\gamma} \in [0,1]^m \end{cases}, \quad i = 1,...,t .$$

These tasks can be solved with, for example, simplex method. So the precise solution of the initial task can be found.

γ,ε-optimization

The second task is more complex. As in previous chapter we choose parametrical subset $\{B\}_{\gamma,\varepsilon}$ of ECA first:

- $x_0 = 0, x_1 = 1$,
- Ω_A consists of the single support set (the method can be simply generalized to include cases of small number of support sets),
- $\tilde{\gamma} \in [0,1]^m$,
- $\varepsilon_1,...,\varepsilon_n \geq 0$.

The task is the same as in previous section, i.e. to find in $\{B\}_{\gamma,\varepsilon}$ the algorithm with the maximum value of $d(i, j, B)$.

The algorithm for solving of this task consists of two parts. First one is the construction of auxiliary finite system of parallelepipeds P:

1. Build new sequence of objects $\{S'_1,...,S'_t\}$: for all $S \in \tilde{K}_j$ add differences $S^i - S$ to the sequence.
2. Find the minimal system P of parallelepipeds $[-\varepsilon_1,\varepsilon_1]\times...\times[-\varepsilon_n,\varepsilon_n]$ containing all different combinations of objects from $\{S'_1,...,S'_t\}$.

To construct the system P we must for all subsets $S \subset \{S'_1,...,S'_t\}$ find out if its combination is possible, i.e. if there exists any parallelepiped $E = [-\varepsilon_1,\varepsilon_1]\times...\times[-\varepsilon_n,\varepsilon_n]$ such that $S' \in S$ if and only if $S' \in E$, and for all possible combinations add the minimal parallelepiped spanning it to the system. In practice there is no need to enumerate all different subsets of $\{S'_1,...,S'_t\}$. If we have found any impossible one, every combination containing it is impossible too.

The following theorem can be proved: $\max\limits_{\varepsilon\in[0,\infty)^n} \min\limits_{(u,v)\in M_0} (\Gamma_j(S^i) - \Gamma_v(S^u)) = \max\limits_{\varepsilon\in P} \min\limits_{(u,v)\in M_0} (\Gamma_j(S^i) - \Gamma_v(S^u))$. Indeed for any ε-neighborhood $[-\varepsilon_1,\varepsilon_1]\times...\times[-\varepsilon_n,\varepsilon_n]$ the maximum one from P containing in it will give not more estimations.

The second part is to calculate estimations themselves and solve the task. From (1)-(3) we have

$\Gamma_j(S^i) = g_1\gamma_1 + ... + g_n\gamma_n, g_k \in \{0,1\}, k = 1,...,n$,

$\Gamma_v(S^u) = g_1^{u,v}\gamma_1 + ... + g_n^{u,v}\gamma_n, g_s^{u,v} \in \{0,1\}, s = 1,...,n$.

And the difference is

$$\Gamma_j(S^i) - \Gamma_v(S^u) = g_1\gamma_1 + ... + g_n\gamma_n - g_1^{u,v}\gamma_1 - ... - g_n^{u,v}\gamma_n. \quad (8)$$

So the solution is

$$\tilde{\gamma}^* : \gamma_s^* = \begin{cases} 1, & g_s = 1 \\ 0, & otherwise \end{cases}, \quad s = 1,...,n.$$

Indeed for any $\tilde{\gamma} \in [0,1]^n$, difference (8) is smaller than $\Gamma_j^*(S^i) - \Gamma_v^*(S^u) = g_1\gamma_1^* + ... + g_n\gamma_n^* - g_1^{u,v}\gamma_1^* - ... - g_n^{u,v}\gamma_n^*$. The initial task transforms into finding $\arg\max\limits_{\varepsilon\in P} \min\limits_{(u,v)\in M_0} \Gamma_j(S^i) - \Gamma_v(S^u)$ and the precise solution can be found too.

Though the solution is precise the necessity to construct system P makes the task extremely difficult with multidimensional data. In order to make calculation faster we suggest proximate method for the same task.

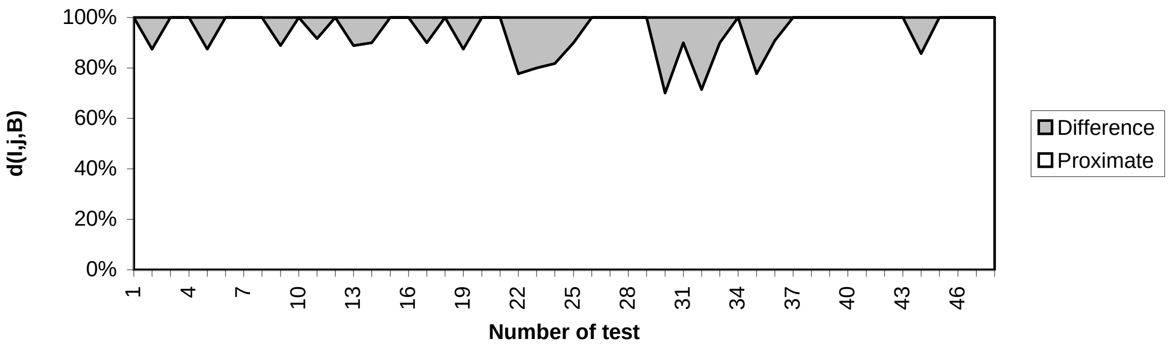
The method starts with the parallelepiped spanning the whole sequence $\{S'_1,...,S'_t\}$. Then on every step we enumerate all admissible combinations of t-1 objects and leave the best one for next step, there we consider neighborhood spanning those best combination. Here t is the number of objects in current parallelepiped. The best combination is one that maximizes the value of $d(i, j, B)$.

The following diagram shows results of hands-on testing of this method in comparison with the precise one. The table of descriptions of forty-eight patients was considered. It consists of three classes of correspondingly seventeen, twenty and twelve objects and thirty-three features. As the M_1 in turns every object was considered. All other objects from its class were considered as the training sequence. All objects from other classes formed M_0. For example the twentieth object generated the following (20-th) test:

$$M_1 = \{(20,2)\}$$

$$M_0 = \{(1,2),(2,2),...,(17,2),(38,2),(39,2),...,(48,2)\}$$

$$\{S_1',...,S_t'\} = \{S_{18}, S_{19}, S_{21}, S_{22},..., S_{37}\}.$$



It's easy to see that in most cases the precise solution or solution of acceptable precision has been found. And while the precise solution takes about two minutes to find (in case of twenty training objects and the difficulty extremely grows with increasing of their number), the proximate algorithm performs all forty-eight tests within about ten seconds.

Conclusion

In this article we have suggested the new approach for optimization of estimations calculating algorithms. It can be used for finding of the correct algorithm of the minimal complexity in the context of the algebraic approach for the pattern recognition.

Also we have considered two parametrical subsets of ECA and have found precise algorithms for solving optimization task for them.

Finally the fast proximate method with acceptable precision has been suggested.

Acknowledgements

The research described in this publication was made possible as a part of Grants 02-01-00558, 00-01-00650, 02-07-90134, 02-07-90137 from the Russian Fund of Fundamental Research, and INTAS 00-650, INTAS 00-370.

Bibliography

[1] Yu.I.Zhuravlev. Ob algebraicheskom podhode k resheniyu zadach raspoznavaniya ili klassifikacii // Problemy kibernetiki. 33, M: Nauka, 1978. (in russian)

[2] Yu.I.Zhuravlev. Korrektnye algebry nad mnozhestvom nekorrektnyh (evristicheskih) algoritmov // Kibernetika. 1977. 4. (in russian)

[3] Yu.I.Zhuravlev, I.V. Isaev. Postroenie algoritmov rasposnavania, korrektnyh dlya zadannoi kontrol'noi vyborki // Zhurnal vychislitel'noi matematiki I matematicheskoi fiziki, t.19, N3, may-june1979. (in russian)

[4] A.A.Dokukin. Induktivnyi metod postroeniya korrektnogo algoritma v algebrah nad modelyu vychisleniya ocenok // Zhurnal vychislitel'noi matematiki I matematicheskoi fiziki, 2003. (in russian)

Author information

Alexander A. Dokukin – Dorodnicyn Computing Centre of the Russian Academy of Sciences, Vavilov st., 40, Moscow GSP-1, 119991, Russia; e-mail: dalex@ccas.ru