Applied Problems of Nonlinear Oscillation and Wave Theory

Izvestiya Vysshikh Uchebnykh Zavedeniy. Applied Nonlinear Dynamics. 2024;32(2)

Article DOI: 10.18500/0869-6632-003089

Learning mechanism for a collective classifier based on competition driven by training examples

A. A. Sutyagin✉, *O. I. Kanakov*

National Research Lobachevsky State University of Nizhny Novgorod, Russia
E-mail: ✉alealesutyagin@gmail.com, okanakov@rf.unn.ru

Received 21.10.2023, accepted 30.11.2023, available online 12.01.2024, published 29.03.2024

Abstract. The *purpose* of this work is to modify the learning mechanism of a collective classifier in order to provide learning by population dynamics alone, without requiring an external sorting device. A collective classifier is an ensemble of non-identical simple elements, which do not have any intrinsic dynamics neither variable parameters; the classifier admits learning by adjusting the composition of the ensemble, which was provided in the preceding literature by selecting the ensemble elements using a sorting device. *Methods.* The population dynamics model of a collective classifier is extended by adding a "learning subsystem", which is controlled by a sequence of training examples and, in turn, controls the strength of intraspecific competition in the population dynamics. The learning subsystem dynamics is reduced to a linear mapping with random parameters expressed via training examples. The solution to the mapping is an asymptotically stationary Markovian random process, for which we analytically find asymptotic expectation and show its variance to vanish in the limit under the specified assumptions, thus allowing an approximate deterministic description of the coupled population dynamics based on available results from the preceding literature. *Results.* We show analytically and illustrate it by numerical simulation that the decision rule of our classifier in the course of learning converges to the Bayesian rule under assumptions which are essentially in line with available literature on collective classifiers. The implementation of the required competitive dynamics does not require an external sorting device. *Conclusion.* We propose a conceptual model for a collective classifier, whose learning is fully provided by its own population dynamics. We expect that our classifier, similarly to the approaches taken in the preceding literature, can be implemented as an ensemble of living cells equipped with synthetic genetic circuits, when a mechanism of population dynamics with synthetically controlled intraspecific competition becomes available.

Keywords: competition, machine learning, classifier, Lotka–Volterra model.

Acknowledgements. This work was supported by the Ministry of Science and Higher Education of the Russian Federation (project No. FSWR-2023-0031).

For citation: Sutyagin AA, Kanakov OI. Learning mechanism for a collective classifier based on competition driven by training examples. Izvestiya VUZ. Applied Nonlinear Dynamics. 2024;32(2):160–179. DOI: 10.18500/0869-6632-003089

Introduction

The classification task consists of assigning a classified object to one of a predetermined set of classes based on the value of some observable feature, which can be either a real value or, for example, a real vector, etc. [1, 2]. The number of classes, generally speaking, is determined by the statement of the problem; the present study is limited to the two-class case. The decision rule of a classifier is a function that maps a set of attribute values to a set of classes. If the conditional probability distributions on the set of attribute values are known according to the condition of the classified object belonging to each of the classes, then the minimum probability of classification error is ensured by Bayes' decision rule [1, section. 1.1–1.2], [2, sec. 3]. If these distributions are unknown, but a set of "training examples" — values of the attribute are given, each of which is equipped with a "class label", that is, an indication that the object belongs to a certain class, then the process of constructing an optimal classifier (for example, in the sense of minimizing the probability of error) based on training examples is called classifier training. The vast majority of research in the field of machine learning is focused on creating algorithms that can be executed by digital computing devices [1, 2]. At the same time, in nature there are non-algorithmic learning systems in which learning is provided by the own dynamics of these systems. These include not only the nervous systems, but also, for example, the adaptive immune systems of living organisms. It is of interest not only to study similar objects in living nature, but also to create artificial learning systems based on similar principles (see literature review in [3]).

In the works [4,5], collective classifiers were introduced into consideration - learning systems, the material basis of which is ensembles of non-identical elements of limited complexity, each of which implements some fixed (and quite simple) function of response to input features that does not have tuning parameters, and the classifier (the ensemble as a whole) is trained by forming an optimal ratio of the numbers of elements of various types in the ensemble. A method for carrying out such learning was described by selecting an ensemble (selectively removing elements) using an external sorting device based on the responses of individual elements to training examples. An implementation of an ensemble element in the form of a bacterial cell with a synthetic gene network was proposed, therefore, for definiteness, we will further call the element of the collective classifier a cell, although this study is of a general nature, without reference to a biological or other implementation. Moreover, the selection procedure described in [4, 5] always leads to a mode of competitive population dynamics with a single winner (in the limit, a single cell type remains), which significantly limits the applicability of this approach; in particular, it is impossible to ensure the asymptotic convergence of the trained classifier in the limit of the learning process to the optimal (for example, Bayesian) decision rule.

In subsequent work [3], an approach was proposed to overcome this limitation through a transition in population dynamics from a regime with a single winner to a regime of competition with coexistence. A model of competitive dynamics was proposed that has the required properties and ensures the formation of a trained ensemble as a stable state of equilibrium in population dynamics; It is also proposed to implement the required competitive dynamics using a selection algorithm controlled by training examples. A common disadvantage of selection methods when training collective classifiers [3–5] is the need to use a sorting device that is external to the system, since this means that the system is not trained using only its own dynamics.

The purpose of this study is to fill this gap. The model of competitive dynamics, previously proposed in [3], is modified and supplemented with a subsystem that ensures, firstly, training the system using examples directly due to the dynamics of the system and, secondly, maintaining the composition of the ensemble after training. It is shown analytically and illustrated by numerical

simulation that the ensemble formed as a result of the proposed population dynamics is a trained collective classifier, the response function of which is an approximation of the Bayesian decision rule.

1. Methodology

1.1. A coexistence competition model based on an external sorter. In the work [3], a collective classifier training strategy was proposed and investigated, based on population dynamics, including intraspecific competition along with competition for a common resource, thereby ensuring competition with coexistence. The dynamics of the numbers of cell types is described by a system of ordinary differential equations

$$\dot{n}_i = b_i n_i \left(1 - \frac{1}{N_c}\sum_{j=1}^{s} n_j\right) - (1-b_i)\frac{n_i^2}{N_c}, \quad i = 1\ldots s, \tag{1}$$

where the numbers of cell types n_i act as continuous dynamic variables, the index i numbers the cell types, s — the number of cell types, b_i — competitiveness parameters, $N_c > 0$ - – "medium capacity". The parameters are subject to the constraint $0 < b_i < 1$, and the phase space is limited by a non-negative orthant (all $n_i \geqslant 0$), which is an invariant region in the complete real space.

The system (1) has an equilibrium state

$$n_i = N_c \frac{b_i}{1-b_i} h, \quad \text{where} \quad h = \left(\sum_{j=1}^{s} \frac{1}{1-b_j} - (s-1)\right)^{-1}, \tag{2}$$

globally stable in the positive orthant of the phase space (where all $n_i > 0$).

In the (1) model, the parameter b_i specifies the ratio of the contributions of the mechanisms of intraspecific competition and competition for a common resource in the population dynamics of the corresponding cell type: with $b_i = 1$ on the right side of the i-th equation in (1) only the first term is non-zero (meaning competition for a common resource), while for $b_i = 0$ only the second term (meaning intraspecific competition) is non-zero. Based on the (1) model, [3] proposed a selection algorithm driven by training examples, generating competitive dynamics that transform into (1) in the continuous limit, resulting in, as shown, a decision rule of the trained classifier approaches the (optimal) Bayes rule.

The mechanisms for training collective classifiers based on selection, presented in [3–5], have a common drawback - the need for an external sorting device to provide the required competitive dynamics. Overcoming this limitation requires solving two problems, which are the focus of this study: firstly, it is required that the collective classifier be trained using the ensemble's own population dynamics, and secondly, the composition of the ensemble must be preserved after training is completed.

The solution to the first problem, that is, the rejection of external selection during training, implies the actual implementation of the mechanism of competitive population dynamics in the ensemble elements (that is, controlled reproduction and removal of elements) together with a mechanism for adjusting competitiveness parameters controlled by training examples. In the case of implementing an ensemble element in the form of a bacterial cell with a synthetic gene network [4], competitiveness control can be organized, for example, by controlling the cell's antibiotic resistance through a synthetic gene network, and the input of training examples and turning on/off the learning mechanism — for example, through the use of chemical channels of intercellular

communication (such as quorum sensing [4]). In any case, the competitiveness coefficients, which in the (1) model are constant parameters, when moving to a learning mechanism based on the element's own dynamics, become additional dynamic variables, the dynamics of which must be described by the corresponding subsystem (hereinafter — "subsystem training"), in addition to the subsystem describing population dynamics. The present study does not aim to describe any specific material implementation of the required competitive dynamics governed by the training examples. We limit ourselves to demonstrating the fundamental possibility of solving the problem posed on the basis of a simple sufficient model.

To solve the second of these problems, that is, to ensure the preservation of the population structure after training, it is necessary that, outside the learning process, the relative proportions of cells of different types in the ensemble do not change, or change on a time scale exceeding the required time for maintaining the trained state of the system. In the original form of the (1) model, this requirement cannot be met, since learning within the meaning of the model is carried out by controlling the competitiveness parameters b_i. The termination of training, therefore, implies the termination of competitiveness control, that is, the return of all parameters b_i to some "initial" value, the same for all cells and determined by the implementation of the element. This, in turn, entails a transition of the system to a stable state of equilibrium, in which the numbers of all types of cells are also the same, which means that the system forgets its trained state. To solve this problem, it is proposed to modify the (1) system in such a way that the parameter controlling the competitiveness of each cell type affects only *intraspecific* competition within a specific cell type, while the competitiveness of all cell types in competition *for share* becomes the same. With this approach, as shown in section 1.2, the termination of training transfers competition to a degenerate mode, preserving the proportions of cell types in the ensemble.

1.2. Modified model of competition with coexistence. To implement the required dynamics, we modify the model (1) by dividing the right-hand side of each equation by b_i and introducing redesignation of the parameters, as a result of which the modified model is written in the form

$$\dot{n}_i = n_i \left(1 - \frac{1}{N_c} \sum_{j=1}^{s} n_j \right) - k_i \frac{n_i^2}{N_c}, \quad \text{where} \quad k_i = \frac{1 - b_i}{b_i}, \quad i = 1 \dots s. \tag{3}$$

The constraint $b_i \in (0, 1)$ goes to $k_i \in (0, +\infty)$. Note that the parameters k_i in the model (3) from the point of view of population dynamics are interpreted as determining intraspecific competition. If the parameters k_i are constant in time and satisfy the specified constraint, then the expression for the equilibrium state (2) remains valid and, taking into account the redesignation of the parameters, is rewritten as

$$n_i = h\, N_c\, k_i^{-1}, \quad \text{where} \quad h = \left(1 + \sum_{j=1}^{s} k_j^{-1} \right)^{-1}. \tag{4}$$

The statement about the global stability of the equilibrium state (4) in the positive orthant remains valid (see paragraph 1 of the Appendix); this means that the coexistence regime determined by the equilibrium state (4) is the only possible result of competitive dynamics in the system (3).

The equilibrium value of the total number of cells does not differ from the known [3, eq.

(6)] for the model (1) and is obtained by summing all numbers n_i in the equilibrium state(4):

$$N_\infty = \sum_{i=1}^{s} n_i = N_c\, h \sum_{i=1}^{s} k_i^{-1} = N_c\, h \left(h^{-1} - 1\right) = N_c\,(1-h). \tag{5}$$

We will assume that the termination of training corresponds to the case when all parameters k_i become zero. The expression for the equilibrium state (4) in this case is not applicable, however the system itself (3) retains its meaning and turns into a degenerate model of competition with the same competitiveness

$$\dot{n}_i = n_i \left(1 - \frac{1}{N_c} \sum_{j=1}^{s} n_j\right), \quad i = 1 \ldots s. \tag{6}$$

This model has a $(s-1)$-dimensional invariant manifold given by the equation

$$\sum_{j=1}^{s} n_j = N_c \tag{7}$$

(a hyperplane cutting off identical segments $n_i = N_c$ on all coordinate axes), which vanishes the expression in brackets on the right side of (6), that is, it consists entirely of equilibrium states, and at the same time is globally stable in the positive orthant (the proof of the last statement follows below).

In addition, an invariant manifold is each ray in phase space emanating from the origin and described by the equation

$$n_i = \nu_i\, N, \tag{8}$$

where ν_i are constants specifying the direction of the ray, and $N \in [0, +\infty)$ is the coordinate along the ray. In fact, substituting (8) into (6) turns each equation of the system (6) into the same equation describing the dynamics of the variable N (motion along the ray)

$$\dot{N} = N \left(1 - \frac{N}{N_c} \sum_{j=1}^{s} \nu_j\right). \tag{9}$$

Without loss of generality, we will assume that the constants ν_i satisfy the additional condition

$$\sum_{j=1}^{s} \nu_j = 1 \tag{10}$$

(which can be done for any ray (8) by multiplying the right side of (8) by a constant), then from (8) we have

$$N = \sum_{j=1}^{s} n_j. \tag{11}$$

In this case, the variable $N(t)$ has the meaning of the total number of cells, and its dynamics are then described by a special case of the equation (9)

$$\dot{N} = N \left(1 - \frac{N}{N_c}\right), \tag{12}$$

having two equilibrium states: unstable $N = 0$ and stable $N = N_c$. Substituting $N = N_c$ into the equation of the ray (8) and taking into account (10), we are convinced that the stable equilibrium state on each ray lies on the invariant manifold (7). Since each point in the positive orthant lies on some ray of the form (8), this reasoning implies the above-mentioned global stability of the invariant manifold (7) in the positive orthant.

The expression (8) when executing (10) can be considered as a solution to the system (6)

$$n_i(t) = \nu_i\, N(t), \tag{13}$$

where $N(t)$ is the solution to the equation (12).

1.3. Training subsystem and method of training a collective classifier. Let us formulate a method for training a collective classifier based on competitive dynamics, based on the assumption that the population dynamics of the ensemble during the learning process is described by the model (3), where the parameters k_i, which determine the strength of intraspecific competition, are now considered time-variable, namely the dynamics variables $k_i(t)$ in the learning process are determined by training examples sequentially presented to the system, and after the end of training all k_i turn to zero (the dynamics of the system are then described by the degenerate model (6)).

Let x be the input of the classifier (a feature or, generally speaking, a vector of features), and for each type of cell a response function $f_i(x) \geqslant 0$ is given, which describes the response of any cell of the i-th type to the input the value of the feature x (everywhere we use the notation corresponding to [3]).

Let there be a training sequence $\{\pi_\alpha\}_{\alpha=1}^{N_{\text{learn}}}$ containing the total number of N_{learn} training examples , each of which is a pair $\pi_\alpha = (x_\alpha, c_\alpha)$, consisting of the attribute value x_α and the class label $c_\alpha \in \{+, -\}$, which specifies whether the example belongs to one of two classes, which are further referred to as "positive" and "negative" and are denoted by the corresponding symbols '+' and '−'.

We will assume that all training examples are presented sequentially, in a random order, the duration of presentation of each example is a fixed value T_s, and there are no time intervals between the presentation of successive examples (the total training time is therefore $N_{\text{learn}} \cdot T_s$).

Let us introduce into consideration the dynamics of variables k_i — "learning subsystem", described by the equations

$$\dot{k}_i = \begin{cases} -\tau_l^{-1}\, k_i & \text{outside the learning process,} \\ \tau_l^{-1}\, f_i(x_\alpha)\,(k^+ - k_i), & \text{if} \quad c_\alpha = +, \\ \tau_l^{-1}\, f_i(x_\alpha)\,(k^- - k_i), & \text{if} \quad c_\alpha = -, \end{cases} \tag{14}$$

where α is the number of the training example presented at the current time, τ_l is a parameter that determines the time scale of the learning process, k^+ and k^- are constants.

If for some k_i only one of the cases on the right side of the equation (14) is implemented for an unlimited time, and $x_\alpha = \text{const}$ (that is, one training example or none is presented for an unlimited time example), then we have for k_i an autonomous system of the 1st order (on the phase line), which for $f_i(x_\alpha) > 0$ has a unique globally stable equilibrium state $k_i = 0$, $k_i = k^+$ or $k_i = k^-$, according to the cases on the right side of (14). The characteristic time for the value $k_i(t)$ to approach the equilibrium state depends on the value $f_i(x_\alpha)$, and in the case of $f_i(x_\alpha) = 0$ (if the ith type cells does not respond to the value of the feature x_α), the dynamics (14) for the variable k_i degenerates into a constant (that is, this training example does not affect the value of the variable k_i).

The intuitive justification for the proposed training methodology is based on the consideration that a sequence of randomly selected training examples leads to stochastic dynamics, as a result of which the steady average value of each variable k_i is determined only by those training examples that fall within the region of the non-zero response of the corresponding (i-th) type of cells, namely — the ratio between the numbers of examples from the positive and negative classes among them. In particular, if in the response region of the i-th cell type there are only examples from the positive (negative) class, then during the learning process in the system (14) the right-hand side turns out to be non-zero only for the case corresponding to $c\ _{alpha} = +$ ($c_\alpha = -$), and then $k_i(t)$ tends to the deterministic limit value $k_i(t) \to k^+$ ($k_i(t) \to k^-$). If cells of the i-th type give a non-zero response to examples from both classes present in the training sequence in random order, then during the learning process random fluctuations of the variable $k_i(t)$ are inevitable in the interval between k^+ and k^-, and we can expect that if among the corresponding (that is, cells of the i type that give a non-zero response) training examples, examples from the positive (negative) class predominate, then the values of $k_i(t)$ on average will be closer to k^+ (k^-). In addition, the fulfillment of the condition

$$T_s \ll \tau_l \tag{15}$$

implies that during the time τ_l the system is presented with a large number of training examples, due to which their effect on the system should be effectively averaged, that is, the variance of fluctuations of the variables $k_i(t)$ should decrease with a decrease in the ratio T_s/τ_l.

The above intuitive reasoning is formalized in the Appendix, which provides an analysis of the dynamics of the learning subsystem (14) both for a deterministic formulation of the problem, in which the training sequence is considered given, and for a stochastic formulation, where the training sequence is considered as a random sample from some general population. It is shown that in the latter case the mathematical expectations of (random) variables $k_i(t)$ converge over time to stationary values

$$\langle k_i(t) \rangle \xrightarrow[t\to\infty]{} \hat{k}_i, \tag{16}$$

Moreover, when the condition (15) is satisfied, the stationary values of the variances $D[k_i(t)]$ tend to zero.

The corresponding competitive dynamics of the numbers of cell types n_i, described by the subsystem (3), where instead of constant parameters k_i random processes $k_i(t)$ must now be substituted, which are the solution of the learning subsystem (14), also becomes stochastic. However, in the steady-state (stationary) mode of the learning subsystem, under the assumption that the variances of the variables $k_i(t)$ are small, which is ensured by the condition (15), one can expect that the variances of the numbers $n_i(t)$ should also be small. For these reasons, we neglect fluctuations when describing the asymptotic state of the subsystem (3), which in this approximation is considered as deterministic and is described by a globally stable equilibrium state (4), where instead of k_i the corresponding stationary values of mathematical expectations $\hat{a}resubstitutedk_i$:

$$\langle n_i(t) \rangle \xrightarrow[t\to\infty]{} \hat{n}_i = h\, N_c\, \hat{k}_i^{-1}. \tag{17}$$

The stationary value of the total number is given by the expression (5), namely

$$\sum_{i=1}^{s} \hat{n}_i = N_c\,(1-h). \tag{18}$$

After the end of training, according to the first case in (14), all variables k_i tend to zero exponentially, with a time constant τ_l. Assuming that this scale is small compared to the

characteristic time scale of population dynamics (3), we can assume that the dynamics of the system instantly switches to the degenerate competition mode (6), for which the asymptotic state of the learning process is (17) acts as an initial condition. As shown in section 1.2, further movement of the representing point in phase space occurs along the corresponding ray (8) and tends to a stable equilibrium state lying at the intersection of this ray with the invariant manifold (7). Movement along the ray (8) implies that the values of the variables in the equilibrium state $n_i = n_i^{\text{eq}}$ are obtained by proportionally scaling the initial values $n_i = \hat{n}_i$ while maintaining the relationships between them, that is

$$n_i^{\text{eq}} = b\,\hat{n}_i, \tag{19}$$

where b is an unknown constant. On the other hand, due to the equation of the invariant manifold (7), we have

$$\sum_{i=1}^{s} n_i^{\text{eq}} = N_c. \tag{20}$$

From the equations (18)–(20) we find $b = (1-h)^{-1}$. Then from the equations (19) and (17) we obtain the steady-state values of the numbers of cell types n_i^{eq}, which determine the resulting composition of the trained classifier

$$n_i^{\text{eq}} = \frac{h}{1-h}\, N_c\, \hat{k}_i^{-1}. \tag{21}$$

1.4. Approximation of a Bayesian decision rule by a trained classifier. If the prior probabilities of the classified object belonging to each of the two classes are the same, then the Bayesian decision rule is optimal from the point of view of minimizing the probability of classification error [1, section. 1.1–1.2], [2, sec. 3], [3, eq. (9)]

$$C_{\text{Bayes}}(x) = \begin{cases} +, & \text{if} \quad w_+(x) > w_-(x), \\ -, & \text{if} \quad w_+(x) \leqslant w_-(x), \end{cases} \tag{22}$$

where $w_+(x)$ and $w_-(x)$ are functions of the conditional probability density distribution of attribute x according to the condition of the object belonging to class c:

$$w_+(x) = w(x|c=+), \quad w_-(x) = w(x|c=-). \tag{23}$$

Within the framework of the learning task, the functions $w_+(x)$ and $w_-(x)$ are not specified a priori, and the decision rule of the classifier must be obtained as a result of training with examples.

Let us show that the decision rule of a classifier trained using the method described in section 1.3, under certain conditions, can be considered as an approximation of the (22) rule. Following the approach used in [3], we will consider the cell response functions $f_i(x)$ to be "narrow" in the sense that they satisfy the approximate relation [3, Eq. (19)] (the issue of formalizing this approximation is discussed below)

$$\frac{\int g(x) f_i(x) dx}{\int f_i(x) dx} \approx g(x_i), \tag{24}$$

where $g(x) = w_+(x)$ or $g(x) = w_-(x)$, x_i — "central" value of the argument (feature) that can be assigned to each function $f_i(x)$ in such a way that "significantly different from zero" (making the main contribution to the integral) values of $f_i(x)$ are achieved for values of the argument x concentrated in the neighborhood of $x = x_i$. The set of values $\{x_i\}$ for all types of cells can

form a uniform or, generally speaking, non-uniform grid on the set of values of the feature x. Note that the relation (24) becomes exact if the response functions are expressed through the Dirac delta function in the form $f_i(x) = c_i\delta(x - x_i)$, where c_i are constants. For real response functions, the relation (24) is satisfied the more accurately, the less the functions $w_+(x)$ and $w_-(x)$ change on the "effective width" (which makes the main contribution to the integral) of the response function $f_i(x)$. The Appendix shows that when (15) and (24) are executed, in the case of equal prior probabilities of two classes (which, however, is not a fundamental requirement; the calculations can be modified for the case of unequal prior probabilities of classes), the stationary state (16) of the learning subsystem, established during the learning process, is approximated by the expression

$$\hat{k}_i \approx \frac{k^+ + k^-}{2} + \frac{k^+ - k^-}{2} \cdot \frac{w_+(x_i) - w_-(x_i)}{w_+(x_i) + w_-(x_i)}, \tag{25}$$

whence, assuming for definiteness $k^+ < k^-$, we obtain the relation

$$\hat{k}_i \left\{ \begin{matrix} > \\ = \\ < \end{matrix} \right\} \frac{k^+ + k^-}{2} \quad \text{at} \quad \begin{cases} w_+(x_i) < w_-(x_i), \\ w_+(x_i) = w_-(x_i), \\ w_+(x_i) > w_-(x_i), \end{cases} \tag{26}$$

performed with accuracy to all approximations made above.

Substituting the result (26) into the expression (21), for the numbers of cell types in the trained classifier n_i^{eq} we have

$$n_i^{\text{eq}} \begin{cases} < \Theta & \text{at} \quad w_+(x_i) < w_-(x_i), \\ = \Theta & \text{at} \quad w_+(x_i) = w_-(x_i), \\ > \Theta & \text{at} \quad w_+(x_i) > w_-(x_i), \end{cases} \quad \text{where} \quad \Theta = \frac{h}{1-h} N_c \frac{2}{k^+ + k^-}. \tag{27}$$

A strict formalization of the approximation (24) requires specification of assumptions about the functions included in it. The Appendix provides one version of such a formalization, presented in the form of an interval estimate (in the form of a double inequality) within the framework of conditions satisfied by the family of response functions $f_i(x)$ used in section 2. The corresponding modification of the relations (26), (27) is also given in the Appendix and can be used when developing a classifier, however, for compactness of calculations, a simplified formulation given by the relations (25)–(27) is considered below.

Let us introduce into consideration the decision rule of the collective classifier, defined similarly to [3, eq. (10)] in the form

$$C_{\text{Collective}}(x) = \begin{cases} +, & \text{if} \quad F(x) > \Theta, \\ -, & \text{if} \quad F(x) \leqslant \Theta, \end{cases} \qquad F(x) = \sum_i n_i f_i(x), \tag{28}$$

where $F(x)$ is the total response of all cells that make up the collective classifier, Θ is the classification threshold. Let us assume that the response functions $f_i(x)$ satisfy the additional requirement

$$f_i(x_j) = \delta_{ij}, \tag{29}$$

where δ_{ij} is the Kronecker symbol, as a result of which we have

$$F(x_i) = n_i. \tag{30}$$

In the example discussed in section 2, the relations (29), (30) are satisfied exactly; for real response functions (for example, when implementing a classifier in the form of an ensemble of

living cells with synthetic gene structures [4,5]), they can be considered as approximate, and the fulfillment of these relations can be ensured with some accuracy.

Applying the expression (30) in the case of a trained classifier, that is, for $n_i = n_i^{\text{eq}}$, and taking into account the relation (27) together with the expression for the threshold Θ introduced in it, we conclude that The decision rule of the trained collective classifier (28) approximates the Bayesian rule (22) in the sense that for feature values x falling on grid nodes $x \in \{x_i\}$, the response of the collective classifier (28) coincides with the Bayesian decision rule (22) with an accuracy determined by the approximations used, and outside the grid nodes depends on the specific choice of response functions $f_i(x)$ (for example, for the case considered in section 2 , is determined by linear interpolation of $F(x)$ values between grid nodes).

In general, the expression for the threshold Θ introduced in (27) is not directly suitable for practical application, since the value h included in it, initially defined in (4), is itself expressed through subsystem variables learning k_i, the equilibrium values of which are established during the learning process and are not known a priori. To solve the problem of practically finding the value of the threshold Θ, consider the limiting case

$$k^+ \ll k^-, \tag{31}$$

at which the expression for Θ becomes

$$\Theta = \frac{h}{1-h} N_c \frac{2}{k^-}. \tag{32}$$

Let us assume that among the cell types there is at least one such cell (let's denote its number as i^*), for which the overwhelming predominance of training examples from the negative class is realized, that is, $w_+(x_{i^*}) \ll w_-(x_{i^*})$. In this limit, the expression (25) for $\hat{k}_{i^*}$ becomes $\hat{k}_{i^*} = k^-$, and the corresponding number of cells of a given type in the trained classifier, according to (21), is

$$n_{i^*}^{\text{eq}} = \frac{h}{1-h} N_c \frac{1}{k^-}. \tag{33}$$

Note that for $k^+ < k^-$ the expression (25) gives $\hat{k}_i \leqslant k^-$ for all cell types i. Then the value (33) can be found as the smallest number among all cell types in the trained classifier (21)

$$n_{i^*}^{\text{eq}} = n_{\min} = \min_i \{n_i^{\text{eq}}\}, \tag{34}$$

or, according to (30), as the smallest value of the total response of the trained classifier $F(x)$ at grid nodes $\{x_i\}$

$$n_{i^*}^{\text{eq}} = n_{\min} = \min_i F(x_i). \tag{35}$$

Comparing the expressions (32) and (33), we note that the desired threshold value Θ can be found as double the value $n_{i^*}^{\text{eq}}$, that is

$$\Theta = 2n_{\min} = 2 \min_i F(x_i). \tag{36}$$

Due to the above, for the practical determination of the Θ threshold, it is necessary to ensure that among the cell types there is at least one "calibration" type for which there is an overwhelming predominance of training examples from the negative class among the examples that cause a non-zero response from cells of this type; in this case, the value of the threshold Θ can be set according to the expression (36).

2. Results

Let us specify the problem as follows. In the population dynamics subsystem (3) the only free parameter is N_c, but changing N_c is equivalent to simultaneously rescaling all state variables n_i, so the value of N_c can be chosen without loss of generality. Let us set $N_c = 1$, which corresponds to measuring the number of cells n_i in fractions of the medium capacity N_c.

In the learning subsystem (14) we set $k^- = 5$, $k^+ = 0.5 \ll k^-$, which satisfies (31). Let us choose the time scale of the learning subsystem $\tau_l = 2$.

We define the response functions of cell types similarly to [3] in the form of a linear interpolation basis on an equidistant grid

$$x_i = x_1 + (i-1)\Delta, \tag{37a}$$

$$f_i(x) = \max\left\{1 - \left|\frac{x - x_i}{\Delta}\right|, 0\right\}, \tag{37b}$$

which satisfies the requirement (29). Then $F(x)$ in (28) is a piecewise linear interpolation of the values of (30) at grid nodes. We assume that the grid step Δ is equal to 1, the number of cell types is $s = 9$, the grid x_i runs through integer values from $x_1 = 2$ to $x_9 = 10$. The graph of one of the functions of this family is shown in Fig. 1.

The initial conditions for the population dynamics subsystem are given by $n_i(0) = N_c/s$, and for the learning subsystem $k_i(0) = 1$.

The duration of presentation of one training example is $T_s = 0.01 \ll \tau_l$, which ensures the execution of (15). The total number of training examples presented is $N_{\text{learn}} = 20000$, so the total training time is $T_s \cdot N_{\text{learn}} = 200$. The duration of the system evolution after completion of training (after presentation of all training examples) is $T_{\text{free}} = 20$.

Training examples during the training process are selected from two classes randomly, with equal probability and independently. The conditional probability density functions of classes to obtain a sample of training examples are specified similarly to the work of [3]: one of these functions (for the "negative" class) is represented by a Gaussian distribution, the second (for the "positive" class) is represented by a bimodal distribution distribution obtained by superposition of two Gaussian functions:

$$w_+(x) = \frac{1}{2\sqrt{2\pi}}\left(\frac{1}{\sigma_1}e^{-\frac{(x-\mu_1)^2}{2\sigma_1^2}} + \frac{1}{\sigma_2}e^{-\frac{(x-\mu_2)^2}{2\sigma_2^2}}\right), \quad w_-(x) = \frac{1}{\sigma_3\sqrt{2\pi}}e^{-\frac{(x-\mu_3)^2}{2\sigma_3^2}}, \tag{38}$$

Simulation results are presented for the following combinations of parameters:

$$\mu_1 = 3, \quad \mu_2 = 9, \quad \mu_3 = 6, \quad \sigma_1 = \sigma_2 = \sigma_3 = 1; \tag{39a}$$

$$\mu_1 = 3, \quad \mu_2 = 7, \quad \mu_3 = 6, \quad \sigma_1 = 1, \quad \sigma_2 = \sigma_3 = 0.5. \tag{39b}$$

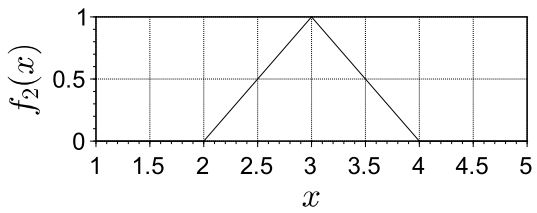

Fig 1. Graph of cell response function $f_i(x)$ for $i = 2$ ($x_i = 3$)

The corresponding graphs of the functions $w_+(x)$ and $w_-(x)$ are shown in the upper panels of Fig. 2. The values of the (39a) parameters are identical to those discussed in [3]. Additional vertical grid lines mark the values $x = x_L$ and $x = x_R$, which are the roots of the equation $w_+(x) = w_-(x)$ and represent the boundaries of the solution domains of the Bayesian decision rule (22) .

Numerical modeling of the dynamics of the complete system has been performed, consisting of the population dynamics subsystem (3), which describes the dynamics of the numbers of cell types $n_i(t)$, and the learning subsystem (14), which describes the dynamics of intraspecific competition coefficients $k_i(t)$. The resulting evolution of the dynamics of all model variables is presented in Fig. 3.

Response graphs of the trained collective classifier $F(x)$ according to (28), where the cell numbers at the final moment of the numerical experiment are taken $n_i = n_i(T_s \cdot N_{\text{learn}} + T_{\text{free}})$, along with the corresponding threshold value Θ according to (36), are shown in the lower panels of Fig. 2. A comparison of the upper and lower panels of Fig. 2 shows that the decision rule of the trained collective classifier (28), as predicted by theory, approximates the Bayesian decision rule (22) in the sense that the inequality $F(x_i) > \Theta$ (positive response of the collective classifier) occurs for feature values x in those grid nodes $x = x_i$ in which the Bayesian classifier gives a positive response ($w_+(x_i) > w_-(x_i)$), and vice versa, $F(x_i) < \Theta$ for $w_+(x_i) < w_-(x_i)$, which confirms the efficiency of the proposed method for training a collective classifier without an external sorter.

Conclusion

The new model of competitive dynamics, which is a modification of the system proposed in [3], provides training of a collective classifier only due to population dynamics, without using an external sorter. To do this, the system is supplemented with a "learning subsystem", the dynamics of which are determined by the sequence of training examples and, in turn, controls the strength of intraspecific competition in population dynamics. Using the example of the considered conceptual

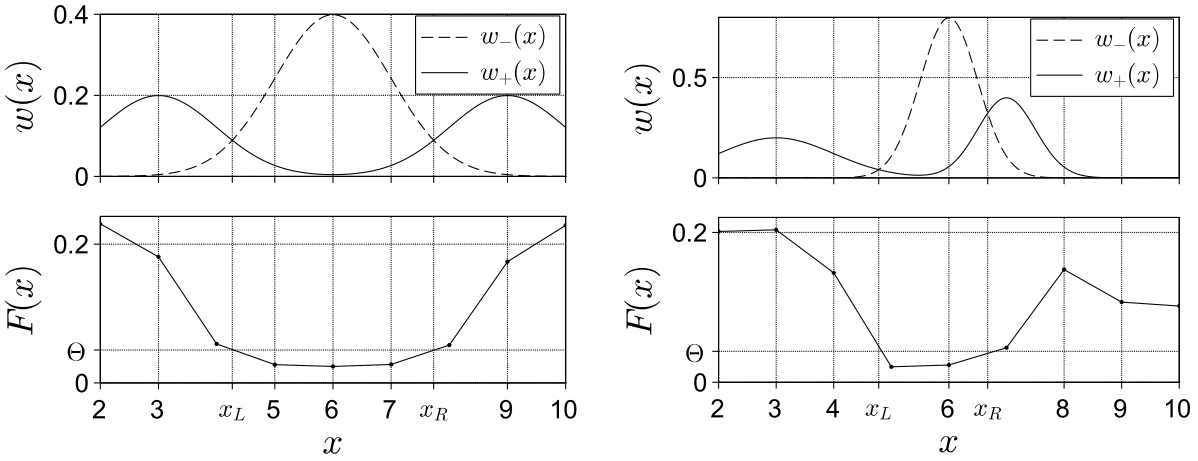

Fig 2. Upper panels: graphs of the conditional probability density functions for the classes $w_+(x)$ and $w_-(x)$ (38). Lower panels: summary response function $F(x)$ (28) of the trained classifier. Additional vertical grid lines denote the decision boundaries x_L, x_R for the Bayesian classification rule (22); additional horizontal grid lines in the lower panels denotes the threshold (36) for the collective classifier decision rule (28). Parameter values: left column (39a), right column (39b)

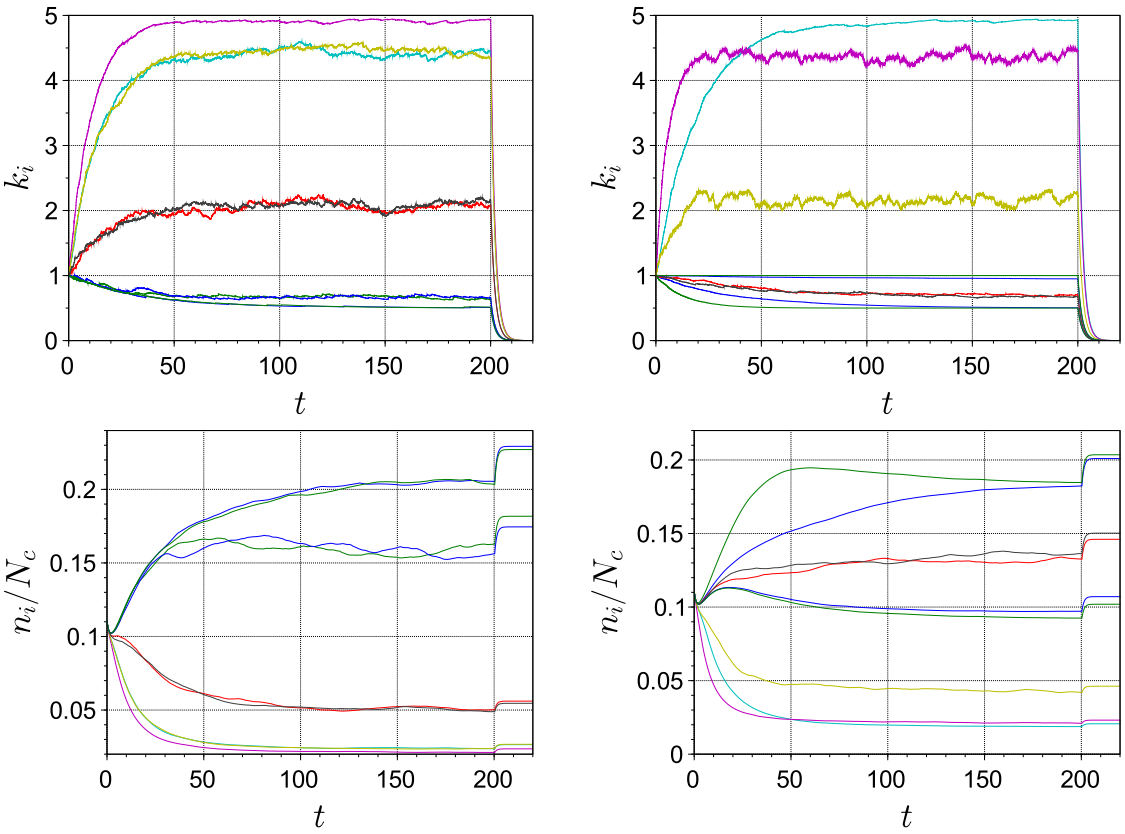

Fig 3. Evolution of the dynamical variables of the learning subsystem (14) $k_i(t)$ (upper panels) and of the population dynamics subsystem (3) $n_i(t)$ (lower panels). Time ranges: $t = 0 \dots 200$ — learning, $t = 200 \dots 220$ — transient process after learning. Parameter values: left column (39a), right column (39b) (color online)

model, it is shown that the decision rule generated by the trained classifier is an approximation of the optimal Bayesian rule if a number of assumptions are satisfied: during the characteristic time scale of the dynamics of the learning subsystem, a large number of training examples, selected randomly and independently, must be presented for their effective averaging (reducing fluctuations); the duration of training must be long enough to ensure convergence to a stable equilibrium state; the competitiveness of cell types outside the learning process must be the same to maintain the proportions of cell types in the trained ensemble; Cell response functions must be narrow in comparison with the conditional distributions of classes in the feature space and together must have the properties of an interpolation basis to form the total response function. These assumptions do not differ from those made in [4], where the implementation of a collective classifier in the form of an ensemble of living cells with synthetic gene structures was considered, which allows one to count on the possibility of such an implementation for the approach proposed in this study, but the problem of implementation remains open < <learning subsystems», that is, managing intraspecific competition based on a sequence of training examples.

Application

1. Global stability of the coexistence regime. In the work [6] the Lotka–Volterra model is written in the form

$$\dot{n}_i = n_i \left(b_i + \sum_{j=1}^{s} a_{ij} n_j \right), \quad i = 1 \dots s.$$

In the case of a non-degenerate coefficient matrix $A = (a_{ij})$, the parentheses simultaneously vanish at a single point in the phase space, which is thus the equilibrium state of the system. The theorem [6] then states that this equilibrium state is globally stable in the positive orthant if it is in it, and there is a constant diagonal matrix with positive coefficients C such that the matrix $CA + (CA)^{\mathrm{T}}$ (where $^{\mathrm{T}}$ is the transposition operation) is negative definite.

To apply this theorem to the system (3) (for clarity, the system dimension is chosen $s = 3$), the matrix A is written in the form

$$A = -\frac{1}{N_c}\begin{pmatrix} 1+k_1 & 1 & 1 \\ 1 & 1+k_2 & 1 \\ 1 & 1 & 1+k_3 \end{pmatrix},$$

and the matrix C is assumed to be singular, then

$$CA + (CA)^{\mathrm{T}} = -\frac{2}{N_c}\left[\begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix} + \begin{pmatrix} k_1 & 0 & 0 \\ 0 & k_2 & 0 \\ 0 & 0 & k_3 \end{pmatrix}\right].$$

Here the matrix of ones is positive semidefinite, the diagonal matrix has positive coefficients and, therefore, is positive definite, therefore the expression as a whole is a negative definite matrix. The remaining conditions of the theorem are also satisfied.

2. Deterministic description of the learning process. The equations that make up the learning subsystem (14) are independent of each other, so further consideration refers to one individual equation of the system, that is, the value of the index i is fixed; everywhere below the notation k is used as a shorthand equivalent for k_i. Since the learning process is described by the second and third cases in (14), they are considered further.

If the sequence of training examples $\{\pi_\alpha\}_{\alpha=1}^{N_{\text{learn}}}$ is given, then the system (14) describes deterministic dynamics with abrupt changes in time parameters on the right side. The moment of time at which the presentation of the training example begins under the number α will be denoted by

$$t_\alpha = T_s(\alpha - 1). \tag{40}$$

At each time interval of presentation of a separate training example $t \in (t_\alpha, t_{\alpha+1})$ the system (14) has constant parameters and can be represented in the form

$$\dot{k} = -\lambda_\alpha(k - k_\alpha^0), \tag{41}$$

where the notation is introduced

$$\lambda_\alpha = \tau_l^{-1} f_i(x_\alpha), \quad k_\alpha^0 = \begin{cases} k^+, & \text{if} \quad c_\alpha = +, \\ k^-, & \text{if} \quad c_\alpha = -. \end{cases} \tag{42}$$

Based on the continuity of the solution $k(t)$ at the break points of the right side $t = t_\alpha$ (that is, at the moments when the training example changes), introducing the notation

$$k_\alpha = k(t_\alpha) \tag{43}$$

for the value of the variable $k(t)$ at the initial moment of presentation of the training example π_α and considering this value as the initial condition for the equation (41), we write the corresponding particular solution in the form

$$k(t) = k_\alpha^0 + (k_\alpha - k_\alpha^0)e^{-\lambda_\alpha(t-t_\alpha)}, \quad t \in [t_\alpha, t_{\alpha+1}], \tag{44}$$

As a result, the value of the variable $k(t)$ at the end of the presentation of the training example π_α (that is, at time $t = t_\alpha + T_s = t_{\alpha+1}$) , which coincides with the initial condition $k_{\alpha+1}$ for the next training example, expressed by dot mapping

$$k_{\alpha+1} = k_\alpha^0 + (k_\alpha - k_\alpha^0)e^{-\lambda_\alpha T_s} \tag{45}$$

or equivalent

$$k_{\alpha+1} = k_\alpha(1 - \varepsilon_\alpha) + k_\alpha^0\varepsilon_\alpha, \tag{46}$$

where the designation is introduced

$$\varepsilon_\alpha = 1 - e^{-\lambda_\alpha T_s} = 1 - e^{-\frac{T_s}{\tau_l} f_i(x_\alpha)}. \tag{47}$$

Since $f_i(x) \geqslant 0$, we always have $\varepsilon_\alpha \in [0, 1)$, and $\varepsilon_\alpha = 0$ is realized only in the case of $f_i(x\ _{alpha}) = 0$, that is, if the i-th cell type gives a zero response to the value of the feature $x = x_\alpha$. In addition, by assuming the response functions $f_i(x)$ to be bounded and introducing the parameter

$$\mu = \frac{T_s}{\tau_l}, \tag{48}$$

which is small within the framework of the assumption (15) about the small duration of presentation of one training example T_s in comparison with the time scale of the training subsystem τ_l, and expanding the exponent in (47) into a series in powers of μ , we find

$$\varepsilon_\alpha = \mu f_i(x_\alpha) + O(\mu^2), \tag{49}$$

which is a small quantity of order $O(\mu)$.

Thus, having a given sequence of training examples $\{\pi_\alpha\}_{\alpha=1}^{N_{\text{learn}}}$, through which the expressions (42) and (47) are determined by the sequences of parameters $\{k_\alpha^0\}$ and $\{\varepsilon_\alpha\}$ in the mapping (46), as well as by specifying the initial condition $k(t = 0) = k_1$, we obtain from (46) the values of the variable $k(t)$ at the moments of changing training examples $k(t_\alpha) = k_\alpha$, then the values of $k(t)$ within the intervals of presentation of training examples are given by the expression (44).

3.Stochastic description of the learning process. Let us now consider a stochastic formulation of the problem, where each training example $\pi_\alpha = (x_\alpha, c_\alpha)$ is selected randomly from a certain general population of training examples, that is, the training sequence $\{\ pi_\alpha\}_{\alpha=1}^{N_{\text{learn}}}$ represents a random sample of some random variable $\pi = (x, c) \in \mathbb{R}\ times\{+, -\}$. Note that in this case, different examples π_α and $\pi_{\beta\neq\alpha}$ are independent of each other and have the same probabilistic properties, but the quantities x_α and c_α within each example are, generally speaking, dependent (it is on this dependence that the possibility of classification is based). The mapping (46) then defines a Markov random process $\{k_\alpha\}$ with discrete time and random parameters k_α^0 and ε_α, and the learning subsystem solution $k(t)$ obtained from (44) is a continuous-time random process.

Since the (random) parameters k_α^0 and ε_α in (46) are uniquely expressed through the example components π_α, they are dependent on each other, however, they are independent of the value k_α, which has the meaning of the initial condition for the learning subsystem when presented with an example π_α, which means k_α can depend on previous examples $pi_{\beta<\alpha}$, but not from π_α. From this follows the factorization of the mathematical expectation $\langle k_\alpha\varepsilon_\alpha\rangle = \langle k_\alpha\rangle\langle\varepsilon_\alpha\rangle$ [7, §25]; then from (46) we obtain a mapping for mathematical expectations

$$\langle k_{\alpha+1}\rangle = \langle k_\alpha\rangle(1 - \langle\varepsilon\rangle) + \langle k^0\varepsilon\rangle, \tag{50}$$

where the notations $\langle\varepsilon\rangle = \langle\varepsilon_\alpha\rangle$ and $\langle k^0\varepsilon\rangle = \langle k^0_\alpha\varepsilon_{areintroduced\alpha}\rangle$, independent of α due to the identity of the probabilistic properties of training examples under all numbers α.

Except for the degenerate case $\langle\varepsilon\rangle = 0$, which means that the i-th cell type does not respond to any example (see the comment to the equation (47); in this case, this type is "redundant" and can be excluded), the mapping (50) has a fixed point

$$\hat{k} = \frac{\langle k^0\varepsilon\rangle}{\langle\varepsilon\rangle}, \tag{51}$$

which is found from the condition $\langle k_{\alpha+1}\rangle = \langle k_\alpha\rangle = \hat{k}$, is stable due to $|1 - \langle\varepsilon\rangle| < 1$ and determines the stationary value to which the mathematical expectation of the process $\{k_\alpha\}$ converges. In addition, the mathematical expectation of a process with continuous time $k(t)$, which is determined by the expression (44) and takes values between k_α and $k_{\alpha+1}$ on each segment $t \in [t_\alpha, t_{\alpha+1}]$, which justifies the formula (16).

According to the formula for the total mathematical expectation [7, §23, Eq. (5)] we find

$$\langle\varepsilon\rangle = \langle\varepsilon_\alpha|c_\alpha = +\rangle P(c_\alpha = +) + \langle\varepsilon_\alpha|c_\alpha = -\rangle P(c_\alpha = -), \tag{52a}$$

$$\langle k^0\varepsilon\rangle = \langle k^0_\alpha\varepsilon_\alpha|c_\alpha = +\rangle P(c_\alpha = +) + \langle k^0_\alpha\varepsilon_\alpha|c_\alpha = -\rangle P(c_\alpha = -). \tag{52b}$$

Note that the values of k^0_α for a given c_α are determined by the expression (42), which means that when calculating conditional mathematical expectations in (52b) they can be considered deterministic values , that is

$$\langle k^0_\alpha\varepsilon_\alpha|c_\alpha = +\rangle = k^+\langle\varepsilon_\alpha|c_\alpha = +\rangle, \quad \langle k^0_\alpha\varepsilon_\alpha|c_\alpha = -\rangle = k^-\langle\varepsilon_\alpha|c_\alpha = -\rangle. \tag{53}$$

For compactness of calculations, we will consider the prior probabilities of the two classes to be the same, that is, $P(c_\alpha = +) = P(c_\alpha = -) = 1/2$. This requirement is not fundamental; further calculations can be modified for the case of unequal prior probabilities of classes.

Introducing notation for (independent of α) conditional mathematical expectations

$$\langle\varepsilon^+\rangle = \langle\varepsilon_\alpha|c_\alpha = +\rangle, \quad \langle\varepsilon^-\rangle = \langle\varepsilon_\alpha|c_\alpha = -\rangle \tag{54}$$

and using the relations (52a,b), (53), we rewrite (51) in the form

$$\hat{k} = \frac{k^+\langle\varepsilon^+\rangle + k^-\langle\varepsilon^-\rangle}{\langle\varepsilon^+\rangle + \langle\varepsilon^-\rangle} = \frac{k^+ + k^-}{2} + \frac{k^+ - k^-}{2} \cdot \frac{\langle\varepsilon^+\rangle - \langle\varepsilon^-\rangle}{\langle\varepsilon^+\rangle + \langle\varepsilon^-\rangle}. \tag{55}$$

Let the general population from which the training examples are selected be described by the conditional probability distribution densities of the feature x according to the condition of the example belonging to each of the two classes, introduced in (23). Then, under the assumption that the parameter μ is small, the conditional mathematical expectations $\langle\varepsilon^+\rangle$ and $\langle\varepsilon^-\rangle$ in (54) can be written (with an accuracy of $O(\ mu^2)$) based on the expression (49) in the form

$$\langle\varepsilon^+\rangle = \mu\int f_i(x)w_+(x)dx, \quad \langle\varepsilon^-\rangle = \mu\int f_i(x)w_-(x)dx. \tag{56}$$

If the functions $f_i(x)$ are "narrow" in the sense of the approximate relation (24), then the expressions (56) can be approximated in the form

$$\langle\varepsilon^+\rangle \approx w_+(x_i)\cdot\mu\int f_i(x)dx, \quad \langle\varepsilon^-\rangle \approx w_-(x_i)\cdot\mu\int f_i(x)dx, \tag{57}$$

as a result, the result (55) is rewritten as (25).

A strict formulation of the approximation (57) requires specification of the properties of the functions included in the expressions. For example, as shown in [3, p. 5 Applications], if on some interval $x \in [a_i, b_i]$ the functions $w_+(x)$ and $f_i(x)$ are integrable, and for $w_+(x)$ the estimate $w_+(x) \in [\underline{w}_i^+, \overline{w}_i^+]$, and for $f_i(x)$ we have $f_i(x) > 0$ on the interval $x \in (a_i, b_i)$, and $f_i(x) = 0$ for $x \notin (a_i, b_i)$ (which holds for the family of functions $f_i(x)$ given by the expressions (37a), (37b), if we put $a_i = x_{i-1}$, $b_i = x_{i+1}$), then the interval estimate is valid

$$\underline{w}_i^+ \cdot \mu \int f_i(x)dx \leqslant \langle \varepsilon^+ \rangle \leqslant \overline{w}_i^+ \cdot \mu \int f_i(x)dx, \tag{58}$$

which (together with a similar estimate for $\langle \varepsilon^- \rangle$) is a rigorous formulation of the approximation (57). The corresponding relations, similar to (26), (27), but based on the formulation of the approximation (58) instead of (57), can be obtained from (55) and are written in the form

$$\hat{k}_i \quad \left\{ \begin{matrix} > \\ < \end{matrix} \right\} \quad \frac{k^+ + k^-}{2} \quad \text{at} \quad \begin{cases} \overline{w}_i^+ < \underline{w}_i^-, \\ \underline{w}_i^+ > \overline{w}_i^-, \end{cases} \tag{59}$$

$$n_i^{\text{eq}} \quad \begin{cases} < \Theta \quad \text{at} \quad \overline{w}_i^+ < \underline{w}_i^-, \\ > \Theta \quad \text{at} \quad \underline{w}_i^+ > \overline{w}_i^-. \end{cases} \tag{60}$$

Let us now write the equation for the variances in the mapping (46) taking into account (50), also factoring the mathematical expectations taking into account the independence of k_α from k_α^0 and ε_α; For brevity, we discard the index α inside the mathematical expectations that do not depend on it:

$$\begin{aligned} D[k_{\alpha+1}] = \langle k_{\alpha+1}^2 \rangle - \langle k_{\alpha+1} \rangle^2 = \\ = \langle (k_\alpha(1-\varepsilon_\alpha) + k_\alpha^0 \varepsilon_\alpha)^2 \rangle - (\langle k_\alpha \rangle (1 - \langle \varepsilon \rangle) + \langle k^0 \varepsilon \rangle)^2 = \\ = \langle k_\alpha^2 \rangle \langle (1-\varepsilon)^2 \rangle + 2\langle k_\alpha \rangle \langle (1-\varepsilon) k^0 \varepsilon \rangle + \langle (k^0 \varepsilon)^2 \rangle - \\ - \langle k_\alpha \rangle^2 (1 - \langle \varepsilon \rangle)^2 - 2\langle k_\alpha \rangle (1 - \langle \varepsilon \rangle) \langle k^0 \varepsilon \rangle - \langle k^0 \varepsilon \rangle^2. \end{aligned} \tag{61}$$

Bringing similar terms taking into account the definition of dispersion and relation

$$\langle (1-\varepsilon)^2 \rangle = (1 - \langle \varepsilon \rangle)^2 + D[\varepsilon], \tag{62}$$

we get

$$D[k_{\alpha+1}] = D[k_\alpha](1 - \langle \varepsilon \rangle)^2 + \langle k_\alpha^2 \rangle D[\varepsilon] - 2\langle k_\alpha \rangle (\langle k^0 \varepsilon^2 \rangle - \langle k^0 \varepsilon \rangle \langle \varepsilon \rangle) + D[k^0 \varepsilon]. \tag{63}$$

In the limit $\mu \to 0$ we have the smallness relation $\varepsilon = O(\mu)$ (49), which implies $D[\varepsilon] = O(\mu^2)$, $D[k^0 \varepsilon] = O(\mu^2)$, and

$$D[k_{\alpha+1}] = D[k_\alpha](1 - 2\langle \varepsilon \rangle) + O(\mu^2), \tag{64}$$

then for the stationary value of the variance $D[k]_s = D[k_{\alpha+1}] = D[k_\alpha]$ we find

$$D[k]_s = \frac{O(\mu^2)}{2\langle \varepsilon \rangle} = O(\mu) \xrightarrow[\mu \to 0]{} 0. \tag{65}$$

References

1. Aivazyan SA, Buchstaber VM, Yenyukov IS, Meshalkin LD. Applied Statistics: Classification and Reduction of Dimensionality. Moscow: Finansy i Statistika; 1989. 608 p. (in Russian).
2. Alpaydin E. Introduction to Machine Learning. Fourth Edition. Cambridge, Massachusetts: MIT Press; 2020. 683 p.
3. Sutyagin AA, Kanakov OI. Collective classifier learning strategy based upon competition in the coexistence regime. Izvestiya VUZ. Applied Nonlinear Dynamics. 2021;29(2):220–239 (in Russian). DOI: 10.18500/0869-6632-2021-29-2-220-239.
4. Didovyk A, Kanakov OI, Ivanchenko MV, Hasty J, Huerta R, Tsimring L. Distributed classifier based on genetically engineered bacterial cell cultures. ACS Synthetic Biology. 2015;4(1):72–82. DOI: 10.1021/sb500235p.
5. Kanakov O, Kotelnikov R, Alsaedi A, Tsimring L, Huerta R, Zaikin A, Ivanchenko M. Multi-input distributed classifiers for synthetic genetic circuits. PLoS ONE. 2015;10(5): e0125144. DOI: 10.1371/journal.pone.0125144.
6. Goh BS. Global stability in many-species systems. The American Naturalist. 1977; 111(977):135–143. DOI: 10.1086/283144.
7. Gnedenko BV. A Course in Probability Theory. Moscow: LENAND; 2022. 456 p. (in Russian).