INTERNATIONAL ATOMIC ENERGY AGENCY
UNITED NATIONS EDUCATIONAL, SCIENTIFIC AND CULTURAL ORGANIZATION
INTERNATIONAL CENTRE FOR THEORETICAL PHYSICS
I.C.T.P., P.O. BOX 586, 34100 TRIESTE, ITALY, CABLE: CENTRATOM TRIESTE

H4.SMR/709-7

Second Workshop on Non-Linear Dynamics
and Earthquake Prediction

22 November - 10 December 1993

Pattern Recognition

V. Keilis-Borok, A. Soloviev
Russian Academy of Sciences
International Institute of Earthquake Prediction
Theory and Mathematical Geophysics
Moscow 113556
Russian Federation

PATTERN RECOGNITION

V.KEILIS-BOROK

A.SOLOVIEV

EXAMPLES OF PATTERN RECOGNITION PROBLEMS

Let it is necessary to decide about any object, phenomenon or process has it some feature or not. This problem could be solved by construction a model on the basis of mechanical, physical, chemical or other scientific laws which could explain the connection between the available information and the feature under consideration. But in many cases the construction of a such model is difficult or practically impossible. In this case it is natural to apply pattern recognition methods.

Examples of some problems for application of pattern recognition methods are the following.

1. *Recognition of earthquake-prone areas* (for example Gelfand et al., 1976).

The objects are the selected structures of a region. The possibility for a strong (with magnitude $M \geq M_0$, where M_0 is some threshold) earthquake to occur near the object is the feature under consideration. The available information is the topographical, geological, geomorphological an geophysical data.

The problem as the pattern recognition one is to divide the selected structures into two classes:

-- structures where earthquakes with $M \geq M_0$ may occur;

-- structures where earthquakes with $M \geq M_0$ may not occur.

2. *Intermediate-term prediction of earthquakes* (for example Keilis-Borok and Rotwain, 1990).

The objects are moments of time. The occurrence of a strong (with magnitude $M \geq M_0$, where M_0 is some threshold) earthquake in the region within the period $(t, t + \tau)$ where τ is a given

constant is the feature under consideration for the moment t. The available information is the values of functions on seismic flow calculated for the moment t.

The problem as the pattern recognition one is to divide the moments of time into two classes:

-- moments for which there is (or will be) a strong earthquake in the region within the period $(t, t + \tau)$;

-- moments for which there are (or will be) not strong earthquakes in the region within the period $(t, t + \tau)$.

2. *Recognition of strata filled with oil.*

The strata encountered by a borehole are the objects. The filling of the strata with oil is the feature under consideration. The geological and geophysical data about the strata are the available information.

The problem as the pattern recognition one is to divide the strata into two classes:

-- strata which contain oil;

-- strata which contain water.

3. *Medical diagnostics.*

The objects are examined people. The feature under consideration is a specific disease. The data obtained through medical examination of people are the available information.

The problem as the pattern recognition one is to divide examined people into two classes:

-- people who have a specific disease;

-- people who do not have it.

GENERAL FORMULATION OF THE PATTERN RECOGNITION PROBLEM

Generalizing the three above examples one may formulate the problem of pattern recognition abstractly as follows:

The set $W = \{ \mathbf{w}^i \}$ is considered, where objects $\mathbf{w}^i = (w_1^i,\ w_2^i,\ \ldots\ ,\ w_m^i)$, $i = 1,\ 2,\ \ldots$ are vectors with real (integer, binary) components. Below these components will be called functions.

The problem is to divide the set W into two or more subsets which differ in certain feature or according to clustering themselves.

There are two kinds of pattern recognition problems and methods:

-- classification without learning;

-- classification with learning.

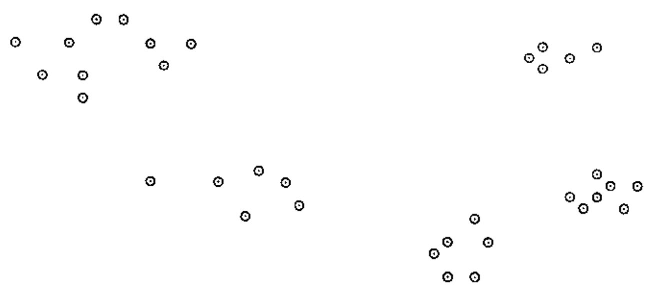

FIGURE 1. Clustering of objects in two-dimensional space.

CLASSIFICATION WITHOUT LEARNING
(CLUSTER ANALYSIS)

The set W is divided into groups (clusters, see Figure 1) on the base of some measure in the m-dimensional space $w_1, w_2, \ldots, w_m$.

Denote $\rho(\mathbf{w},\mathbf{v})$ a distance between two m-dimensional vectors $\mathbf{w} = (w_1,\ w_2,\ \ldots\ ,\ w_m)$ and $\mathbf{v} = (v_1,\ v_2,\ \ldots\ ,\ v_m)$.

To define classification and to estimate at the same time its quality the special function is introduced. The best classification gives the extremum of this function.

Examples of functions.

Let W is a finite set.

$$1.\ J_1 = \frac{(K-1)\sum_{k=1}^{K} \rho_k}{2\sum_{k=1}^{K-1}\ \sum_{j=k+1}^{K} \rho_{\ j}} \longrightarrow \min$$

$$2.\ J_2 = \frac{1}{K}\left(\sum_{k=1}^{K} \rho_k - \frac{2}{K-1}\sum_{k=1}^{K-1}\ \sum_{j=k+1}^{K} \rho_{kj} \right) \longrightarrow \min$$

Here K is the number of groups,

$$\rho_k = \frac{2}{m_k(m_k - 1)} \sum_{i=1}^{m_k - 1}\ \sum_{s=i+1}^{m_k} \rho(\mathbf{w}^i, \mathbf{v}^s),$$

$$\rho_{kj} = \frac{1}{m_k m_j} \sum_{i=1}^{m_k} \sum_{s=1}^{m_j} \rho(\mathbf{w}^i, \mathbf{v}^s),$$

m_k, m_j are the numbers of objects in the k-th and j-th groups respectively,

$\mathbf{w}^1,\ \mathbf{w}^2,\ \ldots\ ,\ \mathbf{w}^{m_k}$ are the objects of the k-th group,

$\mathbf{v}^1,\ \mathbf{v}^2,\ \ldots\ ,\ \mathbf{v}^{m_j}$ are the objects of the j-th group.

After the groups are formed the next problem can be formulated: to find common feature of objects which belong to the same group.

As a rule the set W is divided into two classes, say D and N.

The a priori examples of objects of each class are given. They are called the learning set:

$$W_0 \subset W \,,$$

$$W_0 = D_0 \sqcup N_0 \,.$$

Here D_0 is the learning set (the a priori examples) of objects belonging to class D, N_0 is the learning set of objects belonging to class N.

The result of the pattern recognition is twofold:

-- The rule of recognition; it allows to recognize which class an object belongs to knowing the vector $\mathbf{w}^i$ describing this object.

-- The actual division of objects into separate classes according to this rule:

$$W = D \sqcup N \,.$$

In some cases there are objects with undefined classification, so

$$W = (D \sqcup N) \sqcup U \,.$$

Analysis of the obtained rule of recognition may give information about the connection between the feature which differs the classes D and N on one hand and description of objects (functions of vectors $\mathbf{w}^i$) on another.

For example, analysis of the rule for recognition of earthquake-prone areas gave an insight of the role of fluids in the origin of the earthquakes.

1. *STATISTICAL*

It is assumed that distribution laws are different for vectors from classes D and N (see Figure 2).

Samples D_0 and N_0 are used to define the parameters of these laws.

The recognition rule is based on calculation of an estimation of conditional probabilities for each object $\mathbf{w}^i$ to belong to class D (P_D^i) and N (P_N^i). Classification of the objects according to these probabilities is performed as follows:

$$\mathbf{w}^i \in D, \text{ if } P_D^i - P_N^i \ge \varepsilon,$$

$$\mathbf{w}^i \in N, \text{ if } P_D^i - P_N^i < -\varepsilon,$$

$$\mathbf{w}^i \in U, \text{ if } -\varepsilon \le P_D^i - P_N^i < \varepsilon,$$

where $\varepsilon \ge 0$ is a given constant.

An example of a statistical algorithm is Bayes algorithm.

__BAYES ALGORITHM__

According to Bayes formula

$$P(\mathbf{w}=\mathbf{w}^i|\mathbf{w}\in D)\;P(\mathbf{w}\in D) \;=\; P(\mathbf{w}\in D|\mathbf{w}=\mathbf{w}^i)\;P(\mathbf{w}=\mathbf{w}^i)$$

and

$$P_D^i = P(\mathbf{w}\in D|\mathbf{w}=\mathbf{w}^i) = \frac{P(\mathbf{w}=\mathbf{w}^i|\mathbf{w}\in D)\;P(\mathbf{w}\in D)}{P(\mathbf{w}=\mathbf{w}^i)} \;.$$

Similarly

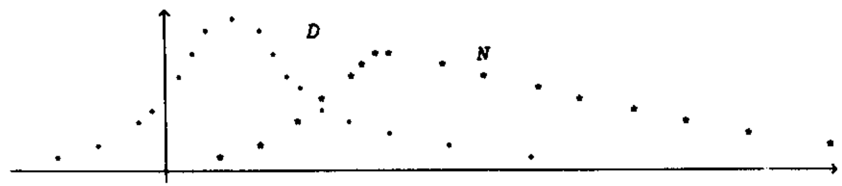

FIGURE 2. Different distribution laws for classes D and N.

$$P_N^i = P(w \in N \mid w = w^i) = \frac{P(w = w^i \mid w \in N)\; P(w \in N)}{P(w = w^i)} \quad .$$

Estimations of probabilities in the right side of these relations are given by following approximate formulae in which the samples D_0 and N_0 are used:

$$P(w = w^i \mid w \in D) \approx P(w = w^i \mid w \in D_0),$$

$$P(w = w^i \mid w \in N) \approx P(w = w^i \mid w \in N_0),$$

$$P(w = w^i) \approx P(w = w^i \mid w \in D_0) P(w \in D) + P(w = w^i \mid w \in N_0) P(w \in N).$$

Probability $P(w \in D)$ is a parameter of the algorithm and has to be given, $P(w \in N) = 1 - P(w \in D)$.

NOTE: The sign of the difference $P_D^i - P_N^i$ does not depend on value of $P(w = w^i)$.

2. *GEOMETRICAL*

In these algorithms surfaces in the space $w_1, w_2, \ldots, w_m$ are constructed to separate classes D and N (see Figure 3).

An example of a geometrical algorithm is the algorithm Hyperplane.

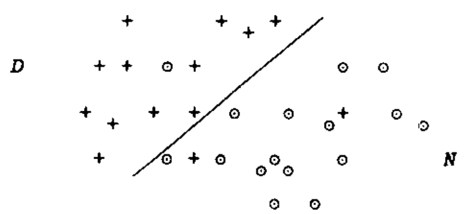

FIGURE 3. Separation of objects from classes D and N in two-dimensional space by the straight line.

ALGORITHM HYPERPLANE

In the space $w_1, w_2, \ldots, w_m$ the hyperplane

$P(w) = a_0 + a_1 w_1 + a_2 w_2 + \ldots + a_m w_m$ is constructed.

This hyperplane has to separate the sets D_0 and N_0 by the best way. It means that some function has to have extremum value.

The example of the function is

$$J(a_0, a_1, \ldots, a_m) = \sum_{i=1}^{n_1} P(w^i) - \sum_{i=1}^{n_2} P(v^i) \longrightarrow \max.$$

Here $w^1, w^2, \ldots, w^{n_1}$ are objects of D_0 ,

$v^1, v^2, \ldots, v^{n_2}$ are objects of N_0 .

The recognition rule as follows:

$w^i \in D$, if $P(w^i) \geq \varepsilon$,

$w^i \in N$, if $P(w^i) < -\varepsilon$,

$w^i \in U$, if $-\varepsilon \leq P(w^i) < \varepsilon$.

Here $\varepsilon \geq 0$ is a given constant.

3. *LOGICAL*

In these algorithms characteristic traits of classes D and N are searched using the sets D_0 and N_0 . Traits are boolean functions on $w_1, w_2, \ldots, w_m$. The object w^i has the trait if the value of the corresponding function calculated for it is true and does not have the trait if it is false. A trait is a characteristic trait of the class D if objects of the set D_0 have this trait more often than objects of the set N_0. A trait is a characteristic trait of the class N if objects of the set N_0 have this trait more often than objects of the set D_0.

Using the searched characteristic traits the recognition rule is formulated as follows:

$w^i \in D$, if $n_D^i - n_N^i \geq \Delta + \varepsilon$,

$w^i \in N$, if $n_D^i - n_N^i < \Delta - \varepsilon$,

$w^i \in U$, if $-\varepsilon \leq n_D^i - n_N^i - \Delta < \varepsilon$.

Here n_D^i and n_N^i are the numbers of characteristic traits of classes D and N which the object w^i has, Δ and $\varepsilon \geq 0$ are given constants.

Logical algorithms are useful to apply in cases then the numbers of objects in sets D_0 and N_0 are small.

As a rule logical algorithms are applied to vectors with binary components. An example of logical algorithm is the algorithm CORA-3. It is applied to geophysical problems in particular to the problems of recognition of earthquake-prone areas and intermediate-term prediction of earthquake. The detailed description of this algorithm can be found in Gelfand et al. (1976) and will be given below.

DISCRETIZATION AND CODING

As it was mentioned above some pattern recognition algorithms (for example CORA-3) are applied only to vectors with binary components. In the case when the set W initially consists of vectors with real components (functions) the discretization and coding are necessary.

After discretization the data become robust. For example if a range of some function is divided into three parts only three gradations for this function ("small", "medium", "large") are used after the discretization instead of its exact value. Do not regret the loss of information. This makes results of recognition stable to variations of data.

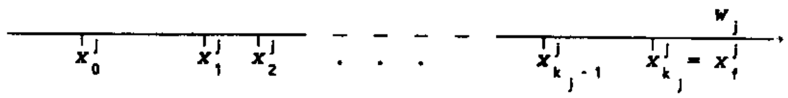

FIGURE 4. Discretization of function w_j.

Let us consider some function w_j of vectors (objects) w which form the set W. Let the range of the function variation is limited with the numbers x_0^j and x_f^j ($x_0^j < x_f^j$). The procedure of *discretization* for the function w_j consists of dividing the range of its variation into k_j intervals by thresholds of discretization (Figure 4):

$$x_1^j,\ x_2^j,\ \ldots,\ x_{k_j-1}^j \quad (x_0^j < x_1^j < x_2^j < \ldots < x_{k_j-1}^j < x_f^j).$$

Assume that the value w_j^i of the j-th function of the i-th object belongs to the s-th interval, if $x_{s-1}^j < w_j^i \leq x_s^j$, where $x_{k_j+1}^j = x_f^j$. In a process of discretization we substitute the exact value of the function by the interval which contains this value.

Usually we divide the range of function variation into two intervals ("small" and "large" values) or into three intervals ("small", "medium" and "large" values).

Thresholds of discretization can be introduce manually on the basis of various considerations for the nature of the given function.

The other way to define the thresholds is to compute them so as to make the numbers of objects within each interval (x_{s-1}^j, x_s^j) $s = 1, 2, \ldots, k_j$ are roughly equal to each other. In this case only the number of intervals k_j has to be defined. Then the thresholds of discretization may be calculated by a computer running a special algorithm. All objects together or only objects of D_0 and N_0 can be considered. This type of discretization is called here and below as *objective or automatic*.

Our purpose is to find such intervals where values of the function w_j for objects from one class occur more often than for objects from another class.

How *informative* is the function w_j in a given discretization can be characterized as follows.

1. Let us compute for each interval (x^j_{s-1}, x^j_s) the numbers P^D_s and P^N_s $(s = 1, 2, \ldots, k_j)$ which gives the percent of objects of D_0 and N_0 respectively, for which value of the function w_j falls within the s-th interval.

Let us denote $P_{max} = \max\limits_{1 \le s \le k_j} |P^D_s - P^N_s|$.

In other words P^D_s and P^N_s are empirical histograms of the value of the function w_j for moments D_0 and N_0. P_{max} is the maximal difference of these histograms.

The larger is P_{max}, the more informative is the function w_j. Functions for which $P_{max} < 20\%$ usually are excluded.

2. Let k = 3. Let us denote:

$$M_D = \frac{|P^D_2 - P^D_1| + |P^D_3 - P^D_2|}{|P^D_3 - P^D_1|} ,$$

$$M_N = \frac{|P^N_2 - P^N_1| + |P^N_3 - P^N_2|}{|P^N_3 - P^N_1|} .$$

If P^D_s changes monotonously with s, $M_D = 1$; the larger is M_D, more jerky is P^D_s. This is clear from the Figure 5. Similar statements are true for M_N, P^N_s.

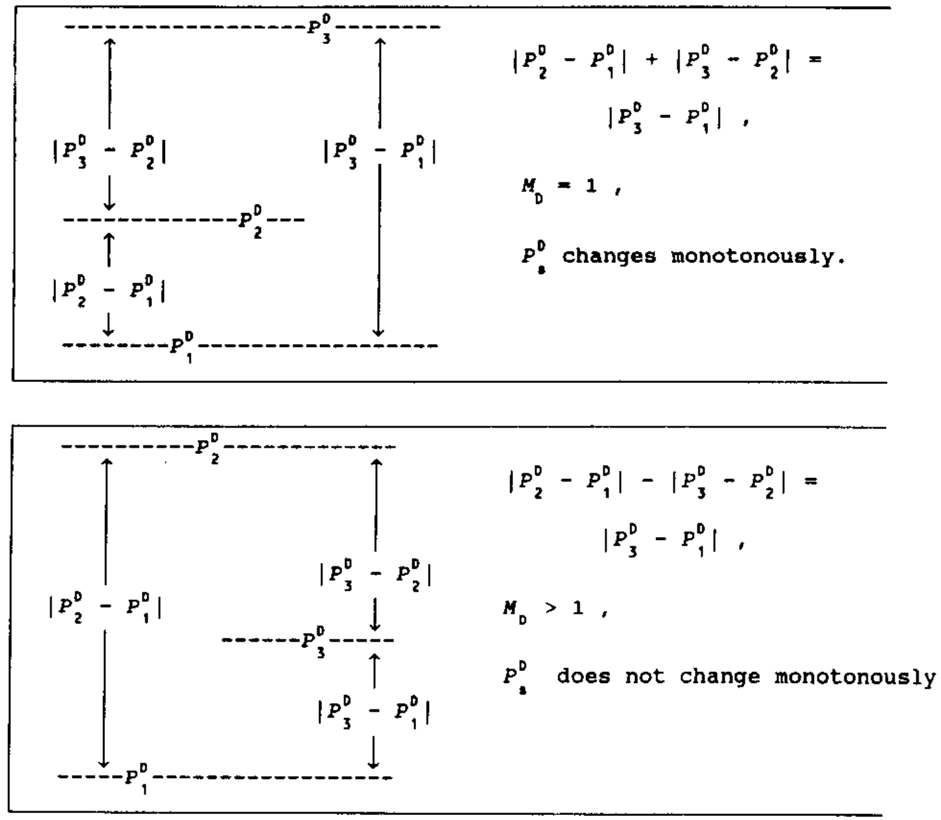

FIGURE 5. Monotonous and non-monotonous changing of P^D_s.

The smaller are M_D and M_N, the better is the function w_j. Functions with both M_D, $M_N \ge 3$ usually are excluded.

3. Samples D_0 and N_0 are often marginally small, so that their observed difference may be random. Therefore the relation between functions P^D_s and P^N_s after discretization should be not absurd according to the problem under consideration, though they may be unexpected indeed.

CODING

With discretization thresholds defined, a procedure of coding of vectors w^i into the form of binary vectors in undertaken. For coding only the functions selected on the stage of discretization are considered. On the stage of coding l_j components of binary vectors are defined for the function w_j. Number l_j depends on the number of thresholds as well as on the type of coding procedure applied to the function w_j.

For coding the following two procedures are used. In the case of I ("impulse") procedure $l_j = k_j$, i.e. the number of binary vector components allocated for the coding of the function w_j is equal to the number of intervals into which the range of its variation is divided after discretization.

Let us denote as $\omega_1, \omega_2, \dots, \omega_{l_j}$ the values of binary vector components which code the function w_j. If the value w_j^i of the function w_j for the i-th object falls within the s-th interval of its discretization, i.e. $x^j_{s-1} < w^i_j \le x^j_s$, then we set

$$\omega_1 = 0,\ \omega_2 = 0,\ \dots,\ \omega_{s-1} = 0,\ \omega_s = 1,\ \omega_{s+1} = 0,\ \dots,\ \omega_{l_j} = 0,$$

In the case of S ("stair") procedure $l_j = k_j - 1$, i.e. the number of binary vector components, allocated for the coding of a function, is equal to the number of thresholds of discretization. If the value w_j^i for the i-th object falls within the s-th interval of its discretization, then we set

$$\omega_1 = 0,\ \omega_2 = 0,\ \dots,\ \omega_{s-1} = 0,\ \omega_s = 1,\ \omega_{s+1} = 1,\ \dots,\ \omega_{l_j} = 1.$$

Below the case when the codes of the function w_j are constructed for $k_j = 2$ is considered.

If the value w_j^i belongs to the first interval ($x_0^j < w_j^i \le x_1^j$) I-coding has the form: 100. For the same value w_j^i S-coding has the form: 11. For the second interval ($x_1^j < w_j^i \le x_2^j$) the codes are 010 (I-method) and 01 (S-method). For the third interval ($x_2^j < w_j^i \le x_3^j$) they are 001 and 00 respectively.

Discretization and coding procedures transform the set of vectors $W = \{ w^i \}$, $i = 1, 2, \dots, n$, which correspond to all objects into a set of vectors with l binary components. Here $l = \sum' l_j$, where summation is implemented only over the functions left after discretization.

Thus, discretization and coding transform the initial problem in the form of the classification within the finite set of l-dimensional vectors with binary components. These vectors will be called *objects of recognition*.

We shall denote by D_0 and N_0 the sets of objects of recognition which correspond to the moments D_0 and N_0 and by W the set of all objects of recognition.

ALGORITHM CORA-3

Algorithm *CORA-3* operates in two steps:

- *selection of characteristic traits (learning);*
- *voting.*

At *the first step* the sets of characteristic traits for classes D and N are constructed on the basis of sets D_0 and N_0.

Matrix

$$A = \begin{Vmatrix} i_1 & i_2 & i_3 \\ \delta_1 & \delta_2 & \delta_3 \end{Vmatrix}$$

is called by a trait. Here i_1, i_2, i_3 are the natural numbers such as $1 \le i_1 \le i_2 \le i_3 \le l$ and δ_1, δ_2, δ_3 are equal to 0 or to 1.

We say that the object which is the binary vector $\omega^i = (\omega_1^i, \omega_2^i, \ldots, \omega_l^i)$ has the trait A if $\omega_{i_1}^i = \delta_1$, $\omega_{i_2}^i = \delta_2$, $\omega_{i_3}^i = \delta_3$.

Let $W' \subseteq W$. We shall denote by $K(W',A)$ the number of objects $\omega^i \in W'$ which have the trait A.

The algorithm has four free parameters k_1, $\bar{k}_1$, k_2, $\bar{k}_2$ which can take integer non-negative values. While the value of free parameters are defined, the notion of characteristic traits is introduced.

The trait A is a characteristic trait of class D if

$K(D_0,A) \ge k_1$, $K(N_0,A) \le \bar{k}_1$.

The trait A is a characteristic trait of class N if

$K(N_0,A) \ge k_2$, $K(D_0,A) \le \bar{k}_2$.

Parameters k_1 and k_2 are called by selection thresholds of characteristic traits for classes D and N respectively. Parameters $\bar{k}_1$ and $\bar{k}_2$ are called by the contradiction thresholds for characteristic traits of classes D and N.

The number of characteristic traits of each class may be large enough. Among them groups of traits which occur on the same learning objects of their class may be. There is no reason to include all traits from a such group in the final list.

Let $\Omega(A)$ be a subset of the set W consisting of the objects which have the trait A. Let, also, A_1 and A_2 be two characteristic traits of class D. We say that the trait A_1 is *weaker* than the trait A_2 (or A_2 is *stronger* than A_1), if

$\Omega(A_1) \cap D_0 \subset \Omega(A_2) \cap D_0$, $(\Omega(A_2) \cap D_0) \setminus (\Omega(A_1) \cap D_0) \neq \emptyset$.

In other words it means that all objects from D_0, having A_1, possess

also A_2. At the same time there is at least one object from D_0, which, having the trait A_2, does not have A_1.

A similar definition we introduce for characteristic traits of class N. Let A_1 and A_2 be two characteristic traits of class N. Then the trait A_1 ,is *weaker* then the trait A_2 (or A_2 is *stronger* than A_1), if

$\Omega(A_1) \cap N_0 \subset \Omega(A_2) \cap N_0$, $(\Omega(A_2) \cap N_0) \setminus (\Omega(A_1) \cap N_0) \neq 0$.

If two characteristic traits A_1 and A_2 of class D are both found in the same objects of the set D_0 i.e.

$\Omega(A_1) \cap D_0 = \Omega(A_2) \cap D_0$,

we call A_1 and A_2 as equivalent.

Similarly, characteristics traits A_1 and A_2 of class N are called equivalent if

$\Omega(A_1) \cap N_0 = \Omega(A_2) \cap N_0$.

The lists of characteristic traits of classes being formed as a result of the learning step by definition include no any trait which is weaker than any trait in the list of its class. Only one trait (selected first) is included from each group of equivalent ones to the final list.

Thus, the learning step results in the set of q_D characteristic traits of class D and the set of q_N of ones of the class N. These sets containing no weaker or equivalent traits in relation to any one from the same set.

The second step of the algorithm involves voting and classification. For every object $\omega^i \in W$ the number n_D^i of the characteristic traits of class D which the object has, the number n_N^i of ones of class N and the difference $\Delta_i = n_D^i - n_N^i$ are calculated.

Classification is performed by the following way.

Class D (the set D) is formed from the objects ω^i for which $\Delta_i \geq \Delta$.

The objects for which $\Delta_i < \Delta$ are included in *class N* (the set N).

Here Δ as k_1, $\bar{k}_1$, k_2 and $\bar{k}_2$ is a parameter of the algorithm.

CLUSTERS ALGORITHM

CLUSTERS algorithm is the modification of *CORA-3* algorithm (Gelfand et al., 1976). It is applied in the case when the set D_0 consists of S nonintersecting subsets (subclasses):

$$D_0 = D_0^1 \sqcup D_0^2 \sqcup \cdots \sqcup D_0^S$$

and it is known a priori that each subclass has at least one object of class D but some objects of the set D_0 may belong to class N.

At *the learning step* algorithm *CLUSTERS* differs from *CORA-3* in the following.

First, by definition a subclass has a trait if at least one object among those which belong to this subclass has this trait.

The trait A is a characteristic trait of class D if

$$K^S(D_0,A) \geq k_1, \; K(N_0,A) \leq \bar{k}_1.$$

Here $K^S(D_0,A)$ is the number of subclasses which have the trait A.

Second, the definition of the weaker and equivalent traits for characteristic traits of class D changes to the following.

A characteristic trait A_1 of class D is weaker than a characteristic trait A_2 of this class if any subclass having the trait A_1 has also A_2, and there is at least one subclass which has the trait A_2 but does not have the trait A_1. Traits A_1 and A_2 are equivalent if they are found in the same subclasses.

CLUSTERS algorithm forms the sets of characteristic traits of classes D and N like *CORA-3*.

The step of voting and classification is the same as in *CORA-3* algorithm.

HAMMING ALGORITHM

Another algorithm applied to geophysical problems is *HAMMING* algorithm (Gvishiani and Kosobokov, 1981). There are also other possible applications of this algorithm (for example Keilis-Borok and Lichtman, 1981).

The application of this algorithm also consists in two steps.

At the first step (learning) for each component ω_k (k = 1, 2, ..., l) of binary vectors the following values are calculated

$q_D(k|0)$ – the number of objects of the set D_0 which have $\omega_k = 0$,

$q_D(k|1)$ – the number of objects of the set D_0 which have $\omega_k = 1$,

$q_N(k|0)$ – the number of objects of the set N_0 which have $\omega_k = 0$,

$q_N(k|1)$ – the number of objects of the set N_0 which have $\omega_k = 1$.

Then the relative number of objects which have this component equal to 1 among the set D_0

$$\alpha_D(k|1) = \frac{q_D(k|1)}{q_D(k|0) + q_D(k|1)}$$

and among the set N_0

$$\alpha_N(k|1) = \frac{q_N(k|1)}{q_N(k|0) + q_N(k|1)}$$

are calculated.

Then *the kernel of class D* $\mathbf{K} = (\kappa_1, \kappa_2, \ldots, \kappa_l)$ is defined as follows

$$\kappa_k = \begin{cases} 1, & \text{if } \alpha_D(k|1) \ge \alpha_N(k|1), \\ 0, & \text{if } \alpha_D(k|1) < \alpha_N(k|1). \end{cases}$$

Values of components of the kernel of class D are more "typical" for the objects of the set D_0 than for the objects of the set N_0. The calculation of the kernel $\mathbf{K}$ completes the first step of applying the algorithm.

NOTE: It may be more reliable to eliminate the components for which $|\alpha_D(k|1) - \alpha_N(k|1)| < \varepsilon$, where ε is a small positive constant.

The voting procedure and actual classification are carried out at *the second stage*. The voting consists of calculating for each object a Hamming's distance ρ_i to the kernel of class D. It is calculated by the formula:

$$\rho_i = \sum_{k=1}^{l} |\omega_k^i - \kappa_k| \ .$$

Classification is performed by the following way.

Class D (the set *D*) is formed from the objects ω^i for which $\rho_i \le R$.

The objects for which $\rho_i > R$ are included in *class N* (the set *N*).

Here *R* is a parameter of the algorithm.

Hamming's distance can calculated with including of the weights of components

$$\rho_i = \sum_{k=1}^{l} \xi_k |\omega_k^i - \kappa_k| \ .$$

Here $\xi_k > 0$ are the weights associated to the components of binary vectors. Weights can be assigned intuitively or computed by the formula:

$$\xi_k = \frac{|\alpha_D(k|1) - \alpha_N(k|1)|}{\max\limits_k |\alpha_D(k|1) - \alpha_N(k|1)|} ,$$

where maximum is taken among the components used in the given run of the algorithm.

TESTS FOR ESTIMATION OF RELIABILITY OF RESULTS

These tests are necessary to be sure in the obtained results. It is especially important in the case of small samples D_0 and N_0. The tests illustrate - how reliable are the results of the pattern recognition. However they do not provide a proof in the strict statistical sense if the learning material is small.

The examples of some tests are listed below.

1. To save the part of objects from W_0 for recognition only, not using it in learning.
2. To check the conditions: $D_0 \subset D$, $N_0 \subset N$.

NOTE: Sometimes this conditions are not valid because the sets D_0 and N_0 are not "clear" enough. For example in the case of recognition of earthquake-prone areas objects of D_0 are structures where epicenters of earthquakes with $M \ge M_0$ are known and objects of N_0 are structures where epicenters of such earthquakes are not known. Objects of N_0 may belong to the class *D*, because in some areas earthquakes with $M \ge M_0$ may be possible, though yet unknown. Objects of D_0 may belong to the class *N* due to the errors in catalog (in epicenters and/or magnitude).

NUMERICAL TESTS. These tests include some variation of the objects, used components of vectors, numerical parameters etc. The test is positive if the results of recognition are stable to these variations.

3. Elimination of objects from D_0 and N_0 - one at a time. Formal criteria of stability - small value of the ratio $\frac{m_D}{|D_0|}$ or $\frac{m_D + m_N}{|D_0| + |N_0|}$. Here m_D and m_N show how many objects of D_0 and N_0 respectively change classification after they were eliminated from learning.

4. Learning on the subsets of the obtained sets D and N.

5. Change the set of used components of binary vectors. In particular elimination of each used component in turn.

Since the danger of selfdeception is not completely eliminated by these tests the design and implementation of new tests should be pursued.

RECOGNITION OF EARTHQUAKE-PRONE AREAS

The problem of recognition of places in the western Alps where earthquakes with $M \geq 5.0$ may occur (Cisternas et al., 1985) will be briefly considered below.

The objects are the intersections of the morphostructural lineaments obtained as the result of the morphostructural zoning of the western Alps. The scheme of the morphostructural zoning of the western Alps and the objects are shown in Figure 6. The total number of objects in the set W is 62. The problem is to classify these objects into two classes: objects where earthquakes with $M \geq 5.0$ may occur (class D) and objects where earthquakes with $M \geq 5.0$ may not occur (class N).

Table 1 contains the list of functions which describe the objects. Values of these functions are components of vectors w^i.

The epicenters of earthquakes with $M \geq 5.0$ or $I_0 \geq 7$ (I_0 is

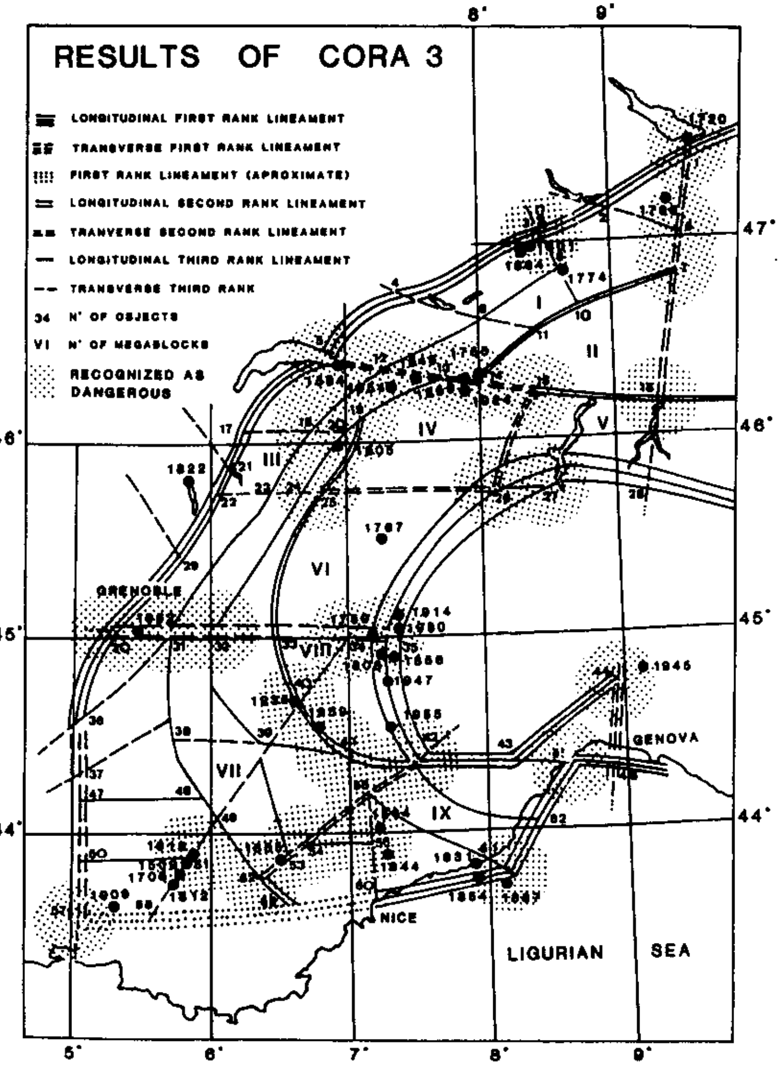

FIGURE 6. The morphostructural scheme of the western Alps and the result of recognition.

TABLE 1. Functions of objects of the western Alps

Characteristics	Discretization thresholds	
	first	second
Maximum altitude H_{max}, m	2686	4807
Minimum altitude H_{min}, m	325	-
Altitude in the lineament intersection point H_0, m	490	900
Distance between points H_{min} and H_{max} l, km	32	42
$\Delta H = H_{max} - H_{min}$, m	2500	-
Altitude gradient $\Delta H/l$, m/km	51	91
Combinations of relief types (yes, no) mountain slope/mountain slope (m/m) mountain slope/plain (m/p) mountain slope/piedmont/plain (m/pd/p) mountain slope/piedmont (m/pd) piedmont/plain (pd/p)		
The portion of the soft (quaternary) deposits Q, %	10	-
The highest rank of the lineament in the intersection R_h	1	2
Number of lineaments forming the intersection n_l	2	-
Number of lineaments in the circle of radius 25 km N_l (3 thresholds)	2	3, 4
Distance to the nearest intersection ρ_{int}, km	20	31
Distance to the nearest first rank lineament ρ_1, km	0	32
Distance to the nearest second rank lineament ρ_2, km	0	40
Maximum value of Bouguer anomaly B_{max}, mgl	-82	-8
Minimum value of Bouguer anomaly B_{min}, mgl	-145	-85
$\Delta B = B_{max} - B_{min}$, mgl	45	65
$\bar{B} = (B_{max} + B_{min})/2$, mgl	-110	-44
$HB = 0.1H_{max}[m] + B_{min}[mgl]$	153	-
Number of Bouguer anomaly isolines N_B	4	7
Number of closed Bouguer anomaly isolines N_{BC}	1	-
Minimum distance between two Bouguer anomaly isolines with values divided by 10 mgl $(\nabla B)^{-1}$, km	2	3

maximum macroseismic intensity) are shown in Figure 6 by dark circles with origin years. The learning set D_0 of class D consists of 14 objects near which instrumental epicenters of earthquakes with $M \geq 5.0$ are known (earthquakes in the 1900-1980 period): 3, 12, 13, 14, 20, 30, 31, 35, 40, 41, 42, 44, 51, 57. The objects (1, 5, 6, 8, 53, 55, 56, 58, 60, 61) which have historic earthquake epicenters (events prior to 1900) with $I_0 \geq 7$ were not included both in D_0 and N_0 learning sets. These objects and objects 18, 19 which are located near the epicenter of 1905 were voted only. The remaining 36 objects constituted the learning set N_0 of class N.

The following functions (Table 1) ought to be considered as the most informative: maximum altitude H_{max}, altitude gradient $\Delta H/l$, the portion of the soft (quaternary) deposits Q, the highest rank of the lineament in the intersection R_h, distance to the nearest second rank lineament ρ_2. For all these functions $P_{max} > 20\%$.

Coding of all the functions, except the combinations of relief types (Table 1), was performed by S-method with the thresholds given in Table 1. Their values have been obtained by the method of objective discretization. Functions describing relief pattern need no additional discretization and coding since they take values 1 (yes) or 0 (no).

Algorithm *CORA-3* was applied with the following values of its parameters: $k_1 = 3$, $\bar{k}_1 = 2$, $k_2 = 11$, $\bar{k}_2 = 1$, and $\Delta = 0$. The selected sets of characteristic traits of classes D and N (D- and N-traits) are given in Table 2. The traits are given in the table as conjunctions of inequalities in the values of object description characteristics. The obtained classification of objects is shown in

TABLE 4. Characteristic traits selected by algorithm *CORA-3* for recognition of objects of the western Alps

Nos.	Q, %	n_i	N_i	ρ_1, km	ρ_2, km	ΔB, mgl	$(\nabla B)^{-1}$ km
D-traits							
1				≤32		≤65	≤2
2				>0		≤65	≤2
3				≤32	0	≤65	
4			>3		0	≤65	
5			>4			>45	≤3
6					>0; ≤40	>45	
7		2		>32		>45	
8		2		>32			
9		>2	≤3				≤3
10	>10		>3		≤40		≤2
N-traits							
1						≤45	>2
2					>0	≤45	
3		2				≤45	
4					>40	≤45	
5					>40		>2
6		2			>40		
7		2	≤3		>0		
8		2		0			

Figure 6. 34 objects are attributed to class *D* and 28 objects are attributed to class *N*. All objects of the learning set D_0 are classified as objects of class *D*. The number of objects of N_0 classified as objects of class *D* is roughly 30% of the their total number in N_0.

INTERMEDIATE-TERM PREDICTION OF EARTHQUAKES

The pattern recognition methods were used to develop the intermediate-term earthquake prediction algorithm **CN** (Keilis-Borok and Rotwain, 1990). This algorithm was initially applied to California-Nevada region.

The objects are moments of the time. These moments are described by the functions defined in the lecture "Functions on Earthquake Flow". The selection of the moments and the forming of the learning sets D_0 and N_0 are described below.

If the earthquake catalog of some region covers the time from t_0 to T_k the three types of time periods can be defined between t_0 and T_k:

- periods which precede strong earthquakes - periods D;
- periods which follow strong earthquakes - periods X;
- periods which are not connected with strong earthquakes - periods N.

The formal definition can be formulated as follows.

Let $t_1, t_2, \ldots, t_m$ $(t_0 < t_1 < t_2 < \ldots < t_m < T_k)$ be the moments of strong earthquakes of the region under consideration. Here strong earthquakes are the main shocks with magnitude $M \geq M_0$, where M_0 is a given threshold.

Periods D are time intervals from $t_i - \Delta t_D$ to t_i.

Periods X are time intervals from t_i to $t_i + \Delta t_X$ out of periods D.

Periods N are intervals from t_0 to T_k which remain after exclusion of all periods D and X.

Here $i = 1, 2, \ldots, m$; Δt_D and Δt_X are given constants.

Example of periods D, X and N is shown in Figure 7. The moments t_i, t_{i+1}, t_{i+2} and t_{i+3} in this figure are the moments of four strong earthquakes.

Moments of time are considered as objects of recognition. For time period from t_0 to T_k three types of moments are defined: D_0, N_0 and X.

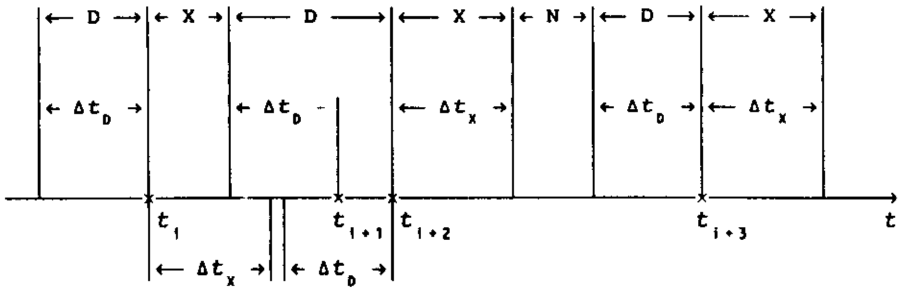

FIGURE 7. Periods D, N and X.

Moments D_0 (the set D_0) are the moments before strong earthquakes. For each strong earthquake with origin time t_i the interval from $t_i - \Delta t_D$ to $t_i - \delta t$ is divided into k equal parts of the length $\Delta t_3 = \Delta t_1 / k$, where $\Delta t_1 = \Delta t_D - \delta t$. Here $\delta \geq 0$ and k are selected so to have the relationship $\delta t \ll \Delta t_3$.

Moments D_0 are the moments

$$t_i^j = t_i - \Delta t_D + j \Delta t_3,$$

where $j = 0, 1, 2, \ldots, k$. The moments D_0 which are earlier than the origin time t_{i-1} of the preceding strong earthquake are eliminated (see Figure 8B).

Moments N are selected within periods N with equal steps, unless there is not specific reason to do otherwise.

Moments N_0 (the set N_0) are selected from moments N to be regularly distributed among them. The number of moments N_0 is usually selected about the same as the number of strong earthquakes in the region.

Moments X are selected from periods X with step Δt_3.

Among the moments D_0 *subclasses* are formed. One subclass includes moments D_0 which precede the same strong earthquake.

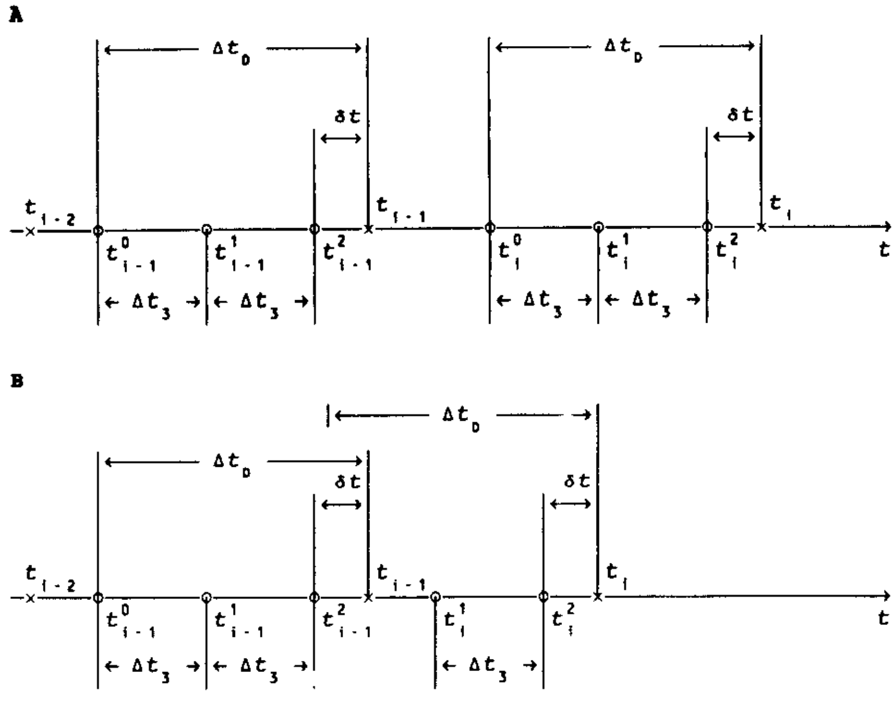

FIGURE 8. Moments D_0 ($k = 2$, the moments D_0 are marked by $\odot$).

includes moments D_0 which precede the same strong earthquake.

Let t_{i-1} and t_i are origin times of two consecutive strong earthquakes. If $t_i - t_{i-1} > \Delta t_0$ then the subclass connected with the i-th strong earthquake consists of moments D_0

$$t_i^j = t_i - \Delta t_D + j \Delta t_3,$$

where $j = 0, 1, 2, \ldots, k$. If $t_i - t_{i-1} \leq \Delta t_D$ then only moments t_i^j which are after t_{i-1} ($t_i^j > t_{i-1}$) are included in the subclass.

In Figure 8A the subclass connected with the strong earthquake with the origin time t_{i-1} consists of three moments D_0: t_{i-1}^0, t_{i-1}^1 and t_{i-1}^2. The subclass connected with the strong earthquake with the origin time t_i also consists of three moments D_0: t_i^0, t_i^1 and t_i^2.

In Figure 8B the subclass connected with the strong earthquake with the origin time t_{i-1} consists of three moments D_0: t_{i-1}^0, t_{i-1}^1 and t_{i-1}^2 and the subclass connected with the strong earthquake with the origin time t_i consists of only two moments D_0: t_i^1 and t_i^2.

These moments were defined for the Southern California for the time period 1938-1984. The threshold magnitude for the strong earthquakes was $M_0 = 6.4$. Table 5 contains the thresholds for discretization of the functions on the earthquake flow calculated for these moments. The coding was performed by *S*-method with these thresholds.

The algorithm *CLUSTERS* was applied to obtain the characteristic traits of classes *D* and *N*. These traits are listed in Table 6. The parameters had the following values: $k_1 = 7$, $\bar{k}_1 = 2$, $k_2 = 10$, and $\bar{k}_2 = 4$. The moments defined for the Southern California are classified by using these traits with $\Delta = 5$. If a moment t is attributed to class *D* then this moment is considered to belong to a period of the time of increased probability (TIP) of a strong earthquake. Formally if t is attributed to class *D* then a TIP diagnosed from t to $t + \tau$ where τ is a given constant. For the

TABLE 5. Thresholds for discretization of functions

Function	Thresholds	
N2	0	-
K	-1	1
G	0.5	0.67
SIGMA	36	71
Smax	7.9	14.2
Zmax	4.1	4.6
N3	3	5
q	0	12
Bmax	12	24

Southern California $\tau = 1$ year was used.

TABLE 6. Characteristic traits of classes *D* and *N* obtained by *CLUSTERS* algorithm for the moments of the Southern California

Traits *D*	N2	K		G		SIGMA		Smax		Zmax		N3		q		Bmax	
1		0															0
2															0		
3												0		0		0	
4											0			0			
5		0										1				0	
6			1								0					0	
7		0									1					0	
8		0		0												0	
9									0		0						
10			1				0				0						
11		0	1								0						
12	0		1							0							
13			0						1								
14			0					0									
15			0			0											
16			0		1												

Traits *N*	N2	K		G		SIGMA		Smax		Zmax		N3		q		Bmax	
1									1							1	
2											1						1
3							1								1		1
4			1												1		1
5												0			1		1
6								1									1
7			1								1						1
8	1										1						1
9							1					0			1		
10								1							1		
11											1		0				
12								1				0					
13			1					1									
14			1			1											
15		1									1						
16			1		1				1								
17		1							1								
18		1					1										

REFERENCES

Cisternas,A., P.Godefroy, A.Gvishiani, A.I.Gorshkov, V.Kosobokov, M.Lambert, E.Ranzman, J.Sallantin, H.Soldano, A.Soloviev, and C.Weber, 1985. A dual approach to recognition of earthquake prone areas in the western Alps. *Annales Geophysicae*, **3**, 2: 249-270.

Gelfand,I.M., Sh.A.Guberman, V.I.Keilis-Borok, L.Knopoff, F.Press, I.Ya.Ranzman, I.M.Rotwain, and A.M.Sadovsky, 1976. Pattern recognition applied to earthquake epicenters in California. *Phys. Earth Planet. Inter.*, 11: 227-283.

Gvishiani,A.D. and V.G.Kosobokov, 1981. On foundations of the pattern recognition results applied to earthquake-prone areas. *Proceedings of Ac. Sci. USSR. Physics of the Earth*, 2: 21-36 (in Russian).

Keilis-Borok,V.I. and A.J.Lichtman, 1981. Pattern recognition applied to presidential elections in the United States, 1860-1980: role of integral social, economic and political traits. *Proceedings of US National Ac. Sci.*, **78**, 11: 7230-7234.

Keilis-Borok,V.I. and I.M.Rotwain, 1990. Diagnosis of Time of Increased Probability of strong earthquakes in different regions of the world: algorithm CN. *Phys. Earth Planet. Inter.*, **61**: 57-72.