Stenina M.M., Kuznetsov M.P., Strijov V.V. Ordinal classification using Pareto fronts // Expert Systems with Applications, 2015, 42(14) : 5947–5953

Instance ranking using partially ordered sets of expert estimations

Medvednikova M. M. MMEDVEDNIKOVA@GMAIL.COM
Moscow Institute of Physics and Technology
Institutskiy lane 9, Dolgoprudny city, Moscow region, 141700, Russia

Kuznetsov M. P.ă MIKHAIL.KUZNECOV@PHYSTECH.EDU
Moscow Institute of Physics and Technology
Institutskiy lane 9, Dolgoprudny city, Moscow region, 141700, Russia

Strijov V. V. STRIJOV@CCAS.RU
National Research University Higher School of Economics
20 Myasnitskaya Ulitsa, Moscow 101000, Russia

Editor:

Abstract

We solve an instance ranking problem using ordinal scaled expert estimations. The experts define a preference binary relation on the set of features. The instance ranking problem is considered as the monotone multiclass classification problem. To solve the problem we use a set of Pareto optimal fronts. The proposed method is illustrated with the problem of categorization of the IUCN Red List threatened species.

Keywords: multiclass classification, ordinal scales, object ranking, pareto-optimal front, feature aggregation

1. Introduction

The problem considered in this paper is an instance ranking problem (Strijov et al., 2011; Liu et al., 2010; Cheng et al., 2010) for categorization of threatened species of animals and plants included in the IUCN Red List. The categorization of threatened species maintaining by the IUCN Red List is as follows. Each species belongs to one of seven possible categories: extinct, extinct in the wild, critically endangered, endangered, vulnerable, near threatened and least concern. This categorization is monotone with respect to the risk of extinction. The category can be defined by one of two methods.

1. A direct designation of the category by the expert concordance method, e.g. Delphi method (Schmidt, 2010).

2. A computation of the species category by the expert-given model using features description (Strijov et al., 2011).

The drawback of the first method is that all experts should possess the entire information about the whole set of objects. The drawback of the second method is model sensitivity. The above problems are the problems of the expert data formalization.

In terms of an instance ranking problem an object is a species included in the Red List, and a feature is a criteria describing the species (for example, population size, area square etc.) An expert makes feature estimations in an ordinal scale. Therefore, the matrix "objects-features" is given. The matrix consists of species description and class labels. The problem is to construct a model estimating the class label by the species' description.

The following assumptions about features structure are considered:

- the given set of features is sufficient to construct an adequate model;
- the partial order relation is defined on the feature values;
- the rule "the bigger the better" is valid, that is the greater feature value causes the greater preference by an object;
- different expert estimations of the same object are allowed.

The set of feature values is a partially ordered set. A partial order is one of the well-known binary relations considered in (Linstone and Turoff, 1975). Ordinal-scaled objects are not points in Euclidean space; their nature is non-numeric. The methods of consideration of the non-numeric information considered in (Agresti, 2007). The considered problem contains an expert information about feature preferences, that is the preference binary relation is defined over the set of features. The number of features in the IUCN problem is comparable with the number of objects. The problems of feature selection are considered in (Park et al., 1995; Nogin, 2004).

The monotone classification of objects using preferences is considered in (Doyle, 2004; Furnkranz and Hullermeier, 2003; Har-Peled et al., 2003; Hullermeier et al., 2008; Hullermeier and Furnkranz, 2004; Xia et al., 2008). The most common method are based on pairwise comparisons. The problem of monotone classification arises in the area of information retrieval. To solve this kind of problems the ordinal regression (Cossock and Zhang, 2006) is used as well as the modified SVM (Yue et al., 2007) and boosting (Freund et al., 2003).

We propose the following method to solve the monotone classification problem. The current version of the IUCN Red List categorization is supposed to be compiled correctly except for some noise objects. Let us find the function mapping the set of objects to the set of class labels using the feature preferences, given by experts and historical class labels. This function is defined over Cartesian product of partially ordered sets of expert estimations. The set of values of this function is a partially ordered set of the class labels.

We propose two-stage categorization method: model construction and classification. To construct the model we propose the Pareto multiclass monotone classification algorithm, POF-MC. The Pareto principle is considered in (Nogin, 2003). We assume that the number of objects of the Pareto front should be minimum for the stable models. The methods of Pareto front reduction are considered in (Nogin, 2003, 2004). We propose to reduce the Pareto front by considering an expert information about feature hierarchy (Podinovsky, 2007).

The proposed method is compared with the decision tree method, with the generalized linear regression and with the copula algorithm (Kuznetsov, 2012).

2. Problem statement

Consider the set of pairs

$$\mathfrak{D} = \{(\mathbf{x}_i, y_i)\}, \quad i \in \mathcal{I} = \{1, \ldots, m\},$$

consisting of objects $\mathbf{x}_i$ and class labels y_i. Each object

$$\mathbf{x} = [\chi_1, \dots, \chi_j, \dots, \chi_d]^\top, \quad j \in \mathcal{J} = \{1, \dots, d\},$$

is described by the ordinal scaled measurements. This means that the set of values for the feature χ_j is the partially ordered set $\mathbb{L}_j = \{l_1, \dots l_{k_j}\}$ with a binary relation $\prec$ such that

$$\chi_j \in \mathbb{L}_j = \{l_1, \dots l_{k_j}\}, \text{ where } l_1 \prec \dots \prec l_{k_j}.$$

The set of values $\mathbb{Y}$ for the class labels y_i is also the partially ordered set $\mathbb{Y} = \{l_1, \dots, l_Y\}$ with a binary relation, $l_1 \prec \dots \prec l_Y$. In this paper we consider only strict total orders, for example total, irreflexive, asymmetric and transitive binary relations. However, the proposed methods remain valid for the strict partial orders.

The problem is to find a *monotonic* function

$$\varphi \colon \mathbf{x} \mapsto \hat{y}, \tag{1}$$

where

$$\mathbf{x} \in \mathbb{X} = \mathbb{L}_1 \times \dots \times \mathbb{L}_d \text{ and } y \in \mathbb{Y}.$$

This mapping should minimize the error function value,

$$S(\varphi) = \sum_{i \in \mathcal{I}} r(y_i, \hat{y}_i), \tag{2}$$

where $\hat{y}_i = \varphi(\mathbf{x}_i)$ and the function

$$r(\cdot, \cdot) \tag{3}$$

is the distance between elements of a partially ordered set.

Distance function between elements of a partially ordered set. Let us define a distance function (3) between elements of a partially ordered set. To do this introduce the binary matrix 1 describing binary relations between elements of the set $\mathbb{Z} = \{l_1, \dots, l_z\}$, $l_1 \prec \dots \prec l_z$. If $l_i \succ l_j$ then the matrix has 1 on the intersection of the row i and the column j. For strict total orders the matrix is lower triangular with a zero diagonal.

Table 1: Matrix of a partial order

Labels	l_1	l_2	...	l_{z-1}	l_z
l_1	0	0	...	0	0
l_2	1	0	...	0	0
...	...	...	...	...	...
l_{z-1}	1	1	...	0	0
l_z	1	1	...	1	0

The element l_i of the set $\mathbb{Z}$ corresponds to the row i of the matrix 1. The distance between l_i and l_j is the Hamming distance between the binary vectors,

$$r(l_i, l_j) = R_{\text{Ham}}(str_i, str_j), \tag{4}$$

where $R_{\text{Ham}}(i, j)$ is a number of unmatched elements of the rows i and$j$. The distance function (4) defines the distance between class labels from the set $\mathbb{Y}$ as well as the distance between feature values from the sets $\mathbb{L}_j$, $j = 1, \dots, d$.

3. Two-class Pareto classification

Consider a special case of the problem (1) such that $\mathbb{Y} = \{l_1, l_2\} = \{\mathfrak{o}, \mathfrak{1}\}$, $\mathfrak{o} \prec \mathfrak{1}$. That is the sample $\mathfrak{D}$ consists of the objects with the class labels $\mathfrak{o}$ or $\mathfrak{1}$. Denote a monotonic function f minimizing (2) for the two classes by

$$f : \mathbf{x} \mapsto \hat{y}. \tag{5}$$

Find the function $f(\mathbf{x})$ using a separable sample set

$$\hat{\mathfrak{D}} = \{(\mathbf{x}_i, y_i)\} \quad i \in \hat{\mathcal{I}} \subseteq \mathcal{I}$$

such that $\hat{\mathfrak{D}}$ is a subset of the entire sample $\mathfrak{D}$. "Separable sample" concept in the case of partial orders means that there exists a hull called "POF" corresponding to each class defined by the binary relation $\succ$ such that the hulls for the two classes do not intersect. First the function f will be defined on the separable sample set $\hat{\mathfrak{D}}$ such that the error function value (2) equals zero on this sample. Second, the the definition of the mapping f will be extended to the entire sample $\mathfrak{D}$ and to the whole set of values $\mathbb{X}$.

3.1 Dominance relation without features hierarchy

Now we introduce the following concepts of the dominance relation: n-domination and p-domination. Split the set of object indices $\hat{\mathcal{I}}$ of the separable sample $\hat{\mathfrak{D}}$ to the two subsets

$$\hat{\mathcal{I}} = \mathcal{N} \bigsqcup \mathcal{P}$$

such that $y_n = 0$ for $n \in \mathcal{N}$, and $y_p = 1$ for $p \in \mathcal{P}$. We say that an object $\mathbf{x}_n = [x_{n1}, \dots, x_{nd}]^\top$ *n-dominates* an object $\mathbf{x}_i = [x_{i1}, \dots, x_{id}]^\top$,

$$\text{or} \quad \mathbf{x}_n \succ_n \mathbf{x}_i,$$

$$\text{if} \qquad x_{nj} \succeq x_{ij} \quad \text{for each} \quad j = 1, \dots, d.$$

We say that an object $\mathbf{x}_p = [x_{p1}, \dots, x_{pd}]^\top$ *p-dominates* an object $\mathbf{x}_k = [x_{k1}, \dots, x_{kd}]^\top$,

$$\text{or} \quad \mathbf{x}_p \succ_p \mathbf{x}_k,$$

$$\text{if} \qquad x_{pj} \succeq x_{kj} \quad \text{for each} \quad j = 1, \dots, d.$$

Assume that an object neither n-dominates nor p-dominates itself,

$$\mathbf{x} \not\succ_n \mathbf{x}, \quad \mathbf{x} \not\succ_p \mathbf{x}.$$

Fig. 1 illustrates dominance relation in the case of two features; x-axis shows feature values from the set $\mathbb{L}_1$, y-axis shows feature values from the set $\mathbb{L}_2$. The yellow color indicates the n-dominance space for the object $\mathbf{x}_n$ and the p-dominance space for the object $\mathbf{x}_p$. The object $\mathbf{x}_n$ n-dominates each object $\mathbf{x}_i$ from the corresponding n-dominance space, as well as the object $\mathbf{x}_p$ p-dominates each object $\mathbf{x}_k$ from the corresponding p-dominance space

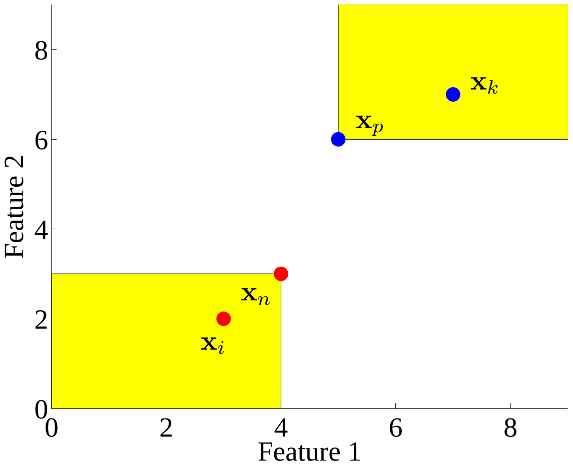


Figure 1: Dominance relation without features hierarchy

3.2 Dominance relation with features hierarchy

Let us introduce the following concepts of dominance relation with features hierarchy, $\succ_{\tilde{n}}$ and $\succ_{\tilde{p}}$. Let the feature χ_r be more preferable than the feature χ_t,

$$r \succ t, \text{ where } r, t \in \mathcal{J}.$$

An object $\mathbf{x}_n = [x_{n1}, \ldots, x_{nr}, \ldots, x_{nt}, \ldots, x_{nd}]^\top$ *$\tilde{n}$-dominates* an object $\mathbf{x}_i = [x_{i1}, \ldots, x_{id}]^\top$,

$$\text{or} \quad \mathbf{x}_n \succ_{\tilde{n}} \mathbf{x}_i,$$

if one of two following conditions holds:

1. $\mathbf{x}_n$ n-dominates $\mathbf{x}_i$ without features hierarchy, $\mathbf{x}_n \succ_n \mathbf{x}_i$, *or*

2. $x_{nr} \succ x_{nt}$ and $\mathbf{x}_n^{tr}$ dominates $\mathbf{x}_i$ without features hierarchy, $\mathbf{x}_n^{tr} \succ_n \mathbf{x}_i$, where $\mathbf{x}_n^{tr} = [x_{n1}, \ldots, x_{nt}, \ldots, x_{nr}, \ldots, x_{nd}]^\top$, that is the object $\mathbf{x}_i$ is n-dominated by the imaginary object $\mathbf{x}_n^{tr}$ corresponding to the object $\mathbf{x}_n$ with the rearranged features r and t.

The object $\mathbf{x}_p = [x_{p1}, \ldots, x_{pr}, \ldots, x_{pt}, \ldots, x_{pd}]^\top$ *$\tilde{p}$-dominates* the object $\mathbf{x}_k = [x_{k1}, \ldots, x_{kd}]^\top$,

$$\text{or} \quad \mathbf{x}_p \succ_{\tilde{p}} \mathbf{x}_k,$$

if one of two following conditions holds:

1. $\mathbf{x}_p$ p-dominates $\mathbf{x}_k$ without features hierarchy, $\mathbf{x}_p \succ_p \mathbf{x}_k$, *or*

2. $x_{pr} \succ x_{pt}$ and $\mathbf{x}_p^{tr}$ dominates $\mathbf{x}_k$ without features hierarchy, $\mathbf{x}_p^{tr} \succ_p \mathbf{x}_k$, where $\mathbf{x}_p^{tr} = [x_{p1}, \ldots, x_{pt}, \ldots, x_{pr}, \ldots, x_{pd}]^\top$, that is the object $\mathbf{x}_i$ is p-dominated by the imaginary object $\mathbf{x}_p^{tr}$ corresponding to the object $\mathbf{x}_p$ with the rearranged features r and t.

Set any object neither n-dominates nor p-dominates itself,

$$\mathbf{x} \not\succ_{\tilde{n}} \mathbf{x}, \quad \mathbf{x} \not\succ_{\tilde{p}} \mathbf{x}.$$

Fig. 2 illustrates dominance example for the case of two features, where the first feature is more preferable than the second one. x-axis shows feature values from the set $\mathbb{L}_1$, y-axis shows feature values from the set $\mathbb{L}_2$. The imaginary objects $\mathbf{x}_n^{21}$ and $\mathbf{x}_p^{21}$ (green points) expand dominance spaces corresponding to the objects $\mathbf{x}_n$ and $\mathbf{x}_p$, respectively.

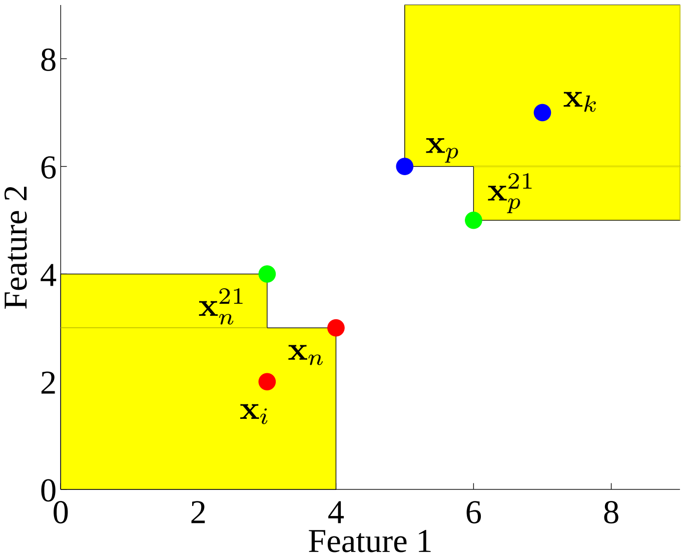


Figure 2: Expansion of dominance spaces due to features hierarchy

Table 2 shows possible dominance spaces according to the feature preferences. The expanded n-dominance space corresponds to the objects having feature 1 more preferable the feature 2. The expanded p-dominance space corresponds to the objects having feature 2 more preferable the feature 1. For the other objects dominance spaces don't expand.

3.3 Pareto optimal front construction

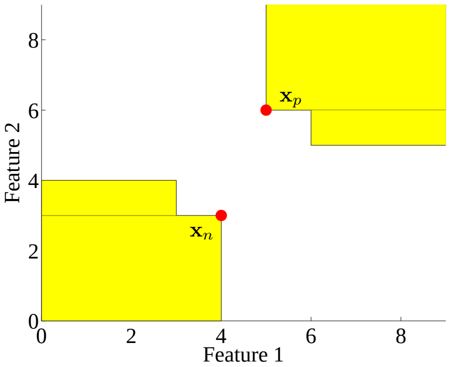
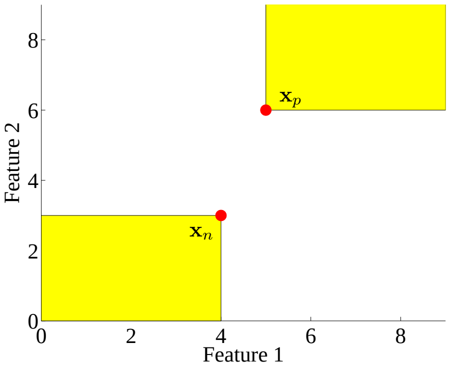
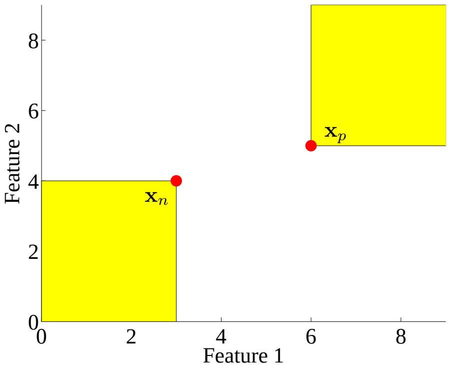
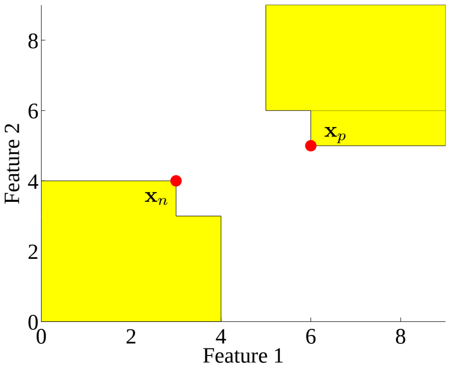
Define the Pareto optimal fronts, the sets defining boundaries of classes of the separable sample.

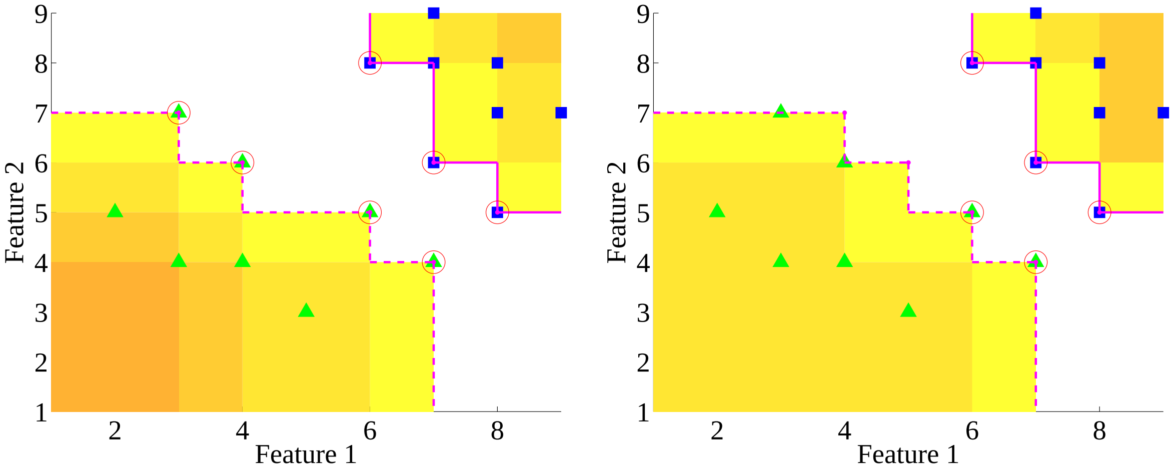
Definition 1 *A set of objects* $\mathbf{x}_n, n \in \mathcal{N}$ *is called Pareto optimal front* POF_n *if for each element* $\mathbf{x}_n \in POF_n$ *doesn't exist any* $\mathbf{x}$ *such that* $\mathbf{x} \succ_n \mathbf{x}_n$ *(*$\mathbf{x} \succ_{\tilde{n}} \mathbf{x}_n$ *for the dominance relation with feature hierarchy).*

Definition 2 *A set of objects* $\mathbf{x}_p, p \in \mathcal{P}$ *is called Pareto optimal front* POF_p *if for each element* $\mathbf{x}_p \in POF_p$ *doesn't exist any* $\mathbf{x}$ *such that* $\mathbf{x} \succ_p \mathbf{x}_p$ *(*$\mathbf{x} \succ_{\tilde{p}} \mathbf{x}_p$ *for the dominance relation with feature hierarchy).*

Fig. 3 illustrates Pareto optimal fronts for the two-class separable sample. Each object is described by the two features. x-axis shows feature values from the set $\mathbb{L}_1$, y-axis shows

Table 2: Dominance spaces corresponding to the feature preferences

	Feature 1 is more preferable than feature 2	Feature 2 is more preferable than feature 1
$x_{n1} \succ x_{n2}$, $x_{p1} \prec x_{p2}$		
$x_{n1} \prec x_{n2}$, $x_{p1} \succ x_{p2}$		



(a) Without feature hierarchy

(b) With feature hierarchy

Figure 3: Pareto optimal fronts

feature values from the set $\mathbb{L}_2$. The green triangles and the blue squares are the objects of different classes. The objects of the Pareto fronts are the red circles. The dotted line indicates the n-front class boundary, the solid line indicates the p-front class boundary. Fig. 3(a) shows Pareto optimal fronts corresponding to the dominance relation without features hierarchy. Fig. 3(b) shows Pareto optimal fronts corresponding to the dominance relation with features hierarchy (feature 1 is more preferable than feature 2).

Fig. 3(b) that dominance space for the front POF_p doesn't change under the assumption of features hierarchy. However, POF_n does change: it contains less objects and has extended dominance space.

In the sequel, we consider dominance relations only with feature hierarchy.

3.4 Two-class classification

Use the constructed Pareto optimal fronts and the corresponding class boundaries to define monotone classifier (5).

Function $f\colon \mathbf{x} \mapsto \hat{y}$ corresponds the class label 0 to the object $\mathbf{x} \in \mathbb{X}$ if there exists an object $\mathbf{x}_n \in \mathrm{POF}_n$, $\tilde{n}$-dominating $\mathbf{x}$. Function f corresponds the class label 1 to the object $\mathbf{x} \in \mathbb{X}$ if there exists an object $\mathbf{x}_p \in \mathrm{POF}_p$, $\tilde{p}$-dominating $\mathbf{x}$.

$$f(\mathbf{x}) = \begin{cases} 0, & \text{if there exists } \mathbf{x}_n \in \mathrm{POF}_n\colon \mathbf{x}_n \succ_{\tilde{n}} \mathbf{x}; \\ 1, & \text{if there exists } \mathbf{x}_p \in \mathrm{POF}_p\colon \mathbf{x}_p \succ_{\tilde{p}} \mathbf{x}. \end{cases} \tag{6}$$

If the set of such elements is empty we extend the definition of the function f to the entire set $\mathbb{X}$ according to the the nearest Pareto optimal front:

$$f(\mathbf{x}) = f\left(\underset{\mathbf{x}' \in \overline{\mathrm{POF}}_n \cup \overline{\mathrm{POF}}_p}{\arg\min} \left(\rho(\mathbf{x}, \mathbf{x}')\right)\right),$$

where the sets $\overline{\mathrm{POF}}_n, \overline{\mathrm{POF}}_p$ include Pareto optimal fronts and boundary points corresponding to the imaginary objects. The function ρ is defined by the function (3) applied to the feature values:

$$\rho(\mathbf{x}, \mathbf{x}') = \sum_{j=1}^{d} r(x_j, x'_j). \tag{7}$$

In other words, the function f classifies an object $\mathbf{x}$ according to the rule of the nearest POF if the object $\mathbf{x}$ isn't dominated with any POF.

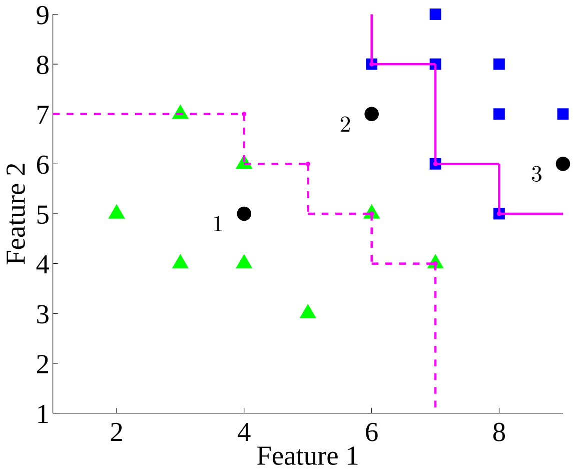


Figure 4: Two-class classification example

Table 3: Two-class classifier example

Object	$\mathbf{x}$	$f(\mathbf{x})$
1	(4,5)	0
2	(6,7)	1
3	(9,6)	1

Fig. 4 shows model data consisting of the two-class objects. The first class objects are the green triangles, the second class objects are the blue squares. Each object is described by the two features. x-axis shows feature values from the set $\mathbb{L}_1$, y-axis shows feature values from the set $\mathbb{L}_2$. The classified objects are the black circles.

Table 3 shows results of classification and consists of three columns. The first column contains object numbers, the second column contains the objects coordinates, the third column contains classifier outputs. The label 0 means that the object is classified as the green triangle, the label 1 means that the object is classified as the blue square.

3.5 Separable sample construction

Consider the method of a set $\hat{\mathcal{I}}$ construction such that the function $f\colon \mathbf{x} \mapsto \hat{y}$ is monotone on the coresponding subsample. Split the set of indices $\mathcal{I}$ into the two subsets

$$\mathcal{I} = \mathcal{N} \bigsqcup \mathcal{P}$$

such that

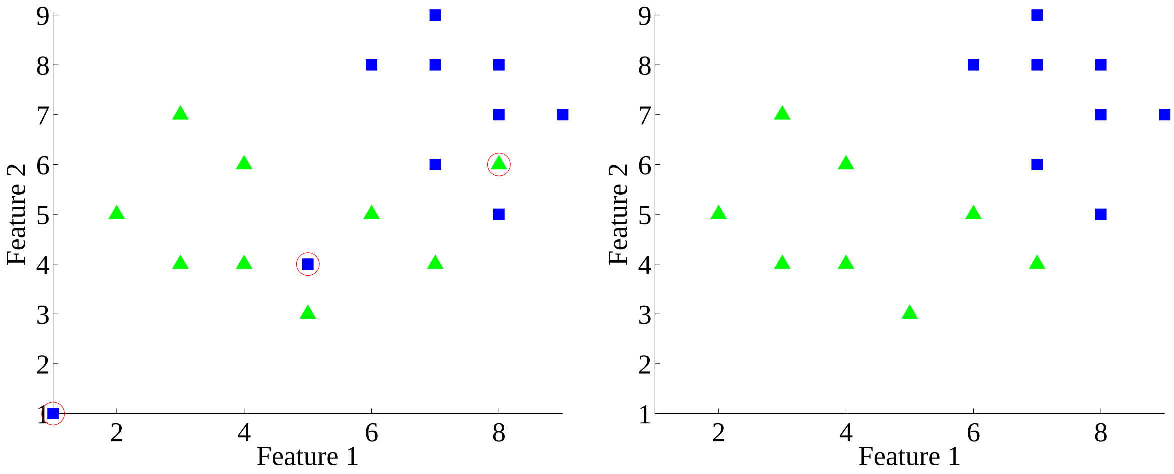
$$y_n = 0, \quad n \in \mathcal{N}, \quad \text{and} \quad y_p = 1, \quad p \in \mathcal{P}.$$

Consider the power μ of the set of objects dominated by the object $\mathbf{x}_i$ and belonging to the foreign class:

$$\mu(\mathbf{x}_i) = \begin{cases} \#\{\mathbf{x}_j \mid \mathbf{x}_i \succ_n \mathbf{x}_j, j \in \mathcal{P}\}, & \text{if } i \in \mathcal{N}; \\ \#\{\mathbf{x}_j \mid \mathbf{x}_i \succ_p \mathbf{x}_j, j \in \mathcal{N}\}, & \text{if } i \in \mathcal{P}, \end{cases}$$

where $\#$ means power of the set. To find the set $\hat{\mathcal{I}}$ we consequently eliminate defective objects from the entire sample $\mathfrak{D}$.

```
$\mathcal{I} = \mathcal{P} \bigsqcup \mathcal{N}$.
return $\hat{\mathcal{I}} = \hat{\mathcal{P}} \bigsqcup \hat{\mathcal{N}}$.
$\hat{\mathcal{I}} := \mathcal{I}$, $\hat{\mathcal{P}} := \mathcal{P}$, $\hat{\mathcal{N}} := \mathcal{N}$; {initialization}
while the sample has the objects $\mathbf{x}_i$, $i \in \hat{\mathcal{I}}$ such that $\mu(\mathbf{x}_i) > 0$ do
    $\hat{i} := \arg\max_{i \in \hat{\mathcal{I}}} \mu(\mathbf{x}_i)$;
    $\hat{\mathcal{I}} := \hat{\mathcal{I}} \backslash \{\hat{i}\}$;
    if $\hat{i} \in \hat{\mathcal{P}}$ then
        $\hat{\mathcal{P}} := \hat{\mathcal{P}} \backslash \{\hat{i}\}$;
    if $\hat{i} \in \hat{\mathcal{N}}$ then
        $\hat{\mathcal{N}} := \hat{\mathcal{N}} \backslash \{\hat{i}\}$.
```



(a) With defective objects

(b) Without defective objects

Figure 5: Defective objects elimination

Fig. 5 shows the model sample set, where objects are described with two features. This sample set consists of two classes (the green triangles and the blue squares). Fig. 5(a) shows the sample set with the defective objects (1;1), (5;4) and (8;6). The defective objects dominate the objects of the opposite class. This objects are marked as the red circles. Fig. 5(b) shows the separable subsample obtained using the defective objects elimination method.

1, 2	...	$u-1$, u	u, $u+1$	...	$Y-1$, Y
1	...	1	0	...	0

Table 4: Monotone classifier illustration

4. Monotone classification

4.1 Monotone classifier construction

Consider a general case of the problem,

$$\mathbb{Y} = \{l_1, \dots, l_u, l_{u+1}, \dots, l_Y\}, \quad l_1 \prec \dots \prec l_u \prec l_{u+1} \prec \dots \prec l_Y.$$

Denote by $\{1, \dots, u, u+1, \dots, Y\}$ class labels indices. Construct a monotone two-class classifier

$$f_{u,u+1} \colon \mathbf{x} \mapsto \hat{y} \in \{0, 1\}, \quad \mathbf{x} \in \mathbb{X},$$

for each pair of adjacent classes $u, u+1$. To construct the two-class classifier we split the sample into two parts: objects with the class labels $\preccurlyeq l_u$ and objects with the class labels $\succcurlyeq l_{u+1}$. Assign the label 0 to the objects from the first part of the sample and the label 1 to the objects from the second one. Wherein the set of object indices of the separable sample $\hat{\mathfrak{D}}$ is splitted into two disjoint subsets,

$$\hat{\mathcal{I}} = \mathcal{N}_u \bigsqcup \mathcal{P}_{u+1}, \text{ where } n \in \mathcal{N}_u, \text{ if } y_n \preceq l_u, \text{ and } p \in \mathcal{P}_{u+1}, \text{ if } y_p \succeq l_{u+1}.$$

The monotone classifier

$$\varphi(\mathbf{x}) = \varphi(f_{1,2}, \dots, f_{Y-1,Y})(\mathbf{x}), \quad \varphi \colon \mathbb{X} \to \mathbb{Y},$$

is defined as follows,

$$\varphi(\mathbf{x}) = \begin{cases} \min\limits_{l_u \in \mathbb{Y}} \{l_u \mid f_{u,u+1}(\mathbf{x}) = 0\}, & \text{if } \{l_u \mid f_{u,u+1}(\mathbf{x}) = 0\} \neq \emptyset; \\ l_Y, & \text{if } \{l_u \mid f_{u,u+1}(\mathbf{x}) = 0\} = \emptyset. \end{cases} \tag{8}$$

Table 4 illustrates formula 8. An output of the monotone classifier $\varphi(\mathbf{x})$ is a first class label l_u where the classifier $f_{u,u+1}$ equals 0. If each output $f_{u,u+1}$ equals 1, assign the label l_Y to the result of monotone classification.

Fig. 6 shows an example of the set of objects from three different classes. The axis show feature values describing objects. The various objects are marked with red circles, green triangles and blue squares. The classes boundaries corresponding to the n-fronts are indicated by the dotted line, the solid line indicates the p-fronts. The classified objects are the black circles. Table 5 shows an example of the set of two-class classifiers $f_{1,2}, f_{2,3}$ included in the monotone classifier $\varphi(\mathbf{x})$ for the illustrated sample. The first column contains object numbers, the second—their coordinates, the third and the fourth—results of two-class classificators for the adjacent classes. The label 0 in the third column means that the classifier $f_{1,2}$ assign the object to the first class. The label 0 in the fourth column means that the classifier $f_{2,3}$ assign the object to the second class. The last column contains the results of monotone classification. The values of this column correspond to the output of the monotone classifier.

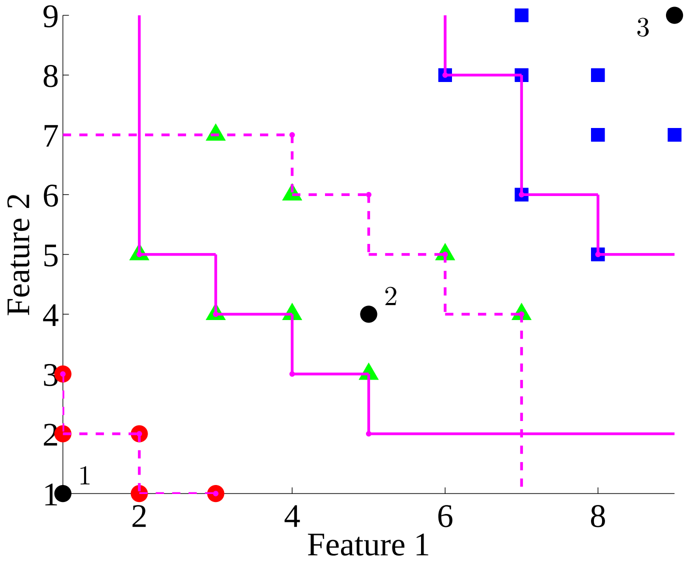


Figure 6: Pareto front, feature 1 is more preferable than feature 2

Table 5: Monotone classifier example

Number	Object, $\mathbf{x}$	$f_{12}(\mathbf{x})$	$f_{23}(\mathbf{x})$	$\varphi(\mathbf{x})$
1	(1,1)	0	0	1
2	(5,4)	1	0	2
3	(9,9)	1	1	3

4.2 Extension of Pareto optimal front definition for monotone classification

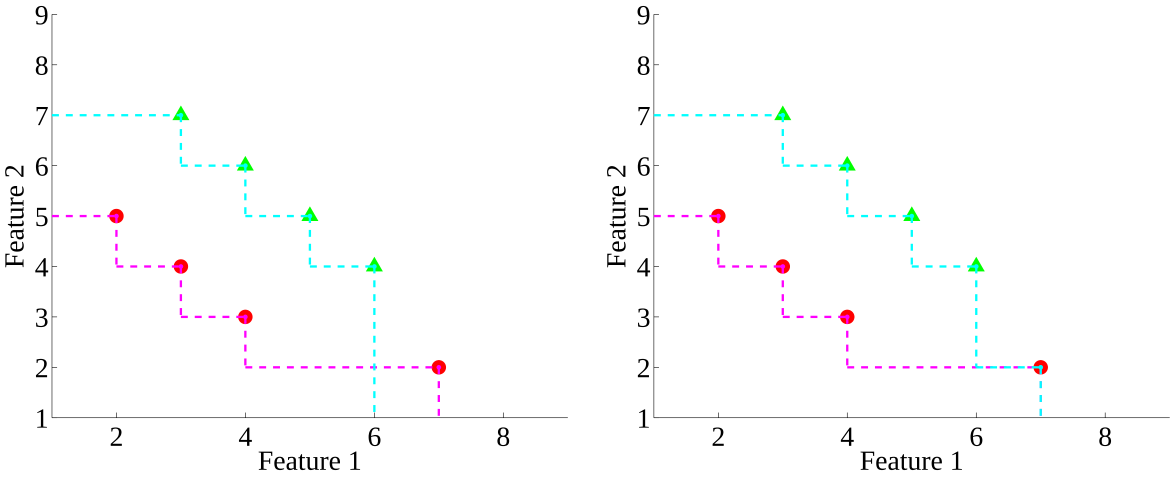
To construct the fronts between classes having labels l_u and l_{u+1} We use objects with the class labels $l_1, \dots, l_u$ to construct an n-front for the class with the label l_u and objects with the class labels $l_{u+1}, \dots, l_Y$ to construct a p-front for the class with the label l_{u+1}. Therefore the same objects belong to fronts for different classes and a front for one class contains objects of different classes. Fig. 7 shows three-class model data sample. The objects of a first class are the red circles, the objects of a second class are the green triangles. The figure shows that the object (7;2) from the first class belongs to the n-fronts of both the first and second classes.

This yields that the definition of the n-front is extended by the objects with the class label not greater than the n-front class label; the definition of the p-front is extended by the objects with the class label not less than the p-front class label.

4.3 Admissible classifiers

Definition 3 *The classifier φ (8) is called* admissible *if for each function $f_{u,u+1}$ the transitivity condition holds:*

$$\begin{cases} \textit{if} \quad f_{u,u+1}(\mathbf{x}) = 0, \quad \textit{than} \quad f_{(u+s)(u+1+s)}(\mathbf{x}) = 0 \qquad s\colon (u+1+s) \leqslant Y, \\ \textit{if} \quad f_{u,u+1}(\mathbf{x}) = 1, \quad \textit{than} \quad f_{(u-s)(u+1-s)}(\mathbf{x}) = 1 \qquad s\colon (u-s) \geqslant 1. \end{cases} \tag{9}$$



(a) Without extension

(b) With extension

Figure 7: An example of a common object for two fronts

Definition 4 *We shall say that Pareto optimal fronts $POF_n(u)$ and $POF_p(u+1)$ don't intersect,*

$$POF_n(u) \quad \cap \quad POF_p(u+1) = \emptyset,$$

if the boundaries of their dominance spaces don't intersect,

$$\overline{POF_n(u)} \quad \cap \quad \overline{POF_p(u+1)} = \emptyset.$$

Fig. 3 shows an example of the non-intersected Pareto optimal fronts.

Theorem 1 *If the Pareto optimal fronts don't intersect,*

$$POF_n(u) \quad \cap \quad POF_p(u+1) = \emptyset, \quad u = 1, \ldots, Y-1,$$

then the transitivity condition (9) holds for any classified object.

Proof Prove the theorem for the case $z = 3$. For more number of classes the proof is similar. The considered dominance relation can be constructed with features hierarchy or without it.

Suppose that Pareto optimal fronts don't intersect,

$$\mathrm{POF}_n(u) \quad \cap \quad \mathrm{POF}_p(u+1) = \emptyset, \quad u = 1, 2;$$

$$\mathrm{POF}_n(u) \quad \cap \quad \mathrm{POF}_p(u+1) = \emptyset, \quad u = 1, 2;$$

and there exists an object $\mathbf{x}$ such that the transitivity condition doesn't hold,

$$f_{1,2}(\mathbf{x}) = 0, \quad f_{2,3}(\mathbf{x}) = 1.$$

(The case $f_{1,2}(\mathbf{x}) = 1, \quad f_{2,3}(\mathbf{x}) = 0$ is similar).

The result $f_{1,2}(\mathbf{x}) = 0$ can be obtained if one of two following conditions holds:

1. $\exists \mathbf{y} \in \mathrm{POF}_n(1): \quad \mathbf{y} \succ_n \mathbf{x}$.

   If $\mathbf{y} \in \mathrm{POF}_n(2)$, then it follows that $f_{2,3}(\mathbf{x}) = 0$ and contradicts with the assumption $f_{2,3}(\mathbf{x}) = 1$.

   If $\mathbf{y} \notin \mathrm{POF}_n(2)$, then

   $$\exists \mathbf{w} \in \mathrm{POF}_n(2): \quad \mathbf{w} \succ_n \mathbf{y},$$

   and it follows that

   $$\mathrm{POF}_n(2) \ni \mathbf{w} \succ_n \mathbf{y} \succ_n \mathbf{x} \quad \Rightarrow \quad f_{2,3}(\mathbf{x}) = 0.$$

2. The fronts $\mathrm{POF}_n(1)$ and $\mathrm{POF}_p(2)$ doesn't dominate the object $\mathbf{x}$. In that case,

   $$\exists \mathbf{y}_0 \in \overline{\mathrm{POF}_n(1)}, \text{ such that } \mathbf{y}_0 = \arg \min_{\mathbf{y} \in \overline{\mathrm{POF}_n(1) \cup \mathrm{POF}_p(2)}} \rho(\mathbf{x}, \mathbf{y}),$$

   where ρ is the distance function (7).

   The assumption $f_{2,3}(\mathbf{x}) = 1$ holds in one of two possible cases.

   (a) $\exists \mathbf{t} \in \mathrm{POF}_p(3): \quad \mathbf{t} \succ_p \mathbf{x}$.

   The object $\mathbf{t}$ doesn't belong to $\mathrm{POF}_p(2)$ since $\mathrm{POF}_p(2)$ doesn't dominate the object $\mathbf{x}$. Then it follows that

   $$\exists \mathbf{t}_0 \in \mathrm{POF}_p(2): \quad \mathbf{t}_0 \succ_p \mathbf{t}.$$

   Therefore we obtain a chain of inequalities,

   $$\mathbf{t}_0 \succ_p \mathbf{t} \succ_p \mathbf{x},$$

   and it follows that $\mathbf{t}_0 \succ_p \mathbf{x}$. That contradicts with the assumption that the front $\mathrm{POF}_p(2)$ doesn't dominate the object $\mathbf{x}$.

   (b) The object $\mathbf{x}$ isn't dominated by the fronts $\mathrm{POF}_n(2)$ and $\mathrm{POF}_p(3)$. In this case the object $\mathbf{x}$ isn't dominated by any front $\mathrm{POF}_n(u), \mathrm{POF}_p(u+1)$ such that $u = 1, 2$.

   Note that there exists an object $\mathbf{y}_1 \in \mathrm{POF}_n(1)$ such that a boundary of dominance space of $\mathbf{y}_1$ contains a point $\mathbf{y_0}$, where $\mathbf{y_0}$ is the nearest points to $\mathbf{x}$ in the sence of Hamming distance (7). The object $\mathbf{y}_1$ can belong to the front $\mathrm{POF}_n(2)$. In this case, the distance between $\mathbf{x}$ and $\mathrm{POF}_n(2)$ is less than the distance between $\mathbf{x}$ and $\mathrm{POF}_n(1)$. Here the distance between a point and a front means the distance between a point and the nearest point of a front. If the object $\mathbf{y}_1$ doesn't belong to the front $\mathrm{POF}_n(2)$, there exists an object

   $$\mathbf{y}_2 \in \mathrm{POF}_n(2): \quad \mathbf{y}_2 \succ_n \mathbf{y}_1.$$

   However, $\mathbf{y}_2 \not\succ_n \mathbf{x}$ since the object $\mathbf{x}$ isn't dominated by any front. Then it follows that the distance between the object $\mathbf{x}$ and the front $\mathrm{POF}_n(2)$ is less than the distance between the object $\mathbf{x}$ and the point $\mathbf{y}_0$ of the front $\mathrm{POF}_n(1)$.

The proof for the pair of fronts $\mathrm{POF}_p(2), \quad \mathrm{POF}_p(3)$ is similar. There exists an object $\mathbf{w}_1 \in \mathrm{POF}_p(2)$ such that a boundary of dominance space of $\mathbf{w}_1$ contains a point $\mathbf{w_0}$, where $\mathbf{w_0}$ is the nearest points to $\mathbf{x}$ in the sence of Hamming distance (7). The distance between the object $\mathbf{x}$ and the front $\mathrm{POF}_p(3)$ is not less that the distance between $\mathbf{x}$ and $\mathbf{w}_0 \in \mathrm{POF}_p(2)$.

We have proved that the distance between $\mathbf{x}$ and $\mathrm{POF}_n(2)$ is not greater than the distance between $\mathbf{x}$ and $\mathrm{POF}_n(1)$, and the distance between $\mathbf{x}$ and $\mathrm{POF}_p(3)$ is not less than the distance between $\mathbf{x}$ and $\mathrm{POF}_p(2)$. From $f_{1,2}(\mathbf{x}) = \mathtt{0}$ it follows that the distance between $\mathbf{x}$ and $\mathrm{POF}_n(1)$ is less than the distance between $\mathbf{x}$ and $\mathrm{POF}_p(2)$. Then $\mathbf{x}$ is nearer to $\mathrm{POF}_n(2)$ than to $\mathrm{POF}_p(3)$. That contradicts with the assumption $f_{2,3}(\mathbf{x}) = \mathtt{1}$ and concludes the proof.

■

Since the method of Pareto optimal fronts uses only separable samples, all fronts are disjoined. Therefore the monotone classifier (8) is admissible and the transitivity condition (9) holds for any classified object.

5. Computational experiment

The goal of the computational experiment is to illustrate the proposed instance ranking method, POF-MC, for the problem of the IUCN Red List categorization using expert estimations. The data contain 110 objects from three categories and are described by 102 features. The set of features is splitted by the experts into 5 subsets. Inside each subset of features experts define features preference binary relation. Below we describe a classification algorithm for subsets of features, the classification error function and the results of the algorithm comparison.

5.1 Monotone classification with feature aggregarion

The set of features is splitted by the experts into five subsets. Each subset contains features describing a certain group of object properties:

- biological condition $\mathcal{A}_1$;
- cumulative threats $\mathcal{A}_2$;
- significance $\mathcal{A}_3$;
- protection level $\mathcal{A}_4$;
- willingness $\mathcal{A}_5$.

The set of feature indices $\mathcal{J}$ is splitted into five disjoint subsets $\mathcal{J} = \mathcal{A}_1 \sqcup \ldots \sqcup \mathcal{A}_5$. The experts set the feature subset preferences: denote $A_i \succ A_j$ if the subset A_i is more preferable than the subset A_j. The preference order on the subsets is

$$\mathcal{A}_1 \succ \ldots \succ \mathcal{A}_5.$$

Table 6: An excerpt from the questionary with expert estimations

Feature	Condition	Trend

population	3 – big;
	2 – small;
	1 – critically small
		4 – grows;
		3 – stable;
		2 – reduces;
		1 – reduces fast

Population structure	2 – complex; **1** – simple	**2** – stable; 1 – disappear

Inside each subset some partial order of feature preferences is defined by the experts.

To get the result classification for the Red List species we construct a monotone classifier (1) for each subset of feature indices $\mathcal{A}_1, \dots, \mathcal{A}_5$. Therefore for each object $\mathbf{x}$ we obtain the set of classification results

$$y_i = \varphi_{\mathcal{A}_i}(\mathbf{x}), \quad i = 1, \dots, 5.$$

Assume the vector $\mathbf{y} = [y_1, \dots, y_n]^\top$ to be the new feature descriptions of the object $\mathbf{x}$,

$$\mathbf{x} \mapsto \mathbf{y} = \begin{bmatrix} \varphi_{\mathcal{A}_1}(\mathbf{x}) \\ \dots \\ \varphi_{\mathcal{A}_5}(\mathbf{x}) \end{bmatrix}.$$

To obtain the final results of classification construct the classifier (1) basing on the new feature description $\mathbf{y}$ and the expert information on the new feature preferences $\mathcal{A}_1 \succ \dots \succ \mathcal{A}_5$:

$$s = \varphi(\mathbf{y}).$$

5.2 Feature aggregation

Table 6 shows an excerpt of the questionary, fulfilled by an expert. This table shows a description of one species described by the four features.

Table 7 shows feature preferences inside the subset "biological condition". An element of the table equals 1 if a row feature is more important that a column feature, and equals 0 otherwise.

Let $\{\chi_1, \dots, \chi_p\}$ belong to the same aggregation subset and be ordinal scaled, $\chi_j \in \mathbb{L}_j = \{1, \dots, k_j\}, \quad j = 1, \dots, p$. To aggregate the features map the natural numbers to the elements of $\mathbb{L}_j$ such that:

$$l_1 = 1, \quad \dots, \quad l_{k_j} = k_j.$$

The value of an aggregated feature ψ is the normalized sum of the values of the features χ_j, $j = 1, \dots, p$:

$$\psi = \sum_{j=1}^{p} \chi_j - p + 1.$$

Table 7: The matrix of feature preferences

1 if row feature $\succ$ column feature	Population size	Population size trend	Population density	Area size	Area structure	Population structure	Genetic diversity	Physiological condition	Habitat condition
Population size	0	1	1	1	1	1	1	1	1
Population size trend	0	0	1	1	1	1	1	1	1
Population density	0	0	0	1	1	1	1	1	1
Area size	0	0	0	0	1	1	1	1	1
Area structure	0	0	0	0	0	1	1	1	0
Population structure	0	0	0	0	0	0	1	1	0
Genetic diversity	0	0	0	0	0	0	0	0	0
Physiological condition	0	0	0	0	0	0	1	0	0
Habitat condition	0	0	0	0	1	1	1	1	0

After the feature aggregation the set of feature indices $\mathcal{J} = \{1, \dots, d\}$ maps to the set $\mathcal{J}' = \{1, \dots, d'\}$ with the less number of elements. A partial order corresponds to each subset of features $j \in \mathcal{J}_{\mathcal{A}_i} \subset \mathcal{J}'$, $i = 1, \dots, 5$. Denote by $j_r \succ j_t$, if the feature j_r is more important than j_t.

The matrix of expert preferences is considered to be self-consistent such that the partial order relation is acyclic. If $j_{i_1} \succ \dots \succ j_{i_n} \succ j_{i_1}$, the features $j_{i_1}, \dots, j_{i_n}$ are non-comparable.

5.3 Algorithm comparison

The POF monotone classifier (POF MC) is compared with the decision trees algorithm, the generalized linear regression and the copula algorithm (Kuznetsov, 2012). To compare the algorithms we use three criteria:

1. mean learn error (2),
2. mean test error, computed by the LOO and 10-fold Cross Validation methods (see the description below),
3. time for model construction.

Table 8 shows the results of the experiment.

Leave-One-Out. At each iteration one object is excluded from the learn sample and becomes the test sample. The following error value is computed:

$$\mathrm{LOO} = \frac{1}{m}\sum_{i=1}^{m} r\big(y_i, \varphi(\mathbf{x}_i, \mathfrak{D} \setminus \{(\mathbf{x}_i, y_i)\})\big), \tag{10}$$

where $\varphi(\mathbf{x}_i, \mathfrak{D}\setminus\{(\mathbf{x}_i, y_i)\})$ is the monotone classifier (8) constructed on the sample $\mathfrak{D}$ without the object $\mathbf{x}_i$, and $r(\cdot,\cdot)$ defined by the formula (4).

10-fold Cross Validation. The set of indices $\mathcal{I} = \{1, \dots, m\}$ of the sample $\mathfrak{D}$ is splitted randomly into 10 disjoint subsets: $\mathcal{I} = \mathcal{B}_1 \bigsqcup \dots \bigsqcup \mathcal{B}_{10}$.

$$\mathrm{CV} = \frac{1}{m}\sum_{k=1}^{10}\sum_{i\in\mathcal{B}_i} r\big(y_i, \varphi(\mathbf{x}_i, \mathfrak{D}(\mathcal{I} \setminus \mathcal{B}_k)\big), \tag{11}$$

where $\varphi(\mathbf{x}_i, \mathfrak{D}(\mathcal{I} \setminus \mathcal{B}_k)$ is the monotone classifier (8), constructed on the sample $\mathfrak{D}$ without the objects with the indices from the set $\mathcal{B}_k$, and $r(\cdot,\cdot)$ is defined by the formula (4).

Table 8: Algorithm comparison

Algorithm	Mean learn error	LOO	Model construction time, sec
POF-MC	0.2157	0.5588	2.1251
Decision trees	0.2451	0.6863	0.4154
Generalized linear model	0.57	0.71	3.6
Copulas	0.57	0.61	0.25

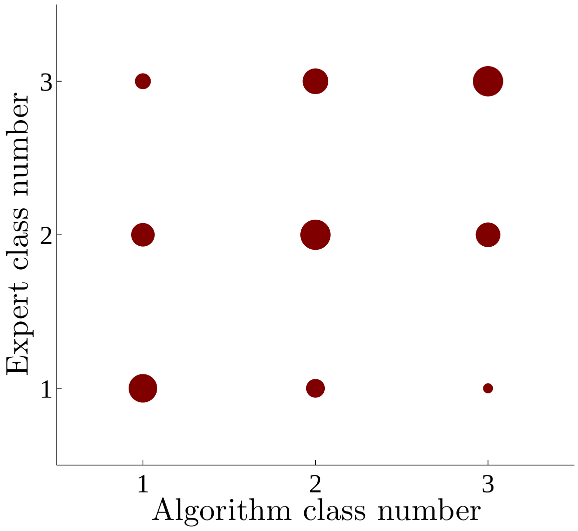


Figure 8: Comparison of the computed and the expert estimated categories

Fig. 8 compares the categories computed by the POF algorithm using the LOO error function. The x-axis shows computed categories, the y-axis shows expert categories. A radius of a point is proportional to the number of objects with the corresponding computed and expert categories. For the significant number of objects the computed and expert categories are the same.

5.4 Testing algorithm on various data sets

Besides the IUCN data the POF-MC algorithm was tested on the data sets Cars and CPU from the UCI repository. The CPU regression problem was transformed to the monotone classification problem by splitting the sample set into four monotone classes according to the target variable values. The LOO (10) and CV (11) error functions were used to measure the quality. Table 9 shows the results of the experiment. The last column of the table shows the result of the algorithm proposed at (Kotlowski and Slowinski, 2009). The missing value in the table means that the corresponding experiment was not executed beacuse of absence of the expert information or small number of features.

Table 9: POF-MC classification results on various data sets

Data	Objects number	Features number	Classes number	POF LOO	POF 10-fold	LPRules 10-fold
IUCN	102	102	3	0.5588	—	—
Cars	1728	6	4	0.3553	0.1933	0.03
CPU	209	6	4	0.6411	0.4833	0.073

6. Conclusion

The authors propose the algorithm for multiclass monotone classification, POF-MC, of objects describing by partially ordered sets of expert estimations. The POF-MC solves the instance ranking problem, the multiclass classification problem where a set of classes is ordered. The algorithm uses Pareto optimal fronts basing on the object dominance relation subject to feature preferences.

The authors propose the method of feature convolution to reduce the feature space dimension. This method is based on expert estimations about feature preferences.

The algorithm is compared with well-known algorithms and shows adequate results. It solved the IUCN Red List categorization problem.

References

Alan Agresti. *An introduction to categorical data analysis*, volume 423. Wiley-Interscience, 2007.

Weiwei Cheng, Michaël Rademaker, Bernard De Baets, and Eyke Hüllermeier. Predicting partial orders: ranking with abstention. *Machine Learning and Knowledge Discovery in Databases*, pages 215–230, 2010.

David Cossock and Tong Zhang. Subset ranking using regression. In Gbor Lugosi and HansUlrich Simon, editors, *Learning Theory*, volume 4005 of *Lecture Notes in Computer Science*, pages 605–619. Springer Berlin Heidelberg, 2006. ISBN 978-3-540-35294-5.

Jon Doyle. Prospects for preferences. *Computational Intelligence*, 20(2):111–136, 2004.

Yoav Freund, Raj Iyer, Robert E. Schapire, and Yoram Singer. An efficient boosting algorithm for combining preferences. *J. Mach. Learn. Res.*, 4:933–969, December 2003. ISSN 1532-4435.

Johannes Furnkranz and Eyke Hullermeier. Pairwise preference learning and ranking. *Machine Learning: ECML 2003*, pages 145–156, 2003.

S. Har-Peled, D. Roth, and D. Zimak. Constraint classification for multiclass classification and ranking. In *NIPS*, pages 785–792, 2003.

Eyke Hullermeier and Johannes Furnkranz. Comparison of ranking procedures in pairwise preference learning. In *Proceedings of the 10th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems (IPMU-04), Perugia, Italy*, 2004.

Eyke Hullermeier, Johannes Furnkranz, Weiwei Cheng, and Klaus Brinker. Label ranking by learning pairwise preferences. *Artificial Intelligence*, 172(16):1897–1916, 2008.

Wojciech Kotlowski and Roman Slowinski. Rule learning with monotonicity constraints. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, pages 537–544, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-516-1.

M. P. Kuznetsov. Integral indicator construction using copulas. *Journal of Machine Learning and Data Analysis*, 1(4):411–419, 2012.

Harold A. Linstone and Murray Turoff. *The Delphi Method: Techniques and Applications.* Addison-Wesley, 1975.

Tie-Yan Liu, Thorsten Joachims, Hang Li, and Chengxiang Zhai. Introduction to special issue on learning to rank for information retrieval. *Information Retrieval*, 13:197–200, 2010. ISSN 1386-4564.

V. D. Nogin. The edgeworth-pareto principle and relative importance of criteria in the case of a fuzzy preference relation. *Computational Mathematics and Mathematical Physics*, 43 (11):16661676, 2003.

V. D. Nogin. A simplified variant of the hierarchy analysis on the ground of nonlinear convolution of criteria. *Computational Mathematics and Mathematical Physics*, 44(7): 11941202, 2004.

Jong Soo Park, Ming-Syan Chen, and Philip S. Yu. An effective hash-based algorithm for mining association rules. *SIGMOD Rec.*, 24(2):175–186, May 1995. ISSN 0163-5808.

V. V. Podinovsky. *Introduction to the importance factors theory in multicriteria decision problem.* Moscow: Fizmatlit, 2007.

Gunther Schmidt. *Relational mathematics*, volume 132. Cambridge University Press, 2010.

Vadim Strijov, Goran Granic, Jeljko Juric, Branka Jelavic, and Sandra Antecevic Maricic. Integral indicator of ecological impact of the croatian thermal power plants. *Energy*, 36 (7):4144–4149, 2011.

Fen Xia, Tie-Yan Liu, Jue Wang, Wensheng Zhang, and Hang Li. Listwise approach to learning to rank: theory and algorithm. In *Proceedings of the 25th international conference on Machine learning*, ICML '08, pages 1192–1199, New York, NY, USA, 2008. ACM. ISBN 978-1-60558-205-4.

Yisong Yue, Thomas Finley, Filip Radlinski, and Thorsten Joachims. A support vector method for optimizing average precision. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '07, pages 271–278, New York, NY, USA, 2007. ACM. ISBN 978-1-59593-597-7.