

# An all-at-once Unimodal SVM Approach for Ordinal Classification

Joaquim F. Pinto da Costa\*, Ricardo Sousa† and Jaime S. Cardoso†

\*CMUP, Faculdade de Ciências da Universidade do Porto

Porto, Portugal

jpcosta@fc.up.pt

†INESC Porto, Faculdade de Engenharia da Universidade do Porto

Porto, Portugal

{rsousa,jsc}@inescporto.pt

**Abstract**—Support vector machines (SVMs) were initially proposed to solve problems with two classes. Despite the myriad of schemes for multiclassification with SVMs proposed since then, little work has been done for the case where the classes are ordered. Usually one constructs a nominal classifier and a posteriori defines the order. The definition of an ordinal classifier leads to a better generalisation. Moreover, most of the techniques presented so far in the literature can generate ambiguous regions. All-at-Once methods have been proposed to solve this issue. In this work we devise a new SVM methodology based on the unimodal paradigm with the All-at-Once scheme for the ordinal classification.

## I. INTRODUCTION

Decision systems which incorporate preference order relations are ubiquitous in everyday life where one prefers a given situation in favour to another. Some application examples goes through recommender systems where one suggest an item to a user by a given order [1], stock market analysis and breast cancer diagnosis [2]. Many supervised learning schemes have been developed in a way to resemble how humans decide. However, the incorporation of an order preference in a decision maker is not always taken into account.

In fact, the use of conventional nominal methods is the most common strategy towards the resolution of ordinal multiclassification problems. Nevertheless, several limitations are inherent to all of them. They do not incorporate totally or in an adequate manner the order, or are too complex. Even though these techniques do not include the order, regarding only to error minimization, also a better classifier generalisation and performance improvements could be attained. The first works date from McCullagh [3] where a regression model was developed incorporating ordinal information on the data eliminating the need for assigning labels. An extension of this work is presented in [4] through the generalization of the additive model [5] by incorporating nonparametric terms. Herbrich [6] applied the Principle of Structural Risk Minimization natural to SVMs [7] to derive a new learning scheme based on large margin bound for the task of ordinal regression. Frank and Hall [8] introduced a simple process to explore the ordinal class information by using conventional binary classifiers. Another approach is presented in [9] which applied a reduction technique to multiple binary problems

that are then solved using a binary margin-based classifier. In [10] it was introduced a generalised formulation for the SVM for ordinal data. More recently, [11] proposed a cascade classification technique encompassing a decision tree classifier and a model tree algorithm. In [12], [13] two new methods were present towards ordinal classification. In [12] a new reduction technique is used allowing to solve the problem of ordinal classification using a single binary classifier. In [13] the class order relation is taken into account by imposing an unimodal distribution to the class a posteriori probabilities.

In this work we introduce a new all-at-once SVM methodology specific for supervised classification with ordered classes,  $\mathcal{C}_1 < \mathcal{C}_2 < \dots < \mathcal{C}_K$ . Based on the work in [13], [14] where a new paradigm mainly in the context of neural networks was developed, here we propose to extend this paradigm for SVM. Basically, our paradigm assumes that the a posteriori probabilities of the  $K$  classes should follow an unimodal distribution, in order to take into account the order relationship. We then formalise this paradigm by introducing appropriate constraints in the usual all-at-once soft margin SVM optimisation functions, both in its primal and dual forms. In Section III-B we present the solution to this mathematical optimisation problem. We consider two situations; a basic and a sophisticated architecture. In Section IV we run some experiments on 6 datasets; one simulated and 5 real. In order to assess the validity of our approach we have compared the two versions of our method with two versions of common all-at-once SVM methods. The performance measures used were the usual misclassification error rate (MER), because of its popularity. Nevertheless, as this measure is inappropriate for the situation under study—ordered classes—we have also used two other performance measures, namely the Spearman’s and Kendall’s tau coefficients (Section IV-B). Afterwards, an extensive discussion follows where we compare all techniques with the different metrics. Finally, in Section V we present the final conclusions and future work.

## II. UNIMODAL PARADIGM

Here we recover the idea of the unimodal paradigm presented in [13], [14]. In the presence of a supervised multiclassification problem where the classes are ordered, like for

instance the four classes in [15], *Excellent* > *Good* > *Fair* > *Poor*, if for a particular instance the class with highest a posteriori probability is *Fair*, then its neighbouring classes, *Good* and *Poor*, should have the second and third highest probabilities. This is the unimodal paradigm which states that the probabilities outputted by a prediction method should increase monotonically, until reaching a maximum value, and then decrease monotonically. In simple words, it doesn't make sense that the most likely class is *Fair* and that the second most likely is *Excellent*; it should be one of the classes closest to *Fair*. This unimodal paradigm has already been introduced in the context of neural networks in [13], [14] and we propose in this work to extend it in another context, namely all-at-once support vector machines (SVM).

### III. ALL-AT-ONCE METHODS

The all-at-once methods were proposed to the scientific community to overcome some vicissitudes present on the standard procedures like the pairwise, one-against-one, one-against-all schemes, DDAG (Decision Directed Acyclic Graph), among others [16]. One of the problems presented on standard heuristics for supervised multiclass classification problems are the unclassifiable regions. These classifiers have

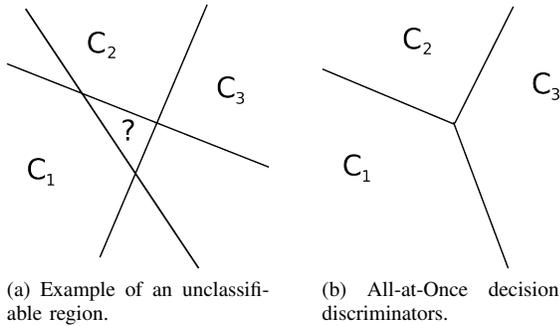


Fig. 1: Different Decision Discriminators.

the feature of not being capable of classifying a point which is within a particular decision region—see Fig. 1a—since each decision function gives a different value for that point. All-at-once schemes solve this issue by determining all the decision functions simultaneously, and therefore do not generate these ambiguity regions.

#### A. Standard Approaches

The standard approaches follow closely the formulation proposed in [17]. However, it should be stated that we are not interested in studying the algorithm complexity that led Crammer [17] to propose an iterative method. The methods implemented in this work are therefore a straightforward implementation of the mathematical formulation.

As referred previously, the technique proposed by Crammer [17] tries to determine all the decision functions simultaneously. More specifically,

$$\mathbf{w}_i^T g(\mathbf{x}) + b_i > \mathbf{w}_j^T g(\mathbf{x}) + b_j, \quad \text{for } j \neq i, i = 1, \dots, K \quad (1)$$

where  $g(x)$  is the mapping function,  $\mathbf{w}_i$  the weight vector for the  $i^{\text{th}}$  class and  $b_i$  its bias term. There are two strategies to attain all the decision planes which we will describe in some detail in the following Sections. These are the basic and sophisticated architectures, as presented in [16].

1) *Basic and Sophisticated Architectures*: All-at-once techniques accomplish the capability to determine simultaneously  $K$  discriminant functions through the definition of one single optimisation function. That is attained by incorporating  $K$  conditions which will serve to separate each class.

In the basic approach the objective function to be minimised is

$$\mathcal{L}(\mathbf{w}, \mathbf{b}, \boldsymbol{\xi}) = \frac{1}{2} \sum_{j=1}^K \|\mathbf{w}_j\|^2 + C \sum_{i=1}^n \sum_{j=1}^K \xi_{i,j}, \quad (2)$$

which uses  $n \times K$  slack variables and, for each point  $(x_i, y_i)$  of the data set, is subject to the constraints,

$$(\mathbf{w}_{y_i} - \mathbf{w}_j)^T g(\mathbf{x}_i) + b_{y_i} - b_j \geq 1 - \xi_{i,j}, \quad \forall j \neq y_i, j = 1, \dots, K, \quad i = 1, \dots, n \quad (3)$$

An alternative to this approach consists in using only  $n$  slack variables. This follows the suggestion of Crammer [17] which replaces the slack variables  $\xi_{i,j}$  by  $\xi_i = \max_j \xi_{i,j}$ . The objective function becomes therefore,

$$\mathcal{L}(\mathbf{w}, \mathbf{b}, \boldsymbol{\xi}) = \frac{1}{2} \sum_{j=1}^K \|\mathbf{w}_j\|^2 + C \sum_{i=1}^n \xi_i \quad (4)$$

subject to the constraints,

$$(\mathbf{w}_{y_i} - \mathbf{w}_j)^T g(\mathbf{x}_i) + b_{y_i} - b_j \geq 1 - \xi_i, \quad \forall j \neq y_i, j = 1, \dots, K, \quad i = 1, \dots, n \quad (5)$$

As it is known, this last problem is easier to solve in the dual Lagrangian formalism.

Focusing for the moment on the basic architecture, the optimisation function becomes,

$$\begin{aligned} \mathcal{L}(\mathbf{w}, \mathbf{b}, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\beta}) = & \frac{1}{2} \sum_{j=1}^K \|\mathbf{w}_j\|^2 + C \sum_{i=1}^n \sum_{j=1}^K \xi_{i,j} - \sum_{i=1}^n \sum_{j=1}^K \beta_{i,j} \xi_{i,j} - \\ & \sum_{i=1}^n \sum_{j=1}^K \alpha_{i,j} ((\mathbf{w}_{y_i} - \mathbf{w}_j)^T g(\mathbf{x}_i) + b_{y_i} - b_j - 1 + \xi_{i,j}) \end{aligned}$$

After some calculus, one obtains the following dual problem,

$$\begin{aligned} \max \mathcal{L}(\boldsymbol{\alpha}) = & \sum_{i=1}^n \sum_{j=1, j \neq y_i}^K \alpha_{i,j} - \frac{1}{2} \sum_{i,k=1}^n \sum_{j=1}^K z_{i,j} z_{k,j} H(x_i, x_k) \\ \text{s.t. } & \begin{cases} \sum_{i=1}^n z_{i,j} = 0 & j = 1, \dots, K \\ 0 \leq \alpha_{i,j} \leq C, & i = 1, \dots, n \quad j \neq y_i, j = 1, \dots, K \end{cases} \end{aligned} \quad (6)$$

where  $H(x_i, x_k)$  is the kernel function and

$$z_{ij} = \begin{cases} \sum_{k=1}^K \alpha_{ik}, & j \neq y_i \\ -\alpha_{ij}, & \text{otherwise} \end{cases} \quad (7)$$

The decision functions are given by

$$D_j(\mathbf{x}) = \sum_{i=1}^n z_{ij} H(x_i, \mathbf{x}) + b_j, \quad j = 1, \dots, K \quad (8)$$

and a new instance  $\mathbf{x}$  is classified into the class  $\arg \max_{j=1, \dots, K} D_j(\mathbf{x})$ .

### B. Unimodal Approach

We have just seen the formulation of all-at-once support vector machines. However, its applicability to ordinal classification is not really appropriate [13], since the order between the classes is not taken into account. The development of ordinal classifiers can lead to more interpretable results and a better generalisation capability.

In a problem with  $K$  ordered classes,  $C_1 < C_2 < \dots < C_K$ , if the maximum *a posteriori* probability is attained at  $\mathcal{P}(C_i|\mathbf{x})$ , the predicted class is  $C_i$ . Then, the unimodal paradigm states that the probabilities should monotonically decrease through  $\mathcal{P}(C_{i+1}|\mathbf{x}) > \dots > \mathcal{P}(C_K|\mathbf{x})$  and  $\mathcal{P}(C_{i-1}|\mathbf{x}) > \dots > \mathcal{P}(C_1|\mathbf{x})$ . This property motivated us to extend the all-at-once methods to ours unimodal paradigm [13].

In the following sections a natural derivation to ordinal classification will be developed inspired by the standard methods presented in the previous section.

1) *Basic Architecture*: Our proposal comes naturally by reformulating the decision functions defined in equation (1) to our problem towards the property mentioned in the Section II. Therefore, the unimodal paradigm for class  $i$  is,

$$\begin{aligned} \mathbf{w}_{j+1}^T g(\mathbf{x}) + b_{j+1} &\geq \mathbf{w}_j^T g(\mathbf{x}) + b_j, \quad j = 1, \dots, i-1 \\ \mathbf{w}_j^T g(\mathbf{x}) + b_j &\geq \mathbf{w}_{j+1}^T g(\mathbf{x}) + b_{j+1}, \quad j = i, i+1, \dots, K-1 \end{aligned} \quad (9)$$

Consequently, the  $L_1$  soft margin SVM can be obtained by minimising

$$\mathcal{L}(\mathbf{w}, \mathbf{b}, \boldsymbol{\xi}) = \frac{1}{2} \sum_{j=1}^K \|\mathbf{w}_j\|^2 + C \sum_{i=1}^n \sum_{j=1}^{K-1} \xi_{i,j} \quad (10)$$

constrained to

$$\begin{aligned} (\mathbf{w}_{j+1} - \mathbf{w}_j)^T g(\mathbf{x}_i) + b_{j+1} - b_j &\geq 1 - \xi_{i,j}, \\ &\forall j = 1, \dots, y_i - 1 \\ (\mathbf{w}_j - \mathbf{w}_{j+1})^T g(\mathbf{x}_i) + b_j - b_{j+1} &\geq 1 - \xi_{i,j}, \\ &\forall j = y_i, \dots, K-1 \end{aligned} \quad (11)$$

To solve this optimisation problem, we will use the Lagrange formalism by introducing the nonnegative Lagrange multipliers  $\alpha_{i,j}$  and  $\beta_{i,j}$  and the quantity to be minimised becomes,

$$\begin{aligned} \mathcal{L}(\mathbf{w}, \mathbf{b}, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\beta}) = & \\ & \frac{1}{2} \sum_{j=1}^K \|\mathbf{w}_j\|^2 + C \sum_{i=1}^n \sum_{j=1}^{K-1} \xi_{i,j} - \sum_{i=1}^n \sum_{j=1}^{K-1} \beta_{i,j} \xi_{i,j} \\ & - \sum_{i=1}^n \sum_{j=1}^{y_i-1} \alpha_{i,j} ((\mathbf{w}_{j+1} - \mathbf{w}_j)^T g(\mathbf{x}_i) + b_{j+1} - b_j - 1 + \xi_{i,j}) \\ & - \sum_{i=1}^n \sum_{j=y_i}^{K-1} \alpha_{i,j} ((\mathbf{w}_j - \mathbf{w}_{j+1})^T g(\mathbf{x}_i) + b_j - b_{j+1} - 1 + \xi_{i,j}) \end{aligned}$$

and after some calculus one obtains the following dual problem:

$$\begin{aligned} \max \mathcal{L}(\boldsymbol{\alpha}) = & \sum_{i=1}^n \sum_{j=1}^K \alpha_{ij} - \frac{1}{2} \sum_{i,k=1}^n \sum_{j=1}^K z_{ij} z_{kj} H(x_i, x_k) \\ \text{s.t.} & \begin{cases} \sum_{i=1}^n z_{ij} = 0 & j = 1, \dots, K-1 \\ 0 \leq \alpha_{ij} \leq C, & i = 1, \dots, n \quad j = 1, \dots, K-1 \end{cases} \end{aligned} \quad (12)$$

where

$$\begin{aligned} z_{ij} = & \alpha_{ij-1} I(j \geq 2) I(j \leq y_i) - \alpha_{ij} I(j \leq y_i - 1) \\ & + \alpha_{ij} I(j \geq y_i) I(j \leq K-1) - \alpha_{ij-1} I(j \geq y_i + 1) \end{aligned} \quad (13)$$

and  $H(x_i, x_k) = g(x_i)^T \cdot g(x_k)$  is the kernel function. The decision functions are given by

$$D_j(\mathbf{x}) = \sum_{i=1}^n z_{ij} H(x_i, \mathbf{x}) + b_j, \quad j = 1, \dots, K \quad (14)$$

and a new instance  $\mathbf{x}$  is classified into the class  $\arg \max_{j=1, \dots, K} D_j(\mathbf{x})$ .

2) *Sophisticated Architecture*: Following Crammer [17] suggestion, one replaces slack variables  $\xi_{ij}$  by  $\xi_i = \max_j \xi_{ij}$ . This produces significant differences in our initial formulation. Therefore, the optimisation function becomes

$$\mathcal{L}(\mathbf{w}, \mathbf{b}, \boldsymbol{\xi}) = \frac{1}{2} \sum_{j=1}^K \|\mathbf{w}_j\|^2 + C \sum_{i=1}^n \xi_i \quad (15)$$

restricted to

$$\begin{aligned} (\mathbf{w}_{j+1} - \mathbf{w}_j)^T g(\mathbf{x}_i) + b_{j+1} - b_j &\geq 1 - \xi_i, \\ &\forall j = 1, \dots, y_i - 1 \\ (\mathbf{w}_j - \mathbf{w}_{j+1})^T g(\mathbf{x}_i) + b_j - b_{j+1} &\geq 1 - \xi_i, \\ &\forall j = y_i, \dots, K-1 \end{aligned} \quad (16)$$

The decision functions are given by

$$\begin{aligned} \max \mathcal{L}(\boldsymbol{\alpha}) = & \sum_{i=1}^n \sum_{j=1}^{K-1} \alpha_{ij} - \frac{1}{2} \sum_{i,k=1}^n \sum_{j=1}^K z_{ij} z_{kj} H(x_i, x_k) \\ \text{s.t.} & \begin{cases} \sum_{i=1}^n z_{ij} = 0 & j = 1, \dots, K-1 \\ 0 \leq \sum_{j=1}^{K-1} \alpha_{ij} \leq C, & i = 1, \dots, n \end{cases} \end{aligned} \quad (17)$$

And the decision functions are given in the same manner as in Equation (14).

## IV. RESULTS

In order to assess the performance of our approach, we performed several experiments. Firstly, we generated a synthetic dataset where the optimal discriminator was known (in this experiment we only needed to find the best parameters values for the objective and kernel function). Afterwards, our

method was evaluated in five real datasets. These datasets are available on the Weka datasets website <sup>1</sup> or in the UCI Machine Learning repository.

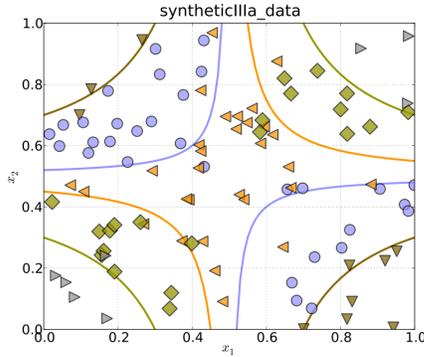
### A. Datasets

On the synthetic dataset, we generated randomly example points  $\mathbf{x} = (x_1, x_2)^t$  in the unit square  $[0, 1] \times [0, 1] \in \mathbb{R}^2$  according to the uniform distribution. To each point was assigned a rank  $y$  from the set  $\{1, 2, 3, 4, 5\}$ , according to

$$y = \min_{r \in \{1, 2, 3, 4, 5\}} \left\{ r : b_{r-1} < 10 \prod_{i=1}^2 (x_i - 0.5) + \varepsilon < b_r \right\}$$

$$(b_0, b_1, b_2, b_3, b_4, b_5) = (-\text{inf}, -1, -0.1, 0.25, 1, +\text{inf}) \quad (18)$$

where  $\varepsilon \sim N(0; 0.125^2)$  simulates the possible existence of error in the assignment of the true class on  $\mathbf{x}$ . A representation of this dataset is present in Fig. 2.



(a) Synthetic dataset in  $\mathbb{R}^2$ ,  $K = 5$ .

Fig. 2: Synthetic dataset representation.

For the real data we tested our method on the SWD, LEV, ESL, Balance and BCCT datasets. The first dataset, SWD, contains real-world assessments of qualified social workers regarding the risk facing children if they stayed with their families at home and is composed by 10 features and 4 classes. LEV dataset contains examples of anonymous lecturer evaluations, taken at the end of MBA courses and is composed by 4 features and 5 classes. These datasets contain 1000 examples each.

Another dataset which we worked on was the ESL dataset containing 488 profiles of applicants for certain industrial jobs. Features are based on psychometric tests results and interviews with the candidates performed by expert psychologists. The class assigned to each applicant was an overall score corresponding to the degree of fitness for the type of job.

Balance dataset available on UCI machine learning repository was also experimented. Created to model psychological experimental results, each example is labelled as having a balance scale tip to the right, left or balanced. Features encompass on left and right weights, and distances.

The last dataset encompasses on 960 observation taken from previous works [15] and expresses the aesthetic evaluation

of Breast Cancer Conservative Treatment (BCCT). For each patient submitted to BCCT, 30 measurements were recorded, capturing visible skin alterations or changes in breast volume or shape. The aesthetic outcome of the treatment for each and every patient was classified in one of the four categories: Excellent, Good, Fair and Poor. In Fig. 3 is depicted the class frequency distribution for each dataset.

### B. Discussion

All the algorithms were put under the same conditions, so that the results could be discussed fairly. The data was divided randomly and distributed through all algorithms. Classes were also equally divided on train (80 instances), validation and test sets to assure that each class was evenly represented. A 5-fold cross validation was performed. In order to assess the variability of the algorithms the experiments were repeated 100 times.

We carried out a straightforward implementation of the formulations presented in Section III and so we did not worry at present with performance issues. We performed a grid search over  $C = 2^{-3}, \dots, 2^{10}$  and  $\gamma = 2^{-3}, \dots, 2^3$  and three measures were used to assess the performance of our models.  $C$  is a penalty factor for each point misclassified and  $\gamma$  controls the fitting of our kernel to the data.

The Misclassification Error Rate (MER), although not very appropriate to our problems with ordered classes (because it considers all errors equally costly) was used due to its popularity. It measures the ratio of the misclassification for some classifier  $f_T$  on a dataset  $\mathcal{O} \subset \mathcal{X}$  as

$$MER = \frac{1}{\text{card}(\mathcal{O})} \sum_{\mathbf{x} \in \mathcal{O}} (1 - \delta_{C_x - f_T(\mathbf{x})}) \quad (19)$$

where  $\delta_{C_x - f_T(\mathbf{x})}$  is the Kronecker delta giving 1 when  $C_x$  is equal to  $f_T(\mathbf{x})$  and zero otherwise.

*Spearman's* rank-order correlation coefficient measures the correlation of two variables which are, or transformed into, two sets of ranks. In our case:

$$\rho_S = \frac{\sum_{\mathbf{x} \in \mathcal{O}} (C_x - \bar{C})(f_T(\mathbf{x}) - \bar{f}_T)}{\sqrt{\sum_{\mathbf{x} \in \mathcal{O}} (C_x - \bar{C})^2} \sqrt{\sum_{\mathbf{x} \in \mathcal{O}} (f_T(\mathbf{x}) - \bar{f}_T)^2}} \quad (20)$$

*Kendall's*  $\tau$  was also used because, although Spearman's coefficient does not consider all errors equally costly, it still depends on the values used to represent the classes. Kendall's coefficient doesn't; it measures the agreement in respect to the *relative* ordering of all possible pairs of data. We call a pair  $(i, j)$  *concordant* if the relative ordering of the true classes  $C_{x_i}$  and  $C_{x_j}$  is the same as the relative ordering of the predicted classes  $f_T(C_{x_i})$  and  $f_T(C_{x_j})$ . We call a pair *discordant* if the relative ordering of the true classes is opposite from the relative ordering of the predicted classes. If there is a tie in either the true or predicted classes, then we do not call the pair either concordant or discordant. If the tie is in the true (or predicted) classes, we will call the pair an "extra true (or predicted) pair",  $e_t$  or  $e_p$ , respectively. If the tie is both on the true and the predicted classes, we ignore the pair.

$$\tau = \frac{c - d}{\sqrt{c + d + e_t} \sqrt{c + d + e_p}} \quad (21)$$

<sup>1</sup>for more information, please see: [http://www.cs.waikato.ac.nz/~ml/weka/index\\_datasets.html](http://www.cs.waikato.ac.nz/~ml/weka/index_datasets.html).

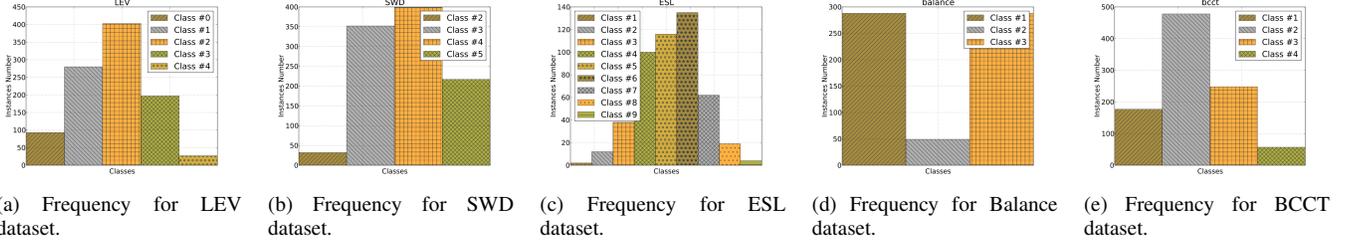


Fig. 3: Real datasets frequency values.

where  $c$  refers to concordant pairs and  $d$  for discordant pairs. In our experiments we have used a RBF kernel and also polynomial kernels with degrees 2 and 3.

The following tables present the best overall results for the four schemes. Note that the postfix I or II refers to the basic and sophisticated architectures, respectively. First we analyse only with the MER measure, due to its common use in classifiers evaluation, and afterwards we will see with the other two measures of accuracy.

Method	standard I	standard II	unimodal I	unimodal II
mer	0.35 (0.09)	<b>0.35 (0.08)</b>	0.38 (0.09)	0.39 (0.11)

(a) mean (std. dev.) for each method, **synthetic dataset**,  $K(\mathbf{x}, \mathbf{y}) = (1 + \mathbf{x}'\mathbf{y})^2$ .

Method	standard I	standard II	unimodal I	unimodal II
mer	0.49 (0.03)	0.49 (0.03)	<b>0.47 (0.03)</b>	0.51 (0.03)

(b) mean (std. dev.) for each method, **SWD dataset**,  $K(\mathbf{x}, \mathbf{y}) = \exp(-\gamma|\mathbf{x} - \mathbf{y}|^2)$ .

Method	standard I	standard II	unimodal I	unimodal II
mer	0.47 (0.04)	0.48 (0.04)	<b>0.46 (0.04)</b>	0.50 (0.04)

(c) mean (std. dev.) for each method, **LEV dataset**,  $K(\mathbf{x}, \mathbf{y}) = \exp(-\gamma|\mathbf{x} - \mathbf{y}|^2)$ .

Method	standard I	standard II	unimodal I	unimodal II
mer	0.49 (0.19)	<b>0.46 (0.12)</b>	0.55 (0.17)	0.50 (0.08)

(d) mean (std. dev.) for each method, **ESL dataset**,  $K(\mathbf{x}, \mathbf{y}) = \exp(-\gamma|\mathbf{x} - \mathbf{y}|^2)$ .

Method	standard I	standard II	unimodal I	unimodal II
mer	0.14 (0.02)	0.14 (0.02)	0.16 (0.03)	<b>0.13 (0.02)</b>

(e) mean (std. dev.) for each method, **Balance dataset**,  $K(\mathbf{x}, \mathbf{y}) = \exp(-\gamma|\mathbf{x} - \mathbf{y}|^2)$ .

Method	standard I	standard II	unimodal I	unimodal II
mer	0.47 (0.02)	0.47 (0.02)	0.47 (0.02)	0.47 (0.02)

(f) mean (std. dev.) for each method, **BCCT dataset**,  $K(\mathbf{x}, \mathbf{y}) = \exp(-\gamma|\mathbf{x} - \mathbf{y}|^2)$ .

TABLE I: Results for MER measure (Misclassification Error Rate).

As we can see in the results on TABLE I, the benefits of our approach are not clear (values in bold correspond to the best results). Our methods obtain the best results in 50% of the datasets, according to MER. One of the reasons is due to the measure used not being appropriate for this problem since it does not take into account the order of the classes. Therefore, we conducted the same experiments but by measuring the performance using the Spearman and Kendall's Tau measures (see TABLE II).

Method	standard I	standard II	unimodal I	unimodal II
spearman	0.86 (0.06)	0.85 (0.07)	<b>0.87 (0.05)</b>	0.86 (0.05)
tau	0.80 (0.06)	0.78 (0.08)	<b>0.81 (0.05)</b>	0.79 (0.06)

(a) mean (std. dev.) for each method, **synthetic dataset**,  $K(\mathbf{x}, \mathbf{y}) = (1 + \mathbf{x}'\mathbf{y})^2$ .

Method	standard I	standard II	unimodal I	unimodal II
spearman	0.45 (0.16)	0.47 (0.06)	<b>0.51 (0.06)</b>	0.47 (0.07)
tau	0.41 (0.07)	0.41 (0.06)	<b>0.46 (0.05)</b>	0.42 (0.05)

(b) mean (std. dev.) for each method, **SWD dataset**,  $K(\mathbf{x}, \mathbf{y}) = \exp(-\gamma|\mathbf{x} - \mathbf{y}|^2)$ .

Method	standard I	standard II	unimodal I	unimodal II
spearman	0.60 (0.05)	0.58 (0.05)	<b>0.63 (0.04)</b>	0.61 (0.05)
tau	0.53 (0.05)	0.52 (0.06)	<b>0.57 (0.04)</b>	0.54 (0.05)

(c) mean (std. dev.) for each method, **LEV dataset**,  $K(\mathbf{x}, \mathbf{y}) = \exp(-\gamma|\mathbf{x} - \mathbf{y}|^2)$ .

Method	standard I	standard II	unimodal I	unimodal II
spearman	0.13 (0.91)	0.75 (0.33)	0.72 (0.42)	<b>0.81 (0.26)</b>
tau	0.02 (0.89)	0.69 (0.34)	0.64 (0.39)	<b>0.76 (0.20)</b>

(d) mean (std. dev.) for each method, **ESL dataset**,  $K(\mathbf{x}, \mathbf{y}) = \exp(-\gamma|\mathbf{x} - \mathbf{y}|^2)$ .

Method	standard I	standard II	unimodal I	unimodal II
spearman	0.85 (0.03)	<b>0.86 (0.03)</b>	<b>0.86 (0.03)</b>	<b>0.86 (0.03)</b>
tau	0.82 (0.04)	<b>0.83 (0.03)</b>	0.82 (0.04)	0.83 (0.04)

(e) mean (std. dev.) for each method, **Balance dataset**,  $K(\mathbf{x}, \mathbf{y}) = \exp(-\gamma|\mathbf{x} - \mathbf{y}|^2)$ .

Method	standard I	standard II	unimodal I	unimodal II
spearman	0.33 (0.04)	0.33 (0.04)	0.25 (0.23)	0.27 (0.23)
tau	0.30 (0.04)	0.30 (0.04)	0.24 (0.19)	0.23 (0.25)

(f) mean (std. dev.) for each method, **BCCT dataset**,  $K(\mathbf{x}, \mathbf{y}) = \exp(-\gamma|\mathbf{x} - \mathbf{y}|^2)$ .

TABLE II: Results for Tau and Spearman measures.

On the synthetic dataset once again the difference between the methods are very dim, although slightly better for our unimodal method. For the real datasets all unimodal schemes attained slightly better results than the corresponding standard all-at-once with exception on the Balance, where there are almost no differences, and on the BCCT dataset.

Despite the results on the Balance dataset being very similar amongst all of the methods it is interesting to see this kind of performance with measures that take into account the order between the classes, whereas with MER, the unimodal approach gives slightly better results. This may be due to the class frequencies distribution on this dataset—see Fig. 3d—because class #2 is slimly represented when compared with the other two.

A far more clear difference is presented on the results for the BCCT dataset, where the results are also very similar according to MER, whereas the standard approaches attain better results according to Spearman and Kendall’s tau. In any case the results are very bad on this dataset. This can be due to the features not being clearly related to the order between the classes. In [15] the authors performed a feature selection in order to select the features that performed best. Based on that study we selected the same features ( $\rho LBC$ ,  $\rho BCE$ ,  $cEMD_a$  and  $s\chi^2 Lab_{3D}$ ) and evaluated our classifier. Results are presented in TABLE III.

Method	standard	standard	unimodal	unimodal
mer	0.20 (0.04)	0.20 (0.04)	0.20 (0.04)	0.20 (0.04)
spearman	<b>0.85 (0.04)</b>	0.85 (0.05)	0.84 (0.04)	0.84 (0.04)
tau	0.81 (0.04)	0.81 (0.05)	<b>0.82 (0.04)</b>	<b>0.82 (0.04)</b>

(a) mean ( std. dev. ) for each method, **BCCT dataset**,  $K(\mathbf{x}, \mathbf{y}) = \exp(-\gamma|\mathbf{x} - \mathbf{y}|^2)$ .

TABLE III: BCCT results when selecting features  $\rho LBC$ ,  $\rho BCE$ ,  $cEMD_a$  and  $s\chi^2 Lab_{3D}$ .

Comparing the results without feature selection— TABLE If and TABLE IIf—and with feature selection— TABLE III—one can assess the improvement not only on the overall performance of all the classifiers but also on our approaches. Even though our methods do not outperform on this particular dataset the standard all-at-once techniques, it attains similar results.

## V. CONCLUSION AND FUTURE WORK

In this work we propose a new multiclass classification formulation for ordinal data. Based on the unimodal paradigm presented in [13], [14] we extended it onto the SVM context using all-at-once strategies. This paradigm states that the probabilities outputted by a prediction method should increase monotonically until reaching a maximum value and then decrease monotonically. With such a strategy we can enforce the ordinal relation amongst the classes.

We have conducted extensive experiments where our method was tested against all-at-once standard techniques. Our Unimodal all-at-once approach was tested on one synthetic and 5 real datasets where, overall, our approach expressed superior results when comparing with standard all-at-once strategies. The classifier performances were assessed with three measures: MER, Spearman and Kendall’s tau.

The methodology here presented can be improved by using different strategies. Crammer in [17] suggested an iterative optimisation technique since the computation of the full problem is highly computationally expensive. This scheme decomposes the problem into sub-problems having therefore the major advantage of being capable to compute for larger datasets. Also, a comparison with Tsochantaridis [18] approach which uses a similar technique as Crammer [17], among others, can be done. Finally, the incorporation of a reject region to remove points that fall in the ambiguity regions can also provide a way to assess the performance of our methods under different conditions.

## ACKNOWLEDGMENTS

The first author was partially supported by Fundação para a Ciência e a Tecnologia (FCT)—Portugal through the Centro de Matemática da Universidade do Porto. The second and third authors would also like to acknowledge the financial support by the ERDF European Regional Development Fund through the COMPETE Programme (operational programme for competitiveness) and by National Funds through the FCT Fundação para a Ciência e a Tecnologia (Portuguese Foundation for Science and Technology) within project FCOMP-01-0124-FEDER-007235.

## REFERENCES

- [1] N. Delannay and M. Verleysen, “Collaborative filtering with interlaced generalized linear models,” *Neurocomputing*, vol. 71, no. 7-9, pp. 1300–1310, 2008.
- [2] A. Tagliafico, G. Tagliafico, S. Tosto, F. Chiesa, C. Martinoli, L. E. Derchi, and M. Calabrese, “Mammographic density estimation: Comparison among bi-rads categories, a semi-automated software and a fully automated one,” *The Breast*, vol. 18, no. 1, pp. 35–40, 2009.
- [3] P. McCullagh, “Regression models for ordinal data,” *Journal Royal Statistical Society, Series B*, vol. 42, pp. 109–142, 1980.
- [4] G. Tutz, “Generalized semiparametrically structured ordinal models,” *Biometrics*, vol. 59, pp. 263–273, 2003.
- [5] T. J. Hastie and R. J. Tibshirani, “Generalized additive models,” *Mono-graphs on Statistics and Applied Probability*, vol. 43, pp. 297–318, 1990.
- [6] R. Herbrich, T. Graepel, and K. Obermayer, “Regression models for ordinal data: A machine learning approach,” Technical University of Berlin, Tech. Rep., 1999.
- [7] V. N. Vapnik, *Statistical Learning Theory*. Wiley-Interscience, 1998.
- [8] E. Frank and M. Hall, “A simple approach to ordinal classification,” in *EMCL ’01: Proceedings of the 12th European Conference on Machine Learning*. London, UK: Springer-Verlag, 2001, pp. 145–156.
- [9] E. L. Allwein, R. E. Schapire, and Y. Singer, “Reducing multiclass to binary: a unifying approach for margin classifiers,” *Journal Machine Learning Research*, vol. 1, pp. 113–141, 2001.
- [10] A. Shashua and A. Levin, “Ranking with large margin principle: Two approaches,” in *Advances in Neural Information Processing Systems 15*, Thrun and K. Obermayer, Eds. Cambridge, MA: MIT Press, 2003, pp. 937–944.
- [11] S. Kotsiantis, “Cascade generalisation for ordinal problems,” *International Journal of Artificial Intelligence and Soft Computing*, vol. 2, pp. 46–57(12), 4 April 2010.
- [12] J. S. Cardoso and J. F. P. da Costa, “Learning to classify ordinal data: the data replication method,” *Journal of Machine Learning Research*, vol. 8, pp. 1393–1429, 2007.
- [13] J. F. P. da Costa, H. Alonso, and J. S. Cardoso, “The unimodal model for the classification of ordinal data,” *Neural Networks*, vol. 21, pp. 78–91, 2008.
- [14] J. F. P. da Costa and J. S. Cardoso, “Classification of ordinal data using neural networks,” *Lecture Notes in Artificial Intelligence*, vol. 3720, pp. 690–697, 2005.
- [15] J. S. Cardoso and M. J. Cardoso, “Towards an intelligent medical system for the aesthetic evaluation of breast cancer conservative treatment,” *Artificial Intelligence in Medicine*, vol. 40, pp. 115–126, 2007.
- [16] S. Abe, *Support Vector Machines for Pattern Classification (Advances in Pattern Recognition)*. Springer, 2005.
- [17] K. Crammer and Y. Singer, “On the Learnability and Design of Output Codes for Multiclass Problems,” *Machine Learning*, vol. 47, no. 2-3, pp. 201–233, 2002.
- [18] I. Tsochantaridis, T. Hofmann, T. Joachims, and Y. Altun, “Support vector learning for interdependent and structured output spaces,” in *International Conference on Machine Learning*, 2004.