A comparative study of machine learning methods for ordinal classification with absolute and relative information

Mengzi Tang[a,*], Raúl Pérez-Fernández[b], Bernard De Baets[a]

[a] *KERMIT, Department of Data Analysis and Mathematical Modelling, Ghent University, Coupure links 653, 9000 Gent, Belgium*
[b] *Department of Statistics and O.R. and Mathematics Didactics, University of Oviedo, C/ Federico García Lorca 18, 33007 Oviedo, Spain.*

Abstract

The performance of an ordinal classifier is highly affected by the amount of absolute information (labelled data) available for training. In order to make up for a lack of sufficient absolute information, an effective way out is to consider additional types of information. In this work, we focus on ordinal classification problems that are provided with additional relative information. We augment several classical machine learning methods by considering both absolute and relative information as constraints in the corresponding optimization problems. We compare these augmented methods on popular benchmark datasets. The experimental results show the effectivenesses of these methods for combining absolute and relative information.

Keywords: Machine learning, absolute information, relative information, ordinal classification

1. Introduction

Classification is undoubtedly the most abundant problem setting in machine learning. Despite the plethora of methods for building a classification model [1], the ultimate predictive performance obviously still depends on the quantity and

*Corresponding author
Email addresses: mengzi.tang@ugent.be (Mengzi Tang), perezfernandez@uniovi.es (Raúl Pérez-Fernández), bernard.debaets@ugent.be (Bernard De Baets)

quality of the training data. Although the era of big data may have rendered this problem obsolete in many cases, still many settings remain where labelled high-quality data is scarce, because of a variety of reasons, such as limited access to expert labellers, high labelling costs, *et cetera* [2, 3]. One possible way out is to tap into another source, for instance by involving novices or by appealing to the crowd [4, 5, 6]. Although this may lead to an improved predictive performance, it also brings along a number of problems, such as source dependence of the reliability of the data (for instance, experts versus novices) and of the gathered type of information (for instance, labels in case of experts, or frequency distributions over the label set in case examples are judged by numerous novices). This calls for a delicate attention when developing a classification model, ensuring the traceability of the impact of the source and type of information on the final model performance. One way to do so is to avoid building a single collective dataset (for instance, as is done by reducing frequency distributions of novices to labelled data by considering the mode, leading to a single dataset of labelled objects [5]), but instead assign a distinctive role to the different sources or types of information (for instance, by using the data from the second source for constraining the learning process on the basis of the data from the first source [7, 8]). Inspiration here can be found in the field of soft-label classification [9, 10, 11], giving us a hint of how frequency distributions stemming from a second source of information could be turned into constraints. For instance, in the case of binary soft-label classification, the class label (positive or negative) comes along with a probability score. These probability scores can be turned into linear constraints on the objective function of a support vector machine, enforcing that examples with a higher probability are projected farther away from the decision boundary than examples with a lower probability of assigning the same class label [10, 11].

The focus in this paper is on *ordinal* classification problems [12, 13], in which the label set comes along with a natural ordering, which obviously needs to be taken into account during model development and prediction. Ordinal classification is an interesting problem setting that has been recognized in many areas

of research, for instance, in social science [14] and in medicine [15]. Various machine learning methods for classification have been adjusted to deal with ordinal classification problems (including naive methods, ordinal binary decomposition methods and threshold methods [16]) and even dedicated methods have been developed (such as distance metric learning methods [17]). Also, ensemble methods have been used to combine several ordinal classifiers into a single best-performing one [18, 19]. Even more so than in the regular classification setting, however, availability of ordinal labelled data is often limited due to its dependence on expensive and time-consuming access to expert labellers, prohibiting the collection of a large amount of labelled data (referred to as *absolute information* from here on, *i.e.*, examples with an explicitly given class label). As mentioned above, one way out is to access an additional source of information, for instance by involving novices in the evaluation. However, in the context of ordinal classification, such novices are usually able to provide *relative information* only, in the form of preference orders for couples of examples, expressing that one example should receive a higher label than another (thus not assigning specific labels, but rather excluding possible labels, therefore already acknowledging the lower reliability of this source of information). Hence, apart from having to deal with two sources of information, we also have to deal with two different types of information, where we additionally have to account for the fact that relative information is much less informative than absolute information. Note that, for obvious reasons, here we do not consider the case (yet) where numerous novices have expressed a preference order for the same couple of objects. From the above description, it is clear that it becomes a challenging research problem to combine absolute and relative information for improving ordinal classification performance, while acknowledging the different information value of both types of information.

The aim of this paper is to solve ordinal classification problems that come along with both absolute and relative information. An obvious approach is to enhance existing methods in order to accommodate relative information as a possible source of additional information. Here, we augment several classical

machine learning methods to combine absolute and relative information for ordinal classification by turning preference orders into linear constraints in the associated optimization problems.

The remainder of this paper is organized as follows. In Section 3, we first introduce the problem setting in which two types of information (absolute and relative) are available. Subsequently, we describe several classical machine learning methods for ordinal regression and propose augmented versions by incorporating constraints arising from the relative information into the corresponding optimization problems. In Section 4, we perform experiments on popular benchmark datasets and analyse the experimental results. Since absolute information is more informative than relative information, yet also more costly, the relationships between cost and performance, and between entropy and performance are explored in detail. We end with some concluding remarks in Section 5.

2. Related work

Although there is a large body of research on ordinal classification, it mainly focuses on tasks based on absolute information only. Only recently, some efforts have been made to adapt existing methods enabling them to exploit different types of information. For instance, Sader et al. [20] proposed an ordinal classification model based on absolute and relative information, the parameters of which are learned by solving a constrained convex optimisation problem. From a different perspective, the present authors recently augmented the nearest neighbor method for ordinal classification to simultaneously cope with absolute and relative information [21]. The latter work was then further improved upon by introducing a distance metric learning method aiming to replace the Euclidean distance metric [22]. The extension of this method to the case in which relative information is gathered from crowds and, therefore, each pairwise comparison is represented by a frequency distribution of preference orders rather than a unique preference order, requires careful consideration and was preliminarily explored in [23]. All mentioned works (perhaps with exception of the latter one,

where the type of information available is slightly different) are closely related to the present study. However, the present work is more ambitious and aims at augmenting other popular ordinal classification methods and comparing their performance. It is anticipated that, as in the classical setting, there is no augmented method that will outperform all others and that the method proposed by the present authors in [22] will stand the test in the comparison with the augmented versions of the most popular methods for ordinal classification found in the literature.

3. Machine learning methods for ordinal classification with absolute and relative information

In this section, we first introduce our problem setting and then augment several machine learning methods for ordinal classification to deal with both absolute and relative information. There is an abundance of machine learning methods for solving ordinal classification problems with absolute information; for a comprehensive overview, see Gutiérrez et al. [16]. Here, we focus on basic methods that allow to incorporate the additional relative information as constraints in the corresponding optimization problems.

3.1. Problem description

Formally, the input data includes two types of information: absolute information and relative information. For the absolute information, we denote the set of input examples by $\mathcal{D} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$. Each input example $\mathbf{x}_i = (x_{i1}, \ldots, x_{id})$ belongs to the input space $\mathcal{X} \subseteq \mathbb{R}^d$ and the corresponding class label y_i belongs to the output space $\mathcal{Y} = \{C_1, C_2, \ldots, C_r\}$, where the class labels are ordered as follows: $C_1 \prec C_2 \prec \ldots \prec C_r$. The absolute information is gathered in a set $\mathcal{A} = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots, (\mathbf{x}_n, y_n)\}$.

In addition, we consider the case that some couples of examples do not have explicitly given class labels but carry relative information. We denote the subset of couples for which the first example is preferred to the second example

by $\mathcal{C}_\succ = \{(\mathbf{a}^1, \mathbf{b}^1), \ldots, (\mathbf{a}^m, \mathbf{b}^m)\} \subseteq \mathcal{X}^2$. A main characteristic of our problem setting is that the amount of absolute information is typically smaller than the amount of relative information, *i.e.*, $n \ll m$.

3.2. Proportional Odds Model with both types of information

The Proportional Odds Model (POM) [24], also called ordinal logistic regression, is a classical approach for solving ordinal classification problems. The underlying idea is to use a logistic function to predict the probabilities of the different class labels. Cumulative probabilities are used for taking into account the ordering among the class labels. Formally, the cumulative probability is modelled as:

$$P(Y_i \preceq C_j) = \phi(\theta_j - \mathbf{w}^\top \mathbf{x}_i) = \frac{1}{1 + \exp\left(\mathbf{w}^\top \mathbf{x}_i - \theta_j\right)}\,, \tag{1}$$

where Y_i is the random (output) variable associated with the input example $\mathbf{x}_i$, the weighing vector $\mathbf{w}$ is common across all class labels, the vector of thresholds $\boldsymbol{\theta}$ is used for separating different classses and $\phi(t) = \frac{1}{1+\exp(-t)}$. Class label C_k is associated with the interval $[\theta_{k-1}, \theta_k[$, where $\theta_0 = -\infty$ and $\theta_r = +\infty$. The probability of example $\mathbf{x}_i$ being assigned class label C_j then is

$$P(Y_i = C_j) = \phi(\theta_j - \mathbf{w}^\top \mathbf{x}_i) - \phi(\theta_{j-1} - \mathbf{w}^\top \mathbf{x}_i)\,. \tag{2}$$

The objective function considered is the negative log-likelihood:

$$\mathcal{L}(\mathbf{w}, \boldsymbol{\theta}) = -\sum_{i=1}^{n} \log\left(\phi(\theta_{y_i} - \mathbf{w}^\top \mathbf{x}_i) - \phi(\theta_{y_i - 1} - \mathbf{w}^\top \mathbf{x}_i)\right)\,, \tag{3}$$

where $\theta_{y_i} := \theta_k$ and $\theta_{y_i - 1} := \theta_{k-1}$ when $y_i = C_k$. The goal is to find the parameters that maximize the likelihood by minimizing the objective function.

In the problem setting considered, additional relative information is provided. For each couple $(\mathbf{a}^\ell, \mathbf{b}^\ell) \in \mathcal{C}_\succ$, we impose that the ordering $\mathbf{a}^\ell \succ \mathbf{b}^\ell$ is respected after projecting both examples, *i.e.*, the inequality $\mathbf{w}^\top \mathbf{a}^\ell > \mathbf{w}^\top \mathbf{b}^\ell$

holds. We incorporate a unit margin and formulate the following constraint:

$$\mathbf{w}^\top \mathbf{a}^\ell - \mathbf{w}^\top \mathbf{b}^\ell \geq 1\,. \tag{4}$$

In order to allow for the violation of some of the inequalities, slack variables are introduced, resulting in the following optimization problem:

$$\begin{aligned} \min_{\mathbf{w},\boldsymbol{\theta}} \quad & \mathcal{F}(\mathbf{w},\boldsymbol{\theta}) = \mathcal{L}(\mathbf{w},\boldsymbol{\theta}) + \frac{\alpha}{m}\sum_{\ell=1}^{m}\eta_\ell + \frac{\lambda}{2}\left\|\mathbf{w}\right\|_2^2 \\ \text{s.t.} \quad & \mathbf{w}^\top \mathbf{a}^\ell - \mathbf{w}^\top \mathbf{b}^\ell \geq 1 - \eta_\ell\,, \\ & \eta_\ell \geq 0, \quad \forall \ell\,, \end{aligned} \tag{5}$$

where m is the number of constraints, α is a parameter controlling the impact of the relative information, $\|\mathbf{w}\|_2$ is a regularizer to avoid overfitting, λ is a regularization parameter and η_ℓ are slack variables.

In order to better understand the objective function in Eq. (5), we rewrite it into the following equivalent form:

$$\mathcal{F}(\mathbf{w},\boldsymbol{\theta}) = \mathcal{L}(\mathbf{w},\boldsymbol{\theta}) + \frac{\alpha}{m}\sum_{\ell=1}^{m}[1 + \mathbf{w}^\top \mathbf{b}^\ell - \mathbf{w}^\top \mathbf{a}^\ell]_+ + \frac{\lambda}{2}\left\|\mathbf{w}\right\|_2^2\,, \tag{6}$$

where $[z]_+ = \max(z, 0)$. The subgradient of $\mathcal{F}(\mathbf{w},\boldsymbol{\theta})$ at $\mathbf{w}$ is given by:

$$\frac{\partial \mathcal{F}(\mathbf{w},\boldsymbol{\theta})}{\partial \mathbf{w}} = \frac{\partial \mathcal{L}(\mathbf{w},\boldsymbol{\theta})}{\partial \mathbf{w}} + \frac{\alpha}{m}\sum_{\ell=1}^{m}\psi_\ell(\mathbf{b}^\ell - \mathbf{a}^\ell) + \lambda \mathbf{w}\,, \tag{7}$$

where

$$\psi_\ell = \begin{cases} 1 \quad , & \text{if } 1 + \mathbf{w}^\top \mathbf{b}^\ell - \mathbf{w}^\top \mathbf{a}^\ell > 0\,; \\ 0 \quad , & \text{if } 1 + \mathbf{w}^\top \mathbf{b}^\ell - \mathbf{w}^\top \mathbf{a}^\ell \leq 0\,. \end{cases} \tag{8}$$

The subgradient of $\mathcal{F}(\mathbf{w},\boldsymbol{\theta})$ at $\boldsymbol{\theta}$ is clearly given by

$$\frac{\partial \mathcal{F}(\mathbf{w},\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} = \frac{\partial \mathcal{L}(\mathbf{w},\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}\,. \tag{9}$$

Finally, we use subgradient descent to update the variables $\mathbf{w}$ and $\boldsymbol{\theta}$ until the

objective function converges. We refer to this augmented Proportional Odds Model with both absolute and Relative information as POM-R.

Note that there is a related work by Sader et al. [20], proposing a method for ordinal classification that combines absolute evaluations from experts and relative evaluations from novices. The method is actually similar to the above augmented method, the only difference being that Sader et al. [20] uses L_1 regularization, which results in models that are simple and interpretable, whereas POM-R uses L_2 regularization, which allows to learn complex data patterns. Due to the similarity between both methods, we do not incorporate this related work in the experiments.

3.3. Support Vector Learning for Ordinal Regression with both types of information

The idea of Support Vector Learning for Ordinal Regression (SVLOR) [25] is to consider a utility function $U(\mathbf{x};\mathbf{w}) = \mathbf{w}^\top\mathbf{x}$ that maps objects from the input space to the real line, thus partitioning the real line in such a way that $U(\mathbf{x};\mathbf{w}) \in [\theta_{k-1}, \theta_k[$ if and only if $y = C_k$ with $\theta_0 = -\infty$ and $\theta_r = +\infty$.

The optimization problem is defined as:

$$\begin{aligned} \min_{\mathbf{w}} \quad & \mathcal{F}(\mathbf{w}) = C\sum_{i=1}^{n}\sum_{j=i+1}^{n}\xi_{i,j} + \frac{1}{2}\left\|\mathbf{w}\right\|_2^2 \\ \text{s.t.} \quad & z_{i,j}(\mathbf{w}^\top\mathbf{x}_i - \mathbf{w}^\top\mathbf{x}_j) \geq 1 - \xi_{i,j}\ , \\ & \xi_{i,j} \geq 0, \quad \forall i,j\ , \end{aligned} \tag{10}$$

where $C > 0$ is a trade-off parameter, $\xi_{i,j}$ are slack variables, $z_{i,j} = +1$ when $y_i \succ y_j$, $z_{i,j} = -1$ when $y_i \prec y_j$ and $z_{i,j} = 0$ when $y_i = y_j$.

After obtaining the optimal weighing vector $\mathbf{w}^*$, the threshold θ_k, $k \in \{1,\ldots,r-1\}$, is computed as

$$\theta_k = \frac{U(\mathbf{u}_k;\mathbf{w}^*) + U(\mathbf{v}_k;\mathbf{w}^*)}{2}\,, \tag{11}$$

where

$$(\mathbf{u}_k, \mathbf{v}_k) = \arg \min_{(i,j), y_i = C_{k+1}, y_j = C_k} \left[U(\mathbf{x}_i; \mathbf{w}^*) - U(\mathbf{x}_j; \mathbf{w}^*) \right],$$

which means that the threshold θ_k for the class label C_k lies in the middle of the utilities of the nearest examples of the k-th class and of the $(k+1)$-th class.

We augment the method in the same way as we did for the proportional odds model. The corresponding optimization problem is:

$$\begin{aligned} \min_{\mathbf{w}} \quad & \mathcal{F}(\mathbf{w}) = C \sum_{i=1}^{n} \sum_{j=i+1}^{n} \xi_{i,j} + \frac{\alpha}{m} \sum_{\ell=1}^{m} \eta_\ell + \frac{1}{2} \|\mathbf{w}\|_2^2 \\ \text{s.t.} \quad & z_{i,j}(\mathbf{w}^\top \mathbf{x}_i - \mathbf{w}^\top \mathbf{x}_j) \geq 1 - \xi_{i,j} \ , \\ & \xi_{i,j} \geq 0, \quad \forall i, j \, , \\ & \mathbf{w}^\top \mathbf{a}^\ell - \mathbf{w}^\top \mathbf{b}^\ell \geq 1 - \eta_\ell \, , \\ & \eta_\ell \geq 0, \quad \forall \ell \, , \end{aligned} \tag{12}$$

where m, α, $\|\mathbf{w}\|_2$ and η_ℓ are as in Eq. (5).

Rewriting the above optimization problem as:

$$\min_{\mathbf{w}} \mathcal{F}(\mathbf{w}) = C \sum_{i=1}^{n} \sum_{j=i+1}^{n} [1 - z_{i,j}(\mathbf{w}^\top \mathbf{x}_i - \mathbf{w}^\top \mathbf{x}_j)]_+ + \frac{\alpha}{m} \sum_{\ell=1}^{m} [1 + \mathbf{w}^\top \mathbf{b}^\ell - \mathbf{w}^\top \mathbf{a}^\ell]_+ + \frac{1}{2} \|\mathbf{w}\|_2^2 \ , \tag{13}$$

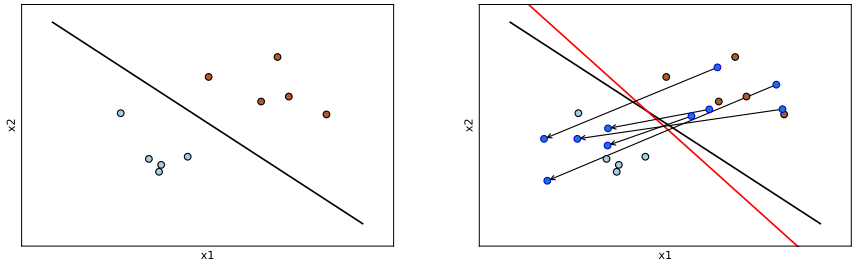
the subgradient of $\mathcal{F}$ at $\mathbf{w}$ is given by:

$$\frac{\partial \mathcal{F}(\mathbf{w})}{\partial \mathbf{w}} = C \sum_{i=1}^{n} \sum_{j=i+1}^{n} \phi_{ij}(-z_{i,j}(\mathbf{x}_i - \mathbf{x}_j)) + \frac{\alpha}{m} \sum_{\ell=1}^{m} \psi_{\ell k}(\mathbf{b}^\ell - \mathbf{a}^\ell) + \mathbf{w} \, , \tag{14}$$

where

$$\phi_{ij} = \begin{cases} 1 & , \quad \text{if } 1 - z_{i,j}(\mathbf{w}^\top \mathbf{x}_i - \mathbf{w}^\top \mathbf{x}_j) > 0 \, ; \\ 0 & , \quad \text{otherwise} \, . \end{cases} \tag{15}$$

The function $\psi_{\ell k}$ is computed as in Eq. (8). We also use subgradient descent to update $\mathbf{w}$ until the objective function converges. We refer to this Support Vector Learning model for Ordinal Regression with both absolute and Relative information as SVLOR-R.



(a) (b)

Figure 1: Example of incorporating absolute and relative information. (a) Best separating hyperplane (black line) based on absolute information only. (b) After considering relative information (each couple is represented by two blue points connected by an arrow pointing from the most preferred to the least preferred example), the original hyperplane (the black line) is replaced by a new hyperplane (the red line).

A graphical illustration is given in Figure 1. The left panel describes the classical support vector machine with absolute information only, while the right panel shows the augmented support vector machine with absolute and relative information with an updated hyperplane.

3.4. Support Vector Ordinal Regression with both types of information

The goal of Support Vector Ordinal Regression (SVOR) [26] is to find an optimal mapping direction $\mathbf{w}$ and $r-1$ thresholds, which determine $r-1$ parallel discriminant hyperplanes for separating the r classes. It is similar to the above-mentioned SVLOR, however, SVLOR automatically sets the thresholds by means of a utility function.

Two different types of thresholds are considered for exploiting the ordering among the class labels. One way is to consider EXplicit constraints on thresholds (this method is referred to as SVOREX). For each threshold θ_j, the empirical errors are computed for the examples with adjacent class labels C_j and C_{j+1}.

The optimization problem is formulated as follows:

$$
\begin{aligned}
\min_{\mathbf{w},\boldsymbol{\theta}} \quad & \mathcal{F}(\mathbf{w},\boldsymbol{\theta}) = C\sum_{j=1}^{r-1}(\sum_{i=1}^{n_j}\xi_i^j + \sum_{i=1}^{n_{j+1}}\xi_i^{*j+1}) + \frac{1}{2}\left\|\mathbf{w}\right\|_2^2 \\
\text{s.t.} \quad & \theta_j - \mathbf{w}^\top \mathbf{x}_i^j \geq 1 - \xi_i^j\,, \\
& \xi_i^j \geq 0, \quad \forall j = 1,\ldots,r-1, \quad \forall i = 1,\ldots,n_j\,, \\
& \mathbf{w}^\top \mathbf{x}_i^{j+1} - \theta_j \geq 1 - \xi_i^{*j+1}\,, \\
& \xi_i^{*j+1} \geq 0, \quad \forall j = 1,\ldots,r-1, \quad \forall i = 1,\ldots,n_{j+1}\,, \\
& \theta_{j-1} \leq \theta_j, \quad \forall j = 2,\ldots,r-1\,,
\end{aligned}
\tag{16}
$$

where $C > 0$ is a trade-off parameter, n_j is the number of examples of the j-th class, $\mathbf{x}_i^j$ is the i-th example of the j-th class, and ξ_i^j, ξ_i^{*j+1} are slack variables.

Similarly as above, we incorporate the additional relative information using the same constraints as in Eq. (4). The new optimization problem then becomes:

$$
\begin{aligned}
\min_{\mathbf{w},\boldsymbol{\theta}} \quad & \mathcal{F}(\mathbf{w},\boldsymbol{\theta}) = C\sum_{j=1}^{r-1}(\sum_{i=1}^{n_j}\xi_i^j + \sum_{i=1}^{n_{j+1}}\xi_i^{*j+1}) + \frac{\alpha}{m}\sum_{\ell=1}^{m}\eta_\ell + \frac{1}{2}\left\|\mathbf{w}\right\|_2^2 \\
\text{s.t.} \quad & \theta_j - \mathbf{w}^\top \mathbf{x}_i^j \geq 1 - \xi_i^j\,, \\
& \xi_i^j \geq 0, \quad \forall j = 1,\ldots,r-1, \quad \forall i = 1,\ldots,n_j\,, \\
& \mathbf{w}^\top \mathbf{x}_i^{j+1} - \theta_j \geq 1 - \xi_i^{*j+1}\,, \\
& \xi_i^{*j+1} \geq 0, \quad \forall j = 1,\ldots,r-1, \quad \forall i = 1,\ldots,n_{j+1}\,, \\
& \theta_{j-1} \leq \theta_j, \quad \forall j = 2,\ldots,r-1\,, \\
& \mathbf{w}^\top \mathbf{a}^\ell - \mathbf{w}^\top \mathbf{b}^\ell \geq 1 - \eta_\ell\,, \\
& \eta_\ell \geq 0, \quad \forall \ell = 1,\ldots,m\,,
\end{aligned}
\tag{17}
$$

where m, α, $\|\mathbf{w}\|_2$ and η_ℓ are as in Eq. (5).

The second way is to consider IMplicit constraints on thresholds (this method is referred to as SVORIM). Here, the examples of all the classes are incorporated in order to estimate the errors for all thresholds. The optimization problem is

formulated as follows:

$$
\begin{aligned}
\min_{\mathbf{w},\boldsymbol{\theta}} \quad & \mathcal{F}(\mathbf{w},\boldsymbol{\theta}) = C\sum_{j=1}^{r-1}(\sum_{k=1}^{j}\sum_{i=1}^{n_k}\xi_{ki}^{j} + \sum_{k=j+1}^{r}\sum_{i=1}^{n_k}\xi_{ki}^{*j}) + \frac{1}{2}\left\|\mathbf{w}\right\|_2^2 \\
\text{s.t.} \quad & \theta_j - \mathbf{w}^\top \mathbf{x}_i^k \geq 1 - \xi_{ki}^{j},\ \xi_{ki}^{j} \geq 0\,, \\
& \forall j = 1,\ldots,r-1, \quad \forall k = 1,\ldots,j, \quad \forall i = 1,\ldots,n_k\,, \\
& \mathbf{w}^\top \mathbf{x}_i^k - \theta_j \geq 1 - \xi_{ki}^{*j},\ \xi_{ki}^{*j} \geq 0\,, \\
& \forall j = 1,\ldots,r-1, \quad \forall k = j+1,\ldots,r, \quad \forall i = 1,\ldots,n_k\,,
\end{aligned}
\tag{18}
$$

where ξ_{ki}^{j}, ξ_{ki}^{*j} are slack variables and $C > 0$, n_j and $\mathbf{x}_i^j$ are as in Eq. (16).

After incorporating additional relative information, the new optimization problem becomes:

$$
\begin{aligned}
\min_{\mathbf{w},\boldsymbol{\theta}} \quad & \mathcal{F}(\mathbf{w},\boldsymbol{\theta}) = C\sum_{j=1}^{r-1}(\sum_{k=1}^{j}\sum_{i=1}^{n_k}\xi_{ki}^{j} + \sum_{k=j+1}^{r}\sum_{i=1}^{n_k}\xi_{ki}^{*j}) + \frac{\alpha}{m}\sum_{\ell=1}^{m}\eta_\ell + \frac{1}{2}\left\|\mathbf{w}\right\|_2^2 \\
\text{s.t.} \quad & \theta_j - \mathbf{w}^\top \mathbf{x}_i^k \geq 1 - \xi_{ki}^{j},\ \xi_{ki}^{j} \geq 0\,, \\
& \forall j = 1,\ldots,r-1, \quad \forall k = 1,\ldots,j, \quad \forall i = 1,\ldots,n_k\,, \\
& \mathbf{w}^\top \mathbf{x}_i^k - \theta_j \geq 1 - \xi_{ki}^{*j},\ \xi_{ki}^{*j} \geq 0\,, \\
& \forall j = 1,\ldots,r-1, \quad \forall k = j+1,\ldots,r, \quad \forall i = 1,\ldots,n_k\,. \\
& \mathbf{w}^\top \mathbf{a}^\ell - \mathbf{w}^\top \mathbf{b}^\ell \geq 1 - \eta_\ell\,, \\
& \eta_\ell \geq 0, \quad \forall \ell = 1,\ldots,m\,,
\end{aligned}
\tag{19}
$$

where m, α, $\|\mathbf{w}\|_2$ and η_ℓ are as in Eq. (5).

Similarly as for the preceding methods, we use subgradient descent to optimize the objective functions. We refer to Support Vector Ordinal Regression considering EXplicit (resp. IMplicit) constraints with both absolute and Relative information as SVOREX-R (resp. SVORIM-R).

3.5. Linear Discriminant Learning for Ordinal Regression with both types of information

The underlying idea of Linear Discriminant Learning for Ordinal Regression [27] is to project high-dimensional data onto a low-dimensional space, in such a way that this projection separates data from different classes as much as possible while respecting the ordering among the class labels. More specifically, the aim is to find a projection direction that not only minimizes the within-class distances and maximizes the between-class distances simultaneously, but also preserves the ordering among the class labels.

For ease of discussion, a within-class scatter matrix and a between-class scatter matrix are defined as follows:

$$S_w = \frac{1}{n} \sum_{k=1}^{r} \sum_{\substack{i=1 \\ y_i = C_k}}^{n} (\mathbf{x}_i - \mathbf{m}_k)(\mathbf{x}_i - \mathbf{m}_k)^\top , \tag{20}$$

and

$$S_b = \frac{1}{n} \sum_{k=1}^{r} n_k (\mathbf{m}_k - \mathbf{m})(\mathbf{m}_k - \mathbf{m})^\top , \tag{21}$$

where n_k is the number of examples of the k-th class, $\mathbf{m}_k = \frac{1}{n_k} \sum_{\substack{i=1 \\ y_i = C_k}}^{n} \mathbf{x}_i$ denotes the mean vector of the examples of the k-th class and $\mathbf{m} = \frac{1}{n} \sum_{i=1}^{n} \mathbf{x}_i$ represents the mean vector of all the examples. Linear discriminant learning is formalized as solving the following optimization problem:

$$\begin{aligned} \min_{\mathbf{w}, \rho} \quad & \mathbf{J}(\mathbf{w}, \rho) = \mathbf{w}^\top S_w \mathbf{w} - C\rho \\ \text{s.t.} \quad & \mathbf{w}^\top (\mathbf{m}_{k+1} - \mathbf{m}_k) \geq \rho, \quad \forall k = 1, 2, \ldots, r-1 , \end{aligned} \tag{22}$$

where C is a penalty coefficient and $\rho > 0$. This method aims to minimize the variance of the data points of the same class and separate the projected mean vectors of two neighboring classes. The projected mean vectors of all classes are expected to respect the ordering among the class labels in the projected data space. We refer to Linear Discriminant Learning for Ordinal Regression with

absolute information only as LDLOR.

Similarly as above, we incorporate the additional relative information using the same constraints as in Eq. (4). The new optimization problem becomes:

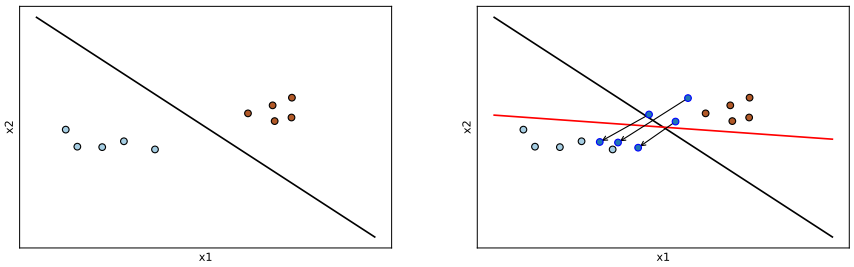
$$\begin{aligned} \min_{\mathbf{w},\rho} \quad & \mathbf{J}(\mathbf{w},\rho) = \mathbf{w}^\top S_w \mathbf{w} - C\rho + \frac{\alpha}{m}\sum_{\ell=1}^{m}\eta_\ell \\ \text{s.t.} \quad & \mathbf{w}^\top(\mathbf{m}_{k+1} - \mathbf{m}_k) \geq \rho, \quad \forall k = 1, 2, \ldots, r-1 \ , \\ & \mathbf{w}^\top \mathbf{a}^\ell - \mathbf{w}^\top \mathbf{b}^\ell \geq 1 - \eta_\ell \ , \\ & \eta_\ell \geq 0, \quad \forall \ell \ , \end{aligned} \tag{23}$$

where m, α and η_ℓ are as in Eq. (5).

We again use subgradient descent to minimize the objective function. After obtaining the optimal direction $\mathbf{w}$, the class label of a test example is predicted by the following decision rule:

$$f(\mathbf{x}) = \min\{k \in \{1, \ldots, r\} \mid \mathbf{w}^\top \mathbf{x} - b_k < 0\}\,, \tag{24}$$

where $b_k = \mathbf{w}^\top(\mathbf{m}_{k+1} + \mathbf{m}_k)/2$. We refer to Linear Discriminant Learning for Ordinal Regression with both absolute and Relative information as LDLOR-R.



(a) (b)

Figure 2: Example of incorporating absolute and relative information. (a) Best projection direction (black line) based on absolute information only. (b) After considering relative information, the original projection direction (the black line) is changed to a new direction (the red line).

A graphical illustration is given in Figure 2. The left panel describes the classical Linear Discriminant Learning for Ordinal Regression with absolute information only. The right panel shows the augmented Linear Discriminant Learning for Ordinal Regression with absolute and relative information with an updated projection direction.

3.6. Distance Metric Learning with both types of information

The aim of Distance Metric Learning [28] is to learn a distance metric that is adapted to the hidden structure of the input data. Here, we restrict the search to the family of Mahalanobis distance metrics. Formally, the squared Mahalanobis distance between two examples is defined as

$$d^2_{\mathbf{M}}(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i - \mathbf{x}_j)^\top \mathbf{M} (\mathbf{x}_i - \mathbf{x}_j) \ , \tag{25}$$

where $\mathbf{M} \succcurlyeq 0$ is a symmetric positive semidefinite (PSD) matrix. In order to learn the matrix $\mathbf{M}$, some constraints need to be set. These constraints are commonly represented in a pairwise [29] or triplet [30] form. Here, we exploit triplet constraints only.

For absolute information, the learned distance metric is expected to satisfy some natural constraints. Firstly, distances between examples of the same class should be small and distances between examples of different classes should be large. The corresponding triplets are given by

$$\mathcal{R}_{A_1} = \{(i, j, \ell) \in \{1, \ldots, n\}^3 \mid \mathbf{x}_j, \mathbf{x}_\ell \in \mathcal{N}(\mathbf{x}_i), y_i = y_j \neq y_\ell\} \ , \tag{26}$$

where $\mathcal{N}(\mathbf{x}_i)$ is the neighborhood of $\mathbf{x}_i$ containing its k nearest neighbor examples. Secondly, the ordering among the class labels needs to be preserved. In particular, the larger the difference between the class label of a given example and that of a neighbor example, the larger the distance between these two examples should be. For instance, in case $y_i \succ y_j \succ y_\ell$ or $y_i \prec y_j \prec y_\ell$, the distance between $\mathbf{x}_i$ and $\mathbf{x}_\ell$ should be larger than the distance between $\mathbf{x}_i$ and $\mathbf{x}_j$. The

corresponding triplets are given by

$$\mathcal{R}_{A_2} = \{(i,j,\ell) \in \{1,\ldots,n\}^3 \mid \mathbf{x}_j, \mathbf{x}_\ell \in \mathcal{N}(\mathbf{x}_i), (y_i \succ y_j \succ y_\ell) \vee (y_i \prec y_j \prec y_\ell)\}\,. \quad (27)$$

With each triplet (i,j,ℓ) in $\mathcal{R}_A = \mathcal{R}_{A_1} \cup \mathcal{R}_{A_2}$, an inequality constraint (incorporating a unit margin) is associated:

$$d^2_{\mathbf{M}}(\mathbf{x}_i, \mathbf{x}_j) + 1 \leq d^2_{\mathbf{M}}(\mathbf{x}_i, \mathbf{x}_\ell)\,. \quad (28)$$

Since our ultimate goal is to use an augmented version of the method of nearest neighbours that incorporates relative information, we also need to establish triplet constraints related to relative information. To that end, for any pair of examples $\mathbf{a}$ and $\mathbf{b}$, we consider both couples $(\mathbf{a},\mathbf{b})$ and $(\mathbf{b},\mathbf{a})$. In addition to the already-defined set $\mathcal{C}_\succ$, here we also consider the set $\mathcal{C}_\prec = \{(\mathbf{a}^{m+1},\mathbf{b}^{m+1}),\ldots,(\mathbf{a}^{2m},\mathbf{b}^{2m})\} = \{(\mathbf{b}^1,\mathbf{a}^1),\ldots,(\mathbf{b}^m,\mathbf{a}^m)\}$. Note that if a couple $(\mathbf{a},\mathbf{b})$ belongs to $\mathcal{C}_\succ$, then the couple $(\mathbf{b},\mathbf{a})$ belongs to $\mathcal{C}_\prec$. The entire set of input couples of examples is denoted by $\mathcal{C} = \mathcal{C}_\succ \cup \mathcal{C}_\prec = \{(\mathbf{a}^1,\mathbf{b}^1),\ldots,(\mathbf{a}^{2m},\mathbf{b}^{2m})\}$.

The Mahalanobis distance between couples of examples is defined as a special case of the product distance (see [31]):

$$d_{*\mathbf{M}}((\mathbf{u},\mathbf{v}),(\mathbf{w},\mathbf{t})) = d_{\mathbf{M}}(\mathbf{u},\mathbf{w}) + d_{\mathbf{M}}(\mathbf{v},\mathbf{t})\,. \quad (29)$$

The learned distance metric should satisfy that the distances between any given couple and the couples in its neighborhood with the same preference order are smaller than the distances between that couple and the couples in its neighborhood with the opposite preference order, where the neighborhood is considered to reduce the number of constraints. The corresponding triplets are given by

$$\mathcal{R}_R = \{(p,q,t) \in \{1,\ldots,m\}^2 \times \{m+1,\ldots,2m\} \mid (\mathbf{a}^q,\mathbf{b}^q),(\mathbf{a}^t,\mathbf{b}^t) \in \mathcal{N}((\mathbf{a}^p,\mathbf{b}^p))\}\,, \quad (30)$$

where $\mathcal{N}((\mathbf{a}^p,\mathbf{b}^p))$ is the neighborhood of the couple $(\mathbf{a}^p,\mathbf{b}^p)$. With each triplet

$(p, q, t) \in \mathcal{R}_R$, an inequality constraint (incorporating a unit margin) is associated:

$$d^2_{*\mathbf{M}}((\mathbf{a}^p, \mathbf{b}^p), (\mathbf{a}^q, \mathbf{b}^q)) + 1 \leq d^2_{*\mathbf{M}}((\mathbf{a}^p, \mathbf{b}^p), (\mathbf{a}^t, \mathbf{b}^t)) \,, \tag{31}$$

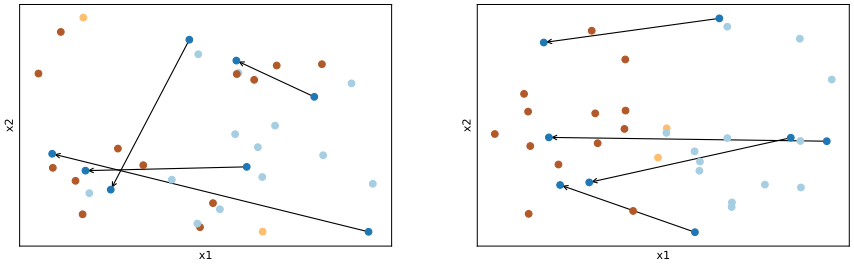
where we replace $d^2_{*\mathbf{M}}((\mathbf{a}^s, \mathbf{b}^s), (\mathbf{a}^r, \mathbf{b}^r))$ by $d^2_{\mathbf{M}}(\mathbf{a}^s, \mathbf{a}^r) + d^2_{\mathbf{M}}(\mathbf{b}^s, \mathbf{b}^r)$ to reduce the computational complexity.

As it might be difficult to satisfy all the constraints, a soft margin is considered to tolerate some violations. The proposed distance metric learning method with both absolute and relative information then corresponds to the following optimization problem:

$$\begin{aligned}
\min_{\mathbf{M},\xi,\eta} \quad & f_L(\mathbf{M}) = \lambda \mathrm{Tr}(\mathbf{M}) + \frac{\alpha}{|\mathcal{R}_A|} \sum_{(i,j,\ell)\in\mathcal{R}_A} \xi_{ij\ell} + \frac{\beta}{|\mathcal{R}_R|} \sum_{(p,q,t)\in\mathcal{R}_R} \eta_{pqt} \\
\text{s.t.} \quad & d^2_{\mathbf{M}}(\mathbf{x}_i, \mathbf{x}_\ell) - d^2_{\mathbf{M}}(\mathbf{x}_i, \mathbf{x}_j) \geq 1 - \xi_{ij\ell} \\
& d^2_{*\mathbf{M}}((\mathbf{a}^p, \mathbf{b}^p), (\mathbf{a}^t, \mathbf{b}^t)) - d^2_{*\mathbf{M}}((\mathbf{a}^p, \mathbf{b}^p), (\mathbf{a}^q, \mathbf{b}^q)) \geq 1 - \eta_{pqt} \\
& \xi_{ij\ell} \geq 0, \quad \forall (i,j,\ell) \in \mathcal{R}_A \\
& \eta_{pqt} \geq 0, \quad \forall (p,q,t) \in \mathcal{R}_R \\
& \mathbf{M} \succcurlyeq 0 \,,
\end{aligned} \tag{32}$$

where $\lambda \geq 0$ is a trade-off parameter, α is a parameter to control the impact of the absolute information, β is a parameter to control the impact of the relative information, $|\mathcal{R}_A|$ is the number of constraints in $\mathcal{R}_A$ and $|\mathcal{R}_R|$ is the number of constraints in $\mathcal{R}_R$, and $\xi_{ij\ell}$, η_{pqt} are slack variables. More details can be found in [22]. A graphical illustration is given in Figure 3. The left panel shows the original data points in the Euclidean space. The right panel shows the data points in the Mahalanobis space with the learned Mahalanobis distance metric.

Ultimately, the learned Mahalanobis distance metric is used for replacing the Euclidean distance metric in either the classical k-NN or the augmented version of k-NN presented in [21], depending on whether only absolute or both absolute and relative information is involved. We refer to Distance Metric Learning with only absolute information within the classical k-NN as DMLNN and to



(a) (b)

Figure 3: Example of incorporating absolute and relative information. The ordering among the class labels is $red \succ yellow \succ blue$. (a) The original data distribution in the Euclidean space. (b) The new data distribution in the Mahalanobis space with the learned Mahalanobis distance metric for absolute and relative information.

Distance Metric Learning with both absolute and Relative information within the augmented k-NN as DMLNN-R.

4. Experiments

4.1. Datasets

We perform extensive experiments on selected datasets stemming from real ordinal classification problems as well as datasets obtained by discretizing standard regression problems. Datasets of the first type were downloaded from repositories such as the UCI (University of California, Irvine) machine learning repository [32] and mldata.org [33], while datasets of the second type were provided by Chu [34]. As real-life classification datasets usually need to be collected by experts, their size is typically small. Datasets from discretized regression problems are larger, as they are typically obtained by discretizing the response variables into ordinal classes of the same cardinality. Table 1 describes the characteristics of the datasets used, including the number of examples, features, classes and class distribution. All the features have been standardized (i.e. to have zero mean and unit standard deviation) to avoid the impact of the scale of features. In all experiments, we use ten-fold cross-validation to compute the performance.

Table 1: Description of the benchmark datasets.

Dataset	#Examples	#Features	#Classes	#Class distribution
Real ordinal classification datasets				
Toy (TO)	300	2	5	(35, 87, 79, 68, 31)
Balance-scale (BS)	625	4	3	(288, 49, 288)
Eucalyptus (EU)	736	91	5	(180, 107, 130, 214, 105)
Swd (SW)	1000	10	4	(32, 352, 399, 217)
Lev (LE)	1000	4	5	(93, 280, 403, 197, 27)
Winequality-red (WR)	1599	11	6	(10, 53, 681, 638, 199, 18)
Car (CA)	1728	21	4	(210, 384, 69, 65)
Discretized regression datasets				
Housing5 (HO5)	506	14	5	$\approx$ 101 per class
Abalone5 (AB5)	4177	11	5	$\approx$ 836 per class
Bank1-5 (BA1-5)	8192	8	5	$\approx$ 1639 per class
Bank2-5 (BA2-5)	8192	32	5	$\approx$ 1639 per class
Computer1-5 (CO1-5)	8192	12	5	$\approx$ 1639 per class
Computer2-5 (CO2-5)	8192	21	5	$\approx$ 1639 per class
Housing10 (HO10)	506	14	10	$\approx$ 51 per class
Abalone10 (AB10)	4177	11	10	$\approx$ 418 per class
Bank1-10 (BA1-10)	8192	8	10	$\approx$ 820 per class
Bank2-10 (BA2-10)	8192	32	10	$\approx$ 820 per class
Computer1-10 (CO1-10)	8192	12	10	$\approx$ 820 per class
Computer2-10 (CO2-10)	8192	21	10	$\approx$ 820 per class

Note that none of these datasets contains relative information. In order to compare the augmented methods, we simulate our problem setting by generating relative information from the available absolute information (see Figure 4). For each dataset, we initially divide all labelled examples into ten folds, represented by the ten rows in Figure 4 (left). As in classical cross-validation, we keep one fold for testing (the red part) and use the remaining folds for training. The folds used for training are further split to generate absolute and relative information. More specifically, we randomly select 5% of the labelled examples in the remaining folds and keep them unchanged as absolute information (the yellow part) and use the remaining 95% of the labelled examples (the blue part) to generate relative information (the green part) by transforming the class labels into preference orders between examples. For this purpose, we divide the labelled examples in the blue part into ten parts of equal size (the ten orange parts). We then pairwisely compare the class labels of the ten examples (one from each orange part) in each row to obtain the corresponding preference orders. For more details on the process of generating relative information, we refer to our previous work [21]. In summary, we construct two datasets for each

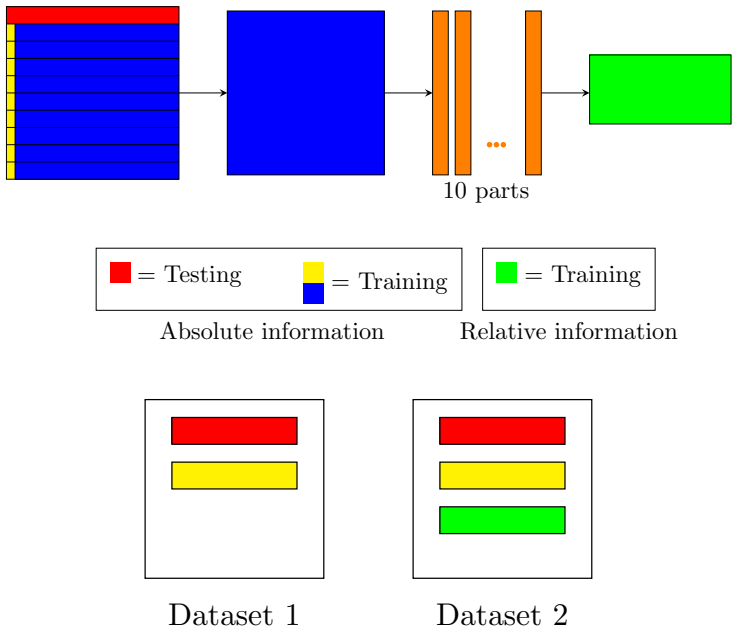


Figure 4: Process of generating relative information from absolute information.

original dataset as schematically shown in Figure 4.

4.2. Performance measures

There exists a variety of performance measures used for evaluating ordinal classification methods [25, 35, 36]. Here, we choose the three most common ones: the Mean Zero-one Error (MZE), the Mean Absolute Error (MAE) and the C-index. The MZE describes the error rate of the classifier computed as

$$\mathrm{MZE} = \frac{1}{T}\sum_{i=1}^{T}\delta(y_i^* \neq y_i) = 1 - \mathrm{Acc}\ , \tag{33}$$

where T is the number of test examples, y_i is the real class label and y_i^* is the predicted class label; Acc is the accuracy of the classifier. The value of MZE ranges from 0 to 1. It describes the global performance, but neglects the ordering among the class labels.

The MAE is the mean absolute error between y_i and y_i^* and is computed as

$$\text{MAE} = \frac{1}{T}\sum_{i=1}^{T}|y_i - y_i^*| \ . \tag{34}$$

The value of MAE ranges from 0 to $r-1$ (maximum absolute error between classes). Because the real distances among the class labels are unknown, the numerical representation of the class labels has a considerable impact on the MAE.

One way to avoid this impact is to consider the relations between the real class label and the predicted class label. Here, we use the concordance index or C-index to represent these relations. The C-index is computed as the proportion of concordant pairs among the comparable pairs (see [37], page 50):

$$\text{C-index} = \frac{1}{\sum_{C_p \prec C_q} T_{C_p} T_{C_q}} \sum_{y_i \prec y_j} \left(\delta(y_i^* \prec y_j^*) + \frac{1}{2}\delta(y_i^* = y_j^*)\right) \ , \tag{35}$$

where T_{C_p} and T_{C_q} are the numbers of test examples with class label C_p and C_q, y_i and y_j are the ordinal class labels and y_i^* and y_j^* the predicted class labels. When there are only two different class labels, the C-index reduces to the area under the Receiver Operating Characteristic (ROC) curve [38]. It is important to mention that the C-index takes no account of the numerical representation of the class labels and is therefore more suitable than the MZE and MAE.

A lower MZE, lower MAE or a higher C-index indicated a better performance. Here, in order to facilitate the discussion of the results and to preserve the analogy with the other evaluation metrics, we replace C-index by $1-$ C-index. In this way, a lower MZE, MAE or $1-$ C-index represents a better performance.

4.3. Experimental settings

Here, we employ the methods mentioned in Section 3 and refer to them as follows:

POM: Proportional Odds Model with absolute information only;

POM-R: Proportional Odds Model with absolute and Relative information;

SVLOR: Support Vector Learning for Ordinal Regression with absolute information only;

SVLOR-R: Support Vector Learning for Ordinal Regression with absolute and Relative information;

SVOREX: Support Vector Ordinal Regression considering EXplicit constraints with absolute information only;

SVOREX-R: Support Vector Ordinal regression considering EXplicit constraints with absolute and Relative information;

SVORIM: Support Vector Ordinal Regression considering IMplicit constraints with absolute information only;

SVORIM-R: Support Vector Ordinal Regression considering IMplicit constraints with absolute and Relative information;

LDLOR: Linear Discriminant Learning for Ordinal Regression with absolute information only;

LDLOR-R: Linear Discriminant Learning for Ordinal Regression with absolute and Relative information;

DMLNN: Distance Metric Learning with only absolute information;

DMLNN-R: Distance Metric Learning with absolute and Relative information.

For the classical and augmented machine learning methods, the parameters α, λ and C are tuned in the set $\{10^{-3}, \ldots, 10^{3}\}$ by three-fold cross-validation based on the three performance measures. For the distance metric learning methods, we set $\lambda = 10^{-4}$ and $\alpha = \beta = 1$.

4.4. Overall performance analysis

In this subsection, we analyze the performance of the different machine learning methods discussed in this paper. The experimental results shown in Tables 2–4 have been obtained using ten-fold cross-validation. From these tables, several conclusions can be drawn.

Not surprisingly, considering both absolute and relative information leads to a better performance than considering absolute information only. For example,

row "Num" in Table 2 (in terms of MZE) shows that POM-R outperforms POM on 15 out of 19 datasets, while SVLOR-R outperforms SVLOR on 18 out of 19 datasets. Looking at the results per dataset, column "Num" shows that 4 out of 6 augmented methods perform better than the corresponding original ones on the TO dataset, while all augmented methods perform better on the BS dataset. Table 3 (in terms of MAE) and Table 4 (in terms of 1 - C-index) can be read in the same way. Globally, these tables show that the augmented methods perform better than the corresponding original ones on most of the datasets, while on each of the datasets, most of the augmented methods outperform the corresponding original ones.

For a more comprehensive view of the improvement obtained by incorporating relative information, Figures 5 and 6 show the absolute and relative differences in performance in terms of the MZE, MAE and $1 - \text{C-index}$ for the augmented methods compared to the original ones. The greener the color (*i.e.*, the more negative the values), the larger the improvement. SVLOR and LDLOR deserve special attention as they greatly improve their performance on most datasets when considering additional relative information. A further look at Tables 2–4 shows that the performance of SVLOR and LDLOR is subpar compared to the other methods on several datasets only consisting of absolute information. However, the expected performance is attained when the relative information is incorporated.

Finally, we test whether there is an overall significant difference in performance among the different methods [39]. More specifically, we apply the non-parametric Friedman test [40] at a significance level of $\alpha = 0.05$. The results are shown in the last row of Tables 2–4. It can be seen that all p-values for the three performance measures are smaller than α, which means that at least two machine learning methods behave differently on average and that there is an overall statistically significant difference between the performances of at least two original methods and between those of at least two augmented ones. Post-hoc analyses are not here shown, but, instead, all pairwise comparisons are performed.

Table 2: MZE for only absolute information or for both absolute and relative information. The best results are highlighted in boldface. Column Num represents the number of augmented methods that outperform the original ones, while row Num represents the number of datasets on which the augmented methods outperform the original ones.

Dataset	Only absolute information						Absolute and relative information						Num
	POM	SVLOR	SVOREX	SVORIM	LDLOR	DMLNN	POM-R	SVLOR-R	SVOREX-R	SVORIM-R	LDLOR-R	DMLNN-R	
TO	0.7267	0.7318	0.6729	0.7398	0.7273	**0.6202**	0.7241	0.7320	0.7138	0.7372	0.6745	**0.4355**	**4 / 6**
BS	0.1858	0.2048	0.6831	**0.1665**	0.2908	0.1807	0.1422	0.1058	**0.1026**	0.1041	0.2745	0.1489	**6 / 6**
EU	0.6153	0.5809	0.5910	0.5868	**0.5610**	0.5952	0.4629	**0.4090**	0.5607	0.5360	0.4122	0.5106	**6 / 6**
SW	0.5721	0.5629	0.5449	**0.5149**	0.5779	0.5282	0.5689	0.4970	**0.4838**	0.4901	0.5744	0.4993	**6 / 6**
LE	0.5448	0.5186	0.5233	**0.4317**	0.5274	0.4686	0.5349	0.4943	**0.3989**	0.4248	0.5457	0.4514	**5 / 6**
WR	0.5915	0.6106	0.5563	**0.4937**	0.6545	0.4954	0.5802	0.5788	0.5578	0.5289	0.6871	**0.4655**	**3 / 6**
CA	0.1684	0.1887	0.1585	**0.1511**	0.3629	0.1575	0.1430	**0.0833**	0.1354	0.1226	0.2117	0.0995	**6 / 6**
HO5	0.5134	0.5396	0.5012	**0.4750**	0.4776	0.5036	0.4565	0.4838	0.4787	0.4723	0.4479	**0.4330**	**6 / 6**
AB5	0.5909	0.6627	**0.5602**	0.5653	0.5854	0.6258	0.5883	0.5786	0.5559	0.5667	**0.5329**	0.6254	**5 / 6**
BA1-5	0.3262	0.6438	0.2175	**0.2174**	0.2549	0.2821	0.2964	0.2327	**0.2191**	0.2196	0.2340	0.2987	**3 / 6**
BA2-5	**0.5303**	0.7318	0.5615	0.5549	0.5869	0.6317	**0.5234**	0.5697	0.5511	0.5582	0.5297	0.6393	**4 / 6**
CO1-5	0.4004	0.6211	**0.3761**	0.3784	0.4215	0.4105	0.4081	0.4236	0.3781	0.3744	**0.3535**	0.3955	**4 / 6**
CO2-5	0.5737	0.4882	0.3265	**0.3236**	0.3788	0.3415	0.5737	0.3939	0.3276	0.3279	**0.2962**	0.3373	**3 / 6**
HO10	0.7508	0.7576	0.7881	0.7937	0.7444	**0.7408**	0.7488	0.7332	0.7541	0.7664	0.6996	**0.6801**	**6 / 6**
AB10	0.7728	0.8226	**0.7411**	0.7631	0.7595	0.7810	0.7707	0.7551	0.7368	0.7571	**0.7332**	0.8006	**5 / 6**
BA1-10	0.5863	0.8090	0.4289	**0.4280**	0.4452	0.4981	0.5618	0.4569	0.4298	**0.4280**	0.4343	0.5337	**3 / 6**
BA2-10	**0.7736**	0.8561	0.7500	0.7571	0.7760	0.7963	0.7677	0.7581	0.7475	0.7612	**0.7389**	0.8034	**4 / 6**
CO1-10	**0.3790**	0.7433	0.5696	0.5801	0.6166	0.6004	**0.3806**	0.6387	0.5800	0.5790	0.5512	0.5896	**4 / 6**
CO2-10	**0.5302**	0.7166	0.5321	0.5342	0.5969	0.5555	0.5321	0.6599	0.5340	0.5340	**0.5014**	0.5404	**4 / 6**
Num							**15 / 19**	**18 / 19**	**11 / 19**	**12 / 19**	**17 / 19**	**14 / 19**	
Average	0.5333	0.6206	0.5307	**0.4976**	0.5445	0.5165	0.5139	0.5044	**0.4866**	0.4889	0.4965	0.4888	
p-value	0.00017						0.00166						

Table 3: MAE for only absolute information or for both absolute and relative information. The best results are highlighted in boldface. Column Num represents the number of augmented methods that outperform the original ones, while row Num represents the number of datasets on which the augmented methods outperform the original ones.

Dataset	Only absolute information						Absolute and relative information						Num
	POM	SVLOR	SVOREX	SVORIM	LDLOR	DMLNN	POM-R	SVLOR-R	SVOREX-R	SVORIM-R	LDLOR-R	DMLNN-R	
TO	1.0575	1.4748	1.0761	0.9905	1.3907	**0.9738**	0.9969	1.4904	0.9971	1.0358	1.1339	**0.6104**	**4 / 6**
BS	0.2386	0.2641	0.6960	**0.2161**	0.3019	0.2321	0.1757	0.1365	**0.1219**	0.1345	0.2888	0.1794	**6 / 6**
EU	0.8230	0.9153	0.9254	0.8895	**0.7535**	0.9013	0.5366	0.4810	0.8318	0.7401	**0.4546**	0.6487	**6 / 6**
SW	0.6120	0.6467	0.5869	**0.5589**	0.7170	0.6102	0.6099	0.5619	**0.5097**	0.5171	0.7381	0.5714	**5 / 6**
LE	0.6146	0.5825	0.6094	**0.4696**	0.6365	0.5423	0.6038	0.5721	**0.4338**	0.4637	0.6970	0.5044	**5 / 6**
WR	0.6742	0.8175	0.6571	0.5757	0.9121	**0.5705**	0.6573	0.7709	0.6644	0.6071	0.9695	**0.5312**	**3 / 6**
CA	0.1817	0.2251	0.1713	**0.1603**	0.4614	0.1801	0.1529	**0.0856**	0.1453	0.1365	0.2146	0.1077	**6 / 6**
HO5	0.6834	0.7780	0.6877	**0.5957**	0.6022	0.6525	0.5913	0.6257	0.6534	0.5909	0.5988	**0.5398**	**6 / 6**
AB5	0.7851	1.2791	0.8365	**0.7505**	0.8649	0.9574	0.7647	0.8686	0.8134	0.7544	**0.7355**	0.9720	**4 / 6**
BA1-5	0.3290	1.0814	0.2195	**0.2193**	0.2580	0.2874	0.2992	0.2341	**0.2212**	0.2217	0.2357	0.3092	**3 / 6**
BA2-5	**0.6089**	1.4453	0.7654	0.7241	0.8291	0.9125	**0.5968**	0.7997	0.7304	0.7119	0.6835	0.9269	**5 / 6**
CO1-5	0.4769	0.9735	0.4241	**0.4192**	0.5050	0.4794	0.4843	0.4860	0.4302	0.4146	**0.3943**	0.4581	**4 / 6**
CO2-5	0.8847	0.5905	0.3621	**0.3541**	0.4304	0.3790	0.8783	0.4324	0.3644	0.3574	**0.3159**	0.3746	**4 / 6**
HO10	**1.4927**	1.8121	1.5298	1.5954	1.6796	1.5446	1.5167	1.4296	1.4196	1.3700	1.3454	**1.2203**	**5 / 6**
AB10	1.5535	2.8641	1.8742	**1.5516**	1.8898	1.9594	**1.5421**	1.7983	1.7313	1.5554	1.6577	2.0882	**4 / 6**
BA1-10	0.7495	2.3995	0.4722	**0.4699**	0.5008	0.5824	0.6972	0.5185	0.4731	**0.4699**	0.4745	0.6600	**3 / 6**
BA2-10	1.5206	3.0915	1.6341	**1.5001**	1.8362	1.8936	1.4874	1.6173	1.6083	**1.4835**	1.5035	1.9286	**5 / 6**
CO1-10	**0.4349**	1.9847	0.8762	0.8513	1.0531	0.9869	**0.4389**	1.0500	0.9060	0.8444	0.8116	0.9645	**4 / 6**
CO2-10	0.7438	1.4444	0.7442	**0.7222**	0.9286	0.8260	0.7371	1.0339	0.7467	0.7209	**0.6752**	0.7947	**5 / 6**
Num							**16 / 19**	**18 / 19**	**12 / 19**	**12 / 19**	**16 / 19**	**13 / 19**	
Average	0.7613	1.2984	0.7973	**0.7165**	0.8711	0.8143	0.7246	0.7891	0.7264	**0.6910**	0.7331	0.7574	
p-value	0.0						8e-05						

Table 4: 1 - C-index for only absolute information or for both absolute and relative information. The best results are highlighted in boldface. Column Num represents the number of augmented methods that outperform the original ones, while row Num represents the number of datasets on which the augmented methods outperform the original ones.

Dataset	Only absolute information						Absolute and relative information						Num
	POM	SVLOR	SVOREX	SVORIM	LDLOR	DMLNN	POM-R	SVLOR-R	SVOREX-R	SVORIM-R	LDLOR-R	DMLNN-R	
TO	0.4932	0.4961	0.4818	0.5067	0.4789	**0.3770**	0.4628	0.4673	0.4858	0.5041	0.3811	**0.2178**	**5 / 6**
BS	0.1036	0.1160	0.3719	0.1003	**0.0881**	0.1057	0.0717	0.0708	**0.0543**	0.0661	0.0858	0.0791	**6 / 6**
EU	0.3105	0.2712	0.2897	0.2801	**0.2028**	0.2705	0.1723	0.1355	0.2556	0.2211	**0.1100**	0.1837	**6 / 6**
SW	0.4243	**0.2883**	0.3973	0.3615	0.3110	0.3435	0.3980	**0.2479**	0.3204	0.3348	0.3230	0.3021	**5 / 6**
LE	0.2262	0.2038	0.3724	0.2042	**0.1849**	0.2407	0.2369	**0.1856**	0.1972	0.2082	0.2127	0.2140	**3 / 6**
WR	0.4191	**0.2728**	0.2829	0.2942	0.2906	0.3249	0.4191	**0.2699**	0.2889	0.3174	0.2802	0.2827	**3 / 6**
CA	0.1713	0.1433	**0.0858**	0.0891	0.1943	0.1035	0.1722	**0.0538**	0.0765	0.0799	0.0625	0.0603	**5 / 6**
HO5	0.1744	0.2125	0.1793	**0.1533**	0.1586	0.1687	0.1547	0.1595	0.1701	0.1576	0.1516	**0.1358**	**5 / 6**
AB5	0.2087	0.3157	0.2245	**0.2084**	0.2338	0.2774	0.2020	0.2230	0.2179	0.2104	**0.1935**	0.2822	**4 / 6**
BA1-5	0.0659	0.2839	**0.0523**	**0.0523**	0.0612	0.0677	0.0605	0.0554	**0.0527**	0.0528	0.0562	0.0736	**3 / 6**
BA2-5	0.2038	0.3952	0.2034	**0.1956**	0.2270	0.2618	0.1853	0.2030	0.1945	0.1929	**0.1783**	0.2658	**5 / 6**
CO1-5	**0.1034**	0.2095	0.1062	0.1049	0.1297	0.1228	0.1049	0.1179	0.1089	0.1039	**0.0987**	0.1165	**4 / 6**
CO2-5	0.0914	0.1337	0.0905	**0.0887**	0.1089	0.0951	0.0928	0.1003	0.0915	0.0891	**0.0776**	0.0939	**3 / 6**
HO10	**0.1874**	0.2250	0.2046	0.2248	0.2258	0.2183	0.2024	0.1750	0.1771	0.1809	0.1742	**0.1619**	**5 / 6**
AB10	**0.2116**	0.3175	0.2377	0.2161	0.2368	0.2830	0.2103	0.2174	0.2250	0.2159	**0.2072**	0.2991	**5 / 6**
BA1-10	0.0554	0.3015	0.0507	**0.0503**	0.0545	0.0641	0.0540	0.0554	0.0507	**0.0503**	0.0506	0.0746	**2 / 6**
BA2-10	0.2060	0.3829	0.2105	**0.2012**	0.2404	0.2706	0.1997	0.2035	0.2079	0.1993	**0.1922**	0.2754	**5 / 6**
CO1-10	**0.1031**	0.2098	0.1078	0.1051	0.1322	0.1274	0.1011	0.1251	0.1113	0.1041	**0.0995**	0.1244	**5 / 6**
CO2-10	**0.0814**	0.1496	0.0889	0.0863	0.1136	0.1030	**0.0795**	0.1087	0.0891	0.0860	0.0806	0.0991	**5 / 6**
Num							**13 / 19**	**19 / 19**	**11 / 19**	**12 / 19**	**17 / 19**	**13 / 19**	
Average	0.2021	0.2594	0.2125	**0.1854**	0.1933	0.2014	0.1884	0.1671	0.1777	0.1776	**0.1587**	0.1759	
p-value	0.009						0.00019						

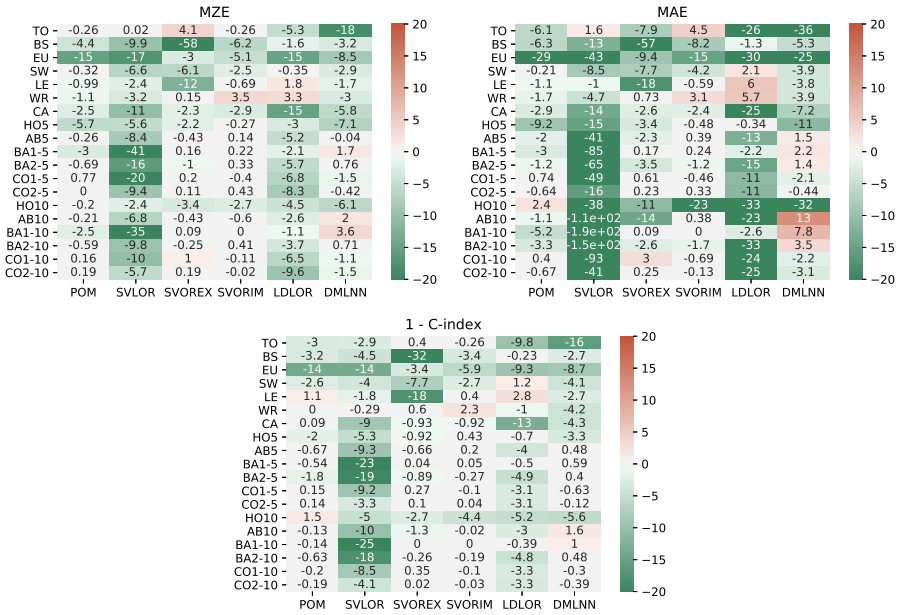


Figure 5: Absolute difference in performance in terms of MZE, MAE and 1 - C-index for the augmented methods compared to the original ones on all datasets. The values are recorded in percentage.

More specifically, in order to detect whether there is a significant difference in performance between every two of the augmented methods, we perform the Wilcoxon unilateral signed-rank test [41] at a significance level of $\alpha = 0.05$. The results are shown in Table 5. We conclude that for the $1 - \text{C-index}$ there is statistical evidence that SVLOR-R performs better than DMLNN-R, and that LDLOR-R performs better than the other augmented methods, except for SVLOR-R.

In addition, in order to detect whether there is a significant difference in performance between an augmented method and the corresponding original one, we compute the median difference and perform the Wilcoxon unilateral signed-rank test at a significance level of $\alpha = 0.05$. Table 7 shows that, except for the p-values for the MZE and $1 - \text{C-index}$ for the methods SVORIM and SVORIM-R, all other p-values are smaller than α, which means that there is a statistically significant difference between the performances. Overall, the median differences

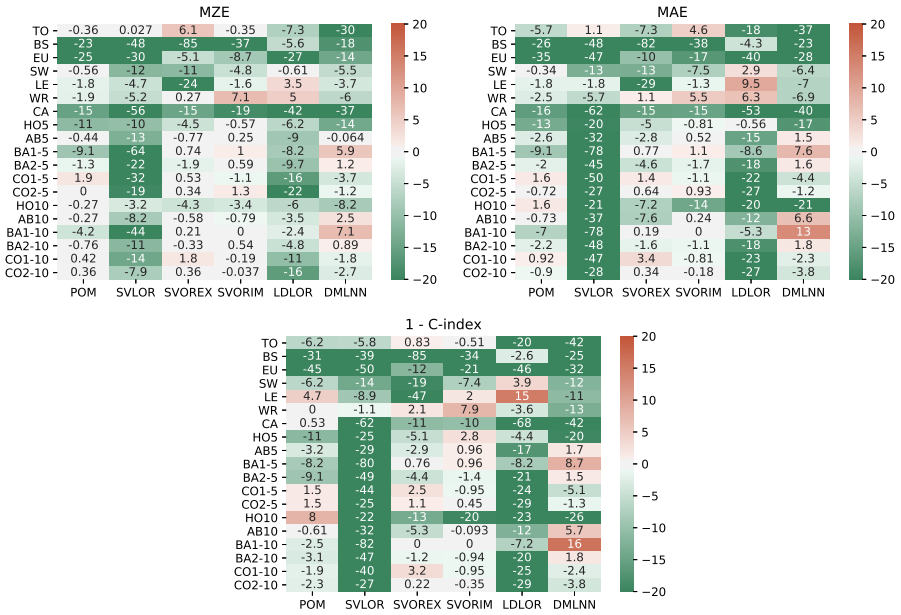


Figure 6: Relative difference in performance in terms of MZE, MAE and 1 - C-index for the augmented methods compared to the original ones on all datasets. The values are recorded in percentage.

in Table 6 and the p-values in Table 7 show that there is statistical evidence that all the augmented methods outperform the original ones for the three performance measures.

4.5. Performance analysis from the point of view of cost and information content

In this subsection, we start by illustrating the obvious positive impact of an increasing amount of relative information on the performance of the augmented methods. Note that one can clearly expect the act of labelling to be more expensive than that of assigning a preference order, while one can also expect a labelled example to be more informative than a preference-ordered couple. Hence, it seems worthwhile to explore the relationship between performance and cost, where cost refers to the actual cost of data collection or to the amount of information provided (measured in terms of entropy). For the sake of brevity, we restrict our attention in this section to the $1 - \text{C-index}$ (as it is the most

Table 5: p-values according to the Wilcoxon unilateral test among the augmented methods based on the three performance measures. p-values smaller than 0.05 are highlighted in boldface.

Method		MZE	MAE	1 − C-index
POM-R	SVLOR-R	0.90793	0.16707	0.65631
	SVOREX-R	0.96492	0.5	0.5954
	SVORIM-R	0.94173	0.93175	0.77175
	LDLOR-R	0.83293	0.27304	0.99958
	DMLNN-R	0.80097	0.42027	0.51605
SVLOR-R	POM-R	0.09207	0.83293	0.34369
	SVOREX-R	0.98353	0.9937	0.22225
	SVORIM-R	0.96169	0.99812	0.34369
	LDLOR-R	0.75305	0.87008	0.94405
	DMLNN-R	0.62625	0.5	**0.03508**
SVOREX-R	POM-R	**0.03508**	0.5	0.4046
	SVLOR-R	**0.01647**	**0.0063**	0.77775
	SVORIM-R	0.22299	0.99438	0.54805
	LDLOR-R	0.62625	0.57973	0.99295
	DMLNN-R	0.17733	0.17733	0.18799
SVORIM-R	POM-R	0.05827	0.06825	0.22825
	SVLOR-R	**0.03831**	**0.00188**	0.65631
	SVOREX-R	0.77701	**0.00562**	0.45195
	LDLOR-R	0.65631	0.18799	0.99835
	DMLNN-R	0.17733	0.06825	0.28658
LDLOR-R	POM-R	0.16707	0.72696	**0.00042**
	SVLOR-R	0.24695	0.12992	0.05595
	SVOREX-R	0.37375	0.42027	**0.00705**
	SVORIM-R	0.34369	0.81201	**0.00165**
	DMLNN-R	0.5	0.27304	**0.0267**
DMLNN-R	POM-R	0.19903	0.57973	0.48395
	SVLOR-R	0.37375	0.5	0.96492
	SVOREX-R	0.82267	0.82267	0.81201
	SVORIM-R	0.82267	0.93175	0.71342
	LDLOR-R	0.5	0.72696	0.9733

suitable performance measure and results are similar for MZE and MAE).

4.5.1. Impact of the amount of relative information

In Figure 7, we illustrate the impact of the amount of relative information on the performance on the EU and BS datasets, by fixing the absolute information and continuously adding couples to the already present ones. As expected, the performance levels off after an initial improvement. More specifically, on the EU

Table 6: Median difference between each augmented method and the corresponding original one based on the three different performance measures.

Test	Methods	MZE	MAE	$1 - $ C-index
Median difference	POM-R and POM	0.00590	0.01690	0.00200
	SVLOR-R and SVLOR	0.09430	0.41050	0.08470
	SVOREX-R and SVOREX	0.00430	0.02600	0.00660
	SVORIM-R and SVORIM	0.00260	0.00480	0.00100
	LDLOR-R and LDLOR	0.04480	0.12940	0.03130
	DMLNN-R and DMLNN	0.0151	0.03130	0.00630

Table 7: p-value according to the Wilcoxon unilateral test for each augmented method and the corresponding original one based on the three different performance measures.

Test	Methods	MZE	MAE	$1 - $ C-index
Wilcoxon	POM-R and POM	0.00080	0.00097	0.01049
	SVLOR-R and SVLOR	7.75e-05	9.09e-05	6.59e-05
	SVOREX-R and SVOREX	0.02796	0.00500	0.00990
	SVORIM-R and SVORIM	0.05596	0.02908	0.07227
	LDLOR-R and LDLOR	0.00042	0.00074	0.00048
	DMLNN-R and DMLNN	0.01211	0.02929	0.014887

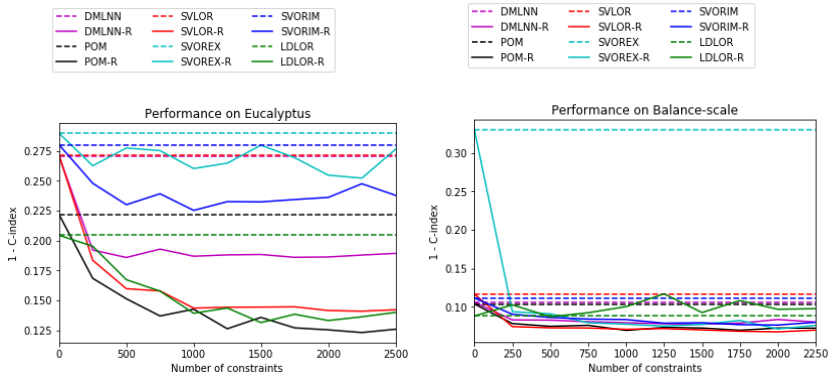


Figure 7: $1-$C-index for all the methods provided with a fixed amount of absolute information and an increasing amount of relative information on the EU and BS datasets.

dataset, there is a clear decreasing trend for the augmented methods DMLNN-R, POM-R, LDLOR-R and SVLOR-R, while this effect is minimal for SVOREX-R and SVORIM-R. However, this is not the case for the BS dataset. On this dataset, SVOREX-R and SVORIM-R are competitive with other augmented ordinal classification methods, although SVOREX performs badly compared to the other original ordinal classification methods (Note that SVOREX and

SVORIM still perform great on most of the datasets, as can be seen in Tables 2–4.). The results show that it is not necessary to incorporate as many constraints as possible, and that it might be beneficial to try to balance the amounts of absolute and relative information, and thus also the cost of data collection.

4.5.2. Cost of collecting different types of information

We denote the cost of assigning a class label to an example as C_A and the cost of assigning a preference order to a couple as C_R. The total cost of data collection is then given by

$$C_T = nC_A + mC_R \ , \tag{36}$$

where n is the number of labelled examples and m is the number of preference-ordered couples. Here, we fix C_A at 1 and denote C_R by ρ in order to emphasize that we are referring to the cost of relative information per unitary cost of absolute information. The total cost then simply is $C_T = n + \rho m$. It is assumed that $\rho \leq 1$ since absolute information is in general more informative than relative information and, thus, the latter is only useful when collected at a cheaper cost.

Based on the generation process in Figure 4, from the generated absolute and relative information for each fold, we randomly sample different numbers of examples and couples, corresponding to different amounts of absolute and relative information, for instance $n \in \{20, 25, 30, \ldots, 65\}$ and $m \in \{100, 200, 300, \ldots, 1000\}$ for the CA dataset. For each combination of n and m, we apply the augmented machine learning methods and obtain the performance via ten-fold cross-validation.

In Figure 8, we show a scatterplot of the performance versus the total cost for different values of ρ (here, 1, 0.01 and 0.0001) for DMLNN and DMLNN-R on the CA dataset. Obviously, the performance of DMLNN-R is better than that of DMLNN for every value of ρ. Depending on the problem-specific cost ρ of relative information per unitary cost of absolute information, we can distinguish three settings. For values of ρ close to 1, both absolute and relative

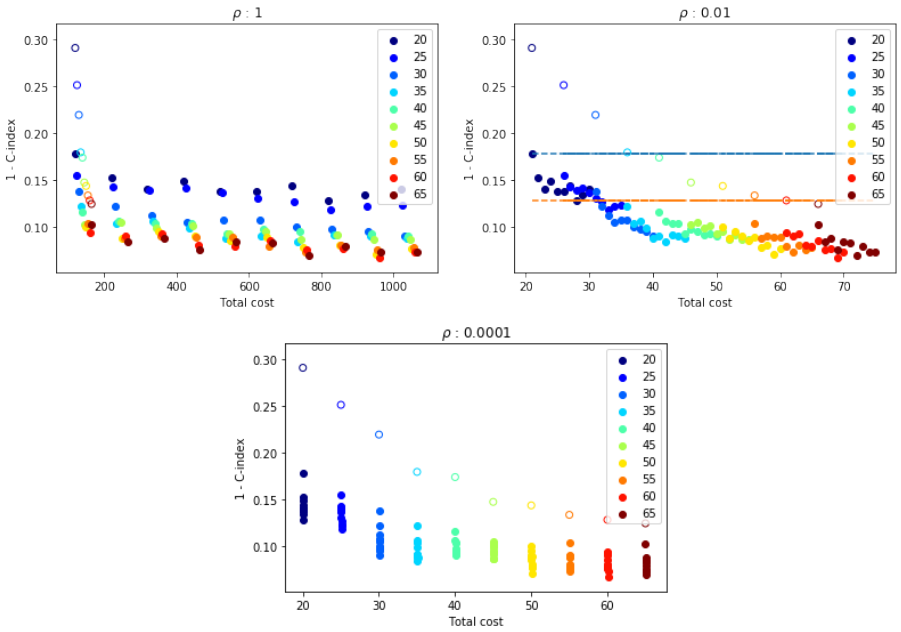


Figure 8: $1 - $ C-index versus total cost for different values of ρ for DMLNN (empty circles) and DMLNN-R (solid circles) on the CA dataset. The color gradient represents the amount of absolute information. At a fixed color, a larger cost corresponds to a larger amount of relative information.

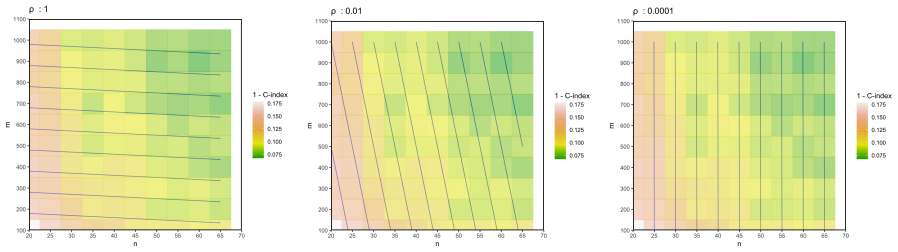


Figure 9: Heatmap representing the $1-$C-index (color gradient) and lines identifying cells with the same total cost (obtained from different amounts of absolute and relative information) for three different values of ρ for DMLNN-R on the CA dataset.

information equally contribute to the total cost but the performance is mainly impacted by the amount of absolute information (since absolute information is typically more informative than relative information). For moderate values of ρ (here 0.01), the performance depends on the amounts of both absolute and relative information. As an illustration, the gap between the two horizontal lines in

the middle part of Figure 8 shows the improvement when considering additional relative information for a fixed amount of absolute information. As can be seen, the incorporation of more relative information improves the performance w.r.t. the performance obtained for datasets with a larger total cost (in particular a larger amount of absolute information and a smaller amount of relative information). Typically, realistic values of ρ are of this type, thus encouraging to jointly use both absolute and relative information. For small values of ρ (here 0.0001), the total cost is dominated by the cost of absolute information and the performance is mainly impacted by the amount of relative information. In such cases, collecting as much relative information as possible is always advisable since it provides valuable information at an almost free cost. Obviously, a value of ρ of the first or third type hints that investing in the collection of one of the two types of information does not actually pay off.

These results are also illustrated in Figure 9, where we show a heatmap of the performance of DMLNN-R on the CA dataset for different combinations of n and m. The superimposed lines connect cells with the same total cost for the corresponding value of ρ. Obviously, when n and m become larger, the performance gets better. For values of ρ close to 1, the lines are nearly horizontal and can be seen not to relate to the improvement in performance. For moderate values of ρ (here 0.01), both absolute and relative information contribute to the improvement in performance. In particular, increasing the budget (i.e., the allowed total cost) results in a better performance. For small values of ρ (here 0.0001), the lines are nearly vertical and the performance is mainly dominated by the fixed amount of absolute information, thus obtaining a boost in performance at a very low cost when additionally considering relative information.

4.5.3. Information entropy

Since the information brought by a labelled example differs from that brought by a preference-ordered couple, we propose to quantify this information content in terms of information entropy [42]. Note that in a real-world problem, the

distribution of the class labels is usually not uniform. Due to the ordinal nature of the scale, it is mostly the case that the probability that an example is assigned an extreme class label is low compared to that of being assigned a more central one, although this clearly depends on the problem domain. In medical diagnosis [43], for instance, the lowest class (absence of disease) typically has a high probability, while the higher classes (indicating different levels of intensity of the disease) come with lower probabilities.

Here, we make use of the prior distribution of the dataset and denote the probability of an object $\mathbf{x}_i$ belonging to the k-th class as $P(y_i = C_k)$. The entropy for a labelled object is then computed as:

$$H_{\mathcal{A}} = -\sum_{k=1}^{r} P(y_i = C_k) \log_2 P(y_i = C_k)\,. \tag{37}$$

As an illustration, note that in the unlikely case of a uniform distribution, it holds that $H_{\mathcal{A}} = \log_2 r$, increasing quickly at lower values (the typical range for ordinal classification problems), while slowing down for larger values. For the CA dataset ($r = 4$), the class distribution is $(210, 384, 69, 65)$ and $H_{\mathcal{A}} = 1.64$.

For a preference-ordered couple $(\mathbf{a}^j, \mathbf{b}^j)$, we denote the class label of $\mathbf{a}^j$ as $y_{\mathbf{a}^j}$ and the class label of $\mathbf{b}^j$ as $y_{\mathbf{b}^j}$. The preference order for this couple may either be $\mathbf{a}^j \succ \mathbf{b}^j$ implying $y_{\mathbf{a}^j} \succ y_{\mathbf{b}^j}$, or $\mathbf{a}^j \prec \mathbf{b}^j$ implying $y_{\mathbf{a}^j} \prec y_{\mathbf{b}^j}$. Obviously, it holds that $P(y_{\mathbf{a}^j} \preceq y_{\mathbf{b}^j}) = P(y_{\mathbf{b}^j} \preceq y_{\mathbf{a}^j}) = 0.5$. The entropy for a preference-ordered couple is then given by

$$H_{\mathcal{R}} = -P(y_{\mathbf{a}^j} \preceq y_{\mathbf{b}^j}) \log_2 P(y_{\mathbf{a}^j} \preceq y_{\mathbf{b}^j}) - P(y_{\mathbf{b}^j} \preceq y_{\mathbf{a}^j}) \log_2 P(y_{\mathbf{b}^j} \preceq y_{\mathbf{a}^j}) = 1\,. \tag{38}$$

Since entropy is additive for independent observations, the total entropy is computed as

$$H = nH_{\mathcal{A}} + mH_{\mathcal{R}} = nH_{\mathcal{A}} + m\,, \tag{39}$$

where n is the number of labelled examples and m is the number of preference-ordered couples.

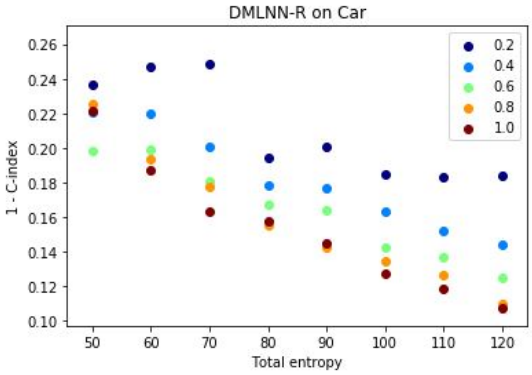


Figure 10: $1 -$ C-index versus total entropy for DMLNN-R on the CA dataset. The color gradient represents the ratio of the entropy for absolute information to the total entropy. At a fixed color, a larger entropy corresponds to larger amounts of absolute and relative information.

We denote the ratio of the entropy for absolute information to the total entropy as $\lambda = \frac{nH_{\mathcal{A}}}{H}$ and the ratio of the entropy for relative information to the total entropy as $1 - \lambda = \frac{m}{H}$. For the CA dataset, the range of λ is set to $\{0.2, 0.4, 0.6, 0.8, 1\}$ and the range of the total entropy H is set to $\{50, 60, 70, \ldots, 120\}$. We explore the impact of λ on the performance for each fixed value of the total entropy. More specifically, for each value of H and each value of λ, we get the corresponding combination of n and m, then apply the augmented machine learning methods and obtain the performance via ten-fold cross-validation. Figure 10 shows a scatterplot of the performance of DMLNN-R versus the total entropy on the CA dataset ($H = 1.64n + m$). The performance is highly correlated with λ. In general, when λ is larger, the performance tends to get better. However, note that, for many fixed values of the total entropy, the performance for $\lambda = 0.8$ is similar to that for $\lambda = 1$, which means that relative information can replace absolute information to some extent and combining both types of information is highly encouraged, even at a lower information entropy level.

5. Conclusion

For the interesting problem setting in which both absolute and relative information are available for ordinal classification, we augmented several ordinal classification methods by incorporating the relative information as constraints in the corresponding optimization problems. We extensively tested these methods on existing benchmark datasets. While POM-R is very fast to train, its performance is generally a bit worse than that of other methods. Similarly, SVLOR-R is also fast to train compared to other augmented ordinal classification methods and generally leads to a better performance. For this reason, POM-R and SVLOR-R are good options when dealing with large datasets. SVOREX and SVORIM are good threshold models that usually perform very well on absolute information, however, their improvement on most datasets when considering additional relative information is not so great. Still, both options remain competitive. DMLNN-R is the easiest and most explainable option, since it is based on the classical k-NN, and leads to a great performance. Even though learning the distance metric might be a time-consuming task (something that might be considered as a disadvantage at first), once accomplished the running time is similar to that of the other methods. Finally, LDLOR-R is the method leading to the best results in terms of average performance on the datasets considered. As a final comment, while all augmented methods perform better than the original counterparts, it comes as no surprise that there is no absolute best augmented method, since there is also no obvious winner in the case of absolute information only [16]. The final choice is thus left to the user as a matter of augmenting her personal favourite method.

Since additional relative information effectively helps to improve the ordinal classification performance, our approach paves the way for potential cost savings in real-life ordinal classification problems where relative information might be obtained at a lower cost. However, although the performance improves when increasing the amount of relative information, this effect clearly levels off, implying that the amount of relative information should be dosed appropriately.

The trade-off between absolute and relative information is an intricate one, not only depending on cost but also information content, which is linked with the cardinality of the ordinal scale.

For future work, it is of interest to explore active learning in this hybrid setting of absolute and relative information and identify effective objects to be labelled or couples to be preference ordered. Also, as public crowdsourcing platforms could provide an effective way of collecting relative information, the question arises how to deal with couples that have been preference-ordered multiple times, resulting in a frequency distribution of opinions; or, more formally, how to extend the present framework to the hybrid setting of absolute information and relative information in the form of frequency distributions. In a similar direction, the case in which the novices are asked to provide intensities preference may also be worth to be taken into consideration. However, it is admittedly true that this case appears to be less attractive for the setting of this paper bearing in mind that the main appeal for introducing additional relative information is that it can be easily gathered from novices. From a different point of view, the relative information is here understood as a collection of preference-ordered couples, rather than as a binary relation on the set of examples. In real-life applications, this binary relation might contain inconsistencies ($\mathbf{x}_i$ is preferred to $\mathbf{x}_j$ and $\mathbf{x}_j$ is preferred to $\mathbf{x}_\ell$, but $\mathbf{x}_\ell$ is preferred to $\mathbf{x}_i$) and might be missing some key information drawn from the transitivity of an order relation ($\mathbf{x}_i$ is preferred to $\mathbf{x}_j$ and $\mathbf{x}_j$ is preferred to $\mathbf{x}_\ell$ implies that $\mathbf{x}_i$ is preferred to $\mathbf{x}_\ell$). How to deal with and possibly exploit such inconsistencies and missing information could also be of interest. All these topics have been addressed within Multi-Criteria Decision Aiding (MCDA) [44, 45, 46, 47], a field from which inspiration for future work may surely be drawn.

Acknowledgments

Mengzi Tang is supported by the China Scholarship Council (CSC). Raúl Pérez-Fernández acknowledges the support of KERMIT, Department of Data

Analysis and Mathematical Modelling, Ghent University, the Research Foundation of Flanders (FWO17/PDO/160) and the Spanish MINECO (TIN-2017-87600-P). This research received funding from the Flemish Government under the "Onderzoeksprogramma Artificiële Intelligentie (AI) Vlaanderen" programme.

References

[1] S. B. Kotsiantis, I. Zaharakis, P. Pintelas, Supervised machine learning: A review of classification techniques, Emerging Artificial Intelligence Applications in Computer Engineering 160 (1) (2007) 3–24.

[2] G. He, B. Li, H. Wang, W. Jiang, Cost-effective active semi-supervised learning on multivariate time series data with crowds, IEEE Transactions on Systems, Man, and Cybernetics: Systems (2020) 1–14.

[3] Z.-H. Zhou, A brief introduction to weakly supervised learning, National Science Review 5 (1) (2018) 44–53.

[4] R. Kwitt, S. Hegenbart, N. Rasiwasia, A. Vécsei, A. Uhl, Do we need annotation experts? a case study in celiac disease classification, in: Proceedings of the 17th International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer, Boston, MA, USA, 2014, pp. 454–461.

[5] Y. Baba, H. Kashima, K. Kinoshita, G. Yamaguchi, Y. Akiyoshi, Leveraging non-expert crowdsourcing workers for improper task detection in crowdsourcing marketplaces, Expert Systems with Applications 41 (6) (2014) 2678–2687.

[6] P.-Y. Hsueh, P. Melville, V. Sindhwani, Data quality from crowdsourcing: a study of annotation selection criteria, in: Proceedings of the NAACL HLT 2009 workshop on Active Learning for Natural Language Processing, Boulder, Colorado, 2009, pp. 27–35.

[7] Q. Nguyen, H. Valizadegan, M. Hauskrecht, Learning classification with auxiliary probabilistic information, in: Proceedings of the 11th IEEE International Conference on Data Mining, Vancouver, British Columbia, Canada, 2011, pp. 477–486.

[8] M. Tabassian, R. Ghaderi, R. Ebrahimpour, Combining complementary information sources in the Dempster–Shafer framework for solving classification problems with imperfect labels, Knowledge-Based Systems 27 (2012) 92–102.

[9] Q. Nguyen, H. Valizadegan, A. Seybert, M. Hauskrecht, Sample-efficient learning with auxiliary class-label information, in: AMIA Annual Symposium Proceedings, Vol. 2011, American Medical Informatics Association, 2011, pp. 1004–1012.

[10] Q. Nguyen, H. Valizadegan, M. Hauskrecht, Learning classification models with soft-label information, Journal of the American Medical Informatics Association 21 (3) (2014) 501–508.

[11] Y. Xue, M. Hauskrecht, Efficient learning of classification models from soft-label information by binning and ranking, in: Proceedings of the 30th International Florida Artificial Intelligence Research Society Conference, Marco Island, Florida, 2017, pp. 164–169.

[12] F. Fernández-Navarro, P. Campoy-Muñoz, L. P.-M. Mónica-de, C. Hervás-Martínez, X. Yao, Addressing the EU sovereign ratings using an ordinal regression approach, IEEE Transactions on Cybernetics 43 (6) (2013) 2228–2240.

[13] K.-Y. Chang, C.-S. Chen, Y.-P. Hung, Ordinal hyperplanes ranker with cost sensitivities for age estimation, in: Proceedings of the 24th IEEE Conference on Computer Vision and Pattern Recognition, Colorado Springs, Colorado, USA, 2011, pp. 585–592.

[14] A. S. Fullerton, J. Xu, The proportional odds with partial proportionality constraints model for ordinal response variables, Social Science Research 41 (1) (2012) 182–198.

[15] O. M. Doyle, E. Westman, A. F. Marquand, P. Mecocci, B. Vellas, M. Tsolaki, I. Kłoszewska, H. Soininen, S. Lovestone, S. C. Williams, et al., Predicting progression of alzheimer's disease using ordinal regression, PloS One 9 (8) (2014) e105542.

[16] P. A. Gutiérrez, M. Pérez-Ortiz, J. Sánchez-Monedero, F. Fernández-Navarro, C. Hervás-Martínez, Ordinal regression methods: survey and experimental study, IEEE Transactions on Knowledge and Data Engineering 28 (1) (2016) 127–146.

[17] A. Bellet, A. Habrard, M. Sebban, Metric learning, Synthesis Lectures on Artificial Intelligence and Machine Learning 9 (1) (2015) 1–151.

[18] P. Yıldırım, U. K. Birant, D. Birant, Eboc: Ensemble-based ordinal classification in transportation, Journal of Advanced Transportation 2019.

[19] G. Manthoulis, M. Doumpos, C. Zopounidis, E. Galariotis, An ordinal classification framework for bank failure prediction: Methodology and empirical evidence for US banks, European Journal of Operational Research 282 (2) (2020) 786–801.

[20] M. Sader, J. Verwaeren, R. Pérez-Fernández, B. De Baets, Integrating expert and novice evaluations for augmenting ordinal regression models, Information Fusion 51 (2019) 1–9.

[21] M. Tang, R. Pérez-Fernández, B. De Baets, Fusing absolute and relative information for augmenting the method of nearest neighbors for ordinal classification, Information Fusion 56 (2020) 128–140.

[22] M. Tang, R. Pérez-Fernández, B. De Baets, Distance metric learning for augmenting the method of nearest neighbors for ordinal classification with absolute and relative information, Information Fusion 65 (2021) 72–83.

[23] M. Tang, R. Pérez-Fernández, B. De Baets, Combining absolute and relative information with frequency distributions for ordinal classification, in: Proceedings of the 18th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems, Springer, 2020, pp. 594–602.

[24] P. McCullagh, Regression models for ordinal data, Journal of the Royal Statistical Society: Series B (Methodological) 42 (2) (1980) 109–127.

[25] R. Herbrich, T. Graepel, K. Obermayer, Support vector learning for ordinal regression, in: Proceedings of the 9th International Conference on Artificial Neural Networks, Edinburgh, UK, 1999, pp. 97–102.

[26] W. Chu, S. S. Keerthi, Support vector ordinal regression, Neural Computation 19 (3) (2007) 792–815.

[27] B.-Y. Sun, J. Li, D. D. Wu, X.-M. Zhang, W.-B. Li, Kernel discriminant learning for ordinal regression, IEEE Transactions on Knowledge and Data Engineering 22 (6) (2009) 906–910.

[28] B. Nguyen, C. Morell, B. De Baets, Distance metric learning for ordinal classification based on triplet constraints, Knowledge-Based Systems 142 (2018) 17–28.

[29] E. P. Xing, A. Y. Ng, M. I. Jordan, S. Russell, Distance metric learning with application to clustering with side-information, in: Proceedings of the 15th International Conference on Neural Information Processying Systems, Cambride, MA, USA, 2002, pp. 521–528.

[30] M. Schultz, T. Joachims, Learning a distance metric from relative comparisons, in: Proceedings of the 16th International Conference on Neural Information Processying Systems, Cambride, MA, USA, 2003, pp. 41–48.

[31] M. M. Deza, E. Deza, Encyclopedia of distances, Springer Berlin Heidelberg, 2009, pp. 1–583.

[32] A. Asuncion, D. J. Newman, UCI machine learning repository, http://www.ics.uci.edu/~mlearn/MLRepository.html (2007).

[33] PASCAL, (Pattern Analysis, Statistical Modelling and Computational Learning) machine learning benchmarks repository, http://mldata.org/ (2011).

[34] W. Chu, Z. Ghahramani, Gaussian processes for ordinal regression, Journal of Machine Learning Research 6 (7) (2005) 1019–1041.

[35] M. Cruz-Ramírez, C. Hervás-Martínez, J. Sánchez-Monedero, P. A. Gutiérrez, Metrics to guide a multi-objective evolutionary algorithm for ordinal classification, Neurocomputing 135 (2014) 21–31.

[36] S. Baccianella, A. Esuli, F. Sebastiani, Evaluation measures for ordinal regression, in: Proceedings of the 9th IEEE International Conference on Intelligent Systems Design and Applications, Pisa, Italy, 2009, pp. 283–287.

[37] W. Waegeman, B. De Baets, L. Boullart, Learning to rank: a ROC-based graph-theoretic approach, Pattern Recognition Letters 29 (1) (2008) 1–9.

[38] C. Cortes, M. Mohri, AUC optimization vs. error rate minimization, in: Proceedings of the 16th International Conference on Neural Information Processing Systems, Whistler, British Columbia, Canada, 2004, pp. 313–320.

[39] J. Demšar, Statistical comparisons of classifiers over multiple data sets, Journal of Machine Learning Research 7 (2006) 1–30.

[40] M. Friedman, A comparison of alternative tests of significance for the problem of m rankings, The Annals of Mathematical Statistics 11 (1) (1940) 86–92.

[41] F. Wilcoxon, Individual comparisons by ranking methods, Biometrics Bulletin 1 (6) (1945) 80–83.

[42] R. M. Gray, Entropy and Information Theory, Springer Science & Business Media, 2011.

[43] A. M. Durán-Rosal, J. Camacho-Cañamón, P. A. Gutiérrez, M. V. G. Moreno, E. Rodríguez-Cáceres, J. A. V. Casas, C. Hervás-Martínez, Ordinal classification of the affectation level of 3D-images in Parkinson diseases, Scientific Reports 11 (1) (2021) 1–13.

[44] D. Bouyssou, T. Marchant, M. Pirlot, A. Tsoukias, P. Vincke, Evaluation and Decision Models with Multiple Criteria: Stepping Stones for the Analyst, Vol. 86, Springer Science & Business Media, New York, 2006.

[45] M. Rademaker, B. De Baets, Consistent union and prioritized consistent union: New operations for preference aggregation, Annals of Operations Research 195 (1) (2012) 237–259.

[46] S. Corrente, S. Greco, M. Kadziński, R. Słowiński, Robust ordinal regression in preference learning and ranking, Machine Learning 93 (2) (2013) 381–422.

[47] M. Kadziński, K. Martyn, M. Cinelli, R. Słowiński, S. Corrente, S. Greco, Preference disaggregation for multiple criteria sorting with partial monotonicity constraints: Application to exposure management of nanomaterials, International Journal of Approximate Reasoning 117 (2020) 60–80.