V-shaped interval insensitive loss for ordinal classification

Kostiantyn Antoniuk¹ · Vojtěch Franc¹ ·
Václav Hlaváč¹

Received: 25 March 2015 / Accepted: 26 October 2015 / Published online: 7 January 2016
© The Author(s) 2016

Abstract We address a problem of learning ordinal classifiers from partially annotated examples. We introduce a V-shaped interval-insensitive loss function to measure discrepancy between predictions of an ordinal classifier and a partial annotation provided in the form of intervals of candidate labels. We show that under reasonable assumptions on the annotation process the Bayes risk of the ordinal classifier can be bounded by the expectation of an associated interval-insensitive loss. We propose several convex surrogates of the interval-insensitive loss which are used to formulate convex learning problems. We described a variant of the cutting plane method which can solve large instances of the learning problems. Experiments on a real-life application of human age estimation show that the ordinal classifier learned from cheap partially annotated examples can achieve accuracy matching the results of the so-far used supervised methods which require expensive precisely annotated examples.

Keywords Ordinal classification · Partially annotated examples · Risk minimization

1 Introduction

The ordinal classification model (also ordinal regression) is used in problems where the set of labels is fully ordered, for example, the label can be an age category (0–9, 10–19, . . . , 90–99)

Editors: Hang Li, Dinh Phung, Tru Cao, Tu-Bao Ho, and Zhi-Hua Zhou.

✉ Kostiantyn Antoniuk
antonkos@cmp.felk.cvut.cz

Vojtěch Franc
xfrancv@cmp.felk.cvut.cz

Václav Hlaváč
hlavac@fel.cvut.cz

¹ Center for Machine Perception, Department of Cybernetics, Faculty of Electrical Engineering, Czech Technical University in Prague, Technická 2, 166 27 Prague 6, Czech Republic

or a respondent answer to certain question (from strongly agree to strongly disagree). The ordinal classifiers are routinely used in social sciences, epidemiology, information retrieval or computer vision.

Recently, many supervised algorithms have been proposed for discriminative learning of the ordinal classifiers. The discriminative methods learn parameters of an ordinal classifier by minimizing a regularized convex proxy of the empirical risk. A Perceptron-like on-line algorithm PRank has been proposed in [Crammer and Singer \(2001\)](#). A large-margin principle has been applied for learning ordinal classifiers in [Shashua and Levin \(2002\)](#). The paper ([Chu and Keerthi 2005](#)) proposed Support Vector Ordinal Regression algorithm with explicit constraints (SVOR-EXP) and the SVOR algorithm with implicit constraints (SVOR-IMC). Unlike ([Shashua and Levin 2002](#)), the SVOR-EXP and SVOR-IMC guarantee the learned ordinal classifier to be statistically plausible. The same approach have been proposed independently by [Rennie and Srebro \(2005\)](#) who introduce so called immediate-threshold loss and all-thresholds loss functions. Minimization of a quadratically regularized immediate-threshold loss and the all-threshold loss are equivalent to the SVOR-EXP and the SVOR-IMC formulation, respectively. A generic framework proposed in [Li and Lin \(2006\)](#), of which the SVOR-EXP and SVOR-IMC are special instances, allows to convert learning of the ordinal classifier into learning of two-class SVM classifier with weighted examples.

Estimating parameters of a probabilistic model by the Maximum Likelihood (ML) method is another paradigm that can be used to learn ordinal classifiers. A plug-in ordinal classifier can be then constructed by substituting the estimated model to the optimal decision rule derived for a particular loss function (see e.g. [Dembczyński et al. 2008](#) for a list of losses and corresponding decision functions suitable for ordinal classification). Parametric probability distributions suitable for modeling the ordinal labels have been proposed in [McCullagh \(1980\)](#), [Fu and Simpson \(2002\)](#), [Rennie and Srebro \(2005\)](#). Besides the parametric methods, the non-parametric probabilistic approaches like the Gaussian processes were also proposed ([Chu and Ghahramani 2005](#)).

Properties of the discriminative and the ML based methods are complementary to each other. The ML approach can be directly applied in the presence of incomplete annotation (e.g. the setting considered in this paper when label interval is given instead of a single label) by using the expectation–maximization algorithms ([Schlesinger 1968](#); [Dempster et al. 1997](#)). However, the ML methods are sensitive to model mis-specification which complicates their application in modeling complex high-dimensional data. In contrast, the discriminative methods are known to be robust against the model mis-specification while their extension for learning from partial annotations is not trivial. To our best knowledge, the existing discriminative approaches for ordinal classification assume the precisely annotation only, that is, each training instance is annotated by exactly one label.

In this paper, we consider learning of the ordinal classifiers from partially annotated examples. We assume that each training input is annotated by an interval of candidate labels rather than a single label. This setting is common in practice. For example, let us assume a computer vision problem of learning an ordinal classifier predicting age from a facial image (e.g. [Ramanathan et al. 2009](#), [Chang et al. 2011](#)). In this case, examples of face images are typically downloaded from the Internet and the age of depicted persons is estimated by a human annotator. Providing a reliable year-exact age just from a face image is virtually impossible. For humans it is more natural and easier to provide an interval estimate of the age. The interval annotation can be also obtained in an automated way e.g. by the method of [Kotłowski et al. \(2008\)](#) removing inconsistencies in imprecisely annotated data.

To deal with the interval annotations, we propose an interval-insensitive loss function which extends an arbitrary (supervised) V-shaped loss to the interval setting. The interval-

insensitive loss measures a discrepancy between the interval of candidate labels given in the annotation and a label predicted by the classifier. Our interval-insensitive loss can be seen as the ordinal regression counterpart of the ϵ -insensitive loss used in the support vector regression (Vapnik 1998). We prove that under reasonable assumptions on the annotation process, the Bayes risk of the ordinal classifier can be bounded by the expectation of the interval-insensitive loss. This bound justifies learning of the ordinal classifier via minimization of an empirical estimate of the interval-insensitive loss. Tightness of the bound depends on two intuitive parameters characterizing the annotations process. We show how to control these parameters in practice by properly designing the annotation process. We propose a convex surrogate of an arbitrary V-shaped interval-insensitive loss which is then used to formulate a convex learning problem. We also show how to modify the existing supervised methods, the SVOR-EXP and the SVOR-IMC algorithms, in order to minimize a convex surrogate of the interval-insensitive loss associated with the 0/1-loss and the mean absolute error (MAE) loss. Finally, we design a variant of a cutting plane algorithm which enables to solve large instances of the learning problems efficiently.

Discriminative learning from partially annotated examples has been recently studied in the context of a generic multi-class classifiers (Cour et al. 2011), the Hidden Markov Chain based classifiers (Do and Artières 2009), generic structured output models (Lou and Hamprecht 2012), the multi-instance learning (Jie and Orabona 2010) etc. All these methods translate learning to minimization of a partial loss evaluating discrepancy between the classifier predictions and partial annotations. The partial loss is defined as minimal value of a supervised loss (defined on a pair of labels, e.g. 0/1-loss) over all candidate labels consistent with the partial annotation. Our interval-insensitive loss can be seen as an application of such type of partial losses in the context of the ordinal classification. In particular, we analyze the partial annotation in the form of intervals of the candidate labels and the mean absolute error being the most typical target loss in the ordinal classification. The bounds of the Bayes risk via the expectation of the partial loss have been studied in Cour et al. (2011) but only for the 0/1-loss which is much less suitable for ordinal classification. It worth mentioning that the ordinal classification model allows for a tight convex approximations of the partial loss in contrast to previously considered classification models often requiring a hard to optimize non-convex surrogates (Do and Artières 2009; Lou and Hamprecht 2012; Jie and Orabona 2010).

The paper is organized as follows. Formulation of the learning problem and its solution via minimization of the interval insensitive loss is presented in Sect. 2. Algorithms approximating minimization of the interval-insensitive loss by convex optimization problems are proposed in Sect. 3. A cutting plane based method solving the convex programs is described in Sect. 4. Section 5 presents experimental evaluation and Sect. 6 concludes the paper.

2 Learning ordinal classifier from weakly annotated examples

2.1 Learning from completely annotated examples

Let $\mathcal{X} \subset \mathbb{R}^n$ be a space of input observations and $\mathcal{Y} = \{1, \dots, Y\}$ a set of hidden labels endowed with a natural order.¹ We consider learning of an ordinal classifier $h: \mathcal{X} \rightarrow \mathcal{Y}$ of the form

¹ The sequence $1, \dots, Y$ is used just for a notational convenience, however, any other finite and fully ordered set can be used instead.

Fig. 1 The figure visualizes division of the 2-dimensional feature space into four classes realized by an instance of the ordinal classifier (1)

$$h(\mathbf{x}; \mathbf{w}, \boldsymbol{\theta}) = 1 + \sum_{k=1}^{Y-1} \mathbb{I}[\langle \mathbf{x}, \mathbf{w} \rangle > \theta_k] \quad (1)$$

where $\mathbf{w} \in \mathbb{R}^n$ and $\boldsymbol{\theta} \in \Theta = \{\boldsymbol{\theta}' \in \mathbb{R}^{Y-1} \mid \theta'_y \leq \theta'_{y+1}, y = 1, \dots, Y-1\}$ are admissible parameters. The brackets $\langle \cdot, \cdot \rangle$ denote the dot product and the operator $\mathbb{I}[A]$ evaluates to 1 if A holds, otherwise it is 0. The classifier (1) splits the real line of projections $\langle \mathbf{x}, \mathbf{w} \rangle$ into Y consecutive intervals defined by thresholds $\theta_1 \leq \theta_2 \leq \dots \leq \theta_{Y-1}$. The observation $\mathbf{x}$ is assigned a label corresponding to the interval to which the projection $\langle \mathbf{w}, \mathbf{x} \rangle$ falls to. The classifier (1) is a suitable model if the label can be thought of as a rough measurement of a continuous random variable $\xi(\mathbf{x}) = \langle \mathbf{x}, \mathbf{w} \rangle + \text{noise}$ (McCullagh 1980). An example of the ordinal classifier applied to a toy 2D problem is depicted in Fig. 1.

There exist several discriminative methods for learning parameters $(\mathbf{w}, \boldsymbol{\theta})$ of the classifier (1) from examples, e.g. Crammer and Singer (2001), Shashua and Levin (2002), Chu and Keerthi (2005), Li and Lin (2006). To our best knowledge, all the existing methods are fully supervised algorithms requiring a set of completely annotated training examples

$$\mathcal{T}^m = \{(\mathbf{x}^1, y^1), \dots, (\mathbf{x}^m, y^m)\} \in (\mathcal{X} \times \mathcal{Y})^m \quad (2)$$

typically assumed to be drawn from i.i.d. random variables with some unknown distribution $p(\mathbf{x}, y)$. The goal of the supervised learning algorithm is formulated as follows. Given a loss function $\Delta: \mathcal{Y} \times \mathcal{Y} \rightarrow \mathbb{R}$ and the training examples (2), the task is to learn the ordinal classifier h whose *Bayes risk*

$$R(h) = \mathbb{E}_{(\mathbf{x}, y) \sim p(\mathbf{x}, y)} \Delta(y, h(\mathbf{x}; \mathbf{w}, \boldsymbol{\theta})) \quad (3)$$

is as small as possible. The loss functions most commonly used in practice are the mean absolute error (MAE) $\Delta(y, y') = |y - y'|$ and the 0/1-loss $\Delta(y, y') = \mathbb{I}[y \neq y']$. The MAE and 0/1-loss are instances of so called V-shaped losses.

Definition 1 (V-shaped loss). A loss $\Delta: \mathcal{Y} \times \mathcal{Y} \rightarrow \mathbb{R}$ is V-shaped if $\Delta(y, y) = 0$ and $\Delta(y'', y) \geq \Delta(y', y)$ holds for all triplets $(y, y', y'') \in \mathcal{Y}^3$ such that $|y'' - y'| \geq |y' - y|$.

That is, the value of a V-shaped loss grows monotonically with the distance between the predicted and the true label. In this paper we constrain our analysis to the V-shaped losses.

Because the expected risk $R(h)$ is not accessible directly due to the unknown distribution $p(\mathbf{x}, y)$, the discriminative methods like Shashua and Levin (2002), Chu and Keerthi (2005), Li and Lin (2006) minimize a convex surrogate of the empirical risk augmented by a quadratic regularizer. We follow the same framework but with novel surrogate loss functions suitable for learning from partially annotated examples.

2.2 Learning from partially annotated examples

Analogically to the supervised setting we assume that the observation $\mathbf{x} \in \mathcal{X}$ and the corresponding hidden label $y \in \mathcal{Y}$ are generated from some unknown distribution $p(\mathbf{x}, y)$. However, in contrast to the supervised setting the training set does not contain a single label for each instance. Instead, we assume that an annotator provided with the observation $\mathbf{x}$, and possibly with the label y , returns a partial annotation in the form of an interval of candidate labels $[y_l, y_r] \in \mathcal{P}$. The symbol $\mathcal{P} = \{[y_l, y_r] \in \mathcal{Y}^2 \mid y_l \leq y_r\}$ denotes the set of all possible partial annotations. The partial annotation $[y_l, y_r]$ means that the true label y is from the interval $[y_l, y_r] = \{y \in \mathcal{Y} \mid y_l \leq y \leq y_r\}$. We shall assume that the annotator can be modeled by a stochastic process determined by a distribution $p(y_l, y_r \mid \mathbf{x}, y)$. That is, we are given a set of partially annotated examples

$$\mathcal{T}_I^m = \{(\mathbf{x}^1, [y_l^1, y_r^1]), \dots, (\mathbf{x}^m, [y_l^m, y_r^m])\} \in (\mathcal{X} \times \mathcal{P})^m \quad (4)$$

assumed to be generated from i.i.d. random variables with the distribution

$$p(\mathbf{x}, y_l, y_r) = \sum_{y \in \mathcal{Y}} p(y_l, y_r \mid \mathbf{x}, y) p(\mathbf{x}, y)$$

defined over $\mathcal{X} \times \mathcal{P}$. The learning algorithms described below do not require the knowledge of $p(\mathbf{x}, y)$ and $p(y_l, y_r \mid \mathbf{x}, y)$. However, it is clear that the annotation process given by $p(y_l, y_r \mid \mathbf{x}, y)$ cannot be arbitrary in order to make learning possible. For example, in the case when $p(y_l, y_r \mid \mathbf{x}, y) = p(y_l, y_r)$ the annotation would carry no information about the true label. Therefore we will later assume that the annotation is consistent in the sense that $y \notin [y_l, y_r]$ implies $p(y_l, y_r \mid \mathbf{x}, y) = 0$. The consistency of the annotation process is a standard assumption used e.g. in Cour et al. (2011).

The goal of learning from the partially annotated examples is formulated as follows. Given a (supervised) loss function $\Delta: \mathcal{Y} \times \mathcal{Y} \rightarrow \mathbb{R}$ and partially annotated examples (4), the task is to learn the ordinal classifier (1) whose Bayes risk $R(h)$ defined by (3) is as small as possible. Note that the objective remains the same as in the supervised setting but the information about the labels contained in the training set is reduced to intervals.

2.3 Interval insensitive loss

We define an interval-insensitive loss function in order to measure discrepancy between the interval annotation $[y_l, y_r] \in \mathcal{P}$ and the predictions made by the classifier $h(\mathbf{x}; \mathbf{w}, \theta) \in \mathcal{Y}$.

Definition 2 (Interval insensitive loss) Let $\Delta: \mathcal{Y} \times \mathcal{Y} \rightarrow \mathbb{R}$ be a supervised V-shaped loss. The interval insensitive loss $\Delta_I: \mathcal{P} \times \mathcal{Y} \rightarrow \mathbb{R}$ associated with Δ is defined as

$$\Delta_I(y_l, y_r, y) = \min_{y' \in [y_l, y_r]} \Delta(y', y) = \begin{cases} 0 & \text{if } y \in [y_l, y_r], \\ \Delta(y, y_l) & \text{if } y \leq y_l, \\ \Delta(y, y_r) & \text{if } y \geq y_r. \end{cases} \quad (5)$$

The interval-insensitive loss $\Delta_I(y_l, y_r, y)$ does not penalize predictions which are in the interval $[y_l, y_r]$. Otherwise the penalty is either $\Delta(y, y_l)$ or $\Delta(y, y_r)$ depending which border

of the interval $[y_l, y_r]$ is closer to the prediction y . In the special case of the mean absolute error (MAE) $\Delta(y, y') = |y - y'|$, one can think of the associated interval-insensitive loss $\Delta_I(y_l, y_r, y)$ as the discrete counterpart of the ϵ -insensitive loss used in the Support Vector Regression (Vapnik 1998).

Having defined the interval-insensitive loss, we can approximate minimization of the Bayes risk $R(h)$ defined in (3) by minimization of the expectation of the interval-insensitive loss

$$R_I(h) = \mathbb{E}_{(\mathbf{x}, y_l, y_r) \sim p(\mathbf{x}, y_l, y_r)} \Delta_I(y_l, y_r, h(\mathbf{x}; \mathbf{w}, \boldsymbol{\theta})) . \quad (6)$$

In the sequel we denote $R_I(h)$ as the *partial risk*. The question is how well the partial risk $R_I(h)$ approximates the Bayes risk $R(h)$ being the target quantity to be minimized. In the rest of this section we first analyze this question for the 0/1-loss adapting results of Cour et al. (2011), and then we present a novel bound for the MAE loss. In particular, we show that the Bayes risk $R(h)$ for both losses can be upper bounded by a linear function of the partial risk $R_I(h)$.

In the sequel we will assume that the annotated process governed by the distribution $p(y_l, y_r \mid \mathbf{x}, y)$ is consistent in the following sense.

Definition 3 (Consistent annotation process) Let $p(y_l, y_r \mid \mathbf{x}, y)$ be a properly defined distribution over $\mathcal{P}$ for any $(\mathbf{x}, y) \in \mathcal{X} \times \mathcal{Y}$. The annotation process governed by $p(y_l, y_r \mid \mathbf{x}, y)$ is consistent if any $y \in \mathcal{Y}$, $[y_l, y_r] \in \mathcal{P}$ such that $y \notin [y_l, y_r]$ implies $p(y_l, y_r \mid \mathbf{x}, y) = 0$.

The consistent annotation process guarantees that the true label is always contained among the candidate labels in the annotation.

We first apply the excess bound for the 0/1-loss function which has been studied in Cour et al. (2011) for a generic partial annotations when $\mathcal{P}$ is not constrained to be a set of label intervals. The tightness of the resulting bound depends on the annotation process $p(y_l, y_r \mid \mathbf{x}, y)$ characterized by so called *ambiguity degree* ε which, if adopted to our interval-setting, is defined as

$$\varepsilon = \max_{\mathbf{x}, y, z \neq y} p(z \in [y_l, y_r] \mid \mathbf{x}, y) = \max_{\mathbf{x}, y, z} \sum_{[y_l, y_r] \in \mathcal{P}} \mathbb{I}_{y_l \leq z \leq y_r} p(y_l, y_r \mid \mathbf{x}, y) . \quad (7)$$

In words, the ambiguity degree ε is the maximum probability of an extra label z co-occurring with the true label y in the annotation interval $[y_l, y_r]$, over all labels and observations.

Theorem 1 Let $p(y_l, y_r \mid \mathbf{x}, y)$ be a distribution describing a consistent annotation process with the ambiguity degree ε defined by (7). Let $R^{0/1}(h)$ be the Bayes risk (3) instantiated for the 0/1-loss and let $R_I^{0/1}(h)$ be the partial risk (6) instantiated for the interval insensitive loss associated to the 0/1-loss. Then the upper bound

$$R^{0/1}(h) \leq \frac{1}{1 - \varepsilon} R_I^{0/1}(h)$$

holds true for any $h \in \mathcal{X} \rightarrow \mathcal{Y}$.

Theorem 1 is a direct application of Proposition 1 from Cour et al. (2011).

Next we will introduce a novel upper bound for the MAE loss more frequently used in applications of the ordinal classifier. We again consider consistent annotation processes. We will characterize the annotation process by two numbers describing an amount of uncertainty in the training data. First, we use $\alpha \in [0, 1]$ to denote the lower bound of the portion of

exactly annotated examples, that is, examples annotated by an interval having just a single label $[y_l, y_r]$, $y_l = y_r$. Second, we use $\beta \in \{0, \dots, Y - 1\}$ to denote the maximal uncertainty in annotation, that is, $\beta + 1$ is the maximal width of the annotation interval which can appear in the training data with non-zero probability.

Definition 4 ($\alpha\beta$ -precise annotation process) Let $p(y_l, y_r \mid \mathbf{x}, y)$ be a properly defined distribution over $\mathcal{P}$ for any $(\mathbf{x}, y) \in \mathcal{X} \times \mathcal{Y}$. The annotation process governed by $p(y_l, y_r \mid \mathbf{x}, y)$ is $\alpha\beta$ -precise if

$$\alpha \leq p(y, y \mid \mathbf{x}, y) \quad \text{and} \quad \beta \geq \max_{[y_l, y_r] \in \mathcal{P}} \mathbb{I}[p(y_l, y_r \mid \mathbf{x}, y) > 0] (y_r - y_l)$$

hold for any $(\mathbf{x}, y) \in \mathcal{X} \times \mathcal{Y}$.

To illustrate the meaning of the parameters α and β let us consider the extreme cases. If $\alpha = 1$ or $\beta = 0$ then all examples are annotated exactly and we are back in the standard supervised setting. On the other hand, if $\beta = Y - 1$ then it may happen (but does not have to) that the annotation brings no information about the hidden label because the intervals contain all labels in $\mathcal{Y}$. With the definition of $\alpha\beta$ -precise annotation we can upper bound the Bayes risk in terms of the partial risk as follows:

Theorem 2 Let $p(y_l, y_r \mid \mathbf{x}, y)$ be a distribution describing a consistent $\alpha\beta$ -precise annotation process. Let $R^{\text{MAE}}(h)$ be the Bayes risk (3) instantiated for the MAE-loss and let $R_I^{\text{MAE}}(h)$ be the partial risk (6) instantiated for the interval insensitive loss associated to the MAE-loss. Then the upper bound

$$R^{\text{MAE}}(h) \leq R_I^{\text{MAE}}(h) + (1 - \alpha)\beta \quad (8)$$

holds true for any $h \in \mathcal{X} \rightarrow \mathcal{Y}$.

Proof of Theorem 2 is deferred to Appendix 1.

The bound (8) is obtained by the worst case analysis hence it may become trivial in some cases, for example, if all examples are annotated with wide intervals because then $\alpha = 0$ and β is large. The experimental study presented in section 5 nevertheless shows that the partial risk R_I is a good proxy even in cases when the bound upper bound is large. This suggests that better bounds might be derived, for example, when additional information about $p(y_l, y_r \mid \mathbf{x}, y)$ is available.

In order to improve the performance of the resulting classifier via the bound (8) one needs to control the parameters α and β . A possible way which allows to set the parameters (α, β) exactly is to control the annotation process. For example, given a set of unannotated randomly drawn input samples $\{\mathbf{x}_1, \dots, \mathbf{x}_m\} \in \mathcal{X}^m$ we can proceed as follows:

1. We generate a vector of binary variables $\boldsymbol{\pi} \in \{0, 1\}^m$ according to Bernoulli distribution with probability α that the variable is 1.
2. We instruct the annotator to provide just a single label for each input example with index from $\{i \in \{1, \dots, m\} \mid \pi_i = 1\}$ while the remaining inputs (with $\pi_i = 0$) can be annotated by intervals but not larger than $\beta + 1$ labels. That means that approximately $m \cdot \alpha$ inputs will be annotated exactly and $m \cdot (1 - \alpha)$ with intervals.

This simple procedure ensures that the annotation process is $\alpha\beta$ -precise though the distribution $p(y_l, y_r \mid \mathbf{x}, y)$ itself is unknown and depends on the annotator.

3 Algorithms

In the previous section we argue that the partial risk defined as an expectation of the interval insensitive loss is a reasonable proxy of the target Bayes risk. In this section we design algorithms learning the ordinal classifier via minimization of the quadratically regularized empirical risk used as a proxy for the expected risk. Similarly to the standard supervised case, we cannot minimize the empirical risk directly due to a discrete domain of the interval insensitive loss. For this reason we derive several convex surrogates which allow to translate the risk minimization to tractable convex problems.

We first show how to modify two existing supervised methods in order to learn from partially annotated examples. Namely, we extend the Support Vector Ordinal Regression algorithm with explicit constraints (SVOR-EXP) and the Support Vector Ordinal Regression algorithm with implicit constraints (SVOR-IMC) (Chu and Keerthi 2005). The extended interval-insensitive variants are named II-SVOR-EXP (Sect. 3.1) and II-SVOR-IMC (Sect. 3.2), respectively. The II-SVOR-EXP is a method minimizing a convex surrogate of the interval-insensitive loss associated to the 0/1-loss while the II-SVOR-IMC is designed for the minimization of MAE loss.

In Sect. 3.3, we show how to construct a generic convex surrogate of the interval-insensitive loss associated to an arbitrary V-shaped loss. We call a method minimizing this generic surrogate as the V-shaped interval insensitive loss minimization algorithm (VILMA). We prove that the VILMA subsumes the II-SVOR-IMC (as well as the SVOR-IMC) as a special case.

3.1 Interval insensitive SVOR-EXP algorithm for optimization of 0/1-loss

The original SVOR-EXP algorithm (Chu and Keerthi 2005) learns parameters of the ordinal classifier (1) from completely annotated examples $\mathcal{T}^m$ by solving the following convex problem

$$(\mathbf{w}^*, \boldsymbol{\theta}^*) = \underset{\mathbf{w} \in \mathbb{R}^n, \boldsymbol{\theta} \in \hat{\Theta}}{\operatorname{argmin}} \left[\frac{\lambda}{2} \|\mathbf{w}\|^2 + \sum_{i=1}^m \ell^{\text{EXP}}(\mathbf{x}^i, y^i, \mathbf{w}, \boldsymbol{\theta}) \right] \quad (9)$$

where the optimized convex surrogate loss reads

$$\ell^{\text{EXP}}(\mathbf{x}^i, y^i, \mathbf{w}, \boldsymbol{\theta}) = \max(0, 1 - \langle \mathbf{x}^i, \mathbf{w} \rangle + \theta_{y^i-1}) + \max(0, 1 + \langle \mathbf{x}^i, \mathbf{w} \rangle - \theta_{y^i})$$

and $\hat{\Theta} = \{\boldsymbol{\theta} \in \mathbb{R}^{Y+1} \mid \theta_0 = -\infty, \theta_Y = \infty, \theta_y \leq \theta_{y+1}, y = 1, \dots, Y-1\}$. Note that the set $\hat{\Theta}$ differs from the previously defined Θ by adding auxiliary constants $\theta_0 = -\infty$ and $\theta_Y = \infty$ used just for a notational convenience. In the original paper (Chu and Keerthi 2005), the SVOR-EXP algorithm is formulated as an equivalent quadratic program which can be easily obtained from (9) by using slack variables to replace the $\max(\cdot)$ functions. We rather use the formulation (9) because it shows the surrogate loss in its explicit form. The surrogate $\ell^{\text{EXP}}(\mathbf{w}, \boldsymbol{\theta}, \mathbf{x}, y)$ is a convex upper bound of the 0/1-loss

$$\Delta^{0/1}(y, h(\mathbf{x}; \mathbf{w}, \boldsymbol{\theta})) = \mathbb{I}[y \neq h(\mathbf{x}; \mathbf{w}, \boldsymbol{\theta})] = \mathbb{I}[\langle \mathbf{x}, \mathbf{w} \rangle < \theta_{y-1}] + \mathbb{I}[\langle \mathbf{x}, \mathbf{w} \rangle \geq \theta_y],$$

obtained by replacing the step function $\mathbb{I}[t \leq 0]$ by the hinge loss $\max(0, 1 - t)$.

Now we apply the same idea to approximate the interval insensitive loss $\Delta_I^{0/1}(y_l, y_r, y)$ associated with the 0/1-loss which according to (5) reads

Fig. 2 The *left* figure shows the interval insensitive loss $\Delta_I^{0/1}(\mathbf{x}, y_l, y_r, h(\mathbf{x}; \mathbf{w}, \theta))$ associated with the 0/1-loss and its surrogate $\ell_I^{\text{EXP}}(\mathbf{x}, y_l, y_r, \mathbf{w}, \theta)$. The *right* figure shows the interval insensitive loss $\Delta_I^{\text{MAE}}(\mathbf{x}, y_l, y_r, h(\mathbf{x}; \mathbf{w}, \theta))$ associated with the MAE loss and its surrogate $\ell_I^{\text{IMC}}(\mathbf{x}, y_l, y_r, \mathbf{w}, \theta)$. The losses are shown as a function of the score $\langle \mathbf{x}, \mathbf{w} \rangle$ evaluated for $\theta_1 = 1, \theta_2 = 2, \dots, \theta_{Y-1} = Y - 1$ and $y_l = 4, y_r = 6$. Note that for this particular setting of θ the surrogate $\ell_I^{\text{EXP}}(\mathbf{x}, y_l, y_r, \mathbf{w}, \theta)$ also appears to upper bound $\Delta_I^{\text{MAE}}(\mathbf{x}, y_l, y_r, h(\mathbf{x}; \mathbf{w}, \theta))$, but this does not hold in general

$$\begin{aligned} \Delta_I^{0/1}(y_l, y_r, h(\mathbf{x}; \mathbf{w}, \theta)) &= \min_{y' \in [y_l, y_r]} \mathbb{I}[y' \neq h(\mathbf{x}; \mathbf{w}, \theta)] \\ &= \mathbb{I}[\langle \mathbf{x}, \mathbf{w} \rangle < \theta_{y_l-1}] + \mathbb{I}[\langle \mathbf{x}, \mathbf{w} \rangle \geq \theta_{y_r}]. \end{aligned}$$

Replacing the step functions by the hinge loss yields the surrogate

$$\ell_I^{\text{EXP}}(\mathbf{x}, y_l, y_r, \mathbf{w}, \theta) = \max(0, 1 - \langle \mathbf{x}, \mathbf{w} \rangle + \theta_{y_l-1}) + \max(0, 1 + \langle \mathbf{x}, \mathbf{w} \rangle - \theta_{y_r})$$

which is clearly a convex upper bound of $\Delta_I^{0/1}(y_l, y_r, h(\mathbf{x}; \mathbf{w}, \theta))$ as can be also seen in Fig. 2.

Given partially annotated examples $\mathcal{T}_I^m$, we can learn parameters $(\mathbf{w}, \theta)$ of the ordinal classifier (1) by solving (9) with the surrogate $\ell_I^{\text{EXP}}(\mathbf{x}, y_l, y_r, \mathbf{w}, \theta)$ substituted for $\ell^{\text{EXP}}(\mathbf{x}, y, \mathbf{w}, \theta)$. We denote the modified variant as the II-SVOR-EXP algorithm.

3.2 Interval insensitive SVOR-IMC algorithm for optimization of MAE loss

The original SVOR-IMC algorithm (Chu and Keerthi 2005) learns parameters of the ordinal classifier (1) from completely annotated examples $\mathcal{T}^m$ by solving the following convex optimization problem

$$(\mathbf{w}^*, \theta^*) = \underset{\mathbf{w} \in \mathbb{R}^n, \theta \in \hat{\Theta}}{\operatorname{argmin}} \left[\frac{\lambda}{2} \|\mathbf{w}\|^2 + \sum_{i=1}^m \ell_I^{\text{IMC}}(\mathbf{x}^i, y^i, \mathbf{w}, \theta) \right] \quad (10)$$

where the convex surrogate reads

$$\ell_I^{\text{IMC}}(\mathbf{x}^i, y^i, \mathbf{w}, \theta) = \sum_{y=1}^{y^i-1} \max(0, 1 - \langle \mathbf{x}^i, \mathbf{w} \rangle + \theta_{y-1}) + \sum_{y=y^i}^{Y-1} \max(0, 1 + \langle \mathbf{x}^i, \mathbf{w} \rangle - \theta_y)$$

and using the convention $\sum_{i=m}^n a_i = 0$ if $m > n$. As in the previous case, the problem (10) is an equivalent reformulation of the quadratic program defining the SVOR-IMC algorithm in Chu and Keerthi (2005). It is seen that the surrogate $\ell_I^{\text{IMC}}(\mathbf{x}, y, \mathbf{w}, \theta)$ is a convex upper bound of the MAE loss

$$\begin{aligned}\Delta^{\text{MAE}}(y, h(\mathbf{x}; \mathbf{w}, \boldsymbol{\theta})) &= |y - h(\mathbf{x}; \mathbf{w}, \boldsymbol{\theta})| \\ &= \sum_{y'=1}^{y-1} \mathbb{I}(\langle \mathbf{x}, \mathbf{w} \rangle < \theta_{y'-1}) + \sum_{y'=y}^{Y-1} \mathbb{I}(\langle \mathbf{x}, \mathbf{w} \rangle \geq \theta_{y'}) .\end{aligned}$$

The surrogate is again obtained by replacing the step functions with the hinge loss. Analogically, we can derive a convex surrogate of the interval insensitive loss associated with the MAE which according to the definition (5) reads

$$\begin{aligned}\Delta_I^{\text{MAE}}(y_l, y_r, h(\mathbf{x}; \mathbf{w}, \boldsymbol{\theta})) &= \min_{y' \in [y_l, y_r]} |y' - h(\mathbf{x}; \mathbf{w}, \boldsymbol{\theta})| \\ &= \sum_{y'=1}^{y_l-1} \mathbb{I}(\langle \mathbf{x}, \mathbf{w} \rangle < \theta_{y'}) + \sum_{y'=y_r}^{Y-1} \mathbb{I}(\langle \mathbf{x}, \mathbf{w} \rangle \geq \theta_{y'}) .\end{aligned}$$

Replacing the step functions by the hinge loss we obtain a convex surrogate

$$\ell_I^{\text{IMC}}(\mathbf{x}, y_l, y_r, \mathbf{w}, \boldsymbol{\theta}) = \sum_{y'=1}^{y_l-1} \max(0, 1 - \langle \mathbf{x}, \mathbf{w} \rangle + \theta_{y'-1}) + \sum_{y'=y_r}^{Y-1} \max(0, 1 + \langle \mathbf{x}, \mathbf{w} \rangle - \theta_{y'}) ,$$

which is obviously an upper bound of $\Delta_I^{\text{MAE}}(y_l, y_r, h(\mathbf{x}; \mathbf{w}, \boldsymbol{\theta}))$ as can be also seen in Fig. 2.

Given the partially annotated examples $\mathcal{T}_I^m$, we can learn parameters $(\mathbf{w}, \boldsymbol{\theta})$ of the ordinal classifier (1) by solving (10) with the proposed surrogate $\ell_I^{\text{IMC}}(\mathbf{x}, y_l, y_r, \mathbf{w}, \boldsymbol{\theta})$ substituted for $\ell^{\text{IMC}}(\mathbf{x}, y, \mathbf{w}, \boldsymbol{\theta})$. We denote the modified variant as the II-SVOR-IMC algorithm. Note that due to the equality $\ell_I^{\text{IMC}}(\mathbf{x}, y, y, \mathbf{w}, \boldsymbol{\theta}) = \ell^{\text{IMC}}(\mathbf{x}, y, \mathbf{w}, \boldsymbol{\theta})$ it is clear that the proposed II-SVOR-IMC subsumes the original supervised SVOR-IMC as a special case.

3.3 VILMA: V-shaped interval insensitive loss minimization algorithm

In this section we propose a generic method for learning the ordinal classifiers with arbitrary interval insensitive V-shaped loss. We start by introducing an equivalent parametrization of the ordinal classifier (1) originally proposed in Antoniuk et al. (2013). The ordinal classifier (1) can be re-parametrized as a multi-class linear classifier, in the sequel denoted as multi-class ordinal (MORD) classifier, which reads

$$h'(\mathbf{x}; \mathbf{w}, \mathbf{b}) = \operatorname{argmax}_{y \in \mathcal{Y}} (\langle \mathbf{x}, \mathbf{w} \rangle \cdot y + b_y) \quad (11)$$

where $\mathbf{w} \in \mathbb{R}^n$ and $\mathbf{b} = (b_1, \dots, b_Y) \in \mathbb{R}^Y$ are parameters. Note that the MORD classifier has $n + Y$ parameters and any pair $(\mathbf{w}, \mathbf{b}) \in (\mathbb{R}^n \times \mathbb{R}^Y)$ is admissible. Note that the standard parametrization of the ordinal classifier (1) has $n + Y - 1$ parameters, however, the admissible parameters must satisfy a set of linear constraints $(\mathbf{w}, \boldsymbol{\theta}) \in (\mathbb{R}^n \times \Theta)$. The following theorem states that both parametrizations are equivalent.

Theorem 3 *The ordinal classifier (1) and the MORD classifier (11) are equivalent in the following sense. For any $\mathbf{w} \in \mathbb{R}^n$ and admissible $\boldsymbol{\theta} \in \Theta$ there exists $\mathbf{b} \in \mathbb{R}^Y$ such that $h(\mathbf{x}, \mathbf{w}, \boldsymbol{\theta}) = h'(\mathbf{x}, \mathbf{w}, \mathbf{b})$, $\forall \mathbf{x} \in \mathbb{R}^n$. For any $\mathbf{w} \in \mathbb{R}^n$ and $\mathbf{b} \in \mathbb{R}^n$, there exists admissible $\boldsymbol{\theta} \in \Theta$ such that $h(\mathbf{x}, \mathbf{w}, \boldsymbol{\theta}) = h'(\mathbf{x}, \mathbf{w}, \mathbf{b})$, $\forall \mathbf{x} \in \mathbb{R}^n$.*

Proof of Theorem 3 is given in Antoniuk et al. (2013). The proof is constructive in that it provides analytical formulas for conversion between the two parametrizations. For the sake of completeness the conversion formulas are shown in Appendix 2.

The MORD parametrization allows to adopt existing techniques for linear classification. Namely, we can replace the interval insensitive loss by a convex surrogate similar to the margin-rescaling loss used in the structured output learning (Tsochantaridis et al. 2005). Given a V-shaped supervised loss $\Delta: \mathcal{Y} \times \mathcal{Y} \rightarrow \mathbb{R}$, we propose to approximate the value of the associated interval insensitive loss $\Delta_I(y_l, y_r, h'(\mathbf{x}; \mathbf{w}, \mathbf{b}))$ by a surrogate loss $\ell_I: \mathcal{X} \times \mathcal{P} \times \mathbb{R}^n \times \mathbb{R}^Y \rightarrow \mathbb{R}$ defined as

$$\ell_I(\mathbf{x}, y_l, y_r, \mathbf{w}, \mathbf{b}) = \max_{y \leq y_l} \left[\Delta(y, y_l) + \langle \mathbf{x}, \mathbf{w} \rangle (y - y_l) + b_y - b_{y_l} \right] + \max_{y \geq y_r} \left[\Delta(y, y_r) + \langle \mathbf{x}, \mathbf{w} \rangle (y - y_r) + b_y - b_{y_r} \right]. \quad (12)$$

It is seen that for fixed $(\mathbf{x}, y_l, y_r)$ the function $\ell_I(\mathbf{x}, y_l, y_r, \mathbf{w}, \mathbf{b})$ is a sum of two point-wise maxima over linear functions hence it is convex in the parameters $(\mathbf{w}, \mathbf{b})$. The following proposition states that the surrogate is like the previous surrogates an upper bound of the interval insensitive loss.

Proposition 1 For any $\mathbf{x} \in \mathbb{R}^n$, $[y_l, y_r] \in \mathcal{P}$, $\mathbf{w} \in \mathbb{R}^n$ and $\mathbf{b} \in \mathbb{R}^Y$ the inequality

$$\Delta_I(y_l, y_r, h'(\mathbf{x}; \mathbf{w}, \mathbf{b})) \leq \ell_I(\mathbf{x}, y_l, y_r, \mathbf{w}, \mathbf{b})$$

holds where $h'(\mathbf{x}; \mathbf{w}, \mathbf{b})$ denotes response of the MORD classifier (11).

Proof is deferred to Appendix 3.

Given partially annotated training examples T_I^m , we can learn parameters $(\mathbf{w}, \mathbf{b})$ of the MORD classifier (11) by solving the following unconstrained convex problem

$$(\mathbf{w}^*, \mathbf{b}^*) = \underset{\mathbf{w} \in \mathbb{R}^n, \mathbf{b} \in \mathbb{R}^Y}{\operatorname{argmin}} \left[\frac{\lambda}{2} \|\mathbf{w}\|^2 + \frac{1}{m} \sum_{i=1}^m \ell_I(\mathbf{x}^i, y_l^i, y_r^i, \mathbf{w}, \mathbf{b}) \right] \quad (13)$$

where $\lambda \in \mathbb{R}_{++}$ is a regularization constant. A suitable value of the regularization constant is typically tuned on the validation set. In the sequel we denote the method based on solving (13) as the V-shape Interval insensitive Loss Minimization Algorithm (VILMA).

As an important example let us consider the surrogate (12) instantiated for the MAE loss. In this case, the surrogate becomes

$$\ell_I^{\text{MAE}}(\mathbf{x}, y_l, y_r, \mathbf{w}, \mathbf{b}) = \max_{y \leq y_l} \left[y_l - y + \langle \mathbf{x}, \mathbf{w} \rangle (y - y_l) + b_y - b_{y_l} \right] + \max_{y \geq y_r} \left[y - y_r + \langle \mathbf{x}, \mathbf{w} \rangle (y - y_r) + b_y - b_{y_r} \right]. \quad (14)$$

It is interesting to compare the VILMA instantiated for the MAE loss with the II-SVOR-IMC algorithm which optimizes a different surrogate of the same loss. Note that the II-SVOR-IMC learns the parameters $(\mathbf{w}, \boldsymbol{\theta})$ of the ordinal classifier (1) while the VILMA parameters $(\mathbf{w}, \mathbf{b})$ of the MORD rule (11). The following proposition states that surrogates of both methods are equivalent.

Proposition 2 Let $\mathbf{w} \in \mathbb{R}^n$, $\boldsymbol{\theta} \in \Theta$, $\mathbf{b} \in \mathbb{R}^Y$ be a triplet of vectors such that $h(\mathbf{x}; \mathbf{w}, \boldsymbol{\theta}) = h'(\mathbf{x}; \mathbf{w}, \mathbf{b})$ holds for all $\mathbf{x} \in \mathcal{X}$ where $h(\mathbf{x}; \mathbf{w}, \boldsymbol{\theta})$ denotes the ordinal classifier (1) and $h'(\mathbf{x}; \mathbf{w}, \mathbf{b})$ the MORD classifier (11). Then the equality

$$\ell_I^{\text{IMC}}(\mathbf{x}, y_l, y_r, \mathbf{w}, \boldsymbol{\theta}) = \ell_I^{\text{MAE}}(\mathbf{x}, y_l, y_r, \mathbf{w}, \mathbf{b})$$

holds true for any $\mathbf{x} \in \mathcal{X}$ and $[y_l, y_r] \in \mathcal{P}$.

Proof is deferred to Appendix 4. Proposition 2 ensures that the II-SVOR-IMC algorithm and the VILMA with MAE loss both return the same classification rules although differently parametrized.

We finish with listing core properties of the generic method, the VILMA, proposed in this section:

- VILMA is applicable for an arbitrary V-shaped loss.
- VILMA subsumes the II-SVOR-IMC algorithm optimizing the MAE loss as a special case.
- VILMA converts learning into an unconstrained convex optimization. Note that the II-SVOR-EXP and the II-SVOR-IMC in contrast to VILMA maintain the set of linear constraints $\theta \in \hat{\Theta}$.

4 Double-loop cutting plane solver

The proposed method VILMA translates learning into a convex optimization problem (13) that can be re-written as

$$(\mathbf{w}^*, \mathbf{b}^*) = \underset{\mathbf{w} \in \mathbb{R}^n, \mathbf{b} \in \mathbb{R}^Y}{\operatorname{argmin}} \left[\frac{\lambda}{2} \|\mathbf{w}\|^2 + R_{\text{emp}}(\mathbf{w}, \mathbf{b}) \right] \quad (15)$$

where $R_{\text{emp}}(\mathbf{w}, \mathbf{b}) = \frac{1}{m} \sum_{i=1}^m \ell_I(\mathbf{x}^i, y^i, \mathbf{w}, \mathbf{b})$ is a non-differentiable convex function of variables $\mathbf{w}$ and $\mathbf{b}$. The task (15) can be reformulated as a quadratic program with $n + m + Y$ variables and $Y \cdot m$ constraints. Generic off-the-shelf QP solvers are applicable only to small problems. In this section we derive an instance of the cutting plane algorithm tailored for the problem (15). The resulting CPA is applicable for large problems and it provides a certificate of the optimality.

More details on the CPA based solvers applied to the machine learning problems can be found for example in Teo et al. (2010), Franc et al. (2012). The standard CPA is suitable for solving convex tasks of the form

$$\mathbf{w}^* = \underset{\mathbf{w} \in \mathbb{R}^n}{\operatorname{argmin}} F(\mathbf{w}) \quad \text{where} \quad F(\mathbf{w}) = \frac{\lambda}{2} \|\mathbf{w}\|^2 + G(\mathbf{w}) \quad (16)$$

and $G: \mathbb{R}^n \rightarrow \mathbb{R}$ is a convex function. In contrast to our problem (15), the objective of (16) contains a quadratic regularization imposed on all variables. It is well known that the CPA applied directly to the un-regularized problem like (15) exhibits a strong zig-zag behavior leading to a large number of iterations. A frequently used ad ad-hoc solution is to impose an artificial regularization on $\mathbf{b}$ which may however significantly spoil the results as demonstrated in Sect. 5.2. In the rest of this section we first outline the CPA algorithm for the problem (16) and then show how it can be used to solve the problem (15).

The core idea of the CPA is to approximate the solution of the master problem (16) by solving a *reduced problem*

$$\mathbf{w}_t \in \underset{\mathbf{w} \in \mathbb{R}^n}{\operatorname{argmin}} F_t(\mathbf{w}) \quad \text{where} \quad F_t(\mathbf{w}) = \frac{\lambda}{2} \|\mathbf{w}\|^2 + G_t(\mathbf{w}). \quad (17)$$

The reduced problem (17) is obtained from (16) by substituting a cutting-plane model $G_t(\mathbf{w})$ for the convex function $G(\mathbf{w})$ while the regularizer remains unchanged. The cutting plane model of $G(\mathbf{w})$ reads

$$G_t(\mathbf{w}) = \max_{i=0, \dots, t-1} [G(\mathbf{w}_i) + \langle G'(\mathbf{w}_i), \mathbf{w} - \mathbf{w}_i \rangle] \quad (18)$$

Algorithm 1: Cutting Plane Algorithm

Input: $\varepsilon > 0$, $\mathbf{w}_0 \in \mathbb{R}^n$, $t \leftarrow 0$
Output: vector $\mathbf{w}_t$ being ε -precise solution of (16)
repeat
 $t \leftarrow t + 1$
 Compute $G(\mathbf{w}_{t-1})$ and $G'(\mathbf{w}_{t-1})$
 Update the model $G_t(\mathbf{w}) \leftarrow \max_{i=0, \dots, t-1} G(\mathbf{w}_i) + \langle G'(\mathbf{w}_i), \mathbf{w} - \mathbf{w}_i \rangle$
 Solve the reduced problem $\mathbf{w}_t \leftarrow \operatorname{argmin}_{\mathbf{w}} F_t(\mathbf{w})$ where $F_t(\mathbf{w}) = \lambda \Omega(\mathbf{w}) + R_t(\mathbf{w})$
until $F(\mathbf{w}_t) - F_t(\mathbf{w}_t) \leq \varepsilon$;

where $G'(\mathbf{w}) \in \mathbb{R}^n$ is a sub-gradient of G at point $\mathbf{w}$. Thanks to the convexity of $G(\mathbf{w})$, $G_t(\mathbf{w})$ is a piece-wise linear underestimator of $G(\mathbf{w})$ which is tight in the points $\mathbf{w}_i, i = 0, \dots, t-1$. In turn, the reduced problem objective $F_t(\mathbf{w})$ is an underestimator of $F(\mathbf{w})$. The cutting plane model is build iteratively by the following simple procedure. Starting from $\mathbf{w}_0 \in \mathbb{R}^n$, the CPA computes a new iterate $\mathbf{w}_t$ by solving the reduced problem (17). In each iteration t , the cutting-plane model (18) is updated by a new cutting plane computed at the intermediate solution $\mathbf{w}_t$ leading to a progressively tighter approximation of $F(\mathbf{w})$. The CPA halts if the gap between $F(\mathbf{w}_t)$ and $F_t(\mathbf{w}_t)$ gets below a prescribed $\varepsilon > 0$, meaning that $F(\mathbf{w}_t) \leq F(\mathbf{w}^*) + \varepsilon$ holds. The CPA is guaranteed to halt after $\mathcal{O}(\frac{1}{\lambda\varepsilon})$ iterations at most (Teo et al. 2010). The CPA is outlined in Algorithm 1.

We can convert our problem (15) to (16) by setting

$$G(\mathbf{w}) = R_{\text{emp}}(\mathbf{w}, \mathbf{b}(\mathbf{w})) \quad \text{where} \quad \mathbf{b}(\mathbf{w}) = \operatorname{argmin}_{\mathbf{b} \in \mathbb{R}^Y} R_{\text{emp}}(\mathbf{w}, \mathbf{b}). \quad (19)$$

It is clear that if $\mathbf{w}^*$ is a solution of the problem (16) with the function $G(\mathbf{w})$ defined by the Eq. (19) then $(\mathbf{w}^*, \mathbf{b}(\mathbf{w}^*))$ must be a solution of (15). Because $R_{\text{emp}}(\mathbf{w}, \mathbf{b})$ is jointly convex in $\mathbf{w}$ and $\mathbf{b}$, the function $G(\mathbf{w})$ in (19) is also convex in $\mathbf{w}$ (see for example Boyd and Vandenberghe 2004). Hence, application of Algorithm 1 to solve (15) will preserve all its convergence guarantees. To this end, we only need to provide a first-order oracle computing $G(\mathbf{w})$ and the sub-gradient $G'(\mathbf{w})$ required to build the cutting plane model. Given $\mathbf{b}(\mathbf{w})$, the subgradient of $G(\mathbf{w})$ reads (Boyd and Vandenberghe 2004)

$$G'(\mathbf{w}) = \frac{1}{m} \sum_{i=1}^m \mathbf{x}^i (\hat{y}_l^i + \hat{y}_r^i - y_l^i - y_r^i) \quad (20)$$

where

$$\begin{aligned} \hat{y}_l^i &= \operatorname{argmax}_{y \leq y_l^i} [\Delta(y, y_l^i) + \langle \mathbf{w}, \mathbf{x}^i \rangle y - b_y(\mathbf{w})], \\ \hat{y}_r^i &= \operatorname{argmax}_{y \geq y_r^i} [\Delta(y, y_r^i) + \langle \mathbf{w}, \mathbf{x}^i \rangle y - b_y(\mathbf{w})]. \end{aligned}$$

The proposed CPA transforms solving of the problem (15) into a sequence of two simpler problems:

1. The reduced problem (17) solved in each iteration of the CPA. The problem (17) is a quadratic program that can be approached via its dual formulation (Teo et al. 2010) having only t variables where t is the number of iterations of the CPA. Since the CPA rarely needs more than a few hundred iterations, the dual of (17) can be solved by off-the-shelf QP libraries.

2. The problem (19) providing $\mathbf{b}(\mathbf{w})$ which is required to compute $G(\mathbf{w}) = R_{\text{emp}}(\mathbf{w}, \mathbf{b}(\mathbf{w}))$ and the sub-gradient $G'(\mathbf{w})$ via Eq. (20). The problem (19) has only Y (the number of labels) variables. Hence it can be approached by generic convex solvers like the Analytic Center Cutting Plane algorithm (Gondzio et al. 1996).

Because we use another cutting plane method in the inner loop to implement the first-order oracle, we call the proposed solver as the *double-loop CPA*.

Finally, we point out that the convex problems associated with the II-SVOR-EXP and the II-SVOR-IMC can be solved by a similar method. The only change is in using additional constraints $\theta \in \hat{\Theta}$ in (15) which propagate to the problem (19).

5 Experiments

We evaluate the proposed methods on a real-life computer vision problem of estimating age of a person from his/her facial image. The age estimation is a prototypical problem calling for the ordinal classification as well as learning from interval annotations. The set of labels corresponds to individual ages which form an ordered set. Training examples of the facial images are cheap, for example, they can be downloaded from the Internet. On the other hand obtaining ground truth age for a given facial image is often very complicated for obvious reasons. A typical solution used in practice is to annotate the age manually and use it as a replacement for the true age. Creating a year-precise annotation manually is however tedious process. Moreover, precision and consistency of manual annotations can be large. Using the interval annotation instead of the year-precise one can significantly ease the mentioned problems. We demonstrate on real-life data that the proposed methods can effectively exploit cheap interval annotations for learning precise age estimators.

The experiments have two parts. First, in Sect. 5.2, we present results on precisely annotated examples. By conducting these experiments we i) set a baseline for the latter experiments on partially annotated examples, ii) numerically verify that the VILMA subsumes the SVOR-IMC algorithm as a special case and iii) justify usage of the proposed double-loop CPA. Second, in Sect. 5.3, we thoroughly analyze performance of the VILMA on partially annotated examples. We emphasize that all tested algorithms are designed to optimize the MAE loss being the standard evaluation metric of age estimation systems.

5.1 Databases and implementation details

We use two large face databases with year-precise annotation of the age:

1. MORPH database (Ricanek and Tesafaye 2006) is the standard benchmark for age estimation. It contains 55,134 face images with exact age annotation ranging from 16 to 77 years. Because the age category 70+ is severely under-represented (only 9 examples in total) we removed faces with age higher than 70. The database contains frontal police mugshots taken under controlled conditions. The images have resolution 200×240 pixels and most of them are of very good quality.
2. WILD database is a collection of three public databases: Labeled Faces in the Wild (Huang et al. 2007), PubFig (Kumar et al. 2009) and PAL (Minear and Park 2004). The images are annotated by several independent annotators. We selected a subset of near-frontal images (yaw angle in $[-30^\circ, 30^\circ]$) containing 34,259 faces in total with the age from 1 to 80 years. The WILD database contains challenging “in-the-wild” images

exhibiting a large variation in the resolution, illumination changes, race and background clutter.

The faces were split randomly three times into training, validation and testing part in the ratio 60/20/20. We made sure that images of the same identity never appear in different parts simultaneously.

Preprocessing The feature representation of the facial images was computed as follow. We first localized the faces by a commercial face detector² and consequently applied a Deformable Part Model based detector (Uříčář et al. 2012) to find facial landmarks like the corners of eyes, mouth and tip of the nose. The found landmarks were used to transform the input face by an affine transform into its canonical pose. Finally, the canonical face of size 60×40 pixels was described by multi-scale LBP descriptor (Sonnenburg and Franc 2010) resulting in $n = 159$, 488-dimensional binary sparse vector serving as an input of the ordinal classifier.

Implementation of the solver We implemented the double-loop CPA and the standard CPA in C++ by modifying the code from the Shogun machine learning library (Sonnenburg et al. 2010). To solve internal problem (19) we used the Oracle Based Optimization Engine (OBOE) implementation of the Analytic Center Cutting Plane algorithm being a part of COMputational INfrastructure for Operations Research project (COIN-OR) (Gondzio et al. 1996).

5.2 Supervised setting

The purpose of experiments conducted on fully supervised data is three fold. First, to present results of the standard supervised setting which is later used as a baseline. Second, to numerically verify Proposition 2 which states that the VILMA instantiated for the MAE loss subsumes the SVOR-IMC algorithm. Third, to show that imposing an extra quadratic regularization on the biases $\mathbf{b}$ of the MORD rule (11) severely harms the results which justifies usage the proposed double-loop CPA.

Here we used images with the year-precise age annotations from the MORPH database. We constructed a sequence of training sets with the number of examples m varying from $m = 3300$ to $m = 33,000$ (the total number of training example in the MORPH). For each training set we learned the ordinal classifier with the regularization parameters set to $\lambda \in \{1, 0.1, 0.01, 0.001\}$. The classifier corresponding to λ with the smallest validation MAE was applied on the testing examples. This process was repeated for the three random splits. We report the averages and the standard deviations of the MAE computed on the test examples over the three splits. The same evaluation procedure was used for the three compared algorithms: (i) the proposed method VILMA, (ii) the standard SVOR-IMC and (iii) the VILMA-REG which solves the problem (13) but using the regularization term $\frac{\lambda}{2}(\|\mathbf{w}\|^2 + \|\mathbf{b}\|^2)$ instead of $\frac{\lambda}{2}\|\mathbf{w}\|^2$. We used the double-loop CPA for the VILMA and the SVOR-IMC and the standard CPA for the VILMA-REG. Table 1 summarizes the results.

We observe that the prediction error steeply decreases with adding new precisely annotated examples. The MAE for the largest training set is 4.55 ± 0.02 which closely matches the state-of-the-art methods like Guo and Mu (2010) reporting MAE 4.45 on the same database. The next section shows that similar results can be obtained with cheaper partially annotated examples.

² Courtesy of Eydea Recognition Ltd, www.eydea.cz.

Table 1 The test MAE of the ordinal classifier learned from the precisely annotated examples by the VILMA, the standard SVOR-IMC and the VILMA-REG using the $\frac{\lambda}{2}(\|\mathbf{w}\|^2 + \|\mathbf{b}\|)$ regularizer

	$m = 3300$	$m = 6600$	$m = 13,000$	$m = 23,000$	$m = 33,000$
VILMA	5.56 ± 0.02	5.12 ± 0.02	4.83 ± 0.02	4.66 ± 0.01	4.55 ± 0.02
SVOR-IMC	5.56 ± 0.03	5.14 ± 0.02	4.83 ± 0.01	4.68 ± 0.03	4.54 ± 0.01
VILMA-REG	9.57 ± 0.03	9.21 ± 0.06	9.07 ± 0.05	9.04 ± 0.05	9.06 ± 0.02

The results are shown for the training sets generated from the MORPH database by randomly selecting different number of the training examples m

Although the VILMA and the SVOR-IMC learn different parametrizations of the ordinal classifier the resulting rules are equivalent up a numerical error as predicted by Proposition 2. We repeated the same experiment applying the VILMA and the II-SVOR-IMC on the partially annotated examples as described in the next section. The results of both methods were the same up a numerical error. Hence in the next section we only include the results for the VILMA.

The test MAE of the classifier learned by the VILMA-REG is almost doubled compared to the classifier learned by VILMA via the double-loop CPA. This shows that pushing the biases $\mathbf{b}$ towards zero by the quadratic regularizer, which is necessary if the standard CPA is to be used, has a detrimental effect on the accuracy.

5.3 Learning from partially annotated examples

The goal is to evaluate the VILMA when it is applied to learning from partially annotated examples. The MORPH and WILD databases contain the year-precise annotation. The partial annotation is generated in a way which simulates a practical setting:

- m_P randomly selected examples were annotated precisely by taking the annotation from the databases.
- m_I randomly selected examples were annotated by intervals. The admissible annotation intervals were chosen so that they partition the set of ages and have the same width (up to the border cases). The interval width was varied from $u \in \{5, 10, 20\}$. The interval annotation was obtained by rounding the true age from the databases to the admissible intervals. For example, in case of $(u = 5)$ -years wide intervals the true ages $y \in \{1, 2, \dots, 5\}$ were transformed to the interval annotation $[1, 5]$, the ages $y \in \{6, 7, \dots, 10\}$ to $[6, 10]$ and so on.

The described annotation process is approximately $\alpha\beta$ -precise (c.f. Definition 4) with $\alpha = \frac{m_P}{m_P + m_I}$ and $\beta = u - 1 \in \{4, 9, 19\}$. We varied $m_P \in \{3300, 6600\}$ and m_I from 0 to $m_{\text{total}} - m_P$ where m_{total} is the total number of the training examples in the corresponding database.

For each training set we run the VILMA with the regularization constant λ set to $\{1, 0.1, 0.01, 0.001\}$ and selected the best value according to the MAE computed on the validation examples. The best model was then evaluated on the test part. This process was repeated for the three random splits. The reported errors are the averages and the standard deviations of the MAE computed on the test examples. The results are summarized in Fig. 3 and Table 2.

We observe that adding the partially annotated examples monotonically improves the accuracy. This observation holds true for all tested combinations of m_I , m_P , u and both

Fig. 3 The figures show test MAE for the ordinal classifiers learned by the VILMA from different training sets. The x-axis corresponds to the total number of examples in the training set. In the case of partial annotation, x-axis corresponds $m_p + m_l$ where m_p is the number of partial and m_l the number of precisely annotated examples, respectively. The figures **a**, **c** show results for $m_p = 3300$ and figures **b**, **d** for $m_p = 6600$, respectively. In the supervised case, the x-axis is just the number of precisely annotated examples. Each figure shows one curve for the supervised setting plus three curves corresponding to the partial setting with different width $u \in \{5, 10, 20\}$ of the annotation intervals. The results for MORPH database are in figures **a**, **b** and the results for WILD in **c**, **d**. **a** MORPH— $m_p = 3300$ precisely annotated. **b** MORPH— $m_p = 6600$ precisely annotated. **c** WILD— $m_p = 3300$ precisely annotated. **d** WILD— $m_p = 6600$ precisely annotated

databases. This observation is of a great practical importance. It suggests that adding cheap partially annotated examples only improves and never worsens the accuracy of the ordinal classifier.

It is seen that the improvement caused by adding the partially annotated examples can be substantial. Not surprisingly the best results are obtained for the annotation with the narrowest (5-years) intervals. In this case, the performance of the classifier learned from the partial annotations closely matches the supervised setting. In particular, the loss in accuracy resulting from using the partial annotation on the WILD database is on the level of standard deviation. Even in the most challenging case, when learning from 20-years wide intervals, the results are practically useful. For example, to get classifier with ≈ 9 MAE on the WILD database one can either learn from $\approx 12,000$ precisely annotated examples or instead from 6,600 precisely annotated plus 14,400 partially annotated with 20-years wide intervals.

Table 2 The table summarizes test MAE of the ordinal classifier learned from the training set with m examples

	Supervised	$m = 3300$ 5.56 ± 0.02	$m = 6600$ 5.12 ± 0.02	$m = 13,000$ 4.83 ± 0.02	$m = 23,000$ 4.66 ± 0.01	$m = 33,000$ 4.55 ± 0.02
<i>MORPH</i>						
m_P	u	$m_I = 0$	$m_I = 3300$	$m_I = 9700$	$m_I = 19,700$	$m_I = 29,700$
3300	5	5.56 ± 0.02	5.21 ± 0.04	4.89 ± 0.03	4.70 ± 0.01	4.62 ± 0.01
	10	5.56 ± 0.03	5.25 ± 0.02	5.15 ± 0.05	4.97 ± 0.01	4.90 ± 0.04
	20	5.56 ± 0.03	5.32 ± 0.03	5.26 ± 0.06	5.06 ± 0.04	4.97 ± 0.01
m_P	u	$m_I = 0$	$m_I = 0$	$m_I = 6400$	$m_I = 16,400$	$m_I = 26,400$
6600	5	—	5.12 ± 0.02	4.86 ± 0.02	4.69 ± 0.00	4.61 ± 0.00
	10	—	5.13 ± 0.02	4.96 ± 0.03	4.81 ± 0.01	4.84 ± 0.04
	20	—	5.13 ± 0.02	5.03 ± 0.02	4.86 ± 0.04	4.86 ± 0.01
	Supervised	$m = 3300$ 10.40 ± 0.03	$m = 6600$ 9.60 ± 0.03	$m = 11,000$ 9.14 ± 0.02	$m = 16,000$ 8.89 ± 0.02	$m = 21,000$ 8.68 ± 0.02
<i>WILD</i>						
m_P	u	$m_I = 0$	$m_I = 3300$	$m_I = 7700$	$m_I = 12,700$	$m_I = 17,700$
3300	5	10.40 ± 0.03	9.69 ± 0.02	9.23 ± 0.05	8.89 ± 0.02	8.71 ± 0.02
	10	10.40 ± 0.03	9.76 ± 0.02	9.42 ± 0.04	9.09 ± 0.02	8.99 ± 0.02
	20	10.40 ± 0.03	9.88 ± 0.03	9.67 ± 0.04	9.51 ± 0.00	9.40 ± 0.01
m_P	u	$m_I = 0$	$m_I = 0$	$m_I = 4400$	$m_I = 9400$	$m_I = 14,400$
6600	5	—	9.60 ± 0.03	9.22 ± 0.06	8.89 ± 0.02	8.71 ± 0.02
	10	—	9.60 ± 0.03	9.22 ± 0.02	9.04 ± 0.03	8.90 ± 0.02
	20	—	9.60 ± 0.03	9.35 ± 0.06	9.14 ± 0.03	9.04 ± 0.02

The upper row shows results of the supervised setting when all m examples are precisely annotated. The bottom rows show results of learning from m_P precisely annotated examples and $m_I = m - m_P$ examples annotated by intervals of width u

Finally, let us define a quantity $\gamma(\alpha, \beta) = \hat{R}^{MAE}(h^{\alpha, \beta}) - \hat{R}^{MAE}(h^*)$ where $\hat{R}^{MAE}(\cdot)$ denotes the test MAE, $h^{\alpha, \beta}$ is the classifier learned from partially annotated examples generated by the $\alpha\beta$ -precise annotation process and h^* is the classifier learned from the precise annotations only. The quantity $\gamma(\alpha, \beta)$ thus measures the loss in test accuracy caused by using the imprecise annotation. The values of $\gamma(\alpha, \beta)$ observed on both databases are shown in Fig. 4. We see that the loss in accuracy grows proportionally with the interval width $u = 1 + \beta$ and with the portion of partially annotated examples $1 - \alpha$. This observation complies with the theoretical upper bound $\gamma(\alpha, \beta) \leq (1 - \alpha)\beta$ given in Theorem 2. Although the slope of real curve $\gamma(\alpha, \beta)$, if seen as a function of $1 - \alpha$, is considerably smaller than β , the tendency is approximately linear at least in the regime $1 - \alpha \in [0, 0.5]$ as predicted by the theoretical bound.

6 Conclusions and future work

We have proposed a V-shaped interval-insensitive loss suitable for risk minimization based learning of ordinal classifiers from partially annotated examples. We proved that under rea-

Fig. 4 The figures show $\gamma(\alpha, \beta) = \hat{R}^{MAE}(h^{\alpha, \beta}) - \hat{R}^{MAE}(h^*)$ which is the loss in accuracy caused by training from partially annotated examples generated by $\alpha\beta$ -precise annotation process relatively to the supervised case. The value of $\gamma(\alpha, \beta)$ is shown for different β (note that $u = \beta + 1$ is the interval width) as a function of the portion of the partially annotated examples $1 - \alpha$. The figure **a**, **b** contains the results obtained on the MORPH and the WILD database, respectively

sonable assumption on the annotation process the Bayes risk of the ordinal classifier can be bounded by the expectation of the associated interval-insensitive loss. We proposed a convex surrogate of the interval-insensitive loss associated to an arbitrary supervised V-shaped loss. We derived a generic V-shaped Interval insensitive Loss Minimization Algorithm (VILMA) which translates learning from the interval annotations to a convex optimization problem. We also derived other convex surrogate losses of the interval insensitive loss by extending the existing methods like the SVOR-EXP and SVOR-IMC algorithm. We have proposed a cutting plane method which can solve large instances of the resulting convex learning problems. The experiments conducted on a real-life problem of human age estimation from facial images show that the proposed method has a practical potential. We demonstrated that a precise ordinal classifier with accuracy matching the state-of-the-art results can be obtained by learning from cheap partial annotations.

Our work is based on the interval insensitive loss and its convex surrogates which turned out to work well empirically. We showed that under certain assumptions the expectation of the interval insensitive loss can be used to upper bound expectation of the associated target loss. However a deeper theoretical understanding is needed. For example, an open issue is whether there exists a distribution for which the upper bound is sharp. Another interesting question is how to weaken the assumptions on the annotation process, e.g. the requiring on the consistency of the annotation. It is also unclear which of the introduced convex surrogates is theoretically better. We believe that this issue could be resolved by analyzing statistical consistency of the surrogates as in (Zhang 2004; Tewari and Bartlett 2007). These issues are left for the future work.

Acknowledgments The authors were supported by the Grant Agency of the Czech Republic under Project P202/12/2071, the project ERC-CZ 1303 and EU project FP7-ICT-609763 TRADR.

Appendix 1: Proof of Theorem 2

We will prove the bound (8) for each observation $\mathbf{x} \in \mathcal{X}$ separately, that is, we prove

$$R^{MAE}(h | \mathbf{x}) \leq R_I^{MAE}(h | \mathbf{x}) + (1 - \alpha)\beta, \quad (21)$$

where $R^{MAE}(h \mid \mathbf{x}) = \mathbb{E}_{y \sim p(y \mid \mathbf{x})} |y - h(\mathbf{x})|$ and

$$R_I^{MAE}(h \mid \mathbf{x}) = \mathbb{E}_{[y_l, y_r] \sim p(y_l, y_r \mid \mathbf{x})} \min_{y' \in [y_l, y_r]} |y' - h(\mathbf{x})|.$$

It is clear that (21) satisfied for all $\mathbf{x} \in \mathcal{X}$ implies (8). Let us define a function which measures a discrepancy between the MAE and the its interval insensitive counterpart:

$$\delta(h(\mathbf{x}), y, y_l, y_r) = |y - h(\mathbf{x})| - \min_{y' \in [y_l, y_r]} |y' - h(\mathbf{x})| = \begin{cases} |h(\mathbf{x}) - y| & \text{if } h(\mathbf{x}) \in [y_l, y_r], \\ y - y_l & \text{if } h(\mathbf{x}) < y_l, \\ y_r - y & \text{if } h(\mathbf{x}) > y_r. \end{cases} \quad (22)$$

Let us denote a set of intervals of unit length as $\mathcal{P}_1 = \{[y_l, y_r] \in \mathcal{P} \mid y_l = y_r\}$. Recall also that due to the assumption that $p(y_l, y_r \mid \mathbf{x}, y)$ is consistent and $\alpha\beta$ -precise then we have $p(y, y \mid \mathbf{x}, y) = \alpha$ and $\sum_{[y_l, y_r] \in \mathcal{P}_1} p(y_l, y_r \mid \mathbf{x}, y) = (1 - \alpha)$. With these definitions we can write the following chain of equations:

$$\begin{aligned} R_I^{MAE}(h \mid \mathbf{x}) &= \sum_{y \in \mathcal{Y}} \sum_{[y_l, y_r] \in \mathcal{P}} p(y \mid \mathbf{x}) p(y_l, y_r \mid \mathbf{x}, y) \min_{y' \in [y_l, y_r]} |y' - h(\mathbf{x})| \\ &= \sum_{y \in \mathcal{Y}} p(y \mid \mathbf{x}) \left[\alpha |y - h(\mathbf{x})| + \sum_{[y_l, y_r] \notin \mathcal{P}_1} p(y_l, y_r \mid \mathbf{x}, y) \min_{y' \in [y_l, y_r]} |y' - h(\mathbf{x})| \right] \\ &= \sum_{y \in \mathcal{Y}} p(y \mid \mathbf{x}) \left[\alpha |y - h(\mathbf{x})| + \sum_{[y_l, y_r] \notin \mathcal{P}_1} p(y_l, y_r \mid \mathbf{x}, y) (|y - h(\mathbf{x})| - \delta(h(\mathbf{x}), y, y_l, y_r)) \right] \\ &= \sum_{y \in \mathcal{Y}} p(y \mid \mathbf{x}) \left[|y - h(\mathbf{x})| - \sum_{[y_l, y_r] \notin \mathcal{P}_1} p(y_l, y_r \mid \mathbf{x}, y) \delta(h(\mathbf{x}), y, y_l, y_r) \right] \\ &= R^{MAE}(h \mid \mathbf{x}) - \sum_{y \in \mathcal{Y}} \sum_{[y_l, y_r] \notin \mathcal{P}_1} p(y \mid \mathbf{x}) p(y_l, y_r \mid \mathbf{x}, y) \delta(h(\mathbf{x}), y, y_l, y_r). \end{aligned} \quad (23)$$

By (22) we have that $\delta(h(\mathbf{x}), y, y_l, y_r) \leq \beta$ for all $\mathbf{x} \in \mathcal{X}$, $y \in \mathcal{Y}$, $[y_l, y_r] \in \mathcal{P}$ and hence

$$\sum_{y \in \mathcal{Y}} \sum_{[y_l, y_r] \notin \mathcal{P}_1} p(y \mid \mathbf{x}) p(y_l, y_r \mid \mathbf{x}, y) \delta(h(\mathbf{x}), y, y_l, y_r) \leq (1 - \alpha)\beta. \quad (24)$$

The bound (21) to be proved is obtained immediately by combing (23) and (24).

Appendix 2: Conversion formulas between the standard and the MORD parametrization

Given parameters of the ordinal classifier (1) $\mathbf{w} \in \mathbb{R}^n$, $\boldsymbol{\theta} \in \Theta$, the equivalent MORD classifier (11) is determined by the same parameter vector $\mathbf{w}$ and the vector of intercepts $\mathbf{b}$ computed according to

$$b_1 = 0 \quad \text{and} \quad b_y = - \sum_{i=1}^{y-1} \theta_i, \quad y = 2, \dots, Y. \quad (25)$$

The conversion from the MORD classifier (11) to the ordinal classifier (1) is done differently for the non-generated and the degenerated classifier.

Classifier $h(\mathbf{x})$ is non-degenerated iff all classes are non-degenerated. In opposite case the classifier is called degenerated. We call class $y \in \mathcal{Y}$ non-degenerated for classifier $h'(\mathbf{x})$ iff

$\mathcal{X}_y = \text{interior}(\{\mathbf{x} \in \mathcal{X} : h'(\mathbf{x}) = y\}) \neq \emptyset$. For more details on (non-)degenerated classifiers we refer reader to [Antoniuk et al. \(2013\)](#).

Given parameters of a non-degenerated MORD classifier (11) $\mathbf{w} \in \mathbb{R}^n$ and $\mathbf{b} \in \mathbb{R}^Y$, we can compute thresholds $\theta \in \Theta$ of the equivalent ordinal classifier (1) by

$$\theta_y = b_y - b_{y+1}, \quad y = 1, \dots, Y-1. \quad (26)$$

Given parameters of a degenerated MORD classifier (11) $\mathbf{w} \in \mathbb{R}^n$ and $\mathbf{b} \in \mathbb{R}^Y$, we compute thresholds $\theta \in \Theta$ of the equivalent ordinal classifier (1) by

$$\theta_{y_i} = \dots = \theta_{y_{i+1}-1} = \frac{b_{y_i} - b_{y_{i+1}}}{(y_{i+1} - y_i)}, \quad i = 1, \dots, p, \quad (27)$$

where $y_i \in \mathcal{Y}$, $i = 1, \dots, p$ is an increasing subsequence of non-degenerated classes.

Appendix 3: Proof of Proposition 1

Let us first consider triplet of labels (y, y_l, y_r) such that $y \notin [y_l, y_r]$. In this case, the left max-term $\max_{y \leq y_l} [\Delta(y, y_l) + \langle \mathbf{x}, \mathbf{w} \rangle (y - y_l) + b_y - b_{y_l}]$ appearing in the surrogate (12) is an instance of margin-rescaling loss instantiated for the supervised loss $\Delta(y, y_l)$ defined on labels $y \in [1, y_l - 1]$. The margin-rescaling loss is known to be an upper bound of the respective supervised loss ([Tsochantaridis et al. 2005](#)) and hence it upper bounds $\Delta_I(y_l, y_r, y)$. Analogically, we can see that the right max-term $\max_{y \geq y_r} [\Delta(y, y_r) + \langle \mathbf{x}, \mathbf{w} \rangle (y - y_r) + b_y - b_{y_r}]$ of (12) is a margin-rescaling upper bound of the loss $\Delta(y, y_r)$ defined on labels $y \in [y_r + 1, Y]$ and hence also an upper bound of $\Delta_I(y_l, y_r, y)$. The V-shaped loss $\Delta(y, y')$ is non-negative by definition and hence both max-terms are also non-negative and their sum upper bounds the value of $\Delta_I(y_l, y_r, y)$ for $y \notin [y_l, y_r]$. In the case when $y \in [y_l, y_r]$ the value of $\Delta_I(y_l, y_r, y)$ is defined to be zero and hence it cannot be greater than a sum of the non-negative max-terms.

Appendix 4: Proof of Proposition 2

To prove proposition it is enough to show that following equalities hold

$$\sum_{\hat{y}=1}^{y_l-1} \max(0, 1 - \langle \mathbf{x}, \mathbf{w} \rangle + \theta_{\hat{y}}) = \max_{y \leq y_l} \left[y_l - y + \langle \mathbf{x}, \mathbf{w} \rangle (y - y_l) + b_y - b_{y_l} \right], \quad (28)$$

$$\sum_{\hat{y}=y_r}^{Y-1} \max(0, 1 + \langle \mathbf{x}, \mathbf{w} \rangle - \theta_{\hat{y}}) = \max_{y \geq y_r} \left[y - y_r + \langle \mathbf{x}, \mathbf{w} \rangle (y - y_r) + b_y - b_{y_r} \right]. \quad (29)$$

We will provide proof only for the equality (28), proof for the equality (29) is similar and thus omitted. Let us do some algebra on (28):

$$\sum_{\hat{y}=1}^{y_l-1} \max(0, 1 - \langle \mathbf{x}, \mathbf{w} \rangle + \theta_{\hat{y}}) \stackrel{(a)}{=} \max_{y \leq y_l} \left[\max(0, \sum_{\hat{y}=y}^{y_l-1} 1 - \langle \mathbf{x}, \mathbf{w} \rangle + \theta_{\hat{y}}) \right] \quad (30a)$$

$$\stackrel{(b)}{=} \max_{y \leq y_l} \left[\max(0, y_l - y + \langle \mathbf{x}, \mathbf{w} \rangle (y - y_l) + \sum_{\hat{y}=y}^{y_l-1} \theta_{\hat{y}}) \right] \quad (30b)$$

$$\stackrel{(c)}{=} \max_{y \leq y_l} \left[\max(0, y_l - y + \langle \mathbf{x}, \mathbf{w} \rangle (y - y_l) + b_y - b_{y_l}) \right] \quad (30c)$$

$$\stackrel{(d)}{=} \max_{y \leq y_l} \left[y_l - y + \langle \mathbf{x}, \mathbf{w} \rangle (y - y_l) + b_y - b_{y_l} \right]. \quad (30d)$$

Equality (a) holds since $\theta_y, y = 1, \dots, Y$ is nondecreasing sequence. Equality (c) is possible due to conversion formulas³ (25), (26). Since $y_l - y + \langle \mathbf{x}, \mathbf{w} \rangle (y - y_l) + b_y - b_{y_l} \geq 0, \forall y \leq y_l$ internal maximum in the equality (d) is redundant and thus can be omitted.

References

- Antoniuk, K., Franc, V., & Hlavac, V. (2013). Mord: Multi-class classifier for ordinal regression. In *Proceedings of European conference on machine learning and principles and practice of knowledge discovery in databases (ECML/PKDD)* (pp. 96–111).
- Boyd, S., & Vandenberghe, L. (2004). *Convex optimization*. New York, NY: Cambridge University Press.
- Chang, K., Chen, C., & Hung, Y. (2011). Ordinal hyperplane ranker with cost sensitivities for age estimation. In *Proceedings of computer vision and pattern recognition (CVPR)*.
- Chu, W., & Ghahramani, Z. (2005). Preference learning with gaussian processes. In *Proceedings of the international conference on machine learning (ICML)*.
- Chu, W., & Keerthi, S.S. (2005). New approaches to support vector ordinal regression. In *Proceedings of the international conference on machine learning (ICML)* (pp. 145–152).
- Cour, T., Sapp, B., & Taskar, B. (2011). Learning from partial labels. *Journal of Machine Learning Research*, 12, 1225–1261.
- Crammer, K., & Singer, Y. (2001). Pranking with ranking. In *Advances in neural information processing systems (NIPS)* (pp. 641–647).
- Dembczyński, K., Kotłowski, W., & Slowinski, R. (2008). Ordinal classification with decision rules. In *Mining complex data. Lecture notes in computer science*, 4944, 169–181.
- Dempster, A., Laird, N., & Rubin, D. (1997). Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society*, 1(39), 1–38.
- Do, T.-M.-T., & Artières, T. (2009). Large margin training for hidden markov models with partially observed states. In *Proceedings of the international conference on Machine Learning (ICML)*.
- Franc, V., Sonnenburg, S., & Werner, T. (2012). *Cutting-plane methods in machine learning* (chapter 7, pp. 185–218). The MIT Press, Cambridge, USA.
- Fu, L., & Simpson, D.G. (2002). Conditional risk models for ordinal response data: Simultaneous logistic regression analysis and generalized score test. *Journal of Statistical Planning and Inference*, 108(1–2), 201–217.
- Gondzio, J., du Merle, O., Sarkissian, R., & Vial, J.-P. (1996). ACCPM—A library for convex optimization based on an analytic center cutting plane method. *European Journal of Operational Research*, 94, 206–211.
- Guo, G., & Mu, G. (2010). Human age estimation: What is the influence across race and gender? In *Proceedings of conference on computer vision and pattern recognition workshops (CVPRW)*.
- Huang, G.B., Ramesh, M., Berg, T., & Learned-Miller, E. (2007). *Labeled faces in the wild: A database for studying face recognition in unconstrained environments*. Technical report 07-49, University of Massachusetts, Amherst.
- Jie, L., & Orabona, F. (2010). Learning from candidate labeling sets. In *Proceedings of advances in neural information processing systems (NIPS)*.
- Kotłowski, W., Dembczynski, K., Greco, S., & Slowinski, R. (2008). Stochastic dominance-based rough set model for ordinal classification. *Journal of Information Sciences*, 178(21), 4019–4037.
- Kumar, N., Berg, A. C., Belhumeur, P. N., & Nayar, S. K. (2009). Attribute and simile classifiers for face verification. In *Proceedings of international conference on computer vision (ICCV)*.
- Li, L., & Lin, H.-T. (2006). Ordinal regression by extended binary classification. In *Proceedings of advances in neural information processing systems (NIPS)*.

³ Here for simplicity we provide proof in non-degenerated case, it can be adopted however for generated case as well.

- Lou, X., & Hamprecht, F.A. (2012). Structured learning from partial annotations. In *Proceedings of the international conference on machine learning (ICML)*.
- McCullagh, P. (1980). Regression models for ordinal data. *Journal of the Royal Statistical Society*, 42(2), 109–142.
- Minear, M., & Park, D. (2004). A lifespan database of adult facial stimuli. *Behavior Research Methods, Instruments, & Computers: A Journal of the Psychonomic Society*, 36, 630–633.
- Ramanathan, N., Chellappa, R., & Biswas, S. (2009). Computational methods for modeling facial aging: A survey. *Journal of Visual Languages and Computing*, 20, 131–144.
- Rennie, J. D., & Srebro, N. (2005). Loss functions for preference levels: Regression with discrete ordered labels. In *Proceedings of the IJCAI multidisciplinary workshop on advances in preference handling*.
- Ricanek, K., & Tesafaye, T. (2006). Morph: A longitudinal image database of normal adult age-progression. In *Proceedings of automated face and gesture recognition*.
- Schlesinger, M. (1968). A connection between learning and self-learning in the pattern recognition (in Russian). *Kibernetika*, 2, 81–88.
- Shashua, A., & Levin, A. (2002). Ranking with large margin principle: Two approaches. In *Proceedings of advances in neural information processing systems (NIPS)*.
- Sonnenburg, S., & Franc, V. (2010). Coffin: A computational framework for linear SVMs. In *Proceedings of the international conference on machine learning (ICML)*.
- Sonnenburg, S., Rätsch, G., Henschel, S., Widmer, C., Behr, J., Zien, A., et al. (2010). The shogun machine learning toolbox. *Journal of Machine Learning Research*, 11, 1799–1802.
- Teo, C. H., Vishwanathan, S., Smola, A. J., & Le, Q. V. (2010). Bundle methods for regularized risk minimization. *Journal of Machine Learning Research*, 11, 311–365.
- Tewari, A., & Bartlett, P. (2007). On the consistency of multiclass classification methods. *Journal of Machine Learning Research*, 8, 1007–1025.
- Tsochantaridis, I., Joachims, T., Hofmann, T., Altun, Y., & Singer, Y. (2005). Large margin methods for structured and interdependent output variables. *Journal of Machine Learning Research*, 6, 1453–1484.
- Uříčář, M., Franc, V., & Hlaváč, V. (2012). Detector of facial landmarks learned by the structured output SVM. In *Proceedings of the international conference on computer vision theory and applications (VISAPP)* (Vol. 1, pp. 547–556).
- Vapnik, V. N. (1998). *Statistical learning theory. Adaptive and learning systems*. New York, New York: Wiley.
- Zhang, T. (2004). Statistical behaviour and consistency of classification methods based on convex risk minimization. *Annals of Statistics*, 31(1), 56–134.