Robust Capped ℓ_p-Norm Support Vector Ordinal Regression

Haorui Xiang[a,b], Zhichang Wu[a,b], Guoxu Li[a], Rong Wang[a], Feiping Nie[a,b,*], Xuelong Li[b]

[a]*School of Computer Science and School of Artificial Intelligence, Optics and Electronics (iOPEN), Northwestern Polytechnical University, Xi'an 710072, Shaanxi, P. R. China.*
[b] *Institute of Artificial Intelligence (TeleAI), China Telecom Corp Ltd, 31 Jinrong Street, Beijing 100033, P. R. China*

Abstract

Ordinal regression is a specialized supervised problem where the labels show an inherent order. The order distinguishes it from normal multi-class problem. Support Vector Ordinal Regression, as an outstanding ordinal regression model, is widely used in many ordinal regression tasks. However, like most supervised learning algorithms, the design of SVOR is based on the assumption that the training data are real and reliable, which is difficult to satisfy in real-world data. In many practical applications, outliers are frequently present in the training set, potentially leading to misguide the learning process, such that the performance is non-optimal. In this paper, we propose a novel capped ℓ_p-norm loss function that is theoretically robust to both light and heavy outliers. The capped ℓ_p-norm loss can help the model detect and eliminate outliers during training process. Adhering to this concept, we introduce a new model, Capped ℓ_p-Norm Support Vector Ordinal Regression(CSVOR), that is robust to outliers. CSVOR uses a weight matrix to detect and eliminate outliers during the training process to improve the robustness to outliers. Moreover, a Re-Weighted algorithm algorithm which is illustrated convergence by our theoretical results is proposed to effectively minimize the corresponding problem. Extensive experimen-

*Corresponding author

Email addresses: haoruixiang@mail.nwpu.edu.cn (Haorui Xiang), wks0774@mail.nwpu.edu.cn (Zhichang Wu), lgx2683109120@gmail.com (Guoxu Li), wangrong07@tsinghua.org.cn (Rong Wang), feipingnie@gmail.com (Feiping Nie), li@mail.nwpu.edu.cn (Xuelong Li)

tal results demonstrate that our model outperforms state-of-the-art(SOTA) methods, particularly in the presence of outliers.

Keywords: Ordinal Regression, Support Vector Ordinal Regression, Capped ℓ_p-norm, Robust to Outlier

1. Introduction

Ordinal regression, also known as ordinal classification, is a machine learning task that involves predicting labels with a natural order, distinguishing it from standard multi-class classification tasks [1–3]. In ordinal regression, the order of the labels in the output space corresponds to the order in the input space, allowing models to utilize the inherent order information for more accurate predictions. For instance, when predicting ages, a 9-year-old child's label is more likely to be closer to 8-year-old than to 7-year-old. This characteristic is challenging for standard multi-class methods, as they treat labels as unrelated categories [4, 5]. Ordinal regression finds applications across various domains, including age prediction [6–9], text classification [10, 11], medical research [12, 13], face recognition [14, 15], and social study [16, 17].

In recent years, ordinal regression, a form of supervised learning where the output variable represents a ranking or rating scale, has gained increasing attention, leading to the proposal of many ordinal regression models [15]. One successful model in this domain is Support Vector Ordinal Regression with Explicit Constraints (SVOREX), which is widely utilized in ordinal regression tasks [18, 19]. SVOREX employs the ordinal hinge loss function and incorporates the ordinal relationship of classes into the optimization problem through explicit inequality constraints on the threshold vector $\boldsymbol{b}$. It learns $K-1$ parallel planes to separate the k classes.

As the volume and diversity of available data continue to expand, the likelihood of encountering outliers, such as sensor errors and mislabeled data, also increases [20]. These outliers can significantly impact data analysis and decision-making processes, highlighting the importance of robust techniques [21]. However, like most supervised learning algorithms, SVOREX is designed under the assumption that the training data are real and reliable, which is often challenging to satisfy in real-world datasets. Real-world datasets often contain numerous outliers, which are data points significantly different from others in the same class or data with incorrect labels. The

presence of these outliers can lead SVOREX to find sub-optimal solutions, resulting in a significant reduction in model performance.

To tackle this challenge, we propose a Capped ℓ_p-Norm Support Vector Ordinal Regression (CSVOR) model that utilizes a capped ℓ_p-norm ordinal hinge loss. The capped ℓ_p-norm ordinal hinge loss, unlike the ordinal hinge loss used in SVOREX, is theoretically robust against both light and heavy outliers. Since outliers often have large residuals [22, 23], SVOREX tends to overly focus on outliers, neglecting true samples and thus significantly reducing performance. The capped ℓ_p-norm mitigates this issue by imposing an upper bound on the ordinal hinge loss, eliminating larger residuals and helping the model mitigate the impact of outliers during training. Moreover, we also provide an intuitive explanation of why CSVOR is robust to outliers. CSVOR uses a binary matrix in each iteration to detect outliers with large residuals and then removes these outliers from the iteration to reduce their impact. To the best of our knowledge, this is the first work that attempts to improve the robustness of SVOREX to outliers.

The capped ℓ_p-norm ordinal hinge loss introduces non-convexity and increases the non-smoothness of the objective, making optimization very challenging. To address this challenge, we introduce a Re-Weighted optimization framework to efficiently solve the proposed minimization problem. The Re-Weighted algorithm is an efficient iterative algorithm tailored for solving certain non-convex problems. Our theoretical results guarantee the convergence of the Re-Weighted optimization algorithm.

Our contributions can be summarized concisely in the following three aspects:

1. A novel Capped ℓ_p-Norm Support Vector Ordinal Regression(CSVOR) is proposed by utilizing a capped ℓ_p-Norm ordinal hinge loss function to eliminate data with large residuals during the training process. Moreover, We explain from an intuitive perspective why CSVOR is robust to outliers. A weight matrix D is used to detect and eliminate outliers implicitly during the training process to resist the influence of outliers.
2. A Re-weighted(RW) optimization framework is proposed to efficiently solve the minimization problem, and then it is employed to solve the proposed capped ℓ_p-norm problem. Theoretical results ensure the convergence of the optimization algorithm.
3. Extensive experiments on two artificial datasets and ten benchmark datasets show that our method outperforms state-of-the-art(SOTA)

methods, especially in the presence of outliers. Moreover, we conducted convergence experiments on the proposed algorithm, which showed that our algorithm can converge after a few iterations. In addition, visualization experiments on the commonly used age estimation dataset FG-NET illustrate the potential of the proposed model in anomaly detection.

The remainder of this paper is organized as follows. In Section 2, we briefly review ordinal regression and SVOR models. Section 3 presents the proposed CSVOR model utilizes the capped ℓ_p-norm ordinal hinge loss. An Re-weighted algorithm for solving the optimization problem is introduced in Section 4. In Section 5, we provide some theoretical analysis. In Section 6, we conduct extensive empirical studies on two synthetic datasets and several benchmark datasets to validate the effectiveness of our method. Finally, Section 7 concludes this paper.

2. Related work

2.1. Ordinal Regression

Ordinal regression has been widely researched in recent years. An excellent survey on ordinal regression is presented in [1], which categorizes existing models into three groups.

The first group of methods directly transforms ordinal regression into either multi-class classification or metric regression. For example, Diaz and Amit [4] propose a new encoding scheme that converts ordinal labels into soft probability distributions based on a distance metric. This encoding method pairs well with common classification loss functions such as cross-entropy. Another method, the Moving Window Regression (MWR) method [15], utilizes both local and global regressors to predict the class range of patterns separately. This approach often leads to more accurate predictions compared to other methods.

The second group, known as ordinal binary decompositions, leverages order information to break down the original ordinal regression into multiple binary classification tasks. Frank and Hall [24] propose that an ordinal regression problem with K classes can be transformed into $K-1$ binary classification problems based on whether a pattern $\boldsymbol{x}$ is greater than k. In the work of Waegeman et al. [25], different weights are applied to each sample of each binary classifier so that samples with predicted labels further away

from the true label are more penalized. Perez-Orti'z et al. [26] propose a new method for this decomposition. For intermediate subsets, labels are divided into three groups (less than k, equal to k, and greater than k). For extreme subsets, labels are divided into two categories. This incorporates sequential information into the sub-problems. A method for ordinal regression model by constructing proximal non-parallel hyperplanes is proposed based on the above decomposition method, which can obtain more flexible separate hyperplanes [27].

The third group, referred to as threshold models, is grounded in the general concept of approximating a real-value predictor and subsequently partitioning the real line into intervals. Cumulative Link Models (CLMs) [28], originated from a statistical background, are one of the first models designed specifically for ordinal regression problems. CLMs use a special cumulative probability function to predict the probability of continuous classes. Vargas et al. [6] combine CLMs with the Continuous Quadratic Weighted Kappa (QWK) loss function to propose an ordinal deep network, successfully applying it to tasks such as age prediction and medical image classification. The CORAL method [29] uses the last layer of the neural network to share weights to limit the posterior probability and ensure rank consistency.

There are also recent studies that do not fall into these three categories. For example, Liu et al. [30] use unimodal constraints to reformulate the ordinal information in ordinal regression. Li et al. [31] introduce an efficient unimodal-concentrated loss. This loss maximizes the probability of an instance at its true value in a fully adaptive manner while ensuring a unimodal distribution.

2.2. Support Vector Ordinal Regression(SVOR)

Initially designed for binary classification problems, Support Vector Machines (SVMs) try to find an optimal direction to map feature vectors to function values on the real line and use a single optimization threshold to divide the real line into two regions. Herbrich et al. [32] propose the first ordinal regression algorithm based on SVM, which considers a pairwise approach and constructs a new dataset using difference vectors. The work by Shashua and Levin [32] introduce two methods: maximizing the margin between the closest neighboring classes and maximizing the sum of margins between classes. Both approaches encounter two primary issues: the model is incompletely specified as the thresholds are not uniquely defined, and at the optimal solution, they may not be properly ordered, since the inequality

$(b_1 \leq b_2 \leq \ldots \leq b_{K-1})$ is not included in the formulation. Consequently, Chu and Keerthi [18, 33] present two distinct reformulations of the same concept, addressing the issue of unordered thresholds in the solution. Li and Lin [34] treat the ordinal regression as a special case of cost-sensitive classification and propose a reduction framework for ordinal regression. They then combine this framework and SVM to propose a new support vector ordinal regression model (REDSVM). In recent years, people have begun to pay attention to the issue of noise sensitivity in support vector learning for ordinal regression, where a small amount of noise can greatly impact the performance of models [21]. Zhong et al. [19] combine the pinball loss with SVOR, introducing a novel support vector model for ordinal regression called PINSVOR. This approach mitigates the sensitivity of SVOR to noise near the classification boundary.

2.3. Briefly Review of Support Vector Learning for Ordinal Regression with Explicit Constraints

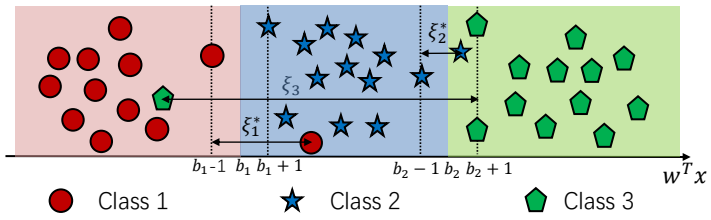


Figure 1: An illustration of the definition of the loss ξ and ξ^*. The samples from different classes,represented as patterns of different colors and shapes, are mapped by $\boldsymbol{w}^T\boldsymbol{x}$ onto the axis of function value.

SVOREX is a widely used ordinal regression method that aims to classify data into K ordered classes by finding $K-1$ parallel classification hyperplanes. It includes a projection vector $\boldsymbol{w}$ and a threshold vector $\boldsymbol{b} = (b_0, b_1, ..., b_K)^T$ which satisfies the constraints $b_1 \leq b_2 \leq \cdots \leq b_{K-1}$ and $b_0 = -\infty$, $b_K = +\infty$. Similar to the binary classification, the projection direction maps feature vectors to values on the real line, and then the threshold vector divides the real line into $K-1$ regions, each corresponding to a class, as shown in Fig. 1. For a data point $(\boldsymbol{x_i}, y_i)$, its mapped value $\boldsymbol{w}^T\boldsymbol{x_i}$ should be assigned to the interval $[b_{y_i-1}+1, b_{y_i}-1]$. If the mapped value is less than the lower bound $b_{y_i-1}+1$ is the error (denoted as $\xi(\boldsymbol{w}, \boldsymbol{b}|\boldsymbol{x_i}, y_i)$),

$b_{y_i-1} + 1 - \boldsymbol{w}^T \boldsymbol{x_i}$ and whereas if the mapped value is greater than the upper bound, $\boldsymbol{w}^T \boldsymbol{x_i} - b_{y_i} - 1$ is the error (denoted as $\xi^*(\boldsymbol{w}, \boldsymbol{b}|\boldsymbol{x_i}, y_i)$). The ordinal hinge loss can be written in the following form:

$$\xi_O(\boldsymbol{w}, \boldsymbol{b}|\boldsymbol{x_i}, y_i) = \xi(\boldsymbol{w}, \boldsymbol{b}|\boldsymbol{x_i}, y_i) + \xi^*(\boldsymbol{w}, \boldsymbol{b}|\boldsymbol{x_i}, y_i), \tag{1}$$

where $\xi(\boldsymbol{w}, \boldsymbol{b}|\boldsymbol{x_i}, y_i) = \max(0, 1 - \boldsymbol{w}^T \boldsymbol{x_i} + b_{y_i-1})$, $\xi^*(\boldsymbol{w}, \boldsymbol{b}|\boldsymbol{x_i}, y_i) = \max(0, 1 + \boldsymbol{w}^T \boldsymbol{x_i} - b_{y_i})$. Similar to many SVM-based methods, SVOREX adopts the l_2 norm of $\boldsymbol{w}$ as the regularization term, which helps the model avoid overfitting. The final optimization problem of SVOREX is as follows:

$$\begin{aligned} &\min_{\boldsymbol{w},\boldsymbol{b}} \sum_{i=1}^{N} \xi_O(\boldsymbol{w}, \boldsymbol{b}|\boldsymbol{x_i}, y_i) + \gamma ||\boldsymbol{w}||_2^2 \\ &\text{s.t.} \quad b_{k-1} \leq b_k \quad k = 1, 2, \ldots K, \end{aligned} \tag{2}$$

where γ is regularization parameter. Once the optimal values of $\boldsymbol{w}$ and $\boldsymbol{b}$ in problem (2) are determined, the prediction rule is defined as follows:

$$r(\boldsymbol{x}) = \sum_{k=1}^{K-1} [[\boldsymbol{w}^T \boldsymbol{x} - b_k \geq 0]] + 1, \tag{3}$$

where $[[\cdot]]$ is an indicator function the output is 1 if the inner condition holds and 0 otherwise.

3. Motivation and Proposed methodology

Despite its significant success, the sensitivity of SVOREX to outliers can not be ignored. Outliers usually have larger residuals and their presence affects model performance [22, 23]. From Fig. 2, it can be observed that when a data point $\boldsymbol{x}$ is misclassified, the loss can grow infinitely large. Therefore, the ordinal hinge loss is not robust enough to data outliers. To address this issue, we propose a capped ℓ_p-norm ordinal hinge loss as a robust and stable loss function to mitigate the impact of outliers. We let $\boldsymbol{\xi_u}(\boldsymbol{w}, \boldsymbol{b}|\boldsymbol{x_i}, y_i) = (\xi(\boldsymbol{w}, \boldsymbol{b}|\boldsymbol{x_i}, y_i), \xi^*(\boldsymbol{w}, \boldsymbol{b}|\boldsymbol{x_i}, y_i))^T$. The capped ℓ_p-norm ordinal hinge loss function is defined as follows:

$$\xi_c(\boldsymbol{w}, \boldsymbol{b}|\boldsymbol{x_i}, y_i) = \min(||\boldsymbol{\xi_u}(\boldsymbol{w}, \boldsymbol{b}|\boldsymbol{x_i}, y_i)||_2^p, \epsilon), \tag{4}$$

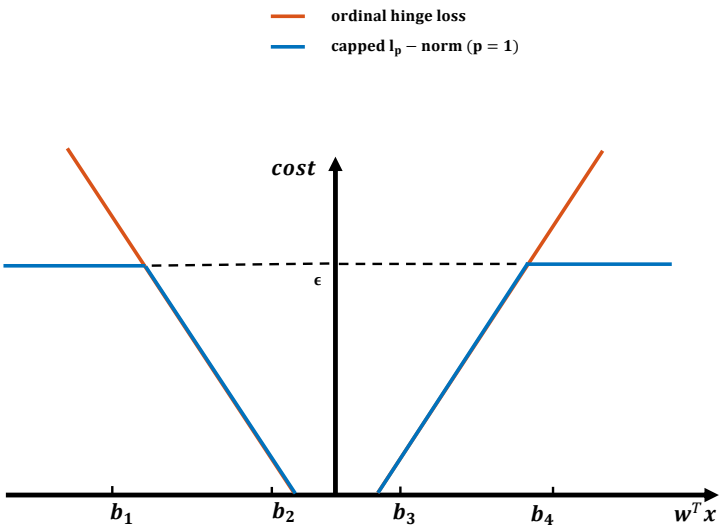


Figure 2: The plots of ordinal hinge loss and capped ℓ_p norm ordinal hinge loss.

where $0 < p \leq 2$ and ϵ determines the upper bound of the loss. We set $\epsilon \geq 1$, because a loss function with $\epsilon < 1$ becomes a constant before exiting the correct interval, it is impossible to distinguish between the correct interval and the wrong interval. To illustrate this loss, we plot the loss function for $p = 1$ and $\epsilon = 1$ in Fig. 2. We can observe that if data point $\boldsymbol{x}$ is misclassified, there will be a loss, but this loss will not exceed ϵ. This attribute is highly robust against data outliers because no matter how large the residual loss is, it will never exceed ϵ. Like SVOREX, we also use the ℓ_2-norm of the projected vector $\boldsymbol{w}$ as the regularization term. Therefore, we propose a new robust support vector ordinal regression method based on capped ℓ_p-norm by solving the following optimization problem:

$$\begin{aligned} &\min_{\boldsymbol{w},\boldsymbol{b}} \sum_{i=1}^{N} \xi_c(\boldsymbol{w}, \boldsymbol{b}|\boldsymbol{x_i}) + \gamma||\boldsymbol{w}||_2^2 \\ &\text{s.t.} \quad b_{k-1} \leq b_k \quad k = 1, 2, \ldots K. \end{aligned} \tag{5}$$

For a single hinge loss, it can be easily demonstrated that it can be written in the following form [35]:

$$\xi(\boldsymbol{w}, b|\boldsymbol{x_i}) = \min_{m_i \geq 0} |\boldsymbol{w}^T \boldsymbol{x_i} + b - y_i - y_i m_i|. \tag{6}$$

Therefore, similar to Eq.(6), the problem (5) can be expressed in the following form:

$$\begin{aligned} \min_{\boldsymbol{w},\boldsymbol{b},M\geq 0} & \sum_{i=1}^{N} \min(||(\boldsymbol{e}\boldsymbol{x_i}^T\boldsymbol{w} - \boldsymbol{\theta_i} - \boldsymbol{y^*} \circ \boldsymbol{m_i}||_2^p, \epsilon) + \gamma||\boldsymbol{w}||_2^2 \\ \text{s.t.} \quad & b_{k-1} \leq b_k \quad k = 1, 2, \ldots, K, \end{aligned} \tag{7}$$

where $\boldsymbol{e} = (1,1)^T$, $\boldsymbol{\theta_i} = (b_{y_i-1} - 1, b_{y_i} + 1)^T$, $\boldsymbol{y^*} = (1,-1)^T$, and $M \in R^{2\times N}$ with the i-th column as $\boldsymbol{m_i}$. $M \geq 0$ means that each element of M is not less than 0. $\circ$ denotes the hadamard product. Due to the non-convex and non-smooth nature of the above problem, the optimization can be quite challenging. In the next section we will introduce an efficient iterative algorithm to solve the optimization problem (7).

4. Optimization Algorithm

In this section, we will introduce an efficient optimization algorithm to optimize the problem (7).

4.1. Algorithm to Solve a General Problem

We first consider the following general problem:

$$\min_{x\in\Omega} h(x) + \sum_i v_i(g_i(x)), \tag{8}$$

where $v_i(x)$ is an arbitrary concave function with the domain of $g_i(x)$, x and $g_i(x)$ can be arbitrary scalars, vectors or matrices. We propose an effective Re-weighted optimization framework to solve problem (8). The details of this optimization algorithm are described in Algorithm 1, where $f_i^{'}(g_i(x))$ denotes any supergradient of the concave function f_i at point $g_i(x)$.

4.2. Optimization Algorithm to Solve Problem (7)

We define the function $v_i(\cdot)$ as follows:

$$v_i(g_i(\boldsymbol{w},\boldsymbol{b},\boldsymbol{m_i})) = \min(g_i(\boldsymbol{w},\boldsymbol{b},\boldsymbol{m_i})^{\frac{p}{2}}, \epsilon), \tag{9}$$

where $g_i(\boldsymbol{w},\boldsymbol{b},\boldsymbol{m_i}) = ||(\boldsymbol{w}^T\boldsymbol{x_i})\boldsymbol{e} + \boldsymbol{\theta_i} - \boldsymbol{y^*} \circ \boldsymbol{m_i}||_2^2$.

Algorithm 1 Re-Weighted optimization framework

Initialize $x \in \Omega$ and $t = 1$
while not converge **do**
Compute the supergradient of the concave function $D_i = v_i'(g_i(x))$ for each i.
Update x by the optimal solution to the problem:
$\min_{x\in\Omega} h(x) + \sum_i Tr(D_i^T g_i(x))$
end while

We can see that the problem (7) represents a special case of problem (8) when $0 < p \leq 2$. Therefore, according to the second step of the Re-Weighted framework, we solve the following optimization problem in each iteration:

$$\begin{aligned} \min_{\boldsymbol{w},\boldsymbol{b},M\geq 0} \quad & \sum_{i=1}^{N} d_i ||(\boldsymbol{e}\boldsymbol{x_i}^T\boldsymbol{w} - \boldsymbol{\theta_i} - \boldsymbol{y^*} \circ \boldsymbol{m_i}||_2^2 + \gamma||\boldsymbol{w}||_2^2 \\ s.t. \quad & b_{k-1} \leq b_k, \quad k = 1, 2, \ldots, K, \end{aligned} \tag{10}$$

where

$$d_i = \begin{cases} \frac{p}{2} g_i(\boldsymbol{w}, \boldsymbol{b}, \boldsymbol{m_i})^{\frac{p-2}{2}} & g_i(\boldsymbol{w}, \boldsymbol{b}, \boldsymbol{m_i}) \leq \epsilon \\ 0 & otherwise. \end{cases} \tag{11}$$

Note that the value of d_i tends to infinity when $g_i(\boldsymbol{w}, \boldsymbol{b}, \boldsymbol{m_i}) = 0$ and $p < 2$. We add an infinitesimal δ for the update of d_i to avoid this situation. More precisely, d_i is update according to:

$$d_i = \begin{cases} \frac{p}{2} (g_i(\boldsymbol{w}, \boldsymbol{b}, \boldsymbol{m_i}) + \delta)^{\frac{p-2}{2}} & g_i(\boldsymbol{w}, \boldsymbol{b}, \boldsymbol{m_i}) \leq \epsilon \\ 0 & otherwise. \end{cases} \tag{12}$$

Eq. (12) is equivalent to Eq. (11) when $\delta \to 0$.

4.2.1. update $\boldsymbol{w}$ and $\boldsymbol{b}$ when M is fixed

When the M is fixed, by setting the derivative of Eq. (10) with respect to $\boldsymbol{w}$ to 0, we have:

$$\boldsymbol{w} = (2XDX^T + \gamma I)^{-1} XD\boldsymbol{e}^T H^T, \tag{13}$$

where $H \in \mathbb{R}^{2\times N}$ is a matrix, each column of which is $\boldsymbol{\theta_i} + \boldsymbol{y^*} \circ \boldsymbol{m_i}$ and $I \in \mathbb{R}^{d\times d}$ is an identity matrix. By substituting Eq. (13) into problem (10), we find that problem (10) is a quadratic programming(QP) problem about $\boldsymbol{b}$. There are many methods that can be used to solve this problem, such as active set, conjugate gradient, and interior point methods.

4.2.2. update M when $\boldsymbol{w}$ and $\boldsymbol{b}$ are fixed

When the $\boldsymbol{w}$ and $\boldsymbol{b}$ are fixed, the problem (10) can be solved by solving the following problem separately for each $\boldsymbol{m_i}$:

$$\min_{\boldsymbol{m_i}\geq 0} ||\boldsymbol{e}\boldsymbol{x_i}^T\boldsymbol{w} - \boldsymbol{\theta_i} - \boldsymbol{y^*} \circ \boldsymbol{m_i}||_2^2. \tag{14}$$

Note that $\boldsymbol{y^*} = (1, -1)^T$, the problem (14) is equivalent to the following form:

$$\min_{\boldsymbol{m_i}\geq 0} ||\boldsymbol{y^*} \circ (\boldsymbol{e}\boldsymbol{x_i}^T\boldsymbol{w}) - \boldsymbol{y^*} \circ \boldsymbol{\theta_i} - \boldsymbol{m_i}||_2^2. \tag{15}$$

Since $\boldsymbol{m_i} \geq 0$, the optimal solution of problem (15) can be easily obtained as follows:

$$\boldsymbol{m_i} = (\boldsymbol{y^*} \circ (\boldsymbol{e}\boldsymbol{x_i}^T\boldsymbol{w}) - \boldsymbol{y^*} \circ \boldsymbol{\theta_i})_+, \tag{16}$$

where the i-th element of vector $(\boldsymbol{u})_+$ is $\max(0, u_k)$. Based on the above analysis, the detailed steps for solving the problem (7) are summarized in Algorithm 2.

5. THEORETICAL ANALYSIS

5.1. Convergence Analysis

In this subsection, we present the convergence analysis of Algorithm 1. Since Algorithm 2 is a special case of Algorithm 1, it is sufficient to provide the convergence analysis of Algorithm 1.

Theorem 1. *The Algorithm 1 will reduce the objective value of problem (8) in each iteration until it converges.*

Proof. Suppose the updated x is x^{new}, in accordance with the second step in Algorithm 1, we have:

Algorithm 2 Optimization algorithm to solve problem (7)

Input the training dataset $S = \{(\boldsymbol{x_i}, y_i)\}_{i=1}^{N}$
Initialize $D = I$ and$M = 0$.
while not converge **do**
1. Update $\boldsymbol{b}$ by solving the QP problem.
2. Update $\boldsymbol{w}$ by Eq.(13):
$\boldsymbol{w} = (2XDX^T + \gamma I)^{-1} XD\boldsymbol{e}^T H^T$.
3. Update M, where the vector corresponding to column i is caculated by Eq.(16):
$\boldsymbol{m_i} = (\boldsymbol{y^*} \circ (\boldsymbol{e}\boldsymbol{x_i}^T\boldsymbol{w}) - \boldsymbol{y^*} \circ \boldsymbol{\theta_i})_+$.
4. Update D by Eq.(11).
end while

$$h(x^{new}) + \sum_i Tr(D_i^T g_i(x^{new})) \leq h(x) + \sum_i Tr(D_i^T g_i(x)). \tag{17}$$

Because $v_i(x)$ is concave for each i, according to the definition of supergradient, the following inequality holds:

$$v(g_i(x^{new})) - v(g_i(x)) \leq Tr(D_i^T g_i(x^{new})) - Tr(D_i^T g_i(x)). \tag{18}$$

Thus, we have:

$$\begin{aligned} &\sum_i v(g_i(x^{new})) - \sum_i Tr(D_i^T g_i(x^{new})) \\ &\leq \sum_i v(g_i(x)) - \sum_i Tr(D_i^T g_i(x)). \end{aligned} \tag{19}$$

Combining the two inequalities Eq. (17) and Eq. (19), the following inequality holds:

$$h(x^{new}) + \sum_i v(g_i(x^{new})) \leq h(x) + \sum_i v(g_i(x)). \tag{20}$$

The equality in Eq.(20) holds only when the algorithm converges. Therefore, Algorithm 1 will monotonically decrease the objective of problem (8) at each iteration until convergence. □

5.2. An intuitive explanation of why CSVOR is robust to outliers

In this subsection, an intuitive explanation is provided to illustrate why CSVOR is robust to outliers.

Like many SVM-based methods, the projection direction corresponding to SVOREX is only determined by a small part of the training data. These points play a crucial role in the model. The projection vector of SVOREX is determined by [33]:

$$\boldsymbol{w} = \frac{1}{2\gamma}\sum_{i=1}^{N}(\alpha_i - \alpha_i^*)\boldsymbol{x_i}. \tag{21}$$

The set of data points corresponding to $\alpha_i - \alpha_i^* \neq 0$ is $A = \{\boldsymbol{x_i}|\xi_i > 0 \text{ and } \xi_i^* = 0\} \cup \{\boldsymbol{x_i}|\xi_i^* > 0 \text{ and } \xi_i = 0\} \cup \{\boldsymbol{x_i}|\boldsymbol{w}^T\boldsymbol{x_i} - b_{y_i-1} - 1 = 0 \text{ or } \boldsymbol{w}^T\boldsymbol{x_i} - b_{y_i} + 1 = 0\}$ [33]. Points outside A have no effect on the calculation of the projection vector $\boldsymbol{w}$. Since outliers usually have larger residuals, they generally belong to set A and play an important role in the calculation of the projection direction $\boldsymbol{w}$. However, according to Eq. (13), it can be seen that in each iteration, the points with $g_i(\boldsymbol{w}, \boldsymbol{b}, \boldsymbol{m_i}) > \epsilon$ do not play a role in the calculation of $\boldsymbol{w}$. In other words, during the training process, CSVOR uses a weight matrix D to identify and remove outliers implicitly so that they do not affect the calculation of the projection direction to resist the influence of outliers. This can be seen as an intuitive explanation for why CSVOR is robust to outliers.

6. Experiments

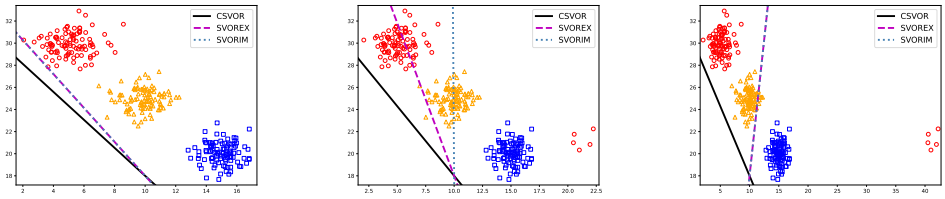
In this section, extensive experiments are conducted on synthetic and benchmark datasets to validate the effectiveness of the proposed method.

6.1. Experiments on Synthetic Datasets

We illustrate the robustness of our algorithm against outliers using two synthetic datasets. The first dataset consists of randomly generated two-dimensional data with three Gaussian clusters, each represented by different colors and shapes. Specifically, the red circles represent samples of the first class, the orange triangles represent samples of the second class, and the blue squares represent samples of the third class. We compare our CSVOR algorithm with SVORIM and SVOREX, and the comparison results are presented

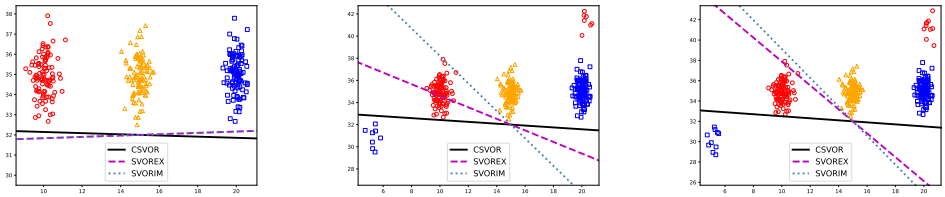
in Fig. 3. This figure displays the projection directions of our CSVOR algorithm and the compared algorithms as the number of outliers varies. From Fig. 3(a), we observe that when there are no outliers, all three algorithms can find a relatively good projection direction. However, when we introduce five outliers into the data, we observe that as the size of the outliers increases, SVOREX and SVORIM become inefficient. In contrast, our CSVOR model consistently performs well, demonstrating its robustness to outliers. This example illustrates that our CSVOR model can effectively avoid the influence of outliers, making it robust to outliers.

The second synthetic dataset also consists of randomly generated two-dimensional data from three Gaussian clusters. We compare our CSVOR algorithm with SVORIM and SVOREX, as shown in Fig. 4. When there are no outliers, all three algorithms perform well. To assess the robustness of these algorithms to outliers, we added 20 and 60 outliers to the data. From Fig. 4(b), it can be seen that with 20 outliers, SVORIM becomes inefficient, and although SVOREX is not inefficient, the projection direction it found has deviated from the optimal projection direction to a certain extent. With 60 outliers, SVOREX becomes completely inefficient. In contrast, our CSVOR model continues to perform well. This example illustrates the robustness of our CSVOR model to outliers as their number increases. As the number of outliers increases, the projection directions found by SVOREX and SVORIM deviate further from the optimal projection direction, while the projection direction found by our model remains close to the optimal one.



(a) non-outlier (b) The value of outliers is relatively large (c) The value of outliers is large

Figure 3: Results on the first synthetic dataset.



(a) non-outlier (b) The number of ouliers is 20 (c) The number of ouliers is 60

Figure 4: Results on the second synthetic dataset.

6.2. Experiments on Benchmark Datasets

6.2.1. Experimental Setting

In this subsection, we conduct experiments on several benchmark datasets to compare the performance of CSVOR and state-of-the-art(SOTA) methods. The SOTA methods include SVOREX [33], SVORIM [33], PINSVOR [19], REDSVM [34], POM [36], KDLOR [37]. We use a total of 10 benchmark datasets, comprising 4 real datasets and 6 discretized regression datasets.

The 4 real datasets are commonly used UCI datasets in the context of ordinal regression [38]. The 6 discretized regression datasets, originally provided by Chu et al. [33], are transformed into ordinal regression datasets by dividing the data into different classes with the same frequency. The details of the datasets are in Table 1. In order to compare the robustness to outliers, we add two different outliers to the real datasets and the discretised regression datasets respectively.

We introduce label noise to the 4 real datasets to compare the robustness of our method and the baselines. Label noise, considered a type of outlier, is the process that pollutes labels [39, 40]. For the real datasets, we consider 30 random stratified splits, allocating 75% of the patterns to the training set and 25% to the test set. Outliers are generated in the training set by randomly changing the labels of 1%, 5%, 10%, 15%, and 20% of the samples. Regarding the discrete regression datasets, which are transformed into ordinal regression by dividing the targets of traditional regression datasets into five different bins with equal frequency, we perform 20 random splits. The number of training and test instances follows the suggestions by Chu et al. [18]. Outliers are generated by normalizing the data to [0, 1] and then randomly setting 50% of the features of 1%, 5%, 10%, 15%, and 20% of the data points to -1.

Table 1: Dataset descriptions.

Datasets	Instances	Attributes	Classes
Contact-Lenses	24	6	3
Squash-Unstored	52	52	3
Tae	151	54	3
Newthyroid	215	5	3
Machine	209	7	5
Housing	506	14	5
Stock	700	9	5
Bank	8192	8	5
Computer	8192	12	5
Cal.housing	20640	8	5

In terms of parameter settings, for SVOR methods (SVOREX, SVORIM, PINSVOR, and REDSVM), the regularization parameter C is selected from the range $[10^{-3}, 10^{-2}, ..., 10^{3}]$. The pinball loss parameter τ in PINSVOR is chosen from $[0.1, 0.2, \ldots, 0.9]$. The penalty coefficient C in KDLOR is selected from $[10^{-1}, 0, 10]$. In our approach, the parameters γ and p are selected from within the specified ranges of $[0.5, 0.7, 1, 1.5, 1.7, 2]$ and $[10^{-3}, 10^{-2}, ..., 10^{3}]$, respectively. Optimal values for all parameters are obtained using five fold cross-validation. Additionally, we set the value of ϵ heuristically by selecting the top 10% of data with the highest residuals in the first 5 iterations.

The Mean Zero Error (MZE) is selected as the evaluation index, which reflects the accuracy of model prediction. The smaller the MZE on the test data set, the better the generalization performance of the ordered regression model. The calculation formula of MZE is as follows:

$$MZE = \frac{1}{N}\sum_{i=1}^{N}[[y_i \neq y_i^{pred}]], \tag{22}$$

where y_i is the true label and y_i^{pred} is the predicted label. We report the performance of CSVOR and 7 baselines on 10 benchmark datasets in Table 2 and Table 3. Fig. 5 shows the critical difference (CD) plot of the Friedman test and Nemenyi post-hoc test at a 5% significance level. The Friedman test with the Nemenyi post-hoc test is a frequently used non-parametric method for comparing the performance of multiple methods across a variety of datasets

[41, 42]. In the CD plot, a smaller average rank (abscissa) indicates better model performance. If the abscissa spacing between two methods is greater than the CD value , we consider there to be a significant difference in their performance. In Fig. 5, we connect the methods that show no significant difference with a red line.

Table 2: Test results for each real dataset and method(average MZE ± standard deviation). Best results are in bold, and the second best results are underlined. (r is the ratio of label noise).

Datasets	r(%)	SVOREX	SVORIM	PINSVOR	REDSVM	POM	KDLOR	CSVOR
Contact-Lenses	0	$\underline{0.294 \pm 0.113}$	0.339 ± 0.108	0.300 ± 0.134	0.339 ± 0.120	0.394 ± 0.178	0.350 ± 0.160	$\mathbf{0.222 \pm 0.126}$
	1	0.322 ± 0.107	0.356 ± 0.129	$\underline{0.311 \pm 0.137}$	0.367 ± 0.068	0.406 ± 0.143	0.344 ± 0.190	$\mathbf{0.278 \pm 0.126}$
	5	0.350 ± 0.166	0.367 ± 0.068	$\underline{0.333 \pm 0.152}$	0.356 ± 0.068	0.406 ± 0.189	0.483 ± 0.202	$\mathbf{0.300 \pm 0.179}$
	10	0.361 ± 0.077	0.422 ± 0.179	$\underline{0.344 \pm 0.115}$	0.367 ± 0.068	0.444 ± 0.166	0.500 ± 0.196	$\mathbf{0.333 \pm 0.214}$
	15	0.367 ± 0.077	0.439 ± 0.198	0.367 ± 0.177	$\underline{0.367 \pm 0.068}$	0.433 ± 0.166	0.456 ± 0.169	$\mathbf{0.356 \pm 0.206}$
	20	0.378 ± 0.075	0.472 ± 0.219	0.367 ± 0.132	0.406 ± 0.150	0.444 ± 0.177	0.517 ± 0.187	$\mathbf{0.367 \pm 0.159}$
Squash-Unstored	0	0.226 ± 0.107	0.226 ± 0.107	$\mathbf{0.208 \pm 0.109}$	0.282 ± 0.135	0.538 ± 0.000	0.238 ± 0.128	$\underline{0.218 \pm 0.123}$
	1	$\mathbf{0.221 \pm 0.110}$	0.223 ± 0.109	0.233 ± 0.138	0.346 ± 0.112	0.538 ± 0.000	0.300 ± 0.133	$\underline{0.221 \pm 0.112}$
	5	$\underline{0.236 \pm 0.103}$	0.238 ± 0.126	0.244 ± 0.125	0.313 ± 0.132	0.538 ± 0.000	0.303 ± 0.138	$\mathbf{0.223 \pm 0.095}$
	10	$\underline{0.228 \pm 0.103}$	0.236 ± 0.116	0.295 ± 0.134	0.326 ± 0.135	0.538 ± 0.000	0.369 ± 0.143	$\mathbf{0.215 \pm 0.116}$
	15	$\underline{0.233 \pm 0.117}$	0.267 ± 0.116	0.290 ± 0.138	0.349 ± 0.145	0.538 ± 0.000	0.472 ± 0.104	$\mathbf{0.226 \pm 0.120}$
	20	$\underline{0.246 \pm 0.102}$	0.308 ± 0.137	0.326 ± 0.155	0.446 ± 0.167	0.538 ± 0.000	0.474 ± 0.162	$\mathbf{0.244 \pm 0.102}$
Tae	0	$\underline{0.443 \pm 0.053}$	0.443 ± 0.056	0.468 ± 0.098	0.462 ± 0.077	0.642 ± 0.086	0.497 ± 0.080	$\mathbf{0.432 \pm 0.058}$
	1	$\underline{0.451 \pm 0.059}$	0.454 ± 0.057	0.470 ± 0.087	0.458 ± 0.067	0.646 ± 0.078	0.504 ± 0.075	$\mathbf{0.439 \pm 0.057}$
	5	$\underline{0.458 \pm 0.062}$	0.466 ± 0.065	0.470 ± 0.083	0.485 ± 0.071	0.645 ± 0.078	0.515 ± 0.080	$\mathbf{0.444 \pm 0.078}$
	10	0.500 ± 0.076	0.507 ± 0.080	0.481 ± 0.081	0.529 ± 0.083	0.649 ± 0.065	0.547 ± 0.090	$\underline{0.486 \pm 0.086}$
	15	0.489 ± 0.070	0.490 ± 0.070	$\underline{0.468 \pm 0.092}$	0.514 ± 0.082	0.649 ± 0.066	0.527 ± 0.095	$\mathbf{0.469 \pm 0.073}$
	20	0.523 ± 0.060	0.529 ± 0.057	0.492 ± 0.086	0.547 ± 0.073	0.651 ± 0.059	0.570 ± 0.089	$\mathbf{0.518 \pm 0.056}$
Newthyroid	0	$\underline{0.031 \pm 0.023}$	$\underline{0.031 \pm 0.023}$	0.052 ± 0.023	$\mathbf{0.027 \pm 0.019}$	0.035 ± 0.023	0.047 ± 0.026	0.037 ± 0.031
	1	$\mathbf{0.031 \pm 0.025}$	0.036 ± 0.030	0.054 ± 0.033	$\underline{0.033 \pm 0.024}$	0.036 ± 0.026	0.049 ± 0.027	0.041 ± 0.027
	5	$\underline{0.043 \pm 0.033}$	0.056 ± 0.042	0.051 ± 0.033	0.054 ± 0.037	0.054 ± 0.035	$\mathbf{0.040 \pm 0.024}$	0.043 ± 0.028
	10	0.067 ± 0.036	0.082 ± 0.037	$\underline{0.049 \pm 0.025}$	0.075 ± 0.037	0.077 ± 0.040	0.049 ± 0.047	$\mathbf{0.038 \pm 0.021}$
	15	0.086 ± 0.043	0.104 ± 0.052	$\underline{0.049 \pm 0.030}$	0.123 ± 0.049	0.106 ± 0.052	0.077 ± 0.053	$\mathbf{0.048 \pm 0.026}$
	20	0.111 ± 0.037	0.132 ± 0.044	$\underline{0.049 \pm 0.028}$	0.169 ± 0.057	0.138 ± 0.046	0.104 ± 0.081	$\mathbf{0.038 \pm 0.023}$

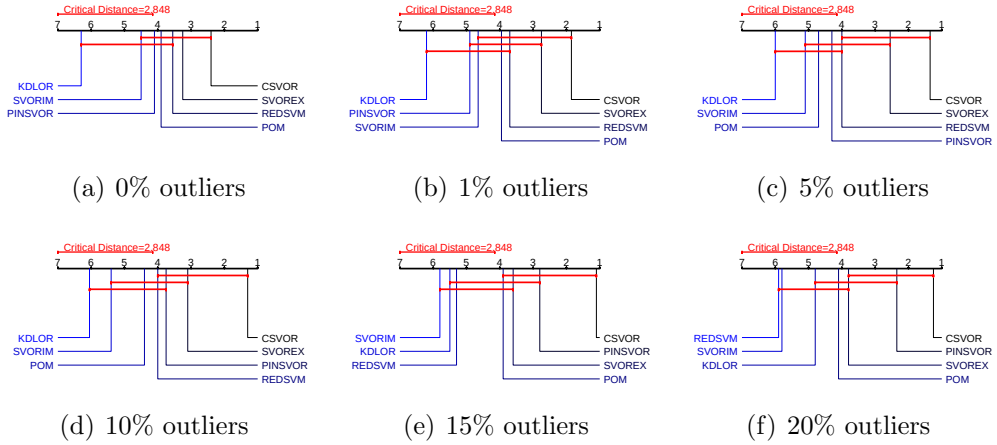
6.2.2. Results Analysis

First, our method exhibits optimal or sub-optimal performance on most datasets with or without outliers, especially on Contact-Lenses and Machine datasets , as shown in Table 2 and Table 3. Moreover, as shown in Fig. 5, the average rank of our method is first among all methods. We also observe that as the number of outliers increases, the average ranking of our method shows an upward trend. This demonstrates that our method outperforms the 7 baselines and is more robust against the influence of outliers. Secondly, as the number of outliers increases, the generalization performance of the model shows a downward trend due to the presence of outliers in the training set. However, our method exhibits slower degradation in generalization performance as the number of outliers increases, indicating its resilience to outlier influence. Lastly, the performance of PINSVOR is unsatisfactory because it

Table 3: Test results for each discretised regression dataset and method(average MZE ± standard deviation). Best results are in bold, and the second best results are underlined. (r is the ratio of data outliers).

Datasets	r(%)	SVOREX	SVORIM	PINSVOR	REDSVM	POM	KDLOR	CSVOR
Machine	0	0.406 ± 0.058	0.405 ± 0.060	0.399 ± 0.076	0.391 ± 0.068	0.397 ± 0.068	0.428 ± 0.064	$\mathbf{0.388 \pm 0.067}$
	1	0.407 ± 0.063	0.418 ± 0.066	0.406 ± 0.061	0.402 ± 0.070	0.415 ± 0.062	0.458 ± 0.063	$\mathbf{0.393 \pm 0.060}$
	5	0.418 ± 0.056	0.449 ± 0.071	0.425 ± 0.055	0.419 ± 0.066	0.442 ± 0.055	0.471 ± 0.062	$\mathbf{0.413 \pm 0.063}$
	10	0.436 ± 0.064	0.524 ± 0.061	0.457 ± 0.071	0.479 ± 0.078	0.455 ± 0.064	0.498 ± 0.056	$\mathbf{0.426 \pm 0.054}$
	15	0.484 ± 0.086	0.600 ± 0.103	0.463 ± 0.074	0.508 ± 0.096	0.460 ± 0.088	0.494 ± 0.079	$\mathbf{0.436 \pm 0.067}$
	20	0.531 ± 0.081	0.649 ± 0.103	0.487 ± 0.072	0.581 ± 0.110	0.460 ± 0.049	0.503 ± 0.083	$\mathbf{0.436 \pm 0.062}$
Housing	0	0.350 ± 0.019	0.357 ± 0.019	$\mathbf{0.285 \pm 0.033}$	0.356 ± 0.024	0.353 ± 0.020	0.467 ± 0.040	0.323 ± 0.033
	1	0.356 ± 0.028	0.366 ± 0.030	0.383 ± 0.029	0.367 ± 0.027	0.355 ± 0.062	0.456 ± 0.039	$\mathbf{0.344 \pm 0.033}$
	5	0.371 ± 0.037	0.401 ± 0.043	0.410 ± 0.039	0.396 ± 0.041	0.386 ± 0.055	0.493 ± 0.062	$\mathbf{0.351 \pm 0.040}$
	10	0.401 ± 0.045	0.449 ± 0.045	0.443 ± 0.048	0.438 ± 0.044	0.421 ± 0.064	0.507 ± 0.041	$\mathbf{0.381 \pm 0.044}$
	15	0.455 ± 0.028	0.500 ± 0.046	0.454 ± 0.028	0.495 ± 0.047	0.454 ± 0.088	0.505 ± 0.042	$\mathbf{0.409 \pm 0.046}$
	20	0.485 ± 0.044	0.535 ± 0.047	0.466 ± 0.034	0.526 ± 0.051	0.477 ± 0.049	0.525 ± 0.046	$\mathbf{0.425 \pm 0.047}$
Stock	0	0.357 ± 0.015	0.367 ± 0.017	0.358 ± 0.095	0.367 ± 0.018	0.365 ± 0.019	0.413 ± 0.022	$\mathbf{0.323 \pm 0.017}$
	1	0.363 ± 0.019	0.390 ± 0.022	0.410 ± 0.021	0.390 ± 0.021	0.384 ± 0.023	0.475 ± 0.036	$\mathbf{0.346 \pm 0.021}$
	5	0.408 ± 0.025	0.412 ± 0.033	0.438 ± 0.042	0.409 ± 0.025	0.445 ± 0.028	0.501 ± 0.041	$\mathbf{0.365 \pm 0.019}$
	10	0.439 ± 0.041	0.456 ± 0.033	0.473 ± 0.030	0.456 ± 0.050	0.473 ± 0.028	0.515 ± 0.033	$\mathbf{0.390 \pm 0.030}$
	15	0.476 ± 0.026	0.541 ± 0.074	0.500 ± 0.031	0.557 ± 0.086	0.494 ± 0.028	0.532 ± 0.032	$\mathbf{0.428 \pm 0.047}$
	20	0.500 ± 0.030	0.632 ± 0.063	0.506 ± 0.047	0.681 ± 0.055	0.503 ± 0.034	0.531 ± 0.043	$\mathbf{0.455 \pm 0.057}$
Bank	0	0.278 ± 0.026	0.278 ± 0.026	0.376 ± 0.046	0.280 ± 0.033	$\mathbf{0.277 \pm 0.025}$	0.317 ± 0.064	0.284 ± 0.109
	1	0.329 ± 0.108	0.360 ± 0.158	0.378 ± 0.078	0.341 ± 0.141	0.340 ± 0.129	0.465 ± 0.140	$\mathbf{0.307 \pm 0.109}$
	5	0.514 ± 0.109	0.575 ± 0.158	0.543 ± 0.091	0.532 ± 0.098	0.534 ± 0.095	0.661 ± 0.061	$\mathbf{0.399 \pm 0.147}$
	10	0.616 ± 0.109	0.627 ± 0.077	0.609 ± 0.097	$\mathbf{0.599 \pm 0.149}$	0.602 ± 0.134	0.675 ± 0.059	0.604 ± 0.145
	15	0.676 ± 0.071	0.700 ± 0.067	0.671 ± 0.060	0.676 ± 0.085	0.667 ± 0.065	0.700 ± 0.041	$\mathbf{0.621 \pm 0.155}$
	20	0.727 ± 0.039	0.749 ± 0.049	$\mathbf{0.695 \pm 0.066}$	0.747 ± 0.057	0.713 ± 0.045	0.727 ± 0.039	0.709 ± 0.113
Computer	0	0.397 ± 0.016	0.397 ± 0.015	0.411 ± 0.029	0.384 ± 0.018	$\mathbf{0.378 \pm 0.014}$	0.434 ± 0.021	0.395 ± 0.018
	1	0.412 ± 0.020	0.413 ± 0.018	0.433 ± 0.028	0.399 ± 0.022	$\mathbf{0.395 \pm 0.023}$	0.448 ± 0.025	0.406 ± 0.020
	2	0.442 ± 0.026	0.447 ± 0.030	0.478 ± 0.032	$\mathbf{0.431 \pm 0.024}$	0.432 ± 0.029	0.478 ± 0.027	0.437 ± 0.020
	3	0.492 ± 0.041	0.500 ± 0.039	0.515 ± 0.039	0.482 ± 0.037	0.481 ± 0.036	0.511 ± 0.026	$\mathbf{0.460 \pm 0.032}$
	4	0.541 ± 0.054	0.566 ± 0.060	0.535 ± 0.041	0.541 ± 0.047	0.521 ± 0.050	0.550 ± 0.051	$\mathbf{0.491 \pm 0.048}$
	5	0.565 ± 0.043	0.608 ± 0.055	0.547 ± 0.023	0.581 ± 0.053	0.541 ± 0.034	0.553 ± 0.040	$\mathbf{0.524 \pm 0.035}$
Cal.housing	0	0.502 ± 0.012	0.507 ± 0.011	0.502 ± 0.019	0.491 ± 0.010	$\mathbf{0.488 \pm 0.010}$	0.505 ± 0.013	0.491 ± 0.016
	1	0.519 ± 0.025	0.526 ± 0.026	0.517 ± 0.026	0.512 ± 0.028	$\mathbf{0.506 \pm 0.024}$	0.523 ± 0.021	0.513 ± 0.030
	2	0.562 ± 0.032	0.603 ± 0.060	0.553 ± 0.025	0.581 ± 0.063	0.547 ± 0.030	0.597 ± 0.049	$\mathbf{0.539 \pm 0.028}$
	3	0.613 ± 0.043	0.669 ± 0.065	0.573 ± 0.029	0.648 ± 0.074	0.578 ± 0.031	0.650 ± 0.059	$\mathbf{0.552 \pm 0.049}$
	4	0.687 ± 0.058	0.747 ± 0.052	0.598 ± 0.036	0.746 ± 0.060	0.615 ± 0.042	0.660 ± 0.047	$\mathbf{0.576 \pm 0.049}$
	5	0.729 ± 0.052	0.790 ± 0.022	0.619 ± 0.040	0.792 ± 0.021	0.638 ± 0.032	0.655 ± 0.057	$\mathbf{0.605 \pm 0.053}$

primarily addresses noise near the classification boundary while disregarding the impact of outliers.



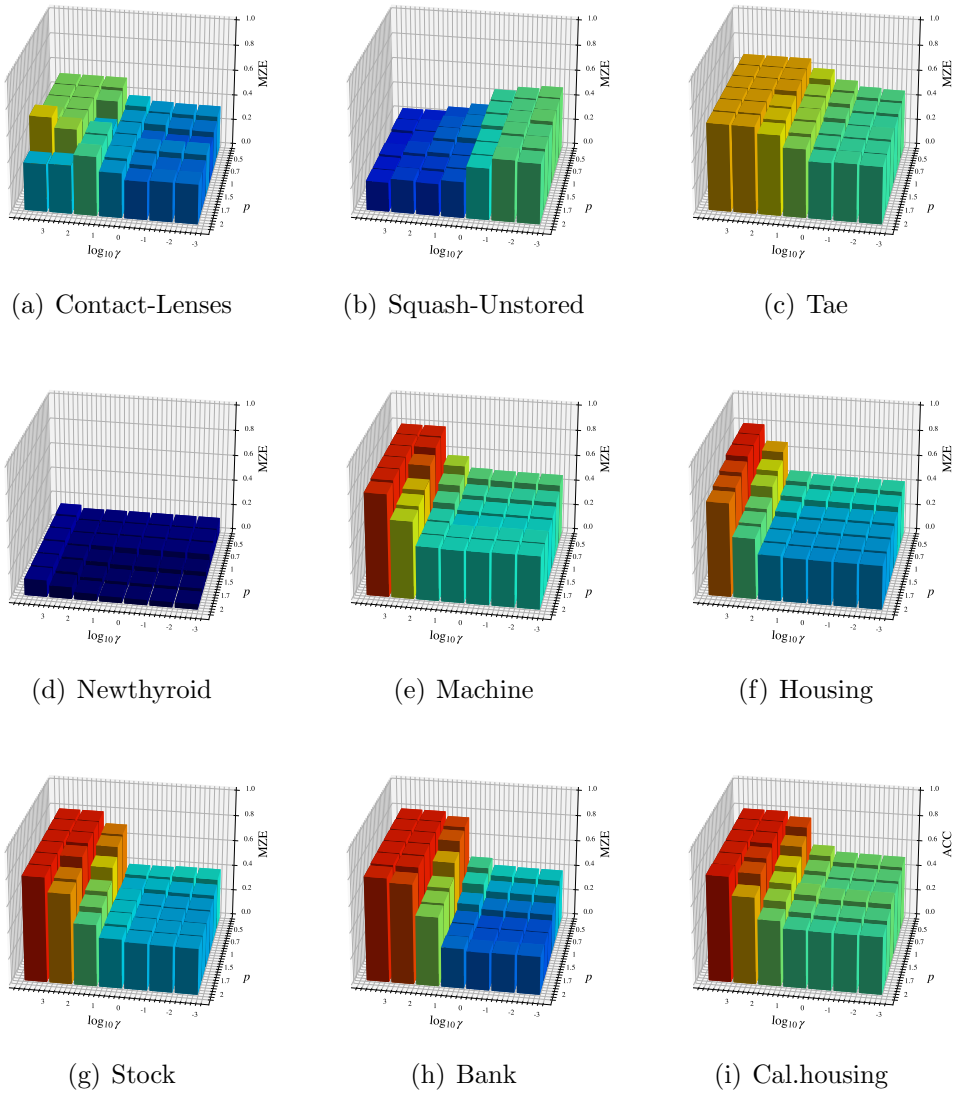
(a) 0% outliers (b) 1% outliers (c) 5% outliers

(d) 10% outliers (e) 15% outliers (f) 20% outliers

Figure 5: The CD diagrams of the Friedman test with Nemenyi post-hoc test at the 5% significance level.

6.3. Visual experiment

In Section 5.2, we provide an intuitive explanation of why CSVOR is robust to outliers. CSVOR utilizes a weight matrix D to detect and remove outliers implicitly. This matrix marks detected outliers as 0, effectively removing them from consideration during training process. The ability of the weight matrix D to identify anomalies directly influences the performance of CSVOR. To illustrate this capability, we conduct visual experiments on the FG-NET dataset, widely used for age estimation. This dataset comprises 1002 images of 82 individuals, ranging in age from 0 to 69 years old [43]. For the experiment, we randomly select 10% of the data and introduce black occlusion blocks of varying sizes and positions. Fig. 8 presents the visualization results, where red boxes indicate abnormal points detected by the model, and green boxes represent normal points. The model successfully identifies images with black occlusion patches as outliers, supporting our previous findings and highlighting the effectiveness of the weight matrix D in anomaly detection.

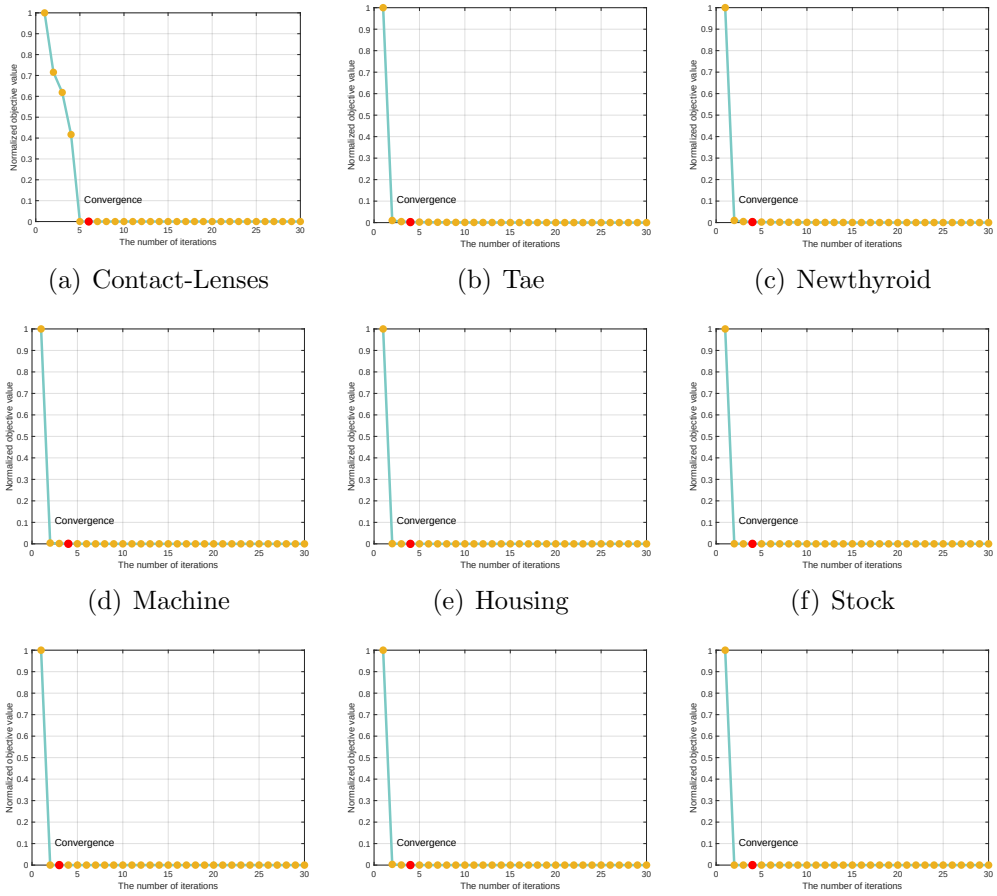


(a) Contact-Lenses (b) Squash-Unstored (c) Tae

(d) Newthyroid (e) Machine (f) Housing

(g) Stock (h) Bank (i) Cal.housing

Figure 6: MZE with respect to parameters p and γ on nine datasets.



(a) Contact-Lenses (b) Tae (c) Newthyroid

(d) Machine (e) Housing (f) Stock

(g) Bank (h) Computer (i) Cal.housing

Figure 7: Convergence curves of CSVOR on nine real-world datasets.

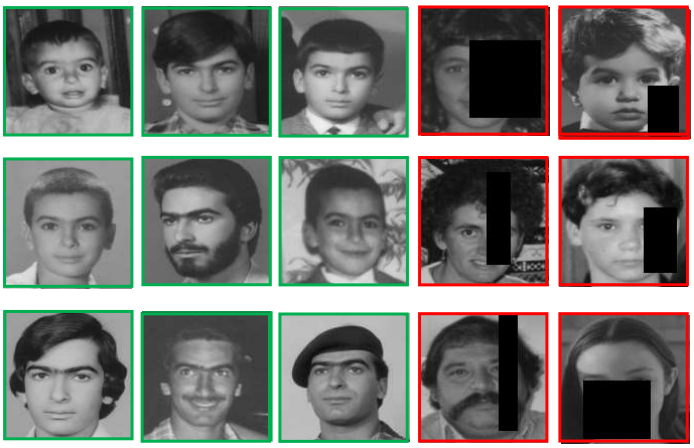


Figure 8: Visualized experimental results on the FG-NET dataset, the first three columns are normal samples, and the last two columns are abnormal samples with black occlusion blocks added.

6.4. Parameters Sensitivity

In this subsection, we study the sensitivity of the parameters p and γ in CSVOR. Fig. 6 shows the Mean Zero-One Error (MZE) on the test sets corresponding to different parameter combinations on nine datasets. (Due to space limitations, we only show the results on nine datasets.) We observe that different parameter combinations have different effects on the performance of the model, and the optimal parameter combinations corresponding to different datasets are different. Therefore, we recommend using five fold cross-validation to select appropriate parameters.

6.5. Convergence Analysis

To address the optimization problem involved in CSVOR, we propose an efficient optimization algorithm based on the Re-Weighted algorithm. The corresponding convergence analysis is given in Section 5.1. Fig. 7 displays the convergence curve of our algorithm on nine datasets. It is evident that CSVOR converges quickly, typically within 10 iterations, across all datasets. The rapid convergence of CSVOR, as illustrated in the convergence curves, highlights the efficiency and effectiveness of our optimization algorithm.

7. CONCLUSION

In this paper, we introduce CSVOR, a robust ordinal regression model that utilizes a novel capped ℓ_p-norm loss function designed to handle both light and heavy outliers. We provide an intuitive explanation for the robustness of CSVOR to outliers CSVOR uses a weight matrix D to identify and eliminate the outliers implicitly during training process. To solve the optimization problem associated with CSVOR, we propose an effective Re-Weighted optimization algorithm, supported by theoretical results demonstrating its convergence. Extensive experiments validate the effectiveness of CSVOR, showcasing its robust performance across various datasets.

Further research is needed to enhance our proposed model. As our current model is linear, its ability to fit complex functions is limited. Therefore, our future research will concentrate on developing nonlinear versions of our model. Additionally, any unbounded loss may be affected by outliers. Although our focus in this paper is on the ordinal hinge loss, the concepts presented herein can be extended to other ordinal regression losses. We plan to explore the application of capped ℓ_p-norm to other ordinal losses, such as ordinal exponential loss and ordinal logistic loss, in future research.

References

[1] P. A. Gutiérrez, M. Perez-Ortiz, J. Sanchez-Monedero, F. Fernandez-Navarro, C. Hervas-Martinez, Ordinal regression methods: survey and experimental study, IEEE Transactions on Knowledge and Data Engineering 28 (1) (2015) 127–146.

[2] 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.

[3] L. Pfannschmidt, J. Jakob, F. Hinder, M. Biehl, P. Tino, B. Hammer, Feature relevance determination for ordinal regression in the context of feature redundancies and privileged information, Neurocomputing 416 (2020) 266–279.

[4] R. Diaz, A. Marathe, Soft labels for ordinal regression, in: in Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 4738–4747.

[5] H. Yan, Cost-sensitive ordinal regression for fully automatic facial beauty assessment, Neurocomputing 129 (2014) 334–342.

[6] V. M. Vargas, P. A. Gutiérrez, C. Hervas-Martinez, Cumulative link models for deep ordinal classification, Neurocomputing 401 (2020) 48–58.

[7] Z. Niu, M. Zhou, L. Wang, X. Gao, G. Hua, Ordinal regression with multiple output cnn for age estimation, in: in Proc. IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 4920–4928.

[8] K.-Y. Chang, C.-S. Chen, Y.-P. Hung, Ordinal hyperplanes ranker with cost sensitivities for age estimation, in: in Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2011, pp. 585–592.

[9] Q. Tian, S. Chen, X. Tan, Comparative study among three strategies of incorporating spatial structures to ordinal image regression, Neurocomputing 136 (2014) 152–161.

[10] W. Li, X. Huang, Z. Zhu, Y. Tang, X. Li, J. Zhou, J. Lu, Ordinalclip: Learning rank prompts for language-guided ordinal regression, in: in Proc.Advances in Neural Information Processing Systems, 2022, pp. 35313–35325.

[11] T. Qian, F. Li, M. Zhang, G. Jin, P. Fan, W. Dai, Contrastive learning from label distribution: A case study on text classification, Neurocomputing 507 (2022) 208–220.

[12] P.-C. Bürkner, M. Vuorre, Ordinal regression models in psychology: A tutorial, Advances in Methods and Practices in Psychological Science 2 (1) (2019) 77–101.

[13] W. Tang, Z. Yang, Y. Song, Disease-grading networks with ordinal regularization for medical imaging, Neurocomputing 545 (2023) 126245.

[14] H. Zhu, H. Shan, Y. Zhang, L. Che, X. Xu, J. Zhang, J. Shi, F.-Y. Wang, Convolutional ordinal regression forest for image ordinal estimation, IEEE Transactions on Neural Networks and Learning Systems 33 (8) (2021) 4084–4095.

[15] N.-H. Shin, S.-H. Lee, C.-S. Kim, Moving window regression: A novel approach to ordinal regression, in: in Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 18760–18769.

[16] N. Pratiwi, et al., Implementing ordinal regression model for analyzing happiness level in indonesia, in: Journal of Physics: Conference Series, Vol. 1320, IOP Publishing, 2019, pp. 012–015.

[17] J. Sun, Z. Gong, D. Zhang, Y. Xu, G. Wei, A robust ordinal regression feedback consensus model with dynamic trust propagation in social network group decision-making, Information Fusion 100 (2023) 101952.

[18] W. Chu, S. S. Keerthi, New approaches to support vector ordinal regression, in: in Proc. International Conference on Machine Learning, 2005, pp. 145–152.

[19] G. Zhong, Y. Xiao, B. Liu, L. Zhao, X. Kong, Ordinal regression with pinball loss, IEEE Transactions on Neural Networks and Learning Systems (2023) 1–15.

[20] J. Wang, X. Zhang, F. Nie, X. Li, Enhanced robust fuzzy k-means clustering joint ℓ_0-norm constraint, Neurocomputing 561 (2023) 126842.

[21] X.-Y. Zhang, C.-L. Liu, C. Y. Suen, Towards robust pattern recognition: A review, Proceedings of the IEEE 108 (6) (2020) 894–922.

[22] F. Nie, X. Wang, H. Huang, Multiclass capped ℓ_p-norm svm for robust classifications, in: in Proc. AAAI Conference on Artificial Intelligence, 2017, pp. 2415–2421.

[23] Z. Wang, H. Hu, R. Wang, Q. Zhang, F. Nie, X. Li, Capped ℓ_p-norm linear discriminant analysis for robust projections learning, Neurocomputing 511 (2022) 399–409.

[24] E. Frank, M. Hall, A simple approach to ordinal classification, in: in Proc. European Conference on Machine Learning, Springer, 2001, pp. 145–156.

[25] W. Waegeman, L. Boullart, et al., An ensemble of weighted support vector machines for ordinal regression, International Journal of Computer Systems Science and Engineering 3 (1) (2009) 47–51.

[26] M. Perez-Ortiź, P. A. Gutiérrez, C. Hervás-Martínez, Projection-based ensemble learning for ordinal regression, IEEE transactions on Cybernetics 44 (5) (2013) 681–694.

[27] H. Jiang, Z. Yang, Z. Li, Non-parallel hyperplanes ordinal regression machine, Knowledge-Based Systems 216 (2021) 106593.

[28] O. Gouvert, T. Oberlin, C. Févotte, Ordinal non-negative matrix factorization for recommendation, in: International Conference on Machine Learning, PMLR, 2020, pp. 3680–3689.

[29] X. Shi, W. Cao, S. Raschka, Deep neural networks for rank-consistent ordinal regression based on conditional probabilities, Pattern Analysis and Applications 26 (3) (2023) 941–955.

[30] X. Liu, F. Fan, L. Kong, Z. Diao, W. Xie, J. Lu, J. You, Unimodal regularized neuron stick-breaking for ordinal classification, Neurocomputing 388 (2020) 34–44.

[31] Q. Li, J. Wang, Z. Yao, Y. Li, P. Yang, J. Yan, C. Wang, S. Pu, Unimodal-concentrated loss: Fully adaptive label distribution learning for ordinal regression, in: in Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 20513–20522.

[32] A. Shashua, A. Levin, Ranking with large margin principle: Two approaches, Advances in neural information processing systems 15 (2002).

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

[34] H.-T. Lin, L. Li, Reduction from cost-sensitive ordinal ranking to weighted binary classification, Neural Computation 24 (5) (2012) 1329–1367.

[35] S. Xiang, F. Nie, G. Meng, C. Pan, C. Zhang, Discriminative least squares regression for multiclass classification and feature selection, IEEE transactions on neural networks and learning systems 23 (11) (2012) 1738–1754.

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

[37] 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.

[38] J. Sánchez-Monedero, P. A. Gutiérrez, M. Pérez-Ortiz, Orca: A matlab/octave toolbox for ordinal regression, Journal of Machine Learning Research 20 (125) (2019) 1–5.

[39] D. Liu, J. Zhao, J. Wu, G. Yang, F. Lv, Multi-category classification with label noise by robust binary loss, Neurocomputing 482 (2022) 14–26.

[40] B. Frenay, M. Verleysen, Classification in the presence of label noise: a survey, IEEE transactions on Neural Networks and Learning Systems 25 (5) (2013) 845–869.

[41] J. Demšar, Statistical comparisons of classifiers over multiple data sets, The Journal of Machine learning research 7 (2006) 1–30.

[42] G. Xu, Z. Cao, B.-G. Hu, J. C. Principe, Robust support vector machines based on the rescaled hinge loss function, Pattern Recognition 63 (2017) 139–148.

[43] A. Lanitis, C. J. Taylor, T. F. Cootes, Toward automatic simulation of aging effects on face images, IEEE Transactions on pattern Analysis and machine Intelligence 24 (4) (2002) 442–455.