Learning from small data: a pairwise approach for ordinal regression

Yanzhu Liu, Xiaojie Li, Adams Wai Kin Kong
School of Computer Science and Engineering
Nanyang Technological University
Block N4, Nanyang Avenue 50, Singapore, 639798
{liuy0109, XLI16}@e.ntu.edu.sg, adamskong@ntu.edu.sg

Chi Keong Goh
Advanced Technology Centre
Rolls-Royce Singapore
6 Seletar Aerospace Rise, Singapore, 797575
ChiKeong.Goh@Rolls-Royce.com

Abstract— **Ordinal regression which aims to classify instances into ordinal categories has numerous applications. As a supervised learning problem, a large number of labeled data is needed to train an accurate model, in particular when the number of categories is large. Learning an effective ordinal classifier from a small dataset is a challenging task. This paper proposes a framework to transform the ordinal regression problem to a binary classification problem and then recover the ordinal information from the binary outputs. The labeled instances are paired up to train a binary classifier, and therefore, the number of training points is squared, which alleviates the lack of training points. The transformed binary classification problem is solved by a pairwise SVM method. Experimental results demonstrate that on 12 widely used benchmarks, the proposed method is effective comparing with the state-of-the-art ordinal regression methods.**

Keywords— ordinal regression; pairwise kernel; small data

I. INTRODUCTION

Ordinal regression is a supervised learning problem, where the objective is to predict discrete labels with a natural order. For example, movie reviews usually involve rating movies based on an ordinal scale such as 1 star to 5 stars and a movie with 4 stars has a better rating than those with 3 stars. Ordinal regression is an active research area because of numerous governmental, commercial and scientific applications, such as disease grading [1], sovereign credit rating [2] and risk rating for beetle infestation [3]. Although in the last decade big data problems attracted great attention, many real-world ordinal regression problems are in fact small data problems. For example, in computer-aided diagnostic problems, datasets of rare disease grading or tumor staging are usually fewer than 100 data points. These diseases affect a relatively small percentage of population and in many cases, collecting data is difficult, expensive and invasive. Therefore, their clinical and experimental records are not large. Even for more common diseases, datasets of medical diagnosis are not large. For example, some researchers explored solutions of automated grading age-related macular degeneration (AMD) [4], which is one of the leading causes of central vision loss in people aged over 50 years. Two popular databases Diaretdb0 [5] and Diaretdb1 [6] of AMD contain only 79 and 43 color fundus images respectively. Besides of the medical field, such small dataset problems also arise in failure prediction in engineering area. Because collecting run-to-failure data is very expensive, datasets are extremely small. Small dataset is an important issue in many applications and is studying by other researchers for classification [7] and foresting [8]. The purpose of this paper is to establish an effective ordinal regression model for small datasets.

Learning from a small dataset is challenging because lack of data increases uncertainty and easily causes overfitting. However, for ordinal regression problems, the ordinal relationship between instances in different categories is valuable information, which can be used to alleviate problems of small training sets. Recently, threshold approaches [9] have been investigated extensively. They assume that there is a latent function $f(\mathrm{x})$ mapping the instances to a real line, and the category of the instances is an interval on the real line. The natural order of interval boundaries on the real line represents the ordinal relationship between categories. Binary decomposition approaches transform an original dataset with ordinal labels into several datasets with binary labels. There are two popular decomposition approaches: training multiple models for sub problems [10] and training a single multi-output model [11]. The derived datasets for both of these approaches and the original dataset have the same number of instances. SVOR [12] and RED_SVM [13] are two state-of-the-art algorithms. SVOR is a threshold approach and RED_SVM is a binary decomposition approach. However, they are neither special for small datasets nor make use of the relationship between individual instances.

To deal with lack of training data, semi-supervised learning and transfer learning have been applied to ordinal regression [14][15]. However, small dataset problems are different from these two settings. Semi-supervised learning aims to make use of unlabeled data for training, typically given a small amount of labeled data with a large amount of unlabeled data, but in small dataset problems, both labeled and unlabeled data are few. Taking rare diseases as an example, the labeled data are instances with grading labels, such as low, moderate and high, and the unlabeled data are from patients with the disease but without the severity level labels. These unlabeled data are also difficult to obtain. Transfer learning aims to make use of other data from related domains for training. However, it is difficult to measure whether a dataset is related or not and hard to guarantee no negative transfer.

This paper proposes a framework to make use of relationship between two instances in every possible pairs to increase training samples. The framework transforms an ordinal regression problem with n training instances to a binary classification problem with $n^2 - \frac{n^2}{C}$ training instances (C is the number of ranks). A decoder is developed to predict the category of an instance from the binary outputs of the classifier. Although in this paper a revised SVM method is employed, other binary classification methods with high performance can be considered.

The contributions of this paper include:

- Increase the number of training points from n to $n^2 - \frac{n^2}{C}$ by transforming the ordinal regression problem to a binary classification problem.
- Modify SVM by employing pairwise kernels and introducing distances between different ranks into the constraints.
- Develop a decoder to predict the rank of a test point from the outputs of SVM.

The rest of this paper is organized as follows. Section II reviews the literature of ordinal regression. Section III describes the proposed ordinal regression framework and introduces the pairwise SVM and the decoder. Section IV reports the experimental results. Section V gives some conclusive remarks.

II. RELATED WORK

A number of machine learning methods have been proposed for ordinal regression. The survey paper published by Gutierrez et al. recently [9] summarized a set of ordinal regression algorithms in a taxonomy containing naive, binary decomposition and threshold approaches. Binary decomposition includes two types of approaches — single multi-output model and multiple binary models. Generally speaking, binary decomposition methods answer the question: "Is the rank of an instance x greater than k?" Frank and Hall (2001) [16] transformed data from m-rank ordinal regression to $m-1$ binary classification problems. In prediction phase, the rank labels of test points were assigned by applying ad-hoc rules on the predications of class probability. This method required training multiple classifiers. RED_SVM [13] improved the performance further. It extended each instance to m instances for m ranks and based on the derived dataset, a binary classifier was trained to predict the rank of an instance is greater or smaller than each rank. The proposed method in this paper converts the original regression problem to a single binary classification problem which answers the question for every two instances: “Whose rank is greater?” and it uses a tailor-made decoder to recover the rank labels from the outputs of the classifier.

Because of the high generalization performance of SVM, several SVM-based formulations have been proposed for ordinal regression. Herbrich et al. (1999) [17] proposed a SVM formulation based on a loss function of pairs of instances. Shashua and Levin (2002) [18] generalized SVM with multiple thresholds. They proposed two formulations: one is fixed-margin-based and the other one is sum-of-margins-based. SVOR [12] improved the fixed-margin-based SVM formulation further. SVOR is a state-of-the-art SVM-based ordinal regression method. The binary classifier in the proposed algorithm is also SVM-based, but other binary classifiers with good performance can be applied.

An important difference between ordinal regression and metric regression is that the distance between ranks are undefined for ordinal regression. Sanchez-Monedero et al. (2013) [19] proposed a method, PCDOC, which explores pairwise rank distances for ordinal regression problems. Sanchez-Monedero et al. modeled the projection from the input space to a 1-dimensional latent space directly using pairwise rank distance calculations. However, PCDOC is limited by scalability and computational time, not all pairs of instances being truly used.

III. A PAIRWISE APPROACH FOR ORDINAL REGRESSION

For the sake of clear presentation, the problem setting and notations are given first. An ordinal regression problem with m ranks denoted by $\mathrm{W} = \{1,2,\dots,m\}$ is considered, where the natural order of numbers indicates the order of ranks. Let (X, Y) be a training set with n labeled instances, i.e., $\mathrm{X} = \{x_1, x_2, \dots, x_n\}$, and $\mathrm{Y} = \{y_1, y_2, \dots, y_n\}$, where $x_i \in R^d$ is an input vector, and $\mathrm{y_i} \in \mathrm{W}$ is the rank label. (x_i, x_j) represents a pair of input vectors, and $I = \{(i,j) | x_i, x_j \in \mathrm{X}\}$ is defined as the index set of input pairs in the training set. $X_k \subseteq X$ is the subset of input vectors whose rank labels are all k. n_k denotes its size and $\mathrm{x_{kj}}$ denotes its j-th input vector. It is assumed that each rank has at least one instance, i.e., $X_k \neq \emptyset$. The task of ordinal regression is to predict the rank label y_t of a new input vector x_t.

A. A pairwise framework

The proposed pairwise ordinal regression (POR) framework is based on a simple intuition. Because there are some training instances in each rank, the prediction of a test point can be obtained by comparing it to each of training instances. More precisely, given a test point x_t, if we can answer "Is $y_t > y_i$ true?" for any $x_i \in \mathrm{X}$, the rank label y_t will be easily inferred from these answers. Figure 1 shows the POR framework, which is a realization of this intuition. The framework contains four steps: the first three steps solve the binary classification problem to answer above question, and the final step is to determine the rank label of a test point based on the answers.

POR starts from deriving new datasets from the original dataset in step 1. Any two instances from different ranks in the original training set are paired up to form one new instance, and the new instance is labeled as positive or negative according to the original ranks of the two entries. For example, two instances x_i and x_j come from different ranks y_i and y_j. If $y_i < y_j$, the new instance (x_i, x_j) will be labeled as +1; otherwise, it will be labeled as -1. The number of instances in the derived dataset is $n^2 - \sum_{k=1}^{m} n_k^2$, which is $\mathrm{O}(n^2)$. The dataset of the new binary problem has been extended quadratically to relieve the difficulties brought by small

datasets. The second step of POR is to train a pairwise SVM on the derived dataset. The input of this SVM is a pair (x_i, x_j), and its output is a binary label +1 or -1. The SVM kernels are pairwise: $\mathrm{K}: (\mathrm{X} \times \mathrm{X}) \times (\mathrm{X} \times \mathrm{X}) \to \mathrm{R}$. The distance between rank y_i and y_j is introduced into the constraints of the SVM, and the details will be discussed in Section III B. In the third step, given a test point x_t, it is paired up with all the training points to form a set of pairs $\{(x_t, x_i) | x_i \in \mathrm{X}\}$. Then the predicted binary label l_{ti} for all (x_t, x_i) will be computed and these labels indicate that the rank of x_t should be larger or smaller than the rank of x_i. Because the ranks of all the training points are known, the fourth step of POR is to decode the estimated rank $\hat{y}_t$ from $\{((x_t, x_i), l_{ti}) | x_i \in \mathrm{X}, l_{ti} \in \{+1, -1\}\}$ for x_t.

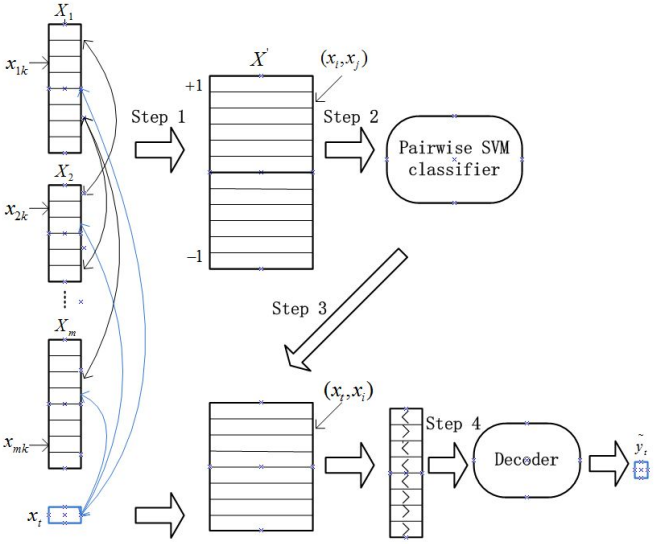

Fig.1 The flowchart of the POR framework

B. A pairwise SVM

Brunner et al. (2012) [20] proposed a pairwise SVM to tackle pairwise classification problems. The task of pairwise classification is to predict whether the instances a and b in a pair (a, b) belong to the same class or not. This pairwise SVM is modified in this paper for the rank comparison problem, which is to predict whether the rank of instance a in a pair (a, b) is smaller than that of instance b. The following formulation is defined for the rank comparison problem on the derived dataset.

$$\begin{aligned} \min_{\mathrm{w},\varepsilon} \frac{1}{2}\|w\|^2 + \mathrm{C}\sum_{(i,j)\in I} \varepsilon_{i,j} \\ \text{s.t.} \quad w \cdot \varphi(x_i, x_j) \geq d_{i,j} - \varepsilon_{i,j} \quad \text{if } y_i > y_j \\ w \cdot \varphi(x_i, x_j) \leq -d_{i,j} + \varepsilon_{i,j} \quad \text{if } y_i < y_j \\ \varepsilon_{i,j} \geq 0 \end{aligned} \tag{1}$$

where I is the index set, $\varepsilon_{i,j}$ is the slack variable with respect to pairs (x_i, x_j), $\varphi(\cdot,\cdot)$ is a function mapping pair (x_i, x_j) to a high dimension space and $d_{i,j} = |y_i - y_j|$. In the first two constraints, $d_{i,j} = |y_i - y_j|$ is used to quantify the distance between rank y_i and y_j. By introducing the distance into the constraints, the contributions of different pairs are distinguished. For example, assuming that x_1, x_2, x_3 are from rank 1, 2, 3 respectively; both (x_1, x_2) and (x_1, x_3) are positive instances for the new binary classification problem, but the contributions of these two instances should not be same because their rank distances are different.

The index set I is symmetric. In other words, if $(i, j) \in \mathrm{I}$, then $(j, i) \in \mathrm{I}$, and $i \neq j$. Therefore, Eq. 1 can be rewritten as following:

$$\begin{aligned} \min_{\mathrm{w},\varepsilon} \frac{1}{2}\|w\|^2 + \mathrm{C}\sum_{(i,j)\in I} \varepsilon_{i,j} \\ \text{s.t.} \quad \mathrm{sgn}(y_i - y_j) w \cdot \varphi(x_i, x_j) \geq d_{i,j} - \varepsilon_{i,j} \quad \forall (i,j) \in I \\ \varepsilon_{i,j} \geq 0 \end{aligned} \tag{2}$$

where $\mathrm{sgn}(\cdot)$ is a sign function. Eq. 2 is the pairwise SVM formulation proposed for rank comparison. The input of this SVM is a pair (x_i, x_j) and the corresponding kernels are defined on pairs: $\mathrm{K}: (\mathrm{X} \times \mathrm{X}) \times (\mathrm{X} \times \mathrm{X}) \to \mathrm{R}$, $\mathrm{K}\big((x_i, x_j), (x_k, x_l)\big) = \langle \varphi(x_i, x_j), \varphi(x_k, x_l) \rangle$. Pairwise kernels for ordinal regression must fulfill the following properties, (a) $\mathrm{K}\big((x_i, x_j), (x_k, x_l)\big) = \mathrm{K}\big((x_k, x_l), (x_i, x_j)\big)$, because according to Mercer's theorem, a kernel matrix must be symmetric and (b) $\mathrm{K}\big((x_i, x_j), (x_k, x_l)\big) = \mathrm{K}\big((x_j, x_i), (x_l, x_k)\big)$. Brunner et al. (2012) gave several examples of pairwise kernels, such as metric learning pairwise kernel, direct sum learning pairwise kernel, and tensor metric learning pairwise kernel. Any traditional kernel $\mathrm{K}: \mathrm{X} \times \mathrm{X} \to \mathrm{R}$ can in fact be used if a new feature vector representing a pair (x_i, x_j) is constructed in advance. In implementation, a lot of methods can be used to construct a feature vector x' from the pair (x_i, x_j), such as $x' = x_i - x_j$ and $x' = [x_i; x_j]$ which appends x_j to x_i.

After training the pairwise SVM, the decision function defined in Eq. 3 is used to predict the rank of a test point.

$$f(x_k, x_l) \triangleq \sum_{(i,j)\in I} \alpha_{i,j} \mathrm{sgn}(y_i - y_j) \mathrm{K}\big((x_i, x_j), (x_k, x_l)\big) \tag{3}$$

where $\alpha_{i,j}$ is the Lagrange multiplier of (x_i, x_j). For the ordinal regression problem, the decision function must fulfill the property $f(x_k, x_l) = -f(x_l, x_k)$, which is called negative symmetric. Obviously, if the kernel used in the SVM fulfills the property (c) $\mathrm{K}\big((x_i, x_j), (x_k, x_l)\big) = -\mathrm{K}\big((x_i, x_j), (x_l, x_k)\big)$, the decision function will be always negative symmetric.

C. A decoder

In the POR framework, once the binary prediction values are obtained, a rule is applied to determine the rank labels. In the prediction phase, a testing point x_t is paired up with all training points, and the pairs are inputted to the pairwise SVM. The decoder is designed to estimate the rank of x_t from the SVM outputs. Algorithm 1 presents the proposed decoding algorithm, where the rank of x_t is determined by majority voting. Assuming that the rank of x_t is equal to c, the decoder calculates how many SVM outputs from the input pairs fit the

assumption. The rank label is assigned to the c who fits the SVM outputs best.

Algorithm 1. Pseudo code for decoder

Input: Y = $\{y_1, \ldots, y_n\}$, and $L = \{l_1, \ldots, l_n | l_i \in \{1, -1\}\}$. l_i is the prediction of the pairwise SVM for (x_t, x_i), where x_t is the test point, and $x_i \in X$. $l_i = 1$ indicates the rank of x_t is higher than that of x_i; otherwise, the rank of x_t is lower.

Output: $\hat{y}_t$, the predicted rank of x_t.

1: for $c = 1$ to m do (where m is the number of ranks)
2: Let r_c be the number of correct predictions in L assuming the rank of x_t is c.
3: Initialize $r_c = 0$.
4: for $i = 1$ to n do
5: if $y_i < c$ and $l_i = -1$ then
6: $r_c = r_c + 1$
7: else if $y_i > c$ and $l_i = 1$ then
8: $r_c = r_c + 1$
9: end if
10: end for
11: Assign prediction accuracy $p_c = \frac{r_c}{n - n_c}$.
12: end for
13: return $\hat{y}_t = \text{argmax}_c(p_c)$.

In the literature, the prediction phase of the decomposition methods for ordinal regression using a single multi-output classifier and multiple binary classifiers can be unified under the Error Correcting Output Codes (ECOC) framework developed by Allwein [21]. The ECOC framework is proposed for multi-class classification originally. Take an one-against-all method for a 4-class classification problem as an example and let $h_j(\cdot), j = 1, \ldots, 4$ be a decision function of the binary classifier distinguishing whether an instance belongs to class j or not. A coding matrix $M \in \{+1, -1, 0\}^{m \times s}$ is defined associating with the decomposition method, where m is the number of classes, and s is the number of the decision functions. Table 1 is a coding matrix for 4 class classification problems solved by one-against-all methods. Each row is for one class and each column is for one decision function. For the sake of clear presentation, the first column is labeled as [1|2,3,4] for $h_1(\cdot)$, which considers class 1 as a positive class and the rest are negative classes. The rest of the columns are for $h_2(\cdot)$, $h_3(\cdot)$ and $h_4(\cdot)$. The elements in the matrix represent the training targets for different functions and different classes. For example, M_{11} is +1, meaning that $h_1(\cdot)$ uses +1 as a training target for class 1. In the prediction phase, given a test point x_t, the estimated values are obtained from all the classifiers, i.e., $h_1(x_t)$,..., $h_s(x_t)$. A predefined similarity metric $d(\cdot, \cdot)$ is used to measure the closeness between the i-th row of the coding matrix $M(i)$ and the estimated vector $Z = [h_1(x_t), \ldots, h_s(x_t)]$. The rank of x_t is determined by $\text{argmax}_i\big(d(M(i), Z)\big)$.

The decoder of the POR framework given in Algorithm 1 can be formulated as a special case under the ECOC framework. Table 2 is the coding matrix M' of POR for 4 ranks ordinal regression. Each row is for one rank and the proposed decision function $f(\cdot,\cdot)$ in Eq. 3 is run on all columns. Different columns represent different subsets of data with different ranks. Column $(x, \{r\})$ represents $f(\cdot,\cdot)$ running on pairs formed by an input vector and all instances from rank r. $M'_{ij} = +1$ indicates that (x_{ik}, x_{jk}) whose ranks are i and j respectively is used as a positive training sample, while $M'_{ij} = -1$ indicates that (x_{ik}, x_{jk}) is used as a negative training sample. $M'_{ii} = 0$ means that samples from the same rank are not used in training. The elements in the i-th row, except for the diagonal elements, are expected outputs for a test sample with rank i.

Table 1. Coding matrix of one-against-all method for 4 class classification

	Coding values			
Class	[1|2,3,4]	[2|1,3,4]	[3|1,2,4]	[4|1,2,3]
1	+1	-1	-1	-1
2	-1	+1	-1	-1
3	-1	-1	+1	-1
4	-1	-1	-1	+1

Table 2. Coding matrix of POR for 4-rank ordinal regression

	Coding values			
Rank	$(x, \{1\})$	$(x, \{2\})$	$(x, \{3\})$	$(x, \{4\})$
1	0	-1	-1	-1
2	+1	0	-1	-1
3	+1	+1	0	-1
4	+1	+1	+1	0

As described before, in the ECOC framework, the rank of a test point is determined by $\text{argmax}_i\big(d(M(i), Z)\big)$, where $M(i)$ is the i-th row vector of M and Z is the estimation vector of a test point x_t. For the decoder of POR, let $Z' = [z_1, \ldots, z_m]$ be the estimation tensor of a test point x_t and each element z_r be the prediction vector for the pairs of x_t and all instances from rank r, i.e., $z_r = \big[f(x_t, x_{r1}), \ldots, f\big(x_t, x_{rn_r}\big)\big]$. Define

$$d(M'(c), Z') = \frac{1}{n - n_c} \sum_{r=1,\ldots m, r \neq c} \frac{n_r + M'_{c,r} \mathbf{1}_{[n_r]} \cdot z_r{}^T}{2} \qquad (4)$$

where $\mathbf{1}_{[n_r]}$ is the all-ones vector with dimension one by n_r, $M'_{c,r}$ is the element in the c-th row and the r-th column in the coding matrix M', and n_c and n_r are the number of instances with rank c and r in the training set. If $M'_{c,r} = 1$, the equation $\frac{n_r + M'_{c,r} \mathbf{1}_{[n_r]} \cdot z_r{}^T}{2}$ inside the summation in Eq. 4 returns the number of ones in z_r, while if $M'_{c,r} = -1$, it returns the number of negative ones in z_r. Therefore, Eq. 4 is a mathematical representation of line 2 to line 11 in the Algorithm 1. In the Algorithm 1 the rank of a test point is determined by $\text{argmax}_c d(M'(c), Z')$. Clearly, the coder of POR is a special case of the ECOC framework.

IV. Experimental Results

To evaluate the performance of the proposed algorithm, it is compared with three state-of-the-art methods on 12 widely used benchmark datasets. Table 3 lists their details. The first

eight datasets are real ordinal regression datasets and the other four datasets are generated from UCI regression datasets [12] by discretizing the target values into ordinal quantities using equal-frequency binning. Each of them has no more than 300

Table 3. Ordinal regression benchmarks

Dataset	Partition	K	Q	Class Distribution
contact-lenses	18/6	6	3	(15,5,4)
pasture	27/9	25	3	(12,12,12)
squash-stored	39/13	51	3	(23,21,8)
squash-unstored	39/13	52	3	(24,24,4)
tae	113/38	54	3	(49,50,52)
newthyroid	161/54	5	3	(30,150,35)
bonrate	42/15	37	5	(6,33,12,5,1)
automobile	153/52	71	6	(3,22,67,54,32,27)
pyrim5	50/24	27	5	(7,28,17,12,10)
machine5	150/59	6	5	(152,27,13,7,10)
pyrim10	50/24	27	10	(2,2,14,14,13,5,10,4,3,7)
machine10	150/59	6	10	(115,37,21,6,8,5,3,4,4,6)

data points. Table 3 lists the partitions between training sets and testing sets. For example, 18/6 means that 18 instances in the database are for training and the rest 6 instances are for testing. K indicates the feature dimensions and Q indicates the number of ranks. The last column shows the number of points in different ranks. Note that some of them are highly imbalanced. More information of the datasets can be found in [12] and [19]. Two metrics are used to evaluate the performance of the methods. The first one is mean zero-one error (MZE) defined by $e = \frac{1}{|T|}\sum_{x_t \in T} [\![\hat{y}_t \neq y_t]\!]$, where T is a testing set, $|T|$ is its size, y_t is the ground truth of x_t, $\hat{y}_t$ is the prediction for y_t and $[\![\cdot]\!]$ is the indicator function. The second one is mean absolute error (MAE) defined by $e = \frac{1}{|T|}\sum_{x_t \in T} |\hat{y}_t - y_t|$.

Three state-of-the-art methods, SVOR [12], RED_SVM [13], and PCDOC [19] are compared with the proposed algorithm POR. Gaussian kernel is used in all of the methods. The hyper-parameters C and γ are selected respectively from $\{2^{-5}, 2^{-4}, 2^{-3}, 2^{-2}, 2^{-1}, 2^{0}, 2^{1}, 2^{2}, 2^{3}, 2^{4}, 2^{5}\}$ and $\{2^{-5}, 2^{-4}, 2^{-3}, 2^{-2}, 2^{-1}, 2^{0}\}$ using 5-fold cross validation. Some features in these datasets are binary but the others are real numbers. Normalization does have impacts on most of classifiers, including SVM. Two types of normalization are considered in this paper: normalizing all features and normalizing only none-binary features. Therefore, in the experiments, there are three types of features for each dataset: original features, normalizing features, and partially normalizing features, and they are also selected through cross-validation. More clearly, 5 fold cross-validation on the training set is used to select the best hyper-parameters and the best corresponding normalization scheme simultaneously. The normalization method used in the experiments is Standard Score, which normalizes a feature v as $\frac{v-\mu}{\sigma}$, where μ is the mean value of this feature in all instances and σ is the standard deviation. For SVOR [12], RED_SVM [13], and PCDOC [19], the model selection methods reported in the original papers are used. For fair comparison, the same experimental settings reported in [19] are applied to the first eight datasets and those reported in [12] are applied to the rest datasets. Tables 4 and 5 list the mean errors and their standard derivations of the first eight databases. The best performance is highlighted. None of the methods can perform the best for all eight datasets. In terms of MZE, POR can achieve the best results on three datasets, while the others can achieve the best results on no more than two datasets. In terms of MAE, PCDOC wins four times, while POR wins three times.

Tables 6 and 7 list the results on the four discrete regression datasets. In terms of both MZE and MAE, POR wins two times; RED_SVM and PCDOC win one time, while SVOR cannot get any best result. Table 8 summarizes the win/loss of the four methods on MZE and MAE for the 12 benchmarks. In terms of MZE, POR performs the best. In terms of MAE, POR and PCDOC perform similarly.

V. Conclusion

In this paper, a new pairwise ordinal regression algorithm is proposed for small data problems. The number of training data

Table 4. Mean zero-one error (MZE) on real ordinal regression datasets

	contact-lenses	pasture	squash-stored	squash-unstored	Tae	newthyroid	bondrate	automobile
RED_SVM	**0.300 ± 0.111**	0.352 ± 0.134	0.336 ± 0.104	0.251 ± 0.086	0.478 ± 0.074	0.031 ± 0.022	**0.447 ± 0.073**	0.316 ± 0.055
SVOR	0.367 ± 0.127	0.333 ± 0.120	0.361 ± 0.118	0.236 ± 0.103	**0.410 ± 0.066**	0.031 ± 0.021	0.453 ± 0.092	0.361 ± 0.076
PCDOC	0.311 ± 0.095	0.344 ± 0.103	**0.315 ± 0.123**	0.305 ± 0.084	0.418 ± 0.064	**0.027 ± 0.020**	0.460 ± 0.101	0.322 ± 0.060
POR	0.344 ± 0.166	**0.304 ± 0.149**	0.359 ± 0.111	**0.226 ± 0.093**	0.420 ± 0.094	0.030 ± 0.024	0.449 ± 0.150	**0.303 ± 0.102**

Table 5. Mean absolute error (MAE) on real ordinal regression datasets

	contact-lenses	pasture	squash-stored	squash-unstored	Tae	newthyroid	bondrate	automobile
RED_SVM	0.378 ± 0.169	0.359 ± 0.142	0.346 ± 0.110	0.251 ± 0.086	0.515 ± 0.087	0.032 ± 0.022	0.598 ± 0.088	**0.393 ± 0.073**
SVOR	0.506 ± 0.167	0.333 ± 0.120	0.372 ± 0.126	0.239 ± 0.109	0.461 ± 0.081	0.031 ± 0.021	0.591 ± 0.102	0.424 ± 0.090
PCDOC	**0.367 ± 0.154**	0.348 ± 0.104	**0.326 ± 0.141**	0.305 ± 0.084	**0.457 ± 0.071**	**0.027 ± 0.020**	0.568 ± 0.126	0.397 ± 0.093
POR	0.489 ± 0.113	**0.304 ± 0.149**	0.369 ± 0.104	**0.226 ± 0.093**	0.539 ± 0.072	0.030 ± 0.024	**0.556 ± 0.114**	0.431 ± 0.062

Table 6. Mean zero-one error (MZE) on discrete regression datasets

	pyrim5	machine5
RED_SVM	**0.413 ± 0.063**	0.264 ± 0.010
SVOR	0.517 ± 0.086	0.431 ± 0.054
PCDOC	0.483 ± 0.088	**0.178 ± 0.036**
POR	0.442 ± 0.075	0.180 ± 0.048
	pyrim10	machine10
RED_SVM	0.762 ± 0.021	0.572 ± 0.013
SVOR	0.719 ± 0.066	0.655 ± 0.045
PCDOC	0.704 ± 0.071	0.385 ± 0.054
POR	**0.641 ± 0.196**	**0.333 ± 0.064**

Table 7. Mean absolute error (MAE) on discrete regression datasets

	pyrim5	machine5
RED_SVM	**0.454 ± 0.086**	0.478 ± 0.031
SVOR	0.615 ± 0.127	0.462 ± 0.062
PCDOC	0.552 ± 0.116	0.202 ± 0.046
POR	0.541 ± 0.049	**0.200 ± 0.038**
	pyrim10	machine10
RED_SVM	1.304 ± 0.040	0.842 ± 0.022
SVOR	1.294 ± 0.046	0.990 ± 0.026
PCDOC	**1.088 ± 0.159**	0.494 ± 0.082
POR	1.133 ± 0.091	**0.476 ± 0.029**

Table 8. Win/loss summary

win/loss	MZE	MAE
RED_SVM	3 / 9	2 / 10
SVOR	1 / 11	0 / 12
PCDOC	3 / 9	5 / 7
POR	5 / 7	5 / 7

is increased quadratically when a pairwise approach is used to overcome the problem of lack of data. SVM is revised such that the ordinal information is embedded in the constraints and a pairwise kernel is used to project data pairs to a high dimensional space. A decoder, which takes the SVM binary outputs from the pairs formed by the test points and all training points as inputs, is designed to recover the ordinal category of test points. The experimental results show that the proposed algorithm is comparable with the state-of-the-art methods.

ACKNOWLEDGMENT

This work was conducted within Rolls-Royce@NTU Corporate Lab with support from the National Research Foundation (NRF) Singapore under the Corp Lab@University Scheme.

REFERENCES

[1] Kuranova, P. and Hajdukova, Z. (2014). Ordinal regression for classification of patients into one of the individual phadiatop test groups. In *Digital Technologies (DT), 2014 10th International Conference on*, 174-178. IEEE.

[2] Fernandez-Navarro, F., Campoy-Munoz, P., la Paz-Marin, M.d., Hervas-Martinez, C., and Yao, X. (2013). Addressing the eu sovereign ratings using an ordinal regression approach. *Cybernetics, IEEE Transactions on*, 43(6), 2228-2240.

[3] Robertson, C., Wulder, M.A., Nelson, T.A., White, J.C., et al. (2009). Preliminary risk rating for mountain pine beetle infestation of lodgepole pine forests over large areas with ordinal regression modelling, volume 2009. Pacific Forestry Centre.

[4] Deepak, K.S. and Sivaswamy, J. (2012). Automatic assessment of macular edema from color retinal images. *Medical Imaging, IEEE Transactions on*, 31(3), 766-776.

[5] DIARETDB, D. (2007). Evaluation database and methodology for diabetic retinopathy algorithms.

[6] Kalviainen, R.V.J.P.H. and Uusitalo, H. (2007). Diaretdb1 diabetic retinopathy database and evaluation protocol. *Medical Image Understanding and Analysis* 2007, 61.

[7] Dougherty, Edward R., Lori A. Dalton, and Francis J. Alexander. "Small data is the problem." *2015 49th Asilomar Conference on Signals, Systems and Computers*. IEEE, 2015.

[8] Chang, Che-Jung, et al. "A forecasting model for small non-equigap data sets considering data weights and occurrence possibilities." *Computers & Industrial Engineering* 67 (2014): 139-145.

[9] Gutierrez, P.A., Perez-Ortiz, M., Sanchez-Monedero, J., Fernandez-Navarro, F., and Hervas-Martinez, C. (2016). Ordinal regression methods: survey and experimental study. *Knowledge and Data Engineering, IEEE Transactions on*, 28(1), 127-146.

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

[11] Cheng, J., Wang, Z., and Pollastri, G. (2008). A neural network approach to ordinal regression. In Neural Networks, 2008. IJCNN 2008.(IEEE World Congress on Computational Intelligence). IEEE International Joint Conference on, 1279-1284. IEEE.

[12] Chu, W. and Keerthi, S.S. (2007). Support vector ordinal regression. *Neural computation*, 19(3), 792-815.

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

[14] Srijith, P., Shevade, S., and Sundararajan, S. (2013). Semi-supervised Gaussian process ordinal regression. In *Machine Learning and Knowledge Discovery in Databases*, 144-159. Springer.

[15] Seah, C.W., Tsang, I.W., and Ong, Y.S. (2012). Transductive ordinal regression. *Neural Networks and Learning Systems, IEEE Transactions on*, 23(7), 1074-1086.

[16] Frank, E. and Hall, M. (2001). A simple approach to ordinal classification. *Springer*.

[17] Herbrich, R., Graepel, T., and Obermayer, K. (1999). Large margin rank boundaries for ordinal regression. *Advances in neural information processing systems*, 115-132.

[18] Shashua, A. and Levin, A. (2002). Taxonomy of large margin principle algorithms for ordinal regression problems. *Advances in neural information processing systems*, 15, 937-944.

[19] Sanchez-Monedero, J., Gutierrez, P.A., Tino, P., and Hervas-Martnez, C. (2013). Exploitation of pairwise class distances for ordinal classification. *Neural computation*, 25(9), 2450-2485.

[20] Brunner, C., Fischer, A., Luig, K., and Thies, T. (2012). Pairwise support vector machines and their application to large scale problems. *The Journal of Machine Learning Research*, 13(1), 2279-2292.

[21] Allwein, E.L., Schapire, R.E., and Singer, Y. (2001). Reducing multiclass to binary: A unifying approach for margin classifiers. *The Journal of Machine Learning Research*, 1, 113-141.