Robust Deep Ordinal Regression under Label Noise

by

Bhanu Garg, Naresh Manwani

in

ACML

Report No: IIIT/TR/2020/-1

Centre for Visual Information Technology
International Institute of Information Technology
Hyderabad - 500 032, INDIA
October 2020

Proceedings of Machine Learning Research 129:782-796, 2020 ACML 2020

Robust Deep Ordinal Regression under Label Noise

Bhanu Garg BHANUGARG05@GMAIL.COM and **Naresh Manwani** NARESH.MANWANI@IIIT.AC.IN
International Institute of Information Technology, Hyderabad, India

Editors: Sinno Jialin Pan and Masashi Sugiyama

Abstract

The real-world data is often susceptible to label noise, which might constrict the effectiveness of the existing state of the art algorithms for ordinal regression. Existing works on ordinal regression do not take label noise into account. We propose a theoretically grounded approach for class conditional label noise in ordinal regression problems. We present a deep learning implementation of two commonly used loss functions for ordinal regression, which is - 1) robust to label noise, and 2) rank consistent for a good ranking rule. We verify these properties of the algorithm empirically and show robustness to label noise on real data and rank consistency. To the best of our knowledge, this is the first approach for robust ordinal regression models.

Keywords: Ordinal regression, deep learning, robust learning, label noise

1. Introduction

Ordinal regression, or sometimes ranking learning, is a supervised learning problem where the objective is to predict categories or labels on an ordinal scale. Ordinal regression frequently arises in social sciences and information retrieval, where human preferences play a significant role. The label space does not have a distance metric defined over it, distinguishing it from regression problems. The relative ordering among the labels distinguishes it from multiclass classification.

Applications of ordinal regression include age detection from face images, predicting credit rating (Hirk et al., 2018), predicting the progress of diseases such as Alzheimer's (Doyle et al., 2014), decoding information on neural activity from fMRI scans (Satake et al., 2018) etc. Such varied and high impact applications make ordinal regression an important learning model.

Ordinal regression is commonly described by a real-valued function and a set of ordered thresholds. Many state-of-the-art supervised learning methods use risk-minimization techniques to learn the model, which requires a suitable loss function. Commonly used zero-one loss for classification problems would ignore the ordinal nature of the labels. Instead, mean absolute error (MAE), defined as the absolute difference between the ranks of the predicted and the true label, is used to evaluate ordinal regression approaches' performance. However, MAE is not a continuous function, which makes risk minimization computationally hard. As a consequence, convex surrogates of MAE are used for risk minimization. One such loss function is the implicit constrained loss (l_{IMC}) proposed in Chu and Keerthi (2005). It is used to learn maximum margin ordinal regression function (Chu and Keerthi, 2005; Antoniuk et al., 2016). Perceptron based online ranking algorithms are proposed in Crammer and Singer (2001); Manwani (2019). The l_{IMC} and the above online algorithms preserve the ordering of thresholds on risk minimization. Ordering of thresholds can also be forced by posing the constraints explicitly (Chu and Keerthi, 2005). Li and Lin (2006) propose an approach that converts ordinal regression learning into extended binary classification. Neural networks have also

been used to learn ordinal regression (Cao et al., 2019; Cheng, 2007). In Cao et al. (2019), authors use cross entropy-based loss (l_{CE}) for ordinal regression and show that l_{CE} intrinsically maintains the ordering among the thresholds. A deep neural network model for ordinal regression is proposed in Liu et al. (2018). All the works above assume that the data used for the training does not suffer from label noise.

Because of practical constraints with the way data is collected, the data labels might be noisy. Subjective errors, measurement errors, manual errors, etc. are why we get noisy labels. Because of this label noise in the data, we may not learn the correct underlying ordinal regression function. Thus, we need to develop robust methods to determine the actual underlying classifier even when we have label noise in the training data.

Label noise problems in the context of binary and multiclass classification problems is an active area of research. A thorough literature survey of label noise-robust methods for classification is provided in Frenay and Verleysen (2014). In Manwani and Sastry (2013); Ghosh et al. (2017), authors provide sufficient symmetry conditions on loss functions that would ensure robustness to label noise for classification. It is shown in Manwani and Sastry (2013); Ghosh et al. (2017) that convex loss functions are not robust under label noise for binary classification. Similar results are shown by Ghosh et al. (2017) for multiclass classification. On the other hand, the approach in Natarajan et al. (2013) assumes the knowledge of noise rates and finds an unbiased estimator of the true risk under noisy labels. The authors also show that the approach generalizes well on the unseen data. Liu and Tao (2016) uses importance reweighting for learning in the presence of class conditional noise, and provide a method to estimate noise rates using density ratio estimation.

Robust learning of ordinal regression models in the presence of label noise remains an unaddressed problem. In this paper, we propose an approach for learning robust ordinal regression with label noise. Our approach is inspired by the method of the unbiased estimators (Natarajan et al., 2013). We have made the following contributions.

Contributions

1. We propose a label noise model for ordinal regression, namely inversely decaying noise. When the noise parameter is equal for all classes, we call it uniformly decaying noise. When the parameter changes with the class, we call it class conditional inversely decaying noise.

2. We propose an unbiased estimator based approach for label noise-robust ordinal regression. We work with losses l_{CE} and l_{IMC}. We show that unbiased estimators $\tilde{l}_{CE}$ and $\tilde{l}_{IMC}$ are also rank consistent.

3. We propose deep learning methods for robust ordinal regression which use $\tilde{l}_{CE}$ and $\tilde{l}_{IMC}$ as loss functions. We further show that stochastic gradient descent (SGD) on $\tilde{l}_{CE}$ and $\tilde{l}_{IMC}$ preserves the ordering of the thresholds and results in a rank consistent model.

4. We also provide generalization bounds for the proposed approach.

5. We experimentally show the effectiveness of the proposed approach on various datasets. We show that our method can learn robust deep ordinal regression models well.

This is the first attempt to address the label noise issue in ordinal regression to the best of our knowledge.

2. Ordinal Regression

Each example is of the form $(\mathbf{x}_i, y_i) \in \mathcal{X} \times \mathcal{Y}$, where $\mathcal{X} \subseteq \mathbb{R}^d$ and $\mathcal{Y} = \{1, \ldots, K\}$. The labels in $\mathcal{Y}$ are ordered, i.e. $1 \prec 2 \prec \ldots \prec K$. Let $\mathcal{D}$ be the unknown joint distribution on $\mathcal{X} \times \mathcal{Y}$ from which N i.i.d. samples are drawn. An ordinal regression function $f : \mathcal{X} \to \mathcal{Y}$ is described using a function $g : \mathcal{X} \to \mathbb{R}$ and thresholds $b_1, \ldots, b_K$ as follows.

$$f(\mathbf{x}) = 1 + \sum_{k=1}^{K-1} \mathbb{I}_{\{g(\mathbf{x})+b_k>0\}} = \min_{i \in [K]} \{i : g(\mathbf{x}) + b_i \leq 0\}$$

Let $\mathbf{b} = [b_1 \ \ldots \ b_{K-1}]^T \in \mathbb{R}^{K-1}$. Thus, function $g(.)$ and thresholds $\mathbf{b}$ are the parameters to be optimized upon. We assume $b_K = -\infty$. We must ensure that $b_1 \geq \ \ldots \ \geq b_{K-1}$ to maintain the ordering among the classes (Crammer and Singer, 2001; Li and Lin, 2006).

2.1. Loss Functions for Ordinal Regression

We now describe commonly used loss functions which capture the discrepancy between the predicted label and the true label.

1. l_{MAE}**:** Mean absolute error finds the absolute difference between the predicted label and the true label (Antoniuk et al., 2016).

$$l_{MAE}(g(\mathbf{x}), \mathbf{b}, y) = \sum_{i=1}^{y-1} \mathbb{I}_{\{g(\mathbf{x})+b_i<0\}} + \sum_{i=y}^{K-1} \mathbb{I}_{\{g(\mathbf{x})+b_i \geq 0\}} \tag{1}$$

   $l_{MAE} = 0$ whenever $-b_{y-1} \leq g(\mathbf{x}) \leq -b_y$. Optimizing l_{MAE} is computationally hard as it is not continuous. Thus, in practice, we use convex surrogates of l_{MAE} as loss functions to minimize the risk and find the parameters of $g(.)$ and thresholds $b_1, \ldots, b_{K-1}$.

2. l_{IMC}**:** It is a convex surrogate of l_{MAE} (Chu and Keerthi, 2005) which implicitly maintains the ordering of the thresholds b_i's.

$$l_{IMC}(g(\mathbf{x}), \mathbf{b}, y) = \sum_{i=1}^{y-1} \max\left(0, 1 - g(\mathbf{x}) - b_i\right) + \sum_{i=y}^{K-1} \max\left(0, 1 + g(\mathbf{x}) + b_i\right) \tag{2}$$

   For a given example-label pair $\{\mathbf{x}, y\}$, $l_{IMC}(g(\mathbf{x}), \mathbf{b}, y) = 0$ only when $g(\mathbf{x}) + b_i \geq 1, \forall i < y$ and $g(\mathbf{x}) + b_i \leq -1, \forall i \geq y$. Let $z_i = \mathbb{I}_{\{i<y\}} - \mathbb{I}_{\{i \geq y\}}, \ i \in [K]$. Thus, $z_i = 1, \ \forall i < y$ and $z_i = -1, \ \forall i \geq y$. Thus, $l_{IMC}(g(\mathbf{x}), \mathbf{b}, y) = 0$ requires that $z_i(g(\mathbf{x}) + b_i) \geq 1, \ \forall i \in [K-1]$. Thus,

$$l_{IMC}(f(\mathbf{x}), \mathbf{b}, y) = \sum_{i=1}^{K-1} \max\left[0, 1 - z_i \left(g(\mathbf{x}) + b_i\right)\right].$$

   In Chu and Keerthi (2005), it is shown that l_{IMC} is implicitly rank consistent. Thus, at the optimal solution, $b_1 \leq \ldots \leq b_{K-1}$.

3. l_{CE}: Cross entropy loss (Cheng, 2007; Cao et al., 2019) for ordinal regression is described as follows.

$$l_{CE}(g(\mathbf{x}), \mathbf{b}, y) = -\sum_{j=1}^{K-1} [z_j \log(\sigma(g(\mathbf{x}) + b_j)) + (1 - z_j) \log(1 - \sigma(g(\mathbf{x}) + b_j))] \quad (3)$$

where $\sigma(a) = (1+e^{-a})^{-1}$ is the sigmoid function. Also, $z_j = 1,\ \forall j < y$ and $z_j = 0,\ \forall j \geq y$. It is shown that l_{CE} is rank consistent (Cao et al., 2019). Thus, minimizer of the risk under l_{CE}, will satisfy condition $b_1 \geq \ldots \geq b_{K-1}$.

3. Label Noise Generation Model in Ordinal Regression

Real-world datasets are seldom perfect and often suffer from various noise issues. One kind of noise in the data has noisy labels, where we get corrupted samples $(\mathbf{x}_i, \tilde{y}_i),\ i = 1 \ldots N$, where $\tilde{y}_i$ are the noisy labels. The noisy label $\tilde{y}_i$ could be different from the true label y_i. A detailed discussion of the sources of noise can be found in Frenay and Verleysen (2014). For classification problems, learning in presence of label noise is a well studied problem (Manwani and Sastry, 2013; Natarajan et al., 2013; Ghosh et al., 2017; Liu and Tao, 2016).

Let $P(\tilde{y} = j|y = i, \mathbf{x}) = \eta_{(i,j)}(\mathbf{x})$ be the probability of observing label j for example $\mathbf{x}$ whose true label is i. Uniform label noise ($P(\tilde{y} = j|y = i, \mathbf{x}) = \eta,\ \forall i \neq j$ and $\forall \mathbf{x}$) and class conditional label noise ($P(\tilde{y} = j|y = i, \mathbf{x}) = \eta_{(i,j)},\ \forall \mathbf{x} \in C_i, i \neq j$) are some of the commonly used noise models (Frenay and Verleysen, 2014; Ghosh et al., 2017). For class conditional noise model, the noise model is entirely represented by noise matrix $\mathbf{N}$ such that $\mathbf{N}_{i,j} = \eta_{(i,j)}$.

3.1. Label Noise Models for Ordinal Regression

We note that the uniform and class conditional noise models described above do not take the label's ordinal aspect into account due to the following reasoning. In practice, when humans annotate the data (say rating a product), it is likely that even if there is an error in the labeling, the human has a sense of label "category". So they might be able to classify the product as good or bad, but there might be an error in imputing the correct rank. Thus, when they make errors in the ranking, they are more likely to choose neighboring classes more often than the far away rank. Therefore, it would make sense to study label noise models in which the noise probability of a label far away is less than that of a label nearer. With this in mind, we propose the following noise model. In the proposed noise model, the noise rate does not depend on $\mathbf{x}$.

- **Inversely decaying noise:** Here, the probability of mislabeling is inversely proportional to the absolute difference between the true rank and the rank of incorrect label. Thus, $\eta_{(i,j)} = \frac{\rho_i}{|i-j|},\ \forall i \neq j$ where ρ_i is a parameter for class i. The diagonal element $\eta_{(i,i)}$ is defined as $\eta_{(i,i)} = 1 - \sum_{j=1, j\neq i}^{K} \eta_{(i,j)}$. If $\rho_i = \rho,\ \forall i$ then the noise model is called uniformly inversely decaying.

  Example 1: Here, we see the noise matrix corresponding to a uniformly inversely decaying noise model. Let $\rho = 0.15$, and there are four classes, then the noise matrix and its inverse are as follows.

$$\underset{\mathbf{N}}{\begin{bmatrix} 0.725 & 0.15 & 0.075 & 0.05 \\ 0.15 & 0.625 & 0.15 & 0.075 \\ 0.075 & 0.15 & 0.625 & 0.15 \\ 0.05 & 0.075 & 0.15 & 0.725 \end{bmatrix}} \quad \underset{\mathbf{N}^{-1}}{\begin{bmatrix} 1.45 & -0.32 & -0.08 & -0.05 \\ -0.32 & 1.77 & -0.36 & -0.09 \\ -0.08 & -0.36 & 1.77 & -0.32 \\ -0.05 & -0.09 & -0.32 & 1.45 \end{bmatrix}}$$

Observe that in the uniform version of the noise model, the probability of not flipping the label $\eta_{(i,i)}$ is maximum at the extremes and is minimum for mid-labels. Labels in the middle of the label range are more susceptible to noise than labels at the end, which conforms to human behavior while ranking objects on an ordinal scale. Say a human is asked to rate a product on a scale of 1 to 10, 1 being poor quality, and 10 being excellent. The human would be more confident when assigning extreme ratings, i.e., an excellent or terrible product is easy to distinguish from the rest. Thus, $\eta_{i,i}$ for extreme ratings would be high. On the other hand, the distinction between labels in the middle range is ambiguous. And hence identifying the true rating becomes more difficult in the middle range compared to the extremes. Hence, the decreasing values of $\eta_{i,i}$ in the middle range. With this being said, real-world noise may not follow these properties and may be of any form. A noise model where $\mathbf{N}$ is asymmetric is called an asymmetric noise model. While we assume symmetric noise matrix in our proofs on rank consistency, we empirically show that the results hold good under practical noise models.

3.2. Properties of Noise Matrix of the Proposed Noise Models

We observe the following properties of the matrix $\mathbf{N}$.

- Since $\eta_{(i,j)}$ is a function of $|i-j|$, matrix $\mathbf{N}$ becomes symmetric. $\mathbf{N}^{-1}$ is also symmetric, because inverse of a symmetric matrix is symmetric.
- Each row (and column) has a sum of 1 as it represents a probability distribution of a random variable. $\sum_i \eta_{(i,j)} = \sum_j \eta_{(i,j)} = 1$. Thus, matrix $\mathbf{N}$ is doubly-stochastic.
- If $\eta_{(i,i)} > 0.5$, then $\eta_{(i,i)} > \sum_{j,j\neq i} \eta_{(j,i)}$. This implies that the matrix $\mathbf{N}$ is (strictly) diagonally dominant. With this assumption, matrix $\mathbf{N}$ becomes non-singular (Horn and Johnson, 2012).
- Row sum (and column sum) of $\mathbf{N}^{-1}$ is 1 as follows. Since $\mathbf{N}^{-1}\mathbf{N} = \mathbf{I}$, for all $j, k \in [K]$, we get, $\sum_{i=1}^{K} \eta_{(j,i)}\mathbf{N}^{-1}_{(i,k)} = \mathbb{I}_{\{j=k\}}$. Now, summing over $j = 1 \ldots K$, we get $\sum_{j=1}^{K}\sum_{i=1}^{K} \eta_{(j,i)}\mathbf{N}^{-1}_{(i,k)} = \sum_{j=1}^{K} \mathbb{I}_{\{j=k\}}$ By rearranging the terms and using the fact that $\sum_{i=1}^{K} \eta_{(i,j)} = 1, \ \forall j$, we get, $\sum_{i=1}^{K} \mathbf{N}^{-1}_{(i,k)} = 1, \ \forall k$. Thus column sum of $\mathbf{N}^{-1}$ is 1. Since $(\mathbf{N}^{-1})^T = \mathbf{N}^{-1}$, $\sum_{i=1}^{K} \mathbf{N}^{-1}_{(k,i)} = 1$. Thus, row sum of $\mathbf{N}^{-1}$ is also 1. The same can be seen in Example 1.
- Every column (row) of $\mathbf{N}^{-1}$ has negative entries. This can verify it by contradiction. Suppose $\mathbf{N}^{-1}$ has only non-negative elements in any column (all cannot be zero since $\mathbf{N}^{-1}$ is an invertible matrix). Consider the dot product of i^{th} row of $\mathbf{N}$ and j^{th} $(j \neq i)$ column of $\mathbf{N}^{-1}$, which is $\sum_{t=1}^{K} \eta_{(i,t)}\mathbf{N}^{-1}_{(t,j)}$. Since $\mathbf{N}^{-1}$ has all non-negative elements, we get, $\sum_{t=1}^{K} \eta_{(i,t)}\mathbf{N}^{-1}_{(t,j)} > 0$. But, $\sum_{t=1}^{K} \eta_{(i,t)}\mathbf{N}^{-1}_{(t,j)} = 0, \forall j \neq i$, which is a contradiction. Hence, in every column (and row) of $\mathbf{N}^{-1}$, there is a negative element. The same can be seen in Example 1.

4. Robust Ordinal Regression in the Presence of Label Noise

In this section, we propose a methodology for robust ordinal regression. As discussed earlier, we get corrupted samples $(\mathbf{x}_i, \tilde{y}_i),\ i = 1 \dots N$, where $\tilde{y}_i$ is the noisy label. Our approach is based on an unbiased estimator (Natarajan et al., 2013; Patrini et al., 2017). Thus, we use unbiased estimator $\tilde{l}(f(\mathbf{x}), \tilde{y})$ of the loss $l(f(\mathbf{x}), y)$. We use the noise matrix N to construct the unbiased estimator $\tilde{l}(f(\mathbf{x}), \tilde{y})$ of $l(f(\mathbf{x}), y)$, which means

$$\mathbb{E}_{\tilde{y}}[\tilde{l}(f(\mathbf{x}), \tilde{y})] = l(f(\mathbf{x}), y). \tag{4}$$

Thus, optimising the risk based on $\tilde{l}(f(\mathbf{x}), \tilde{y})$ in presence of label noise results in optimising risk based on $l(f(\mathbf{x}), y)$ in the absence of noise. Using eq.(4), we get the following equation.

$$l(f(\mathbf{x}), y) = \mathbb{E}_{\tilde{y}}[\tilde{l}(f(\mathbf{x}), \tilde{y})] = \sum_{\tilde{y}=1}^{K} \eta_{y,\tilde{y}} \tilde{l}(f(\mathbf{x}), \tilde{y}) \tag{5}$$

Let $\tilde{\mathbf{L}} = \begin{bmatrix}\tilde{l}(f(\mathbf{x}), 1) & \dots & \tilde{l}(f(\mathbf{x}), K)\end{bmatrix}^T$ and $\mathbf{L} = \begin{bmatrix}l(f(\mathbf{x}), 1) & \dots & l(f(\mathbf{x}), K)\end{bmatrix}^T$, then the system of equations in (5) can be written as $\mathbf{N}\tilde{\mathbf{L}} = \mathbf{L}$. Hence, we get $\tilde{\mathbf{L}} = \mathbf{N}^{-1}\mathbf{L}$. Note that the transformation of l to $\tilde{l}$ depends only on the noise rates. Also, function $\tilde{l}$ need not be convex even if we begin with convex l. In this paper, we work with losses l_{CE} and l_{IMC}. It can be easily verified that $\tilde{l}_{CE}$ and $\tilde{l}_{IMC}$ are no more convex functions.

4.1. Rank Consistency of $\tilde{l}_{CE}$ and $\tilde{l}_{IMC}$

The loss functions used in a robust method for ordinal regression also need to be rank consistent. While we know that both l_{CE} and l_{IMC} are rank consistent (Chu and Keerthi, 2005; Cao et al., 2019), it is required to show that $\tilde{l}_{CE}$ and $\tilde{l}_{IMC}$ are also rank consistent. The following theorem proves it.

Theorem 1 *$\tilde{l}_{CE}$ and $\tilde{l}_{IMC}$ are rank consistent.*

Proof of this Theorem is provided in the Supplementary file. We now discuss the deep-learning approach for learning robust ordinal regression models.

4.2. Deep Learning Model for Robust Ordinal Regression

In this paper, we propose a deep neural network-based approach to ordinal regression using $\tilde{l}_{CE}$ and $\tilde{l}_{IMC}$ as loss functions. These approaches are robust to label noise.

4.2.1. Approach 1: Based on Loss $\tilde{l}_{CE}$

We use the neural network architecture described in Figure 1. The penultimate layer, whose output is denoted as $g(\mathbf{x})$, shares a single weight (but different bias) with all nodes in the pre-final layer. Let $h_j(\mathbf{x}) = \sigma(g(\mathbf{x}) + b_j)$, where $g(\mathbf{x})$ is a function of input vector $\mathbf{x}$ computed using initial layers of the network. The pre-final layer in the network has $K-1$ nodes where $P(y > j|\mathbf{x}) = h_j(\mathbf{x}) = \sigma(g(\mathbf{x}) + b_j)$ is the output of j^{th} node in that layer. b_j is the bias term corresponding to the j^{th} node

in the pre-final layer. We use back-propagation algorithm (SGD) to minimize the loss function $\tilde{l}_{CE}$ as follows.

$$\tilde{l}_{CE}(g(\mathbf{x}), \mathbf{b}, \tilde{y}) = \sum_{j=1}^{K} \mathbf{N}^{-1}_{(\tilde{y},j)} l_{CE}(g(\mathbf{x}), \mathbf{b}, j) = -\sum_{j=1}^{K} \mathbf{N}^{-1}_{(\tilde{y},j)} \sum_{i=1}^{K-1} \left(\log h_i(\mathbf{x})^{z_i^j} + \log(1 - h_i(\mathbf{x}))^{(1-z_i^j)} \right)$$

Where $\mathbf{N}^{-1}$ is the inverse of the noise matrix and $z_i^j = 1,\ \forall i < j$ and $z_i^j = 0,\ \forall i \geq j$.

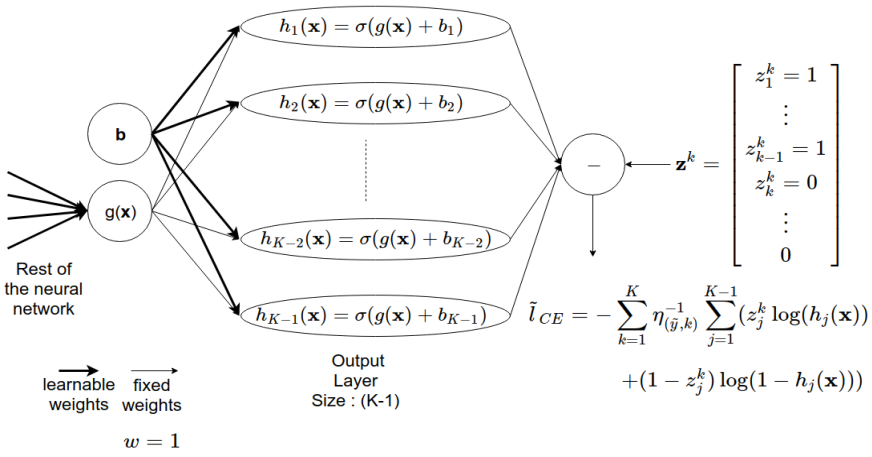


Figure 1: Neural network for robust ordinal regression based on $\tilde{l}_{CE}$.

We observe that the back-propagation algorithm for training the above network ensures the thresholds' orderings in the expected sense as follows.

Theorem 2 *SGD on $\tilde{l}_{CE}$ maintains ordering among the thresholds. Let $b_i^t,\ i \in [K-1]$ be the thresholds at the t^{th} round and $b_i^t - b_{i+1}^t \geq 0,\ i \in [K-1]$ holds true. Then, we observe that $\mathbb{E}_{\tilde{y}^t}[b_i^{t+1} - b_{i+1}^{t+1}] \geq 0,\ \forall i \in [K-1]$.*

The proof is given in the supplementary file. Note that the ordering consistency proof can be shown only in the expected sense because the back-propagation updates involve the terms containing $\mathbf{N}^{-1}_{(\tilde{y},j)}$ which is a random variable. To normalize it, we need to take expectation with respect to $\tilde{y}$. The theorem shows the correctness of the robust ordinal regression approach based on loss $\tilde{l}_{CE}$.

4.2.2. APPROACH 2: BASED ON LOSS $\tilde{l}_{IMC}$

We now give a neural network architecture for robust ordinal regression based on $\tilde{l}_{IMC}$. The architecture is described in Figure 2. Similar to Approach 1, here also, the penultimate layer shares a single weight (but different bias) with all nodes in the pre-final layer. Pre-final layer has $K-1$ nodes whose outputs are $h_1(\mathbf{x}), \ldots, h_{K-1}(\mathbf{x})$. Note that, here, $h_j(\mathbf{x}) = g(\mathbf{x}) + b_j$ where $g(\mathbf{x})$ is some function of the weights of neural network leading to all but last layer and the input vector $\mathbf{x}_i$. We minimize the following loss function using back-propagation.

$$\tilde{l}_{IMC}(g(\mathbf{x}^t), \mathbf{b}, \tilde{y}^t) = \sum_{j=1}^{K} \mathbf{N}^{-1}_{(\tilde{y}^t,j)} \sum_{i=1}^{K-1} \left[1 - z_i^j \left(g(\mathbf{x}^t) + b_i \right) \right]_+$$

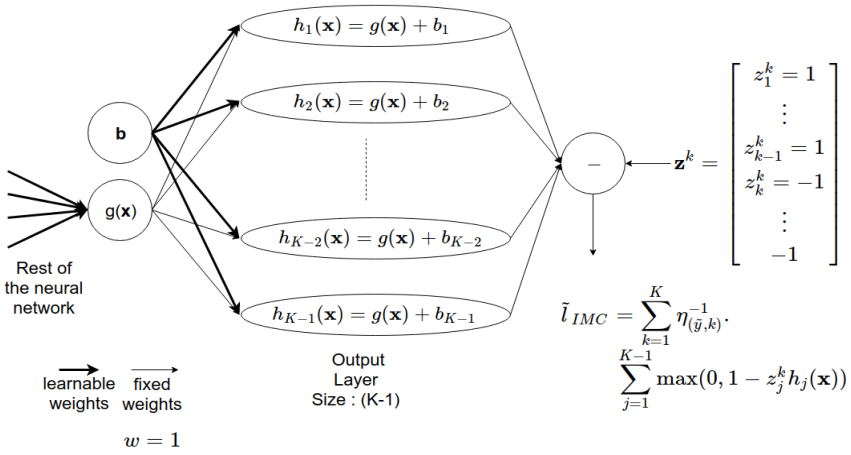


Figure 2: Neural network for robust ordinal regression using $\tilde{l}_{IMC}$.

Where $z_i^j = 1,\ \forall i < j$ and $z_i^j = -1,\ \forall i \geq j$.

Theorem 3 *SGD on $\tilde{l}_{IMC}$ maintains ordering among the thresholds. Let $b_i^t,\ i \in [K-1]$ be the thresholds at t^{th} round and $b_i^t - b_{i+1}^t \geq 0,\ i \in [K-1]$ holds true. Then, we observe that $\mathbb{E}_{\tilde{y}^t}[b_i^{t+1} - b_{i+1}^{t+1}] \geq 0,\ \forall i \in [K-1]$.*

The proof is given in the supplementary file. Note that the ordering consistency proof can be shown only in the expected sense due to the similar reasons as Theorem 2.

4.3. Estimating Noise Rates

We use the noise rate estimation method proposed in Patrini et al. (2017) for our problem. To estimate the noise rates, we treat ordinal regression as a multiclass problem. Equations for calculating noise rates involve solving for n^2 variables from n equations. Patrini et al. (2017) resolves this problem by making the assumption that there exists a sample x_i for every class i in the clean data such that $P(y = i|x_i) = 1$ (Theorem 3 in Patrini et al. (2017)), which works well for multiclass problems. We describe some of the possible approaches to estimate noise rates.

- Approach 1 (failed): We tried using a similar approach for noise estimation for ordinal regression cases using the architecture presented in Fig 1 and Fig 2. However, the "perfect assumption" used in Patrini et al. (2017) is no more valid here. We believe that is because in ordinal regression, the probability domains of two labels (the domain of P(x—y¿i)!=0) overlap, unlike in the case of multi-class problems (where the domain is P(x—y=i)!=0) and it's a more reasonable assumption. To visualize, think a dog and car images to come from non-overlapping sets. Because of this reason, the noise rates estimated using this approach resulted in poor practical results, and hence we do not show the experimental results.

- Approach 2 (used and works): Here, we treat ordinal regression as a multiclass problem. This approach gave reasonable noise rate estimates, and the results are discussed in Section 5. This approach requires us to train another neural network, which is inconvenient and time-consuming.

Since we get reasonable noise rate estimates using Approach 2, we proceed to present the results of Approach 2 in our paper, by using the same experimental setting used in Patrini et al. (2017).

4.4. Generalization Bounds

We represent the total risk of loss functions l_{IMC} and l_{CE} as sum of risks of $K-1$ binary classifiers i.e

$$R_{l,D}(g,\mathbf{b}) = \mathbb{E}_D[l(g(\mathbf{x}),\mathbf{b},y)] = \sum_{i=1}^{K-1} \mathbb{E}_D[l^i(g(\mathbf{x}),\mathbf{b},z_i^y)] = \sum_{i=1}^{K-1} R_{l^i,D}(g,\mathbf{b}))$$

where l^i , $1 \leq i \leq K-1$ represents the loss at the i^{th} binary classifier. Let $\hat{f} \in \arg\min_{f\in\mathcal{F}} \hat{R}_{\tilde{l},S}(f)$ and $f^* \in \arg\min_{f\in\mathcal{F}} R_{l,D}(f)$ where $\mathcal{F}$ is the hypothesis class of the function f. This proof is inspired from Theorem 3 in Natarajan et al. (2013).

Theorem 4 *If $\mathfrak{R}(\mathcal{F})$ is the maximum Rademacher complexity of the function class $\mathcal{F}$ among the $K-1$ binary classifiers, and the loss l is L-Lipschitz, then with probability at least $1-\delta$,*

$$R_{l,D}(\hat{f}) \leq R_{l,D}(f^*) + 2(K-1)\Big(2ML_\rho\mathfrak{R}(\mathcal{F}) + \sqrt{\frac{log(1/\delta)}{2n}}\Big)$$

where $\mathfrak{R}(\mathcal{F}) \coloneqq \max_{i=1}^{K-1} \mathbb{E}_{X_t,\epsilon_t}[\sup_{f_i\in\mathcal{F}} \frac{1}{n}\epsilon_t f_i(X_t)]$, and ϵ_t are i.i.d. symmetric Bernoulli random variables.

The proof is available in the supplementary file. Theorem 4 shows that the risk (on clean distribution D) of classifier $\hat{f}$ learned under $\tilde{l}$ with label noise is bounded by the risk of the classifier from l without label noise. Using an unbiased estimator, we pay the higher Lipschitz-constant price for $\tilde{l}$ and thus needs a larger training sample to generalize well. We discuss further in Section 5.

5. Experiments

We conduct experiments on synthetic and real datasets and compare them with benchmark algorithms.

Dataset Description: We perform experiments on one synthetic dataset and five real datasets. The details of the datasets are as below.

- **Synthetic Dataset:** Synthetic dataset used in the paper is generated as follows. Let f_i for $i = 1,2,3,4$ be four mixture density functions corresponding to 4 classes defined as follows.

$$f_1(\mathbf{x}) = 0.45\mathcal{U}([-1,0]\times[-1,1]) + 0.5\mathcal{U}([-4,-3]\times[0,1]) + 0.05\mathcal{U}([-10,0]\times[-5,5])$$
$$f_2(\mathbf{x}) = 0.45\mathcal{U}([0,1]\times[-1,1]) + 0.5\mathcal{U}([6,7]\times[-1,0]) + 0.05\mathcal{U}([0,10]\times[-5,5])$$
$$f_3(\mathbf{x}) = 0.45\mathcal{U}([19,20]\times[-1,1]) + 0.5\mathcal{U}([16,17]\times[-2,-1]) + 0.05\mathcal{U}([10,20]\times[-5,5])$$
$$f_4(\mathbf{x}) = 0.45\mathcal{U}([20,21]\times[-1,1]) + 0.5\mathcal{U}([26,27]\times[-3,-2]) + 0.05\mathcal{U}([20,30]\times[-5,5])$$

  We generate 2000 i.i.d points from each of the f_i and label the points using the following hyperplanes: $\mathbf{w} = \begin{bmatrix}1 & 0\end{bmatrix}$, $\mathbf{b} = \begin{bmatrix}0 & 10 & 20\end{bmatrix}$.

- **Real Datasets:** We show results on California dataset[1]. Other than that, we also show results on Boston, Abalone, Computer and MSLR[2]. The details of these datasets are given in Table 1.

Table 1: Data Description

Data	Instances	Features	Labels
Synthetic	8000	2	4
Boston	506	13	5
Abalone	4177	8	4
Computer	8192	21	5
California	20640	8	4
MSLR-WEB-10K	1200192	136	5

Data Pre-Processing and Hyper-Parameter Tuning: Each feature is scaled to have zero mean and unit variance coordinate wise. For hyperparameter tuning, we make a grid for two parameters: the learning rate and the hidden layer's size. We use 5-fold cross-validation to select the optimal parameters for the model. The number of epochs is chosen to be 300, and we observe that the loss converges for all models. We chose AdamW optimiser (Kingma and Ba, 2014; Loshchilov and Hutter, 2019) as the optimising algorithm with $\beta_1, \beta_2 = (0.9, 0.999)$ and L_2 penalty with weight decay of 0.01 - the default parameters (Kingma and Ba, 2014). We use ReLu as the activation function in the hidden layers for all datasets except for synthetic. We use a Linear function to demonstrate our method pictographically. All codes are written in PyTorch. The hyperparameters are tuned for noisy labels (both training and testing) using loss l, and the same parameters are used for the other two models, as described below. This is to ensure a stricter test for the performance of the proposed unbiased estimator.

We assume that the obtained datasets have no prior noise in the labels to make it possible to see the effects of noise on the ordinal regression models. We split the dataset into 80% and 20% independently 20 times, and train the following models corresponding to both l_{CE} and l_{IMC} on clean and noisy data. (1) l:trained using loss function l; (2) $\tilde{l}$-KR: trained using $\tilde{l}$ with known noise rates; (3) $\tilde{l}$-EST: trained using $\tilde{l}$ with estimated noise rates. The mean of MAE and $0-1$ error and the standard deviation of these over 20 trials are presented. We artificially induce noise in the training dataset labels and then evaluate the models on clean test data.

Since our proofs for rank consistency require symmetric noise matrix, we present results on noisy labels with the uniform inversely decaying model using $\rho = 0.15$, which is symmetric. The results are presented in Table 3. Further, since real-world noise may not be symmetric, we check rank consistency on asymmetric noise. The results are shown in Table 4.

Estimating noise rates: We train a multiclass neural network with negative log-likelihood loss. Following (Patrini et al., 2017), instead of taking $\arg\max$ to chose $\mathbf{x}_i$ we use 99 percentile. Table 2 shows the estimated noise matrix for California housing dataset.

Discussion on Synthetic Dataset: We compare our method of unbiased estimator with the benchmark deep learning ordinal regression method of using l (Cheng, 2007; Cao et al., 2019). To see the efficacy of unbiased estimator graphically, we use synthetic dataset as described in the Section

[1]The California Housing dataset can be found http://lib.stat.cmu.edu/datasets/

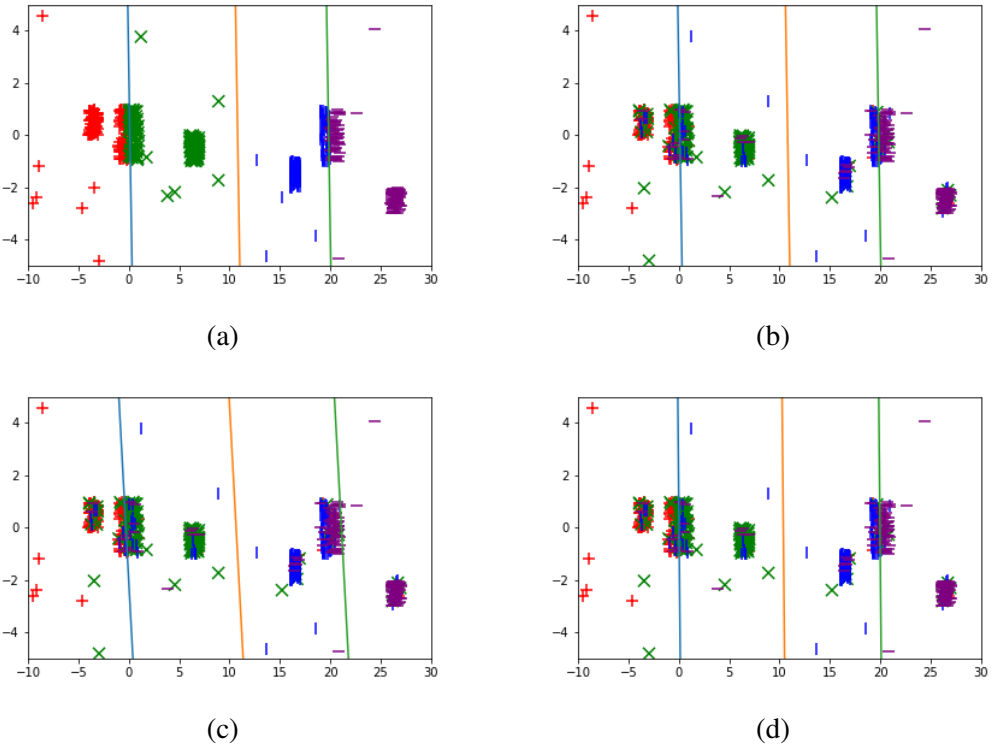
[2]The other datasets can be found at https://www.dcc.fc.up.pt/ ltorgo/Regression/DataSets.html

$$\begin{bmatrix} 0.725 & 0.15 & 0.075 & 0.05 \\ 0.15 & 0.625 & 0.15 & 0.075 \\ 0.075 & 0.15 & 0.625 & 0.15 \\ 0.05 & 0.075 & 0.15 & 0.725 \end{bmatrix} \quad \begin{bmatrix} 0.80 & 0.14 & 0.03 & 0.02 \\ 0.1 & 0.64 & 0.15 & 0.06 \\ 0.03 & 0.11 & 0.63 & 0.23 \\ 0.02 & 0.11 & 0.18 & 0.69 \end{bmatrix}$$

(a) (b)

Table 2: (a) Actual noise matrix used for California housing dataset, and (b) Estimated noise matrix.

5. In Figure 3.(c), we see that the noise changes the orientation of the model when trained using l on noisy data, giving sub-optimal results. On the other hand, $\tilde{l}$ produces robust classifier as shown in Figure 3.(d).



(a) (b)

(c) (d)

Figure 3: Results of different algorithms on Synthetic Dataset. (a) True classifier on clean data, (b) True classifier shown on noisy labels, (c) Classifier trained on noisy labels using l, (d) Classifier trained on noisy labels using $\tilde{l}$ with known noise rates

Discussion on Real Datasets: The performance of l_{CE} is consistently better than l_{IMC} across datasets in Table 3. Further, the performance of l with noise is seen to degrade more for l_{IMC} compared to l_{CE} which can be distinctly seen in Abalone and MSLR datasets, whereas in other datasets l_{CE} performs at least as good as l_{IMC}.

	Loss fn	Mean Absolute Error		Mean Zero-one Error	
		Clean Data	Noisy data	Clean Data	Noisy data
Boston	l_{CE}	$\mathbf{0.36 \pm 0.05}$	0.54 ± 0.09	0.33 ± 0.04	0.47 ± 0.07
	$\tilde{l}_{CE}$-KR	$\mathbf{0.36 \pm 0.05}$	$\mathbf{0.52 \pm 0.05}$	0.33 ± 0.04	0.43 ± 0.04
	$\tilde{l}_{CE}$-EST	0.38 ± 0.06	0.57 ± 0.06	0.34 ± 0.05	0.51 ± 0.05
	l_{IMC}	0.37 ± 0.04	0.54 ± 0.10	0.34 ± 0.03	0.46 ± 0.08
	$\tilde{l}_{IMC}$-KR	0.37 ± 0.04	$\mathbf{0.50 \pm 0.05}$	0.34 ± 0.03	0.41 ± 0.04
	$\tilde{l}_{IMC}$-EST	0.37 ± 0.05	0.53 ± 0.08	0.33 ± 0.04	0.47 ± 0.06
Abalone	l_{CE}	$\mathbf{0.42 \pm 0.02}$	0.46 ± 0.03	0.40 ± 0.02	0.44 ± 0.03
	$\tilde{l}_{CE}$-KR	$\mathbf{0.42 \pm 0.02}$	$\mathbf{0.44 \pm 0.02}$	0.40 ± 0.02	0.41 ± 0.02
	$\tilde{l}_{CE}$-EST	0.44 ± 0.03	0.47 ± 0.04	0.42 ± 0.03	0.45 ± 0.04
	l_{IMC}	$\mathbf{0.42 \pm 0.02}$	0.54 ± 0.03	0.40 ± 0.02	0.50 ± 0.02
	$\tilde{l}_{IMC}$-KR	$\mathbf{0.42 \pm 0.02}$	$\mathbf{0.43 \pm 0.02}$	0.40 ± 0.02	0.41 ± 0.02
	$\tilde{l}_{IMC}$-EST	0.45 ± 0.02	0.48 ± 0.02	0.42 ± 0.02	0.45 ± 0.01
Computer	l_{CE}	0.27 ± 0.01	0.38 ± 0.02	0.26 ± 0.01	0.37 ± 0.02
	$\tilde{l}_{CE}$-KR	0.27 ± 0.01	0.36 ± 0.01	0.26 ± 0.01	0.33 ± 0.01
	$\tilde{l}_{CE}$-EST	0.27 ± 0.01	$\mathbf{0.35 \pm 0.01}$	0.26 ± 0.01	0.33 ± 0.01
	l_{IMC}	0.27 ± 0.01	0.39 ± 0.03	0.26 ± 0.01	0.38 ± 0.03
	$\tilde{l}_{IMC}$-KR	0.27 ± 0.01	0.35 ± 0.02	0.26 ± 0.01	0.33 ± 0.02
	$\tilde{l}_{IMC}$-EST	0.27 ± 0.01	$\mathbf{0.34 \pm 0.01}$	0.26 ± 0.01	0.32 ± 0.01
California	l_{CE}	0.30 ± 0.01	0.38 ± 0.02	0.29 ± 0.01	0.33 ± 0.01
	$\tilde{l}_{CE}$-KR	0.30 ± 0.01	$\mathbf{0.34 \pm 0.01}$	0.29 ± 0.01	0.33 ± 0.01
	$\tilde{l}_{CE}$-EST	0.30 ± 0.01	0.35 ± 0.01	0.29 ± 0.01	0.34 ± 0.01
	l_{IMC}	0.31 ± 0.01	0.41 ± 0.04	0.30 ± 0.01	0.41 ± 0.04
	$\tilde{l}_{IMC}$-KR	0.31 ± 0.01	$\mathbf{0.34 \pm 0.01}$	0.30 ± 0.01	0.33 ± 0.01
	$\tilde{l}_{IMC}$-EST	0.31 ± 0.00	0.35 ± 0.01	0.30 ± 0.01	0.34 ± 0.01
MSLR	l_{CE}	0.55 ± 0.01	0.63 ± 0.01	0.49 ± 0.01	0.59 ± 0.01
	$\tilde{l}_{CE}$-KR	0.55 ± 0.01	$\mathbf{0.55 \pm 0.01}$	0.49 ± 0.01	0.49 ± 0.02
	$\tilde{l}_{CE}$-EST	$\mathbf{0.53 \pm 0.01}$	0.62 ± 0.01	0.46 ± 0.01	0.52 ± 0.02
	l_{IMC}	$\mathbf{0.55 \pm 0.01}$	0.71 ± 0.01	0.49 ± 0.01	0.68 ± 0.01
	$\tilde{l}_{IMC}$-KR	$\mathbf{0.55 \pm 0.01}$	$\mathbf{0.55 \pm 0.01}$	0.49 ± 0.01	0.50 ± 0.01
	$\tilde{l}_{IMC}$-EST	0.56 ± 0.01	0.66 ± 0.01	0.49 ± 0.01	0.55 ± 0.01

Table 3: Comparison of l, $\tilde{l}$-known noise rate, and $\tilde{l}$-estimated noise rate for each of CE and IMC loss functions on clean and symmetric noise label noise

	Loss fn	Mean Absolute Error		Mean Zero-one Error	
		Clean Data	Noisy data	Clean Data	Noisy data
Abalone	l_{CE}	$\mathbf{0.43 \pm 0.01}$	0.51 ± 0.02	0.40 ± 0.01	0.49 ± 0.03
	$\tilde{l}_{CE}$-KR	$\mathbf{0.43 \pm 0.01}$	$\mathbf{0.44 \pm 0.02}$	0.40 ± 0.01	0.41 ± 0.02
	$\tilde{l}_{CE}$-EST	$\mathbf{0.43 \pm 0.01}$	0.47 ± 0.01	0.41 ± 0.01	0.44 ± 0.01

Table 4: Comparison of l, $\tilde{l}$-known noise rate, and $\tilde{l}$-estimated noise rate for CE loss function on clean and real world asymmetric noise label noise

We also observe that the noise rate estimates were reasonable to work in the unbiased estimator where the datasets had a MAE error of less than 35% on clean data (Synth, Computer, California). Here the performance of $\tilde{l}$-KR and $\tilde{l}$-EST were at par. The large deviations in noise rate estimates come because of violations of Statement-1 in Theorem-3 (Patrini et al., 2017), which is more likely

when the MAE is high for clean data. We also observe that the unbiased estimator performs well, even with approximate noise rates. This is in line with observations made by Natarajan et al. (2013).

The performance of $\tilde{l}$ for Boston data is just at par with l because the Boston dataset is small ($\approx$ 500 samples). For huge dataset MSLR, $\tilde{l}$-KR with noise performs as good as l on clean data. This indicates that a comparatively more significant number of samples are needed for the unbiased estimator to perform well and be noise-robust. Further, if the number of examples is large, then errors in noise estimates effect the performance of $\tilde{l}$-ES more. In the MSLR dataset, the $\tilde{l}$-ES performs only as a little better than l. In Computer and California datasets, $\tilde{l}$-ES has approx. 10% better performance than l.

Further, we see that asymmetric noises in Table 4 deteriorate the performance of l much more than symmetric noise models. Since l_{CE} has been more robust compared to l_{IMC}, for asymmetric noise model we only show results of l_{CE}. For Abalone dataset, the symmetric noise increases the MAE by $\approx 10\%$, while asymmetric noise increases MAE by $\approx 20\%$. The intuition is that in symmetric noise, the push on the thresholds while making gradient descent is similar to both sides. The final model has parameters similar to the actual model. In contrast, the push in the asymmetric noise model is imbalanced so that the final model parameters deviate from the true classifier.

Discussion on rank consistency: The proofs of rank consistency use expectation in the difference between adjacent thresholds. To check for rank consistency, we check the threshold's ordering after each update to the neural network. If the thresholds aren't ordered, we flag the iteration. We have proved theoretically that symmetric $\mathbf{N}$ gives a rank consistent model. We need to understand what happens when noise matrix $\mathbf{N}$ in the unbiased estimator is asymmetric. To save space, we only report the average of the number of iterations with unordered thresholds for $\tilde{l}_{CE}$-EST and $\tilde{l}_{IMC}$-EST for synthetic, California housing datasets, and $\tilde{l}_{CE}$-KR and $\tilde{l}_{CE}$-EST for Abalone in Table 5 over the 20 iterations of training. For Synthetic and housing datasets, we corrupt the labels by asymmetric noise model, but the unbiased estimator's estimated matrix is asymmetric. We use an asymmetric noise model for Abalone to corrupt the labels and report results on this known asymmetric matrix and its estimate used in the unbiased estimator. The results for other datasets are similar. We observe that even if thresholds are reversed for some iteration, they get quickly corrected. All the final models of all the datasets and noise models in this paper were rank consistent.

	Loss fn	Clean Data	Noisy Data
Synth	l_{CE}-EST	0.4/67200	0.7/67200
	l_{IMC}-EST	0.0/67200	0.2/67200
California	l_{CE}-EST	0.0/247680	0.2/247680
	l_{IMC}-EST	0.2/247680	0.2/247680
Abalone (Assym. $\mathbf{N}$)	l_{CE}-KR	0.0/41800	0.3/41800
	l_{CE}-EST	0.6/50160	0.3/50160

Table 5: Iterations with unordered thresholds / total iterations

6. Conclusions and Future Work

In this paper, we propose a label noise model for ordinal regression. We then offer an unbiased estimator approach for learning robust ordinal regression models. We show that the models under $\tilde{l}_{CE}$ and $\tilde{l}_{IMC}$ are also rank consistent, which is a desirable property for ordinal regression. We

empirically verify the efficiency of the proposed end-to-end method on synthetic as well as real datasets. While performing the experiments for $\tilde{l}$-EST, we do not make any assumptions on the noise model.

For further study, we could consider coming up with methods to make estimating noise rates more reliable. We could also look into the effects of non-symmetric label noise on the model. This is the first study of ordinal regression under label noise.

References

Kostiantyn Antoniuk, Vojtěch Franc, and Václav Hlaváč. V-shaped interval insensitive loss for ordinal classification. *Machine Learning*, 103(2):261–283, May 2016.

Wenzhi Cao, Vahid Mirjalili, and Sebastian Raschka. Consistent rank logits for ordinal regression with convolutional neural networks. *CoRR*, abs/1901.07884, 2019.

Jianlin Cheng. A neural network approach to ordinal regression. *CoRR*, abs/0704.1028, 2007. URL http://arxiv.org/abs/0704.1028.

Wei Chu and S. Sathiya Keerthi. New approaches to support vector ordinal regression. In *ICML*, 2005.

Koby Crammer and Yoram Singer. Pranking with ranking. In *NIPS*, pages 641–647, 2001.

Orla M. Doyle, Eric Westman, Andre F. Marquand, Patrizia Mecocci, Bruno Vellas, Magda Tsolaki, Iwona Kłoszewska, Hilkka Soininen, Simon Lovestone, Steve C. R. Williams, and Andrew Simmons. Predicting progression of alzheimer's disease using ordinal regression. *PLOS ONE*, 9 (8):1–10, 08 2014.

B. Frenay and M. Verleysen. Classification in the presence of label noise: A survey. *IEEE Transactions on Neural Networks and Learning Systems*, 25(5):845–869, 2014.

Aritra Ghosh, Himanshu Kumar, and PS Sastry. Robust loss functions under label noise for deep neural networks. In *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.

Rainer Hirk, Kurt Hornik, and Laura Vana. Multivariate ordinal regression models: an analysis of corporate credit ratings. *Statistical Methods & Applications*, Aug 2018.

Roger A. Horn and Charles R. Johnson. *Matrix Analysis*. Cambridge University Press, New York, NY, USA, 2nd edition, 2012.

Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *International Conference on Learning Representations*, 12 2014.

Ling Li and Hsuan-Tien Lin. Ordinal regression by extended binary classification. In *Proceedings of the 19th International Conference on Neural Information Processing Systems*, NIPS'06, pages 865–872, Cambridge, MA, USA, 2006. MIT Press. URL http://dl.acm.org/citation.cfm?id=2976456.2976565.

T. Liu and D. Tao. Classification with noisy labels by importance reweighting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(3):447–461, March 2016.

Yanzhu Liu, Adams Wai-Kin Kong, and Chi Keong Goh. A constrained deep neural network for ordinal regression. *CVPR*, pages 831–839, 2018.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=Bkg6RiCqY7.

Naresh Manwani. PRIL: Perceptron Ranking Using Interval Labeled Data. In *CoDS-COMAD*, pages 78–85, Kolkata, India, 2019.

Naresh Manwani and P. S. Sastry. Noise tolerance under risk minimization. *IEEE Trans. Cybernetics*, 43(3):1146–1151, 2013.

Nagarajan Natarajan, Inderjit S. Dhillon, Pradeep Ravikumar, and Ambuj Tewari. Learning with noisy labels. In *NIPS*, pages 1196–1204, 2013.

Giorgio Patrini, Alessandro Rozza, Aditya Krishna Menon, Richard Nock, and Lizhen Qu. Making deep neural networks robust to label noise: A loss correction approach. In *CVPR*, July 2017.

Emi Satake, Kei Majima, Shuntaro C. Aoki, and Yukiyasu Kamitani. Sparse ordinal logistic regression and its application to brain decoding. *Frontiers in Neuroinformatics*, 12:51, 2018.