Rank consistent ordinal regression for neural networks with application to age estimation

Wenzhi Cao[a], Vahid Mirjalili[b], Sebastian Raschka[a,**]

[a]*University of Wisconsin-Madison, Department of Statistics, 1300 University Ave, Madison, WI 53705, USA*
[b]*Michigan State University, Department of Computer Science & Engineering, 428 South Shaw Lane, East Lansing, MI 48824, USA*

ABSTRACT

In many real-world prediction tasks, class labels include information about the relative ordering between labels, which is not captured by commonly-used loss functions such as multi-category cross-entropy. Recently, the deep learning community adopted ordinal regression frameworks to take such ordering information into account. Neural networks were equipped with ordinal regression capabilities by transforming ordinal targets into binary classification subtasks. However, this method suffers from inconsistencies among the different binary classifiers. To resolve these inconsistencies, we propose the COnsistent RAnk Logits (CORAL) framework with strong theoretical guarantees for rank-monotonicity and consistent confidence scores. Moreover, the proposed method is architecture-agnostic and can extend arbitrary state-of-the-art deep neural network classifiers for ordinal regression tasks. The empirical evaluation of the proposed rank-consistent method on a range of face-image datasets for age prediction shows a substantial reduction of the prediction error compared to the reference ordinal regression network.

1. Introduction

Ordinal regression (also called ordinal classification), describes the task of predicting labels on an ordinal scale. Here, a ranking rule or classifier h maps each object $\mathbf{x}_i \in \mathcal{X}$ into an ordered set $h : \mathcal{X} \rightarrow \mathcal{Y}$, where $\mathcal{Y} = \{r_1 < ... < r_K\}$. In contrast to classification, the labels provide enough information to order objects. However, as opposed to metric regression, the difference between label values is arbitrary.

While the field of machine learning has developed many powerful algorithms for predictive modeling, most algorithms have been designed for classification tasks. The extended binary classification approach proposed by Li and Lin (2007) forms the basis of many ordinal regression implementations. However, neural network-based implementations of this approach commonly suffer from classifier inconsistencies among the binary rankings (Niu et al., 2016). This inconsistency problem among the predictions of individual binary classifiers is illustrated in Figure 1. We propose a new method and theorem for guaranteed classifier consistency that can easily be implemented in various neural network architectures.

Furthermore, along with the theoretical rank-consistency guarantees, this paper presents an empirical analysis of our approach to challenging real-world datasets for predicting the age of individuals from face images using our method with convolutional neural networks (CNNs). Aging can be regarded as a non-stationary process since age progression effects appear differently depending on the person's age. During childhood, facial aging is primarily associated with changes in the shape of the face, whereas aging during adulthood is defined mainly by changes in skin texture (Ramanathan et al., 2009; Niu et al., 2016). Based on this assumption, age prediction can be modeled using ordinal regression-based approaches (Yang et al., 2010; Chang et al., 2011; Cao et al., 2012; Li et al., 2012).

The main contributions of this paper are as follows:

1. The consistent rank logits (CORAL) framework for ordinal regression with theoretical guarantees for classifier consistency;
2. Implementation of CORAL to adapt common CNN architectures, such as ResNet (He et al., 2016), for ordinal regression;
3. Experiments on different age estimation datasets showing that CORAL's guaranteed binary classifier consistency improves predictive performance compared to the reference framework for ordinal regression.

**Corresponding author:
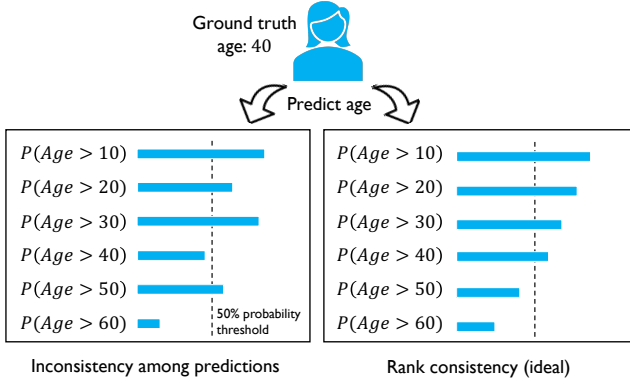*e-mail:* `sraschka@wisc.edu` (Sebastian Raschka)

Fig. 1. Schematic illustration of inconsistencies that can occur among individual classifiers in the general reduction framework from ordinal regression to binary classification: a rank-inconsistent model (left) versus a rank-consistent model where the probabilities decrease consistently (right)

Note that this work focuses on age estimation to study the proposed method's efficacy for ordinal regression. However, the proposed technique can be used for other ordinal regression problems, such as crowd-counting, depth estimation, biological cell counting, customer satisfaction, and others.

2. Related work

2.1. Ordinal regression and ranking

Several multivariate extensions of generalized linear models have been developed for ordinal regression in the past, including the popular proportional odds and proportional hazards models (McCullagh, 1980). Moreover, the machine learning field developed ordinal regression models based on extensions of well-studied classification algorithms, by reformulating the problem to utilize multiple binary classification tasks (Baccianella et al., 2009). Early work in this regard includes the use of perceptrons (Crammer and Singer, 2002; Shen and Joshi, 2005) and support vector machines (Herbrich et al., 1999; Shashua and Levin, 2003; Rajaram et al., 2003; Chu and Keerthi, 2005). Li and Lin (2007) proposed a general reduction framework that unified the view of a number of these existing algorithms.

2.2. Ordinal regression CNN

While earlier works on using CNNs for ordinal targets have employed conventional classification approaches (Levi and Hassner, 2015; Rothe et al., 2015), the general reduction framework from ordinal regression to binary classification by Li and Lin (2007) was recently adopted by Niu et al. (2016) as *Ordinal Regression CNN* (OR-CNN). In the OR-CNN approach, an ordinal regression problem with K ranks is transformed into $K - 1$ binary classification problems, with the k-th task predicting whether the age label of a face image exceeds rank r_k, $k = 1, ..., K - 1$. All $K - 1$ tasks share the same intermediate layers but are assigned distinct weight parameters in the output layer.

While the OR-CNN was able to achieve state-of-the-art performance on benchmark datasets, it does not guarantee consistent predictions, such that predictions for individual binary tasks may disagree. For example, in an age estimation setting, it would be contradictory if the k-th binary task predicted that the age of a person was more than 30, but a previous task predicted the person's age was less than 20. This inconsistency could be suboptimal when the $K - 1$ task predictions are combined to obtain the estimated age.

Niu et al. (2016) acknowledged the classifier inconsistency as not being ideal and also noted that ensuring the $K - 1$ binary classifiers are consistent would increase the training complexity substantially (Niu et al., 2016). The CORAL method proposed in this paper addresses both these issues with a theoretical guarantee for classifier consistency and without increasing the training complexity.

2.3. Other CNN architectures for age estimation

Chen et al. (2017) proposed a modification of the OR-CNN (Niu et al., 2016), known as Ranking-CNN, that uses an ensemble of CNNs for binary classifications and aggregates the predictions to estimate the age label of a given face image. The researchers showed that training an ensemble of CNNs improves the predictive performance over a single CNN with multiple binary outputs (Chen et al., 2017), which is consistent with the well-known fact that an ensemble model can achieve better generalization performance than each individual classifier in the ensemble (Raschka and Mirjalili, 2019).

Recent research has also shown that training a multi-task CNN that shares lower-layer parameters for various face analysis tasks (face detection, gender prediction, age estimation, etc.) can improve the overall performance across different tasks compared to a single-task CNN (Ranjan et al., 2017).

Another approach for utilizing binary classifiers for ordinal regression is the siamese CNN architecture proposed by Polania et al. (2019), which computes the rank from pair-wise comparisons between the input image and multiple, carefully selected anchor images.

3. Proposed method

This section describes our proposed CORAL framework that addresses the problem of classifier inconsistency in the OR-CNN by Niu et al. (2016), which is based on multiple binary classification tasks for ranking.

3.1. Preliminaries

Let $D = \{\mathbf{x}_i, y_i\}_{i=1}^N$ be the training dataset consisting of N training examples. Here, $\mathbf{x}_i \in \mathcal{X}$ denotes the i-th training example and y_i the corresponding rank, where $y_i \in \mathcal{Y} = \{r_1, r_2, ...r_K\}$ with ordered rank $r_K > r_{K-1} > ... > r_1$. The ordinal regression task is to find a ranking rule $h : \mathcal{X} \to \mathcal{Y}$ such that a loss function $L(h)$ is minimized.

Let C be a $K \times K$ *cost matrix*, where C_{y,r_k} is the cost of predicting an example $(\mathbf{x}, y)$ as rank r_k (Li and Lin, 2007). Typically, $C_{y,y} = 0$ and $C_{y,r_k} > 0$ for $y \neq r_k$. In ordinal regression, we generally prefer each row of the cost matrix to be *V-shaped*, that
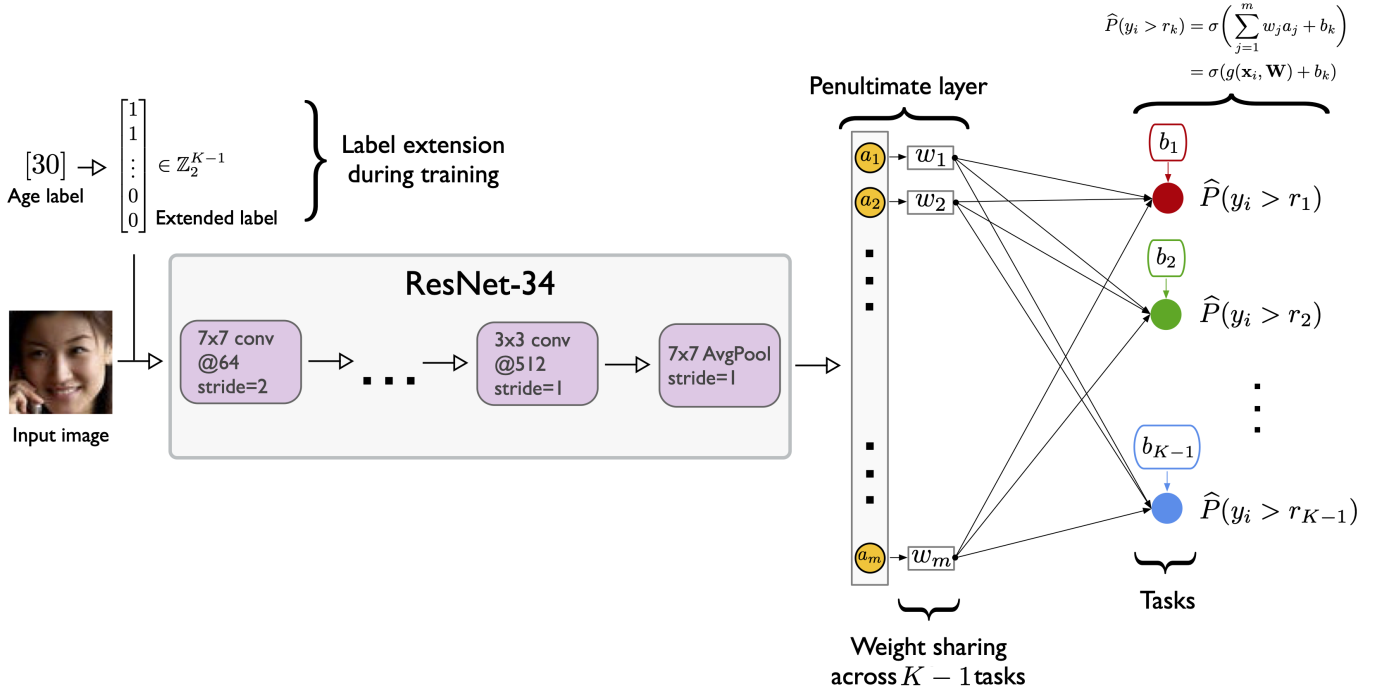
$$\widehat{P}(y_i > r_k) = \sigma\left(\sum_{j=1}^{m} w_j a_j + b_k\right)$$
$$= \sigma(g(\mathbf{x}_i, \mathbf{W}) + b_k)$$

Fig. 2. Illustration of the consistent rank logits CNN (CORAL-CNN) used for age prediction. From the estimated probability values, the binary labels are obtained via Eq. 5 and converted to the age label via Eq. 1.

is, $C_{y,r_{k-1}} \geq C_{y,r_k}$ if $r_k \leq y$ and $C_{y,r_k} \leq C_{y,r_{k+1}}$ if $r_k \geq y$. The *classification* cost matrix has entries $C_{y,r_k} = \mathbb{1}\{y \neq r_k\}$ that do not consider ordering information. In ordinal regression, where the ranks are treated as numerical values, the *absolute* cost matrix is commonly defined by $C_{y,r_k} = |y - r_k|$.

Li and Lin (2007) proposed a general reduction framework for extending an ordinal regression problem into several binary classification problems. This framework requires a cost matrix that is convex in each row ($C_{y,r_{k+1}} - C_{y,r_k} \geq C_{y,r_k} - C_{y,r_{k-1}}$ for each y) to obtain a rank-monotonic threshold model. Since the cost-related weighting of each binary task is specific for each training example, this approach is considered as infeasible in practice due to its high training complexity (Niu et al., 2016).

Our proposed CORAL framework does neither require a cost matrix with convex-row conditions nor explicit weighting terms that depend on each training example to obtain a rank-monotonic threshold model and produce consistent predictions for each binary task.

3.2. Ordinal regression with a consistent rank logits model

In this section, we describe our proposed consistent rank logits (CORAL) framework for ordinal regression. Subsection 3.2.1 describes the label extension into binary tasks used for rank prediction. The loss function of the CORAL framework is described in Subsection 3.2.2. In subsection 3.2.3, we prove the theorem for rank consistency among the binary classification tasks that guarantee that the binary tasks produce consistently ranked predictions.

3.2.1. Label extension and rank prediction

Given a training dataset $D = \{\mathbf{x}_i, y_i\}_{i=1}^{N}$, a rank y_i is first extended into $K - 1$ binary labels $y_i^{(1)}, \ldots, y_i^{(K-1)}$ such that $y_i^{(k)} \in \{0, 1\}$ indicates whether y_i exceeds rank r_k, for instance, $y_i^{(k)} = \mathbb{1}\{y_i > r_k\}$. The indicator function $\mathbb{1}\{\cdot\}$ is 1 if the inner condition is true and 0 otherwise. Using the extended binary labels during model training, we train a single CNN with $K - 1$ binary classifiers in the output layer, which is illustrated in Figure 2.

Based on the binary task responses, the predicted rank label for an input $\mathbf{x}_i$ is obtained via $h(\mathbf{x}_i) = r_q$. The rank index[1] q is given by

$$q = 1 + \sum_{k=1}^{K-1} f_k(\mathbf{x}_i), \tag{1}$$

where $f_k(\mathbf{x}_i) \in \{0, 1\}$ is the prediction of the k-th binary classifier in the output layer. We require that $\{f_k\}_{k=1}^{K-1}$ reflect the ordinal information and are *rank-monotonic*, $f_1(\mathbf{x}_i) \geq f_2(\mathbf{x}_i) \geq \ldots \geq f_{K-1}(\mathbf{x}_i)$, which guarantees consistent predictions. To achieve rank-monotonicity and guarantee binary classifier consistency (Theorem 1), the $K - 1$ binary tasks share the same weight parameters[2] but have independent bias units (Figure 2).

---

[1] While the rank label r_q is application-specific and defined by the user, for example $r_q \in \{$"*bad*", "*okay*", "*good*"$\}$ or $r_q \in \{18\,years, 19\,years, \ldots 70\,years\}$, the rank index q is an integer in the range $\{1, 2, \ldots, K\}$.

[2] To provide further intuition for the weight sharing requirement, we may consider a simplified version, that is, the linear form $logit(p_i) = wx + b_i$ or

3.2.2. Loss function

Let $\mathbf{W}$ denote the weight parameters of the neural network excluding the bias units of the final layer. The penultimate layer, whose output is denoted as $g(\mathbf{x}_i, \mathbf{W})$, shares a single weight with all nodes in the final output layer; $K-1$ independent bias units are then added to $g(\mathbf{x}_i, \mathbf{W})$ such that $\{g(\mathbf{x}_i, \mathbf{W}) + b_k\}_{k=1}^{K-1}$ are the inputs to the corresponding binary classifiers in the final layer. Let

$$\sigma(z) = 1/(1 + \exp(-z)) \tag{2}$$

be the logistic sigmoid function. The predicted empirical probability for task k is defined as

$$\widehat{P}(y_i^{(k)} = 1) = \sigma(g(\mathbf{x}_i, \mathbf{W}) + b_k). \tag{3}$$

For model training, we minimize the loss function

$$
\begin{aligned}
L(\mathbf{W}, \mathbf{b}) = \\
-\sum_{i=1}^{N}\sum_{k=1}^{K-1} \lambda^{(k)} [\log(\sigma(g(\mathbf{x}_i, \mathbf{W}) + b_k)) y_i^{(k)} \\
+ \log(1 - \sigma(g(\mathbf{x}_i, \mathbf{W}) + b_k))(1 - y_i^{(k)})],
\end{aligned}
\tag{4}
$$

which is the weighted cross-entropy of $K-1$ binary classifiers. For rank prediction (Eq. 1), the binary labels are obtained via

$$f_k(\mathbf{x}_i) = \mathbb{1}\{\widehat{P}(y_i^{(k)} = 1) > 0.5\}. \tag{5}$$

In Eq. 4, $\lambda^{(k)}$ denotes the weight of the loss associated with the k-th classifier (assuming $\lambda^{(k)} > 0$). In the remainder of the paper, we refer to $\lambda^{(k)}$ as the importance parameter for task k. Some tasks may be less robust or harder to optimize, which can be considered by choosing a non-uniform task weighting scheme. For simplicity, we carried out all experiments with uniform task weighting, that is, $\forall k : \lambda^{(k)} = 1$. In the next section, we provide the theoretical guarantee for classifier consistency under uniform and non-uniform task importance weighting given that the task importance weights are positive numbers.

3.2.3. Theoretical guarantees for classifier consistency

The following theorem shows that by minimizing the loss L (Eq. 4), the learned bias units of the output layer are non-increasing such that

$$b_1 \geq b_2 \geq \ldots \geq b_{K-1}. \tag{6}$$

Consequently, the predicted confidence scores or probability estimates of the $K-1$ tasks are decreasing, for instance,

$$\widehat{P}\left(y_i^{(1)} = 1\right) \geq \widehat{P}\left(y_i^{(2)} = 1\right) \geq \ldots \geq \widehat{P}\left(y_i^{(K-1)} = 1\right) \tag{7}$$

for all i, ensuring classifier consistency. Consequently, $\{f_k\}_{k=1}^{K-1}$ (Eq. 5) are also rank-monotonic.

$p_i = \sigma(wx + b_i)$ with a single feature x. If the weight w is not shared across the $K-1$ equations, the S-shaped curves of the probability scores p_i will intersect, making the p_i's non-monotone at some given input x. Only if w is shared across the $K-1$ equations, the S-shaped curves are horizontally shifted without intersecting.

Theorem 1 (Ordered bias units). *By minimizing the loss function defined in Eq. 4, the optimal solution $(\mathbf{W}^*, \mathbf{b}^*)$ satisfies $b_1^* \geq b_2^* \geq \ldots \geq b_{K-1}^*$.*

Proof. Suppose $(\mathbf{W}, b)$ is an optimal solution and $b_k < b_{k+1}$ for some k. Claim: replacing b_k with b_{k+1}, or replacing b_{k+1} with b_k, decreases the objective value L. Let

$$
\begin{aligned}
A_1 &= \{n : y_n^{(k)} = y_n^{(k+1)} = 1\}, \\
A_2 &= \{n : y_n^{(k)} = y_n^{(k+1)} = 0\}, \\
A_3 &= \{n : y_n^{(k)} = 1, y_n^{(k+1)} = 0\}.
\end{aligned}
$$

By the ordering relationship, we have

$$A_1 \cup A_2 \cup A_3 = \{1, 2, \ldots, N\}.$$

Denote $p_n(b_k) = \sigma(g(\mathbf{x}_n, \mathbf{W}) + b_k)$ and

$$
\begin{aligned}
\delta_n &= \log(p_n(b_{k+1})) - \log(p_n(b_k)), \\
\delta_n' &= \log(1 - p_n(b_k)) - \log(1 - p_n(b_{k+1})).
\end{aligned}
$$

Since $p_n(b_k)$ is increasing in b_k, we have $\delta_n > 0$ and $\delta_n' > 0$. If we replace b_k with b_{k+1}, the loss terms related to the k-th task are updated. The change of loss L (Eq. 4) is given as

$$\Delta_1 L = \lambda^{(k)}\Big[-\sum_{n \in A_1} \delta_n + \sum_{n \in A_2} \delta_n' - \sum_{n \in A_3} \delta_n \Big].$$

Accordingly, if we replace b_{k+1} with b_k, the change of L is given as

$$\Delta_2 L = \lambda^{(k+1)}\Big[\sum_{n \in A_1} \delta_n - \sum_{n \in A_2} \delta_n' - \sum_{n \in A_3} \delta_n' \Big].$$

By adding $\frac{1}{\lambda^{(k)}}\Delta_1 L$ and $\frac{1}{\lambda^{(k+1)}}\Delta_2 L$, we have

$$\frac{1}{\lambda^{(k)}}\Delta_1 L + \frac{1}{\lambda^{(k+1)}}\Delta_2 L = -\sum_{n \in A_3}(\delta_n + \delta_n') < 0,$$

and know that either $\Delta_1 L < 0$ or $\Delta_2 L < 0$. Thus, our claim is justified. We conclude that any optimal solution $(\mathbf{W}^*, b^*)$ that minimizes L satisfies

$$b_1^* \geq b_2^* \geq \ldots \geq b_{K-1}^*.$$

$\square$

Note that the theorem for rank-monotonicity proposed by Li and Lin (2007), in contrast to Theorem 1, requires a cost matrix C with each row y_n being convex. Under this convexity condition, let $\lambda_{y_n}^{(k)} = |C_{y_n, r_k} - C_{y_n, r_{k+1}}|$ be the weight of the loss associated with the k-th task on the n-th training example, which depends on the label y_n. Li and Lin (2007) proved that by using training example-specific task weights $\lambda_{y_n}^{(k)}$, the optimal thresholds are ordered – Niu et al. (2016) noted that example-specific task weights are infeasible in practice. Moreover, this assumption requires that $\lambda_{y_n}^{(k)} \geq \lambda_{y_n}^{(k+1)}$ when $r_{k+1} < y_n$ and $\lambda_{y_n}^{(k)} \leq \lambda_{y_n}^{(k+1)}$ when $r_{k+1} > y_n$. Theorem 1 is free from this requirement and allows us to choose a fixed weight for each task that does not depend on the individual training examples, which greatly reduces the training complexity. Also, Theorem 1 allows for choosing either a simple uniform task weighting or taking dataset imbalances into account under the guarantee of non-decreasing predicted probabilities and consistent task predictions. Under Theorem 1, the only requirement for guaranteeing rank monotonicity is that the task weights are non-negative.

4. Experiments

4.1. Datasets and preprocessing

The MORPH-2 dataset (Ricanek and Tesafaye, 2006), containing 55,608 face images, was downloaded from `https://www.faceaginggroup.com/morph/` and preprocessed by locating the average eye-position in the respective dataset using facial landmark detection (Sagonas et al., 2016) and then aligning each image in the dataset to the average eye position using *EyepadAlign* function in MLxtend v0.14 (Raschka, 2018). The faces were then re-aligned such that the tip of the nose was located in the center of each image. The age labels used in this study were in the range of 16-70 years.

The CACD dataset (Chen et al., 2014) was downloaded from `http://bcsiriuschen.github.io/CARC/` and preprocessed similar to MORPH-2 such that the faces spanned the whole image with the nose tip at the center. The total number of images is 159,449 in the age range of 14-62 years.

The Asian Face Database (AFAD) by Niu et al. (2016) was obtained from `https://github.com/afad-dataset/tarball`. The AFAD database used in this study contained 165,501 faces in the range of 15-40 years. Since the faces were already centered, no further preprocessing was required.

Following the procedure described in Niu et al. (2016), each image database was randomly divided into 80% training data and 20% test data. All images were resized to 128×128×3 pixels and then randomly cropped to 120×120×3 pixels to augment the model training. During model evaluation, the 128×128×3 RGB face images were center-cropped to a model input size of 120×120×3.

We share the training and test partitions for all datasets, along with all preprocessing code used in this paper in the code repository (Section 4.4).

4.2. Neural network architectures

To evaluate the performance of CORAL for age estimation from face images, we chose the ResNet-34 architecture (He et al., 2016), which is a modern CNN architecture that achieves good performance on a variety of image classification tasks (**?**). For the remainder of this paper, we refer to the original ResNet-34 CNN with standard cross-entropy loss as CE-CNN. To implement a ResNet-34 CNN for ordinal regression using the proposed CORAL method, we replaced the last output layer with the corresponding binary tasks (Figure 2) and refer to this implementation as CORAL-CNN. Similar to CORAL-CNN, we modified the output layer of ResNet-34 to implement the ordinal regression reference approach described in (Niu et al., 2016); we refer to this architecture as OR-CNN.

4.3. Training and evaluation

For model evaluation and comparison, we computed the mean absolute error (MAE) and root mean squared error (RMSE), on the test set after the last training epoch:

$$\text{MAE} = \frac{1}{N} \sum_{i=1}^{N} |y_i - h(\mathbf{x}_i)|,$$

$$\text{RMSE} = \sqrt{\frac{1}{N} \sum_{i=1}^{N} (y_i - h(\mathbf{x}_i))^2},$$

where y_i is the ground truth rank of the i-th test example and $h(\mathbf{x}_i)$ is the predicted rank, respectively.

The model training was repeated three times with different random seeds (0, 1, and 2) for model weight initialization, while the random seeds were consistent between the different methods to allow fair comparisons. Since this study focuses on investigating rank consistency, an extensive comparison between optimization algorithms is beyond the scope of this article, so that all CNNs were trained for 200 epochs with stochastic gradient descent via adaptive moment estimation (Kingma and Ba, 2015) using exponential decay rates $\beta_0 = 0.90$ and $\beta_2 = 0.99$ (default settings) and a batch size of 256. To avoid introducing empirical bias by designing our own CNN architecture for comparing the ordinal regression approaches, we adopted a standard architecture (ResNet-34 (He et al., 2016); Section 4.2) for this comparison. Moreover, we chose a uniform task weighting for the cross-entropy of $K - 1$ binary classifiers in CORAL-CNN, for instance, we set $\forall k : \lambda^{(k)} = 1$ in Eq. 4.

The learning rate was determined by hyperparameter tuning on the validation set. For the various losses (cross-entropy, ordinal regression CNN (Niu et al., 2016), and the proposed CORAL method), we found that a learning rate of $\alpha = 5 \times 10^{-5}$ performed best across all models, which is likely due to using the same base architecture (ResNet-34). All models were trained for 200 epochs. From those 200 epochs, the best model was selected via MAE performance on the validation set. The selected model was then evaluated on the independent test set, from which the reported MAE and RMSE performance values were obtained. For all reported model performances, we reported the best test set performance within the 200 training epochs. We provide the complete training logs in the source code repository (Section 4.4).

4.4. Hardware and software

All loss functions and neural network models were implemented in PyTorch 1.5 (Paszke et al., 2019) and trained on NVIDIA GeForce RTX 2080Ti and Titan V graphics cards. The source code is available at `https://github.com/Raschka-research-group/coral-cnn`.

5. Results and discussion

We conducted a series of experiments on three independent face image datasets for age estimation (Section 4.1) to compare the proposed CORAL method (CORAL-CNN) with the ordinal regression approach proposed by Niu et al. (2016) (OR-CNN). All implementations were based on the ResNet-34 architecture, as described in Section 4.2. We include the standard ResNet-34 classification network with cross-entropy loss (CE-CNN) as a performance baseline.

Table 1. Age prediction errors on the test sets. All models are based on the ResNet-34 architecture.

| Method	Random Seed	MORPH-2		AFAD		CACD	
		MAE	RMSE	MAE	RMSE	MAE	RMSE
CE-CNN	0	3.26	4.62	3.58	5.01	5.74	8.20
	1	3.36	4.77	3.58	5.01	5.68	8.09
	2	3.39	4.84	3.62	5.06	5.53	7.92
	AVG ± SD	3.34 ± 0.07	4.74 ± 0.11	3.60 ± 0.02	5.03 ± 0.03	5.65 ± 0.11	8.07 ± 0.14
OR-CNN (Niu et al., 2016)	0	2.87	4.08	3.56	4.80	5.36	7.61
	1	2.81	3.97	3.48	4.68	5.40	7.78
	2	2.82	3.87	3.50	4.78	5.37	7.70
	AVG ± SD	2.83 ± 0.03	3.97 ± 0.11	3.51 ± 0.04	4.75 ± 0.06	5.38 ± 0.02	7.70 ± 0.09
CORAL-CNN (ours)	0	2.66	3.69	3.42	4.65	5.25	7.41
	1	2.64	3.64	3.51	4.76	5.25	7.50
	2	2.62	3.62	3.48	4.73	5.24	7.52
	AVG ± SD	**2.64 ± 0.02**	**3.65 ± 0.04**	**3.47 ± 0.05**	**4.71 ± 0.06**	**5.25 ± 0.01**	**7.48 ± 0.06**
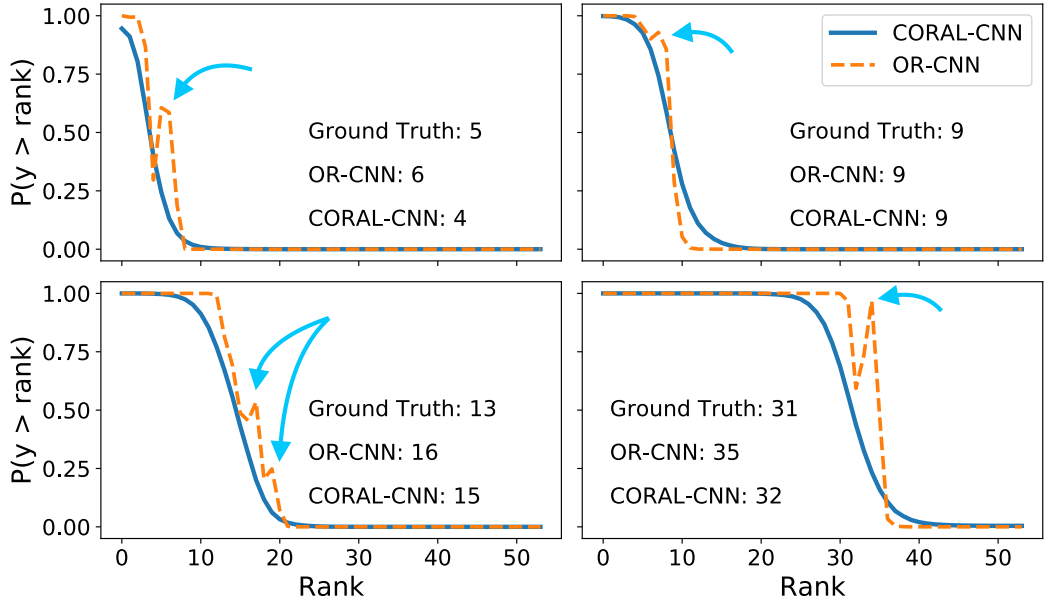
Fig. 3. Graphs of the predicted probabilities for each binary classifier task on four different examples from the MORPH-2 test dataset. In all cases, OR-CNN suffers from one or more inconsistencies (indicated by arrows) in contrast to CORAL-CNN.

Table 2. Average numbers of inconsistencies occurred on the different test datasets for CORAL-CNN and Niu et al's Ordinal CNN. The penultimate column and last column list the average numbers of inconsistencies focusing only on the correct and incorrect age predictions, respectively.

	CORAL-CNN	OR-CNN (Niu et al., 2016)	OR-CNN (Niu et al., 2016)	OR-CNN (Niu et al., 2016)
	All predictions	All predictions	Only correct predictions	Only incorrect predictions
Morph				
Seed 0	0	2.28	1.80	2.37
Seed 1	0	2.08	1.70	2.15
Seed 2	0	0.86	0.65	0.89
AFAD				
Seed 0	0	1.97	1.88	1.98
Seed 1	0	1.91	1.81	1.92
Seed 2	0	1.17	1.02	1.19
CACD				
Seed 0	0	1.24	0.98	1.26
Seed 1	0	1.68	1.29	1.71
Seed 2	0	0.80	0.63	0.81

5.1. Estimating the apparent age from face images

Across all ordinal regression datasets (Table 1) we found that both OR-CNN and CORAL-CNN outperform the standard cross-entropy classification loss (CE-CNN), which does not utilize the rank ordering information. Similarly, as summarized in Table 1, the proposed rank-consistent CORAL method shows a substantial performance improvement over OR-CNN (Niu et al., 2016), which does not guarantee classifier consistency.

Moreover, we repeated each experiment three times using different random seeds for model weight initialization and dataset shuffling to ensure that the observed performance improvement of CORAL-CNN over OR-CNN is reproducible and not coincidental. We can conclude that guaranteed classifier consistency via CORAL has a noticeable positive effect on the predictive performance of an ordinal regression CNN (a more detailed analysis of the OR-CNN's rank inconsistency is provided in Section 5.2).

For all methods (CE-CNN, CORAL-CNN, and OR-CNN), the overall performance on the different datasets appeared in the following order: MORPH-2 > AFAD > CACD (Table 1). A possible explanation is that MORPH-2 has the best overall image quality, and the photos were taken under relatively consistent lighting conditions and viewing angles. For instance, we found that AFAD includes images with very low resolutions (for example, 20x20). CACD also contains some lower-quality images. Because CACD has approximately the same size as AFAD, the overall lower performance achieved on this dataset may also be explained by the wider age range that needs to be considered (CACD: 14-62 years, AFAD: 15-40 years).

5.2. Empirical rank inconsistency analysis

By design, our proposed CORAL guarantees rank consistency (Theorem 1). In addition, we analyzed the rank inconsistency empirically for both CORAL-CNN and OR-CNN (an example of rank inconsistency is shown in Figure 3). Table 2 summarizes the average numbers of rank inconsistencies for the OR-CNN and CORAL-CNN models on each test dataset. As expected, CORAL-CNN has 0 rank inconsistencies. When comparing the average numbers of rank inconsistencies considering only those cases where OR-CNN predicted the age correctly versus incorrectly, the average number of inconsistencies is higher when OR-CNN makes wrong predictions. This observation can be seen as evidence that rank inconsistency harms predictive performance. Consequently, this finding suggests that addressing rank inconsistency via CORAL is beneficial for the predictive performance of ordinal regression CNNs.

6. Conclusions

In this paper, we developed the CORAL framework for ordinal regression via extended binary classification with theoretical guarantees for classifier consistency. Moreover, we proved classifier consistency without requiring rank- or training label-dependent weighting schemes, which permits straightforward implementations and efficient model training. CORAL can be readily implemented to extend common CNN architectures for ordinal regression tasks. The experimental results showed that the CORAL framework substantially improved the predictive performance of CNNs for age estimation on three independent age estimation datasets. Our method can be readily generalized to other ordinal regression problems and different types of neural network architectures, including multilayer perceptrons and recurrent neural networks.

7. Acknowledgments

References

Baccianella, S., Esuli, A., Sebastiani, F., 2009. Evaluation measures for ordinal regression, in: Ninth International Conference on Intelligent Systems Design andApplications, IEEE. pp. 283–287.

Cao, D., Lei, Z., Zhang, Z., Feng, J., Li, S.Z., 2012. Human age estimation using ranking SVM, in: Chinese Conference on Biometric Recognition, Springer. pp. 324–331.

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

Chen, B.C., Chen, C.S., Hsu, W.H., 2014. Cross-age reference coding for age-invariant face recognition and retrieval, in: European Conference on Computer Vision, Springer. pp. 768–783.

Chen, S., Zhang, C., Dong, M., Le, J., Rao, M., 2017. Using Ranking-CNN for age estimation, in: IEEE Conference on Computer Vision and Pattern Recognition, pp. 5183–5192.

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

Crammer, K., Singer, Y., 2002. Pranking with ranking, in: Advances in Neural Information Processing Systems, pp. 641–647.

He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition, in: IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778.

Herbrich, R., Graepel, T., Obermayer, K., 1999. Support vector learning for ordinal regression, in: IET Conference on Artificial Neural Networks, pp. 97–102.

Kingma, D.P., Ba, J., 2015. Adam: A method for stochastic optimization, in: Bengio, Y., LeCun, Y. (Eds.), 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings, pp. 1–8. URL: http://arxiv.org/abs/1412.6980.

Levi, G., Hassner, T., 2015. Age and gender classification using convolutional neural networks, in: IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 34–42.

Li, C., Liu, Q., Liu, J., Lu, H., 2012. Learning ordinal discriminative features for age estimation, in: IEEE Conference on Computer Vision and Pattern Recognition, IEEE. pp. 2570–2577.

Li, L., Lin, H.T., 2007. Ordinal regression by extended binary classification, in: Advances in Neural Information Processing Systems, pp. 865–872.

McCullagh, P., 1980. Regression models for ordinal data. Journal of the Royal Statistical Society. Series B (Methodological) , 109–142.

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

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., Chintala, S., 2019. Pytorch: An imperative style, high-performance deep learning library, in: Advances in Neural Information Processing Systems 32, Curran Associates, Inc.. pp. 8024–8035.

Polania, L., Fung, G., Wang, D., 2019. Ordinal regression using noisy pairwise comparisons for body mass index range estimation, in: IEEE Winter Conference on Applications of Computer Vision, IEEE. pp. 782–790.

Rajaram, S., Garg, A., Zhou, X.S., Huang, T.S., 2003. Classification approach towards ranking and sorting problems, in: European Conference on Machine Learning, Springer. pp. 301–312.

Ramanathan, N., Chellappa, R., Biswas, S., 2009. Age progression in human faces: A survey. Journal of Visual Languages and Computing 15, 3349–3361.

Ranjan, R., Sankaranarayanan, S., Castillo, C.D., Chellappa, R., 2017. An all-in-one convolutional neural network for face analysis, in: IEEE Conference on Automatic Face & Gesture Recognition, pp. 17–24.

Raschka, S., 2018. MLxtend: Providing machine learning and data science utilities and extensions to Python's scientific computing stack. The Journal of Open Source Software 3, 1–2.

Raschka, S., Mirjalili, V., 2019. Python Machine Learning, 3rd Ed. Packt Publishing, Birmingham, UK.

Ricanek, K., Tesafaye, T., 2006. Morph: A longitudinal image database of normal adult age-progression, in: IEEE Conference on Automatic Face and Gesture Recognition, pp. 341–345.

Rothe, R., Timofte, R., Van Gool, L., 2015. DEX: Deep expectation of apparent age from a single image, in: IEEE Conference on Computer Vision Workshops, pp. 10–15.

Sagonas, C., Antonakos, E., Tzimiropoulos, G., Zafeiriou, S., Pantic, M., 2016. 300 faces in-the-wild challenge: database and results. Image and Vision Computing 47, 3–18.

Shashua, A., Levin, A., 2003. Ranking with large margin principle: Two approaches, in: Advances in Neural Information Processing Systems, pp. 961–968.

Shen, L., Joshi, A.K., 2005. Ranking and reranking with perceptron. Machine Learning 60, 73–96.

Yang, P., Zhong, L., Metaxas, D., 2010. Ranking model for facial age estimation, in: 2010 International Conference on Pattern Recognition, IEEE. pp. 3404–3407.

8. Supplementary Material

8.1. Generalization Bounds

Based on well-known generalization bounds for binary classification, we can derive new generalization bounds for our ordinal regression approach that apply to a wide range of practical scenarios as we only require $C_{y,r_k} = 0$ if $r_k = y$ and $C_{y,r_k} > 0$ if $r_k \neq y$. Moreover, Theorem 2 shows that if each binary classification task in our model generalizes well in terms of the standard 0/1-loss, the final rank prediction via h (Eq. 1) also generalizes well.

Theorem 2 (reduction of generalization error). *Suppose C is the cost matrix of the original ordinal label prediction problem, with $C_{y,y} = 0$ and $C_{y,r_k} > 0$ for $k \neq y$. P is the underlying distribution of $(\mathbf{x}, y)$, for instance, $(\mathbf{x}, y) \sim P$. If the binary classification rules $\{f_k\}_{k=1}^{K-1}$ obtained by optimizing Eq. 4 are rank-monotonic, then*

$$\underset{(x,y)\sim P}{E} C_{y,h(\mathbf{x})} \leq \sum_{k=1}^{K-1} |C_{y,r_k} - C_{y,r_{k+1}}| \underset{(x,y)\sim P}{E} \mathbb{1}\{f_k(\mathbf{x}) \neq y^{(k)}\}. \quad (8)$$

Proof. For any $\mathbf{x} \in X$, we have

$$f_1(\mathbf{x}) \geq f_2(\mathbf{x}) \geq \ldots \geq f_{K-1}(\mathbf{x}).$$

If $h(\mathbf{x}) = y$, then $C_{y,h(\mathbf{x})} = 0$.
If $h(\mathbf{x}) = r_q < y = r_s$, then $q < s$. We have

$$f_1(\mathbf{x}) = f_2(\mathbf{x}) = \ldots = f_{q-1}(x) = 1$$

and

$$f_q(\mathbf{x}) = f_{q+1}(\mathbf{x}) = \ldots = f_{K-1}(\mathbf{x}) = 0.$$

Also,

$$y^{(1)} = y^{(2)} = \ldots = y^{(s-1)} = 1$$

and

$$y^{(s)} = y^{(s+1)} = \ldots = y^{(K-1)} = 0.$$

Thus, $\mathbb{1}\{f_k(\mathbf{x}) \neq y^{(k)}\} = 1$ if and only if $q \leq k \leq s - 1$. Since $C_{y,y} = 0$,

$$C_{y,h(\mathbf{x})} = \sum_{k=q}^{s-1} (C_{y,r_k} - C_{y,r_{k+1}}) \cdot \mathbb{1}\{f_k(\mathbf{x}) \neq y^{(k)}\}$$

$$\leq \sum_{k=q}^{s-1} |C_{y,r_k} - C_{y,r_{k+1}}| \cdot \mathbb{1}\{f_k(\mathbf{x}) \neq y^{(k)}\}$$

$$\leq \sum_{k=1}^{K-1} |C_{y,r_k} - C_{y,r_{k+1}}| \cdot \mathbb{1}\{f_k(\mathbf{x}) \neq y^{(k)}\}.$$

Similarly, if $h(x) = r_q > y = r_s$, then $q > s$ and

$$C_{y,h(\mathbf{x})} = \sum_{k=s}^{q-1} (C_{y,r_{k+1}} - C_{y,r_k}) \cdot \mathbb{1}\{f_k(\mathbf{x}) \neq y^{(k)}\}$$

$$\leq \sum_{k=1}^{K-1} |C_{y,r_{k+1}} - C_{y,r_k}| \cdot \mathbb{1}\{f_k(\mathbf{x}) \neq y^{(k)}\}.$$

In any case, we have

$$C_{y,h(\mathbf{x})} \leq \sum_{k=1}^{K-1} |C_{y,r_k} - C_{y,r_{k+1}}| \cdot \mathbb{1}\{f_k(\mathbf{x}) = y^{(k)}\}.$$

By taking the expectation on both sides with $(\mathbf{x}, y) \sim P$, we arrive at Eq. (8). $\square$

In Li and Lin (2007), by assuming the cost matrix to have V-shaped rows, the researchers define generalization bounds by constructing a discrete distribution on $\{1, 2, \ldots, K - 1\}$ conditional on each y, given that the binary classifications are rank-monotonic or every row of C is convex. However, the only case they provided for the existence of rank-monotonic binary classifiers was the ordered threshold model, which requires a cost matrix with convex rows and example-specific task weights. In other words, when the cost matrix is only V-shaped but does not meet the convex row condition, for instance, $C_{y,r_k} - C_{y,r_{k-1}} > C_{y,r_{k+1}} - C_{y,r_k} > 0$ for some $r_k > y$, the method proposed in Li and Lin (2007) did not provide a practical way to bound the generalization error. Consequently, our result does not rely on cost matrices with V-shaped or convex rows and can be applied to a broader variety of real-world use cases.