Rank-consistent Ordinal Regression for Neural Networks

Wenzhi Cao
Department of Statistics
University of Wisconsin-Madison

Vahid Mirjalili
Department of Computer Science & Engineering
Michigan State University

Sebastian Raschka[1]
Department of Statistics
University of Wisconsin-Madison

***Abstract*—Extraordinary progress has been made towards developing neural network architectures for classification tasks. However, commonly used loss functions such as the multi-category cross entropy loss are inadequate for ranking and ordinal regression problems. Hence, approaches that utilize neural networks for ordinal regression tasks transform ordinal target variables into a series of binary classification tasks but suffer from inconsistencies among the different binary classifiers. Thus, we propose a new framework (Consistent Rank Logits, CORAL) with theoretical guarantees for rank-monotonicity and consistent confidence scores. Through parameter sharing, our framework also benefits from lower training complexity and can easily be implemented to extend conventional convolutional neural network classifiers for ordinal regression tasks. Furthermore, the empirical evaluation of our method on a range of face image datasets for age prediction shows a substantial improvement compared to the current state-of-the-art ordinal regression method.**

***Index Terms*—Deep Learning, Ordinal Regression, Convolutional Neural Networks, Age Prediction, Machine Learning, Biometrics.**

I. Introduction

Ordinal regression (sometimes also referred to as *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 $h : \mathcal{X} \to \mathcal{Y}$, where $\mathcal{Y} = \{r_1 \prec ... \prec r_K\}$. In contrast to classification, the ranks include ordering information. In comparison with metric regression, which assumes that $\mathcal{Y}$ is a continuous random variable, ordinal regression regards $\mathcal{Y}$ as a finite sequence where the metric distance between ranks is not defined.

Along with age estimation [1], popular applications for ordinal regression include predicting the progression of various diseases, such as Alzheimer's [2], Crohn's [3], artery [4], and kidney disease [5]. Also, ordinal regression models are common choices for text message advertising [6] and various recommender systems [7].

While the field of machine learning developed many powerful algorithms for predictive modeling, most algorithms were designed for classification tasks. In 2007, Li and Lin proposed a general framework for ordinal regression via extended binary classification [8], which has become the standard choice for extending machine learning algorithms for ordinal regression tasks. However, implementations of this approach commonly suffer from classifier inconsistencies among the binary rankings [1], which we address in this paper with a new method and theorem for guaranteed classifier consistency that can easily be implemented in various machine learning algorithms. Furthermore, we present an empirical study of our approach on challenging real-world datasets for predicting the age of individuals from face images using our method with convolutional neural networks (CNN).

Aging can be regarded as a non-stationary process, since age progression in early childhood is primarily associated with changes in the shape of the face whereas aging during adulthood is largely defined by changes in skin texture. Thus, many traditional approaches for age estimation employed ordinal regression [9]–[12]. In addition to traditional machine learning algorithms that require manual feature extraction, CNNs were proposed that conduct both feature learning and ordinal regression to estimate the apparent age [1]. However, existing CNNs for ordinal regression approaches still suffer from classifier inconsistencies, which we aim to address in this work.

The main contributions of our paper are as follows:

1) the Consistent Rank Logits (CORAL) framework for ordinal regression with theoretical guarantees for classifier consistency and well-defined generalization bounds with and without dataset- and task-specific importance weighting;
2) CNN architectures with CORAL formulation for ordinal regression tasks that come with the added side benefit of reducing the number of parameters to be trained compared to CNNs for classification;
3) experiments showing a substantial improvement of the CORAL method with guaranteed classifier consistency over the state-of-the-art CNN for ordinal regression applied to age estimation from face images.

The remainder of this paper is organized as follows: Section II provides a concise overview of the related work. We explain the theory behind the CORAL framework in Section III, followed by the CNN architecture implementation. Section IV describes the experimental setup of the empirical validation. In Section V, we provide a description and summary of the experimental results. Finally, Section VI concludes the paper

[1]Corresponding Author
Email: sraschka@wisc.edu

with a summary and outlook.

II. Related Work

A. Ordinal Regression and Ranking

Several multivariate extensions of generalized linear models have been developed in the past for ordinal regression, including the popular proportional odds and the proportional hazards models [13]. Moreover, ordinal regression has become a popular topic of study in the field of machine learning to extend classification algorithms by reformulating the problem to utilize multiple binary classification tasks. Early work in this regard includes the use of perceptrons [14], [15] and support vector machines [16]–[19]. A general reduction framework that unified the view of a number of these existing algorithms for ordinal regression was later proposed by Li and Lin [8].

While earlier works on using CNNs for ordinal targets have employed conventional classification approaches [20], [21], the general reduction framework from ordinal regression to binary classification by Li and Lin [8] was recently adopted by Niu et al. [1]. In [1], an ordinal regression problem with K ranks was transformed into $K-1$ binary classification problems, with the kth 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. However, this approach does not guarantee that the predictions are consistent such that predictions for individual binary tasks may disagree. For example, in an age estimation setting, it would be contradictory if the kth binary task predicted that the age of a person was larger than 30, but a previous task predicted the person's age was smaller than 20, which is suboptimal when the $K-1$ task predictions are combined to obtain the estimated age.

While the ordinal regression CNN yielded state-of-the-art results on several age estimation datasets, the authors acknowledged the classifier inconsistency as not being ideal but also noted that ensuring that the $K-1$ binary classifiers are consistent would increase the training complexity substantially [1]. Our proposed method addresses both of these issues with a theoretical guarantee for classifier consistency without increasing the training complexity.

B. CNN Architectures for Age Estimation

Due to its broad utility in social networking, video surveillance, and biometric verification, age estimation from human faces is an active area of research. Likely owed to the rapid advancements in computer vision based on deep learning, most state-of-the-art age estimation methods are now utilizing CNN architectures [1], [21]–[24].

Related to the idea of training binary classifiers separately and combining the independent predictions for ranking [25], a modification of the ordinal regression CNN [1] was recently proposed for age estimation, called Ranking-CNN, that trains an ensemble of CNNs for binary classifications and aggregates the predictions to predict the age label of a given face image [24]. The researchers showed that training a series of CNNs improves the predictive performance over a single CNN with multiple binary outputs. However, ensembles of CNNs come with a substantial increase in training complexity and do not guarantee classifier consistency, which means that the individual binary classifiers used for ranking can produce contradictory results. Another approach for utilizing binary classifiers for ordinal regression is the siamese CNN architecture by Polania et al. [26]. Since this siamese CNN has only a single output neuron, comparisons between the input image and multiple, carefully selected anchor images are required to compute the rank.

Recent research has also shown that training a multi-task CNN for various face analysis tasks, including face detection, gender prediction, age estimation, etc., can improve the overall performance across different tasks compared to a single-task CNN [23] by sharing lower-layer parameters. In [22], a cascaded convolutional neural network was designed to classify face images into age groups followed by regression modules for more accurate age estimation. In both studies, the authors used metric regression for the age estimation subtasks. While our paper focuses on the comparison of different ordinal regression approaches, we hypothesize that such all-in-one and cascaded CNNs can be further improved by our method, since, as shown in [1], ordinal regression CNNs outperform metric regression CNNs in age estimation tasks.

III. Proposed Method

This section describes the proposed CORAL framework that addresses the problem of classifier inconsistency in ordinal regression CNNs based on multiple binary classification tasks for ranking.

A. Preliminaries

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

Let $\mathcal{C}$ be a $K \times K$ *cost matrix* [8], where $\mathcal{C}_{y,r_k}$ is the cost of predicting an example $(\mathbf{x}, y)$ as rank r_k. Typically, $\mathcal{C}_{y,y} = 0$ and $\mathcal{C}_{y,r_k} > 0$ for $y \neq r_k$. In ordinal regression, we generally prefer each row of the cost matrix to be *V-shaped*. That is $\mathcal{C}_{y,r_{k-1}} \geq \mathcal{C}_{y,r_k}$ if $r_k \leq y$ and $\mathcal{C}_{y,r_k} \leq \mathcal{C}_{y,r_{k+1}}$ if $r_k \geq y$. The *classification cost matrix* has entries $\mathcal{C}_{y,r_k} = \mathbb{1}\{y \neq r_k\}$, which does not consider ordering information. In ordinal regression, where the ranks are treated as numerical values, the *absolute cost matrix* is commonly defined by $\mathcal{C}_{y,r_k} = |y - r_k|$.

In [8], the researchers proposed a general reduction framework for extending an ordinal regression problem into several binary classification problems. This framework requires the use of a cost matrix that is convex in each row ($\mathcal{C}_{y,r_{k+1}} - \mathcal{C}_{y,r_k} \geq \mathcal{C}_{y,r_k} - \mathcal{C}_{y,r_{k-1}}$ 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 was described as infeasible in practice

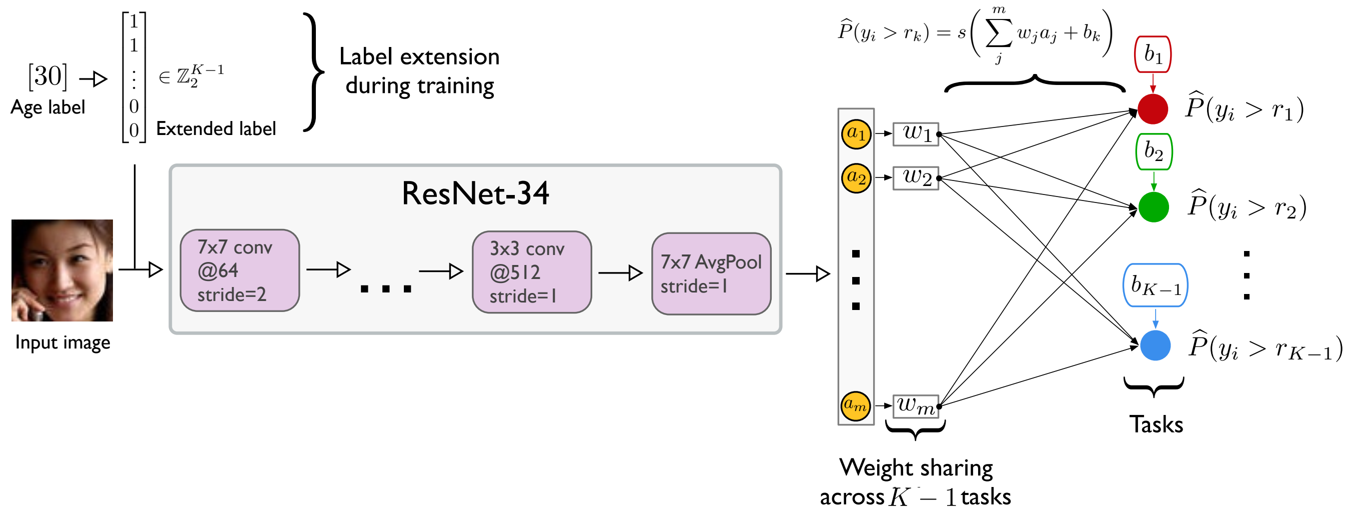

Figure 1. Illustration of the Consistent Rank Logits CNN (CORAL-CNN) used for age prediction. From the estimated probability values, the binary labels are obtained via Eq. (5) and converted to the age label via Eq. (1).

due to its high training complexity [1]. 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 to produce consistent predictions for each binary task. Moreover, CORAL allows for an optional task importance weighting. The optional assignment of non-uniform task importance weights, for example, may be used to address label imbalances (Section III-E), which makes the CORAL framework more applicable to real-world datasets.

B. Ordinal Regression with a Consistent Rank Logits Model

We propose the Consistent Rank Logits (CORAL) model for multi-label CNNs with ordinal responses. Within this framework, the binary tasks produce consistently ranked predictions. The following two subsections describe the label extension into binary tasks performed during training as well as the loss function for parameterizing the neural network to predict ordinal labels.

a) Label Extension and Rank Prediction.: Given the training dataset $D = \{\mathbf{x}_i, y_i\}_{i=1}^N$, we first extend a rank label y_i into $K-1$ binary labels $y_i^{(1)}, \dots, y_i^{(K-1)}$ such that $y_i^{(k)} \in \{0,1\}$ indicates whether y_i exceeds rank r_k, i.e., $y_i^{(k)} = \mathbb{1}\{y_i > r_k\}$. The indicator function $\mathbb{1}\{\cdot\}$ is 1 if the inner condition is true, and 0 otherwise. Providing the extended binary labels as model inputs, we train a single CNN with $K-1$ binary classifiers in the output layer. Here, the $K-1$ binary tasks share the same weight parameter but have independent bias units, which solves the inconsistency problem among the predicted binary responses and reduces the model complexity (Figure 1).

Based on the binary task responses, the predicted rank for an input $\mathbf{x}_i$ is then obtained via

$$h(\mathbf{x}_i) = r_q, \tag{1}$$

$$q = 1 + \sum_{k=1}^{K-1} f_k(\mathbf{x}_i), \tag{2}$$

where $f_k(\mathbf{x}_i) \in \{0,1\}$ is the prediction of the kth binary classifier in the output layer. We require that $\{f_k\}_{k=1}^{K-1}$ reflect the ordinal information and are *rank-monotonic*,

$$f_1(\mathbf{x}_i) \geq f_2(\mathbf{x}_i) \geq \dots, f_{K-1}(\mathbf{x}_i),$$

which guarantees that the predictions are consistent.

b) 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 $s(z) = 1/(1+\exp(-z))$ be the logistic sigmoid function. The predicted empirical probability for task k is defined as

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

For model training, we minimize the loss function

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

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 kth 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 taken into consideration by choosing a non-uniform task weighting scheme. The choice of task importance parameters is covered in more detail in Section III-E. Next, we provide a theoretical guarantee for classifier consistency under uniform and non-uniform task importance weighting given that the task importance weights are positive numbers.

C. Theoretical Guarantees for Classifier Consistency

In the following theorem, we show that by minimizing the loss L (Eq. 4), the learned bias units of the output layer are non-increasing such that $b_1 \geq b_2 \geq \ldots \geq b_{K-1}$. Consequently, the predicted confidence scores or probability estimates of the $K-1$ tasks are decreasing, i.e.,

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

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

Theorem 1 (ordered biases)**.** *By minimizing 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: by either replacing b_k with b_{k+1} or replacing b_{k+1} with b_k, we can decrease 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 $p_n(b_k) = s(g(\mathbf{x}_n, \mathbf{W}) + b_k)$ 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 kth 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, and we conclude that any optimal solution $(\mathbf{W}^*, b^*)$ that minimizes L satisfies

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

□

Note that the theorem for rank-monotonicity in [8], in contrast to Theorem 1, requires the use of a cost matrix $\mathcal{C}$ with each row y_n being convex. Under this convexity condition, let $\lambda_{y_n}^{(k)} = |\mathcal{C}_{y_n, r_k} - \mathcal{C}_{y_n, r_{k+1}}|$ be the weight of the loss associated with the kth task on the nth example, which depends on the label y_n. In [8], the researchers proved that by using example-specific task weights $\lambda_{y_n}^{(k)}$, the optimal thresholds are ordered. 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. Moreover, Theorem 1 allows for choosing either a simple uniform task weighting or taking dataset imbalances into account (Section III-E) while still guaranteeing that the predicted probabilities are non-decreasing and the task predictions are consistent.

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

$$\underset{(x,y) \sim P}{E} \mathcal{C}_{y, h(\mathbf{x})} \leq \sum_{k=1}^{K-1} \left|\mathcal{C}_{y, r_k} - \mathcal{C}_{y, r_{k+1}}\right| \underset{(x,y) \sim P}{E} \mathbb{1}\{f_k(\mathbf{x}) \neq y^{(k)}\}. \tag{6}$$

Proof. For any $\mathbf{x} \in \mathcal{X}$, 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 $\mathcal{C}_{y, h(\mathbf{x})} = 0$.
If $h(\mathbf{x}) = r_q \prec y = r_s$, 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 $\mathcal{C}_{y,y} = 0$,

$$\begin{aligned}
\mathcal{C}_{y,h(\mathbf{x})} &= \sum_{k=q}^{s-1} (\mathcal{C}_{y,r_k} - \mathcal{C}_{y,r_{k+1}}) \cdot \mathbb{1}\{f_k(\mathbf{x}) \neq y^{(k)}\} \\
&\leq \sum_{k=q}^{s-1} |\mathcal{C}_{y,r_k} - \mathcal{C}_{y,r_{k+1}}| \cdot \mathbb{1}\{f_k(\mathbf{x}) \neq y^{(k)}\} \\
&\leq \sum_{k=1}^{K-1} |\mathcal{C}_{y,r_k} - \mathcal{C}_{y,r_{k+1}}| \cdot \mathbb{1}\{f_k(\mathbf{x}) \neq y^{(k)}\}.
\end{aligned}$$

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

$$\begin{aligned}
\mathcal{C}_{y,h(\mathbf{x})} &= \sum_{k=s}^{q-1} (\mathcal{C}_{y,r_{k+1}} - \mathcal{C}_{y,r_k}) \cdot \mathbb{1}\{f_k(\mathbf{x}) \neq y^{(k)}\} \\
&\leq \sum_{k=1}^{K-1} |\mathcal{C}_{y,r_{k+1}} - \mathcal{C}_{y,r_k}| \cdot \mathbb{1}\{f_k(\mathbf{x}) \neq y^{(k)}\}.
\end{aligned}$$

In any case, we have

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

By taking the expectation on both sides with $(\mathbf{x}, y) \sim P$, we arrive at Eq. (6). □

In [8], 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 $\mathcal{C}$ 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, i.e., $\mathcal{C}_{y,r_k} - \mathcal{C}_{y,r_{k-1}} > \mathcal{C}_{y,r_{k+1}} - \mathcal{C}_{y,r_k} > 0$ for some $r_k > y$, the method proposed in [8] 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.

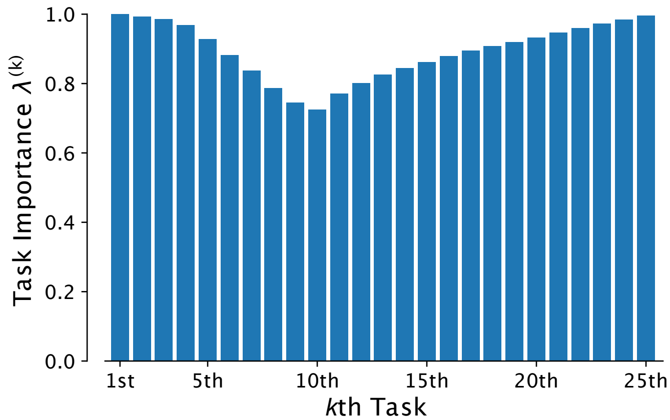

Figure 2. Example of the task importance weighting according to Eq. (7) shown for the AFAD dataset (Section IV-A).

E. Task Importance Weighting

According to Theorem 1, minimizing the loss of the CORAL model guarantees that the bias units are non-increasing and thus the binary classifiers are consistent as long as the task importance parameters are positive: $\forall k \in \{1, ..., K-1\} : \lambda^{(k)} > 0$.

In many real-world applications, features between certain adjacent ranks may have more subtle distinctions. For example, facial aging is commonly regarded as a non-stationary process [27] such that face feature transformations could be more detectable during certain age intervals. Moreover, the relative predictive performance of the binary tasks may also be affected by the degree of binary data imbalance for a given task that occurs as a side-effect of extending a rank label into $K-1$ binary labels. Hence, we hypothesize that choosing non-uniform task weighting schemes improves the predictive performance of the overall model.

We first experimented with a weighting scheme proposed in [1] that aims to address the class imbalance in the face image datasets. However, compared to using a uniform scheme ($\forall k \in \{1, ..., K-1\} : \lambda^{(k)} = 1$), we found that it had a negative effect on the predictive performance of all models evaluated in this study.

Hence, we propose a weighting scheme that takes the rank distribution of the training examples into account but also considers the label imbalance for each classification task after extending the original ranks into binary labels. Note that CORAL, according to Theorem 1, guarantees that the predicted probabilities are non-decreasing and the task predictions are consistent as long as the task importance weights are non-negative as described in Section III-C.

Specifically, our task weighting scheme is defined as follows. Let $S_k = \sum_{i=1}^{N} \mathbb{1}\{y_i^{(k)} = 1\}$ be the number of examples whose ranks exceed r_k. By the rank ordering we have $S_1 \geq S_2 \geq \ldots \geq S_{K-1}$. Let $M_k = \max(S_k, N - S_k)$ be the number of majority binary label for each task. We define the importance of the kth task as the scaled $\sqrt{M_k}$:

$$\lambda^{(k)} = \frac{\sqrt{M_k}}{\max_{1 \leq i \leq K-1}(\sqrt{M_i})}. \tag{7}$$

Under this weighting scheme, the general class imbalance of a dataset is taken into account. Moreover, in our examples, classification tasks corresponding to the edges of the distribution of unique rank labels receive a higher weight than the classification tasks that see more balanced rank label vectors during training, which may help improve the predictive performance of the model. The lowest weight may not always be assigned to the center-rank: if $S_{K-1} > 0.5$, the last task has the lowest weight, and if $S_1 < 0.5$, the first task has the lowest weight. An example of an importance weight distribution is shown in Figure 2.

It shall be noted that the task importance weighting is only used for model parameter optimization; when computing the predicted rank by adding the binary results (Eq. 1), each task has the same influence on the final rank prediction. Since $\lambda^{(k)} > \frac{\sqrt{M_k}}{\sqrt{N}} > \frac{M_k}{N} \geq 0.5$, it prevents tasks from having negligible weights as in [1] when a dataset contains only a small number of examples for certain ranks. We provide an empirical comparison between a uniform task weighting and task weighting according to Eq. (7) in Section V-C.

IV. Experiments

A. Datasets and Preprocessing

The MORPH-2 dataset [28] (55,608 face images; https://www.faceaginggroup.com/morph/) was preprocessed by locating the average eye-position in the respective dataset using facial landmark detection [29] via MLxtend v0.14 [30] and then aligning each image in the dataset to the average eye position. 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 ranged between 16-70 years. The CACD database [31] was preprocessed similar to MORPH-2 such that the faces spanned the whole image with the nose tip being in the center. The total number of images is 159,449 in the age range 14-62 years (http://bcsiriuschen.github.io/CARC/). For both the Asian Face Database [1] (AFAD; 165,501 faces, age labels 15-40 years; https://github.com/afad-dataset/tarball) and the UTKFace database [32] (16,434 images, age labels 21-60 years; https://susanqq.github.io/UTKFace/) centered images were already provided.

Each image database was randomly divided into 80% training data and 20% test data. All images were resized to $128 \times 128 \times 3$ pixels and then randomly cropped to $120 \times 120 \times 3$ pixels to augment the model training. During model evaluation, the $128 \times 128 \times 3$ face images were center-cropped to a model input size of $120 \times 120 \times 3$. The training and test partitions for all datasets, along with all preprocessing code used in this paper, are available at https://github.com/Raschka-research-group/coral-cnn/tree/master/datasets.

B. Convolutional Neural Network Architectures

To evaluate the performance of CORAL for age estimation from face images, we chose the ResNet-34 architecture [33], 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 cross entropy loss as CE-CNN. To implement CORAL, we replaced the last output layer with the corresponding binary tasks (Figure 1) and refer to this CNN as CORAL-CNN. Similar to CORAL-CNN, we replaced the cross-entropy layer of the ResNet-34 with the binary tasks for ordinal regression described in [1] and refer to this architecture as OR-CNN.

Note that next to guaranteed rank-consistency, another advantage of CORAL-CNN method over OR-CNN is the reduction of parameters in the output layer. Suppose that there are m output nodes in the last fully connected layer, which is connected to the output layer. In [1], the output layer consists of $(K-1) \times 2$ output nodes: there are $K-1$ binary classification tasks with 2 neurons each. Thus, the number of parameters in the final layer is $(m+1) \times (K-1) \times 2$. The output layer of the CORAL-network, however, uses one neuron for each task and the weights are shared among all $(K-1)$ tasks. Hence, the number of parameters is $m+K-1$, such that the CORAL-CNN has a substantially lower training complexity as the ORDINAL-CNN. For example, CORAL-CNN has 219,190 fewer parameters than OR-CNN in the case of ResNet-34 (Figure 1) and MORPH-2, where $K = 55$ and $m = 2048$.

C. Training and Evaluation

For model evaluation and comparison, we computed the mean absolute error (MAE) and root mean squared error (RMSE), which are standard metrics used for age prediction, 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 ith test example and $h(\mathbf{x}_i)$ is the predicted rank, respectively.

In addition, we computed the Cumulative Score (CS) as the proportion of images for which the absolute differences between the predicted rank labels and the ground truth are below a threshold T:

$$\text{CS}(T) = \frac{1}{N} \sum_{i=1}^{N} 1\{|y_i - h(\mathbf{x}_i)| \leq T\}.$$

By varying the threshold T, CS curves were plotted to compare the predictive performances of the different age prediction models (the larger the area under the curve, the better).

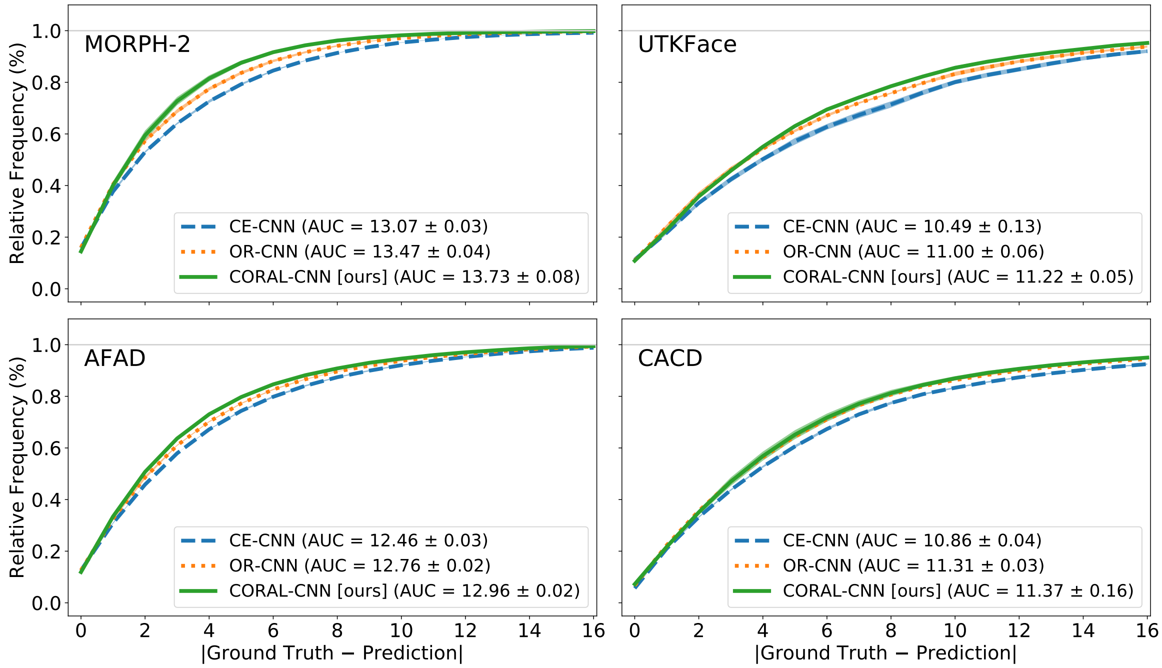

Figure 3. Comparison of age prediction models *without* task importance weighting. CS curves are shown as averages over three independent runs with standard deviation.

The model training was repeated three times with different random seeds for model weight initialization while the random seeds were consistent between the different methods to allow for fair comparisons. All CNNs were trained for 200 epochs with stochastic gradient descent via adaptive moment estimation using exponential decay rates [34] $\beta_0 = 0.90$ and $\beta_2 = 0.99$ (default settings). To avoid introducing empirical bias by designing our own CNN architecture for comparing the ordinal regression approaches, we adopted a standard architecture for this comparison, namely, ResNet-34 [33].

Exploring different learning rates for the different losses (cross-entropy, ordinal regression CNN [1], and the CORAL approach), we found that a learning rate of $\alpha = 5 \times 10^{-5}$ performed best across all models, which is likely due to the similar base architecture (ResNet-34). Also, for all models, the loss converged after 200 epochs.

Comparisons with additional neural network architectures, Inception-v3 [35] and VGG-16 [36], are included in the Supplementary Materials.

D. Hardware and Software

All loss functions and neural network models were implemented in PyTorch 1.1.0 [37] and trained on NVIDIA GeForce 1080Ti and Titan V graphics cards. The source code is available at https://github.com/Raschka-research-group/coral-cnn.

V. Results and Discussion

We conducted a series of experiments on four independent face image datasets for age estimation (Section IV-A) to compare our CORAL approach (CORAL-CNN) with the ordinal regression approach described in [1], denoted as OR-CNN. All implementations were based on the ResNet-34 architecture as described in Section IV-B, including the standard ResNet-34 with cross-entropy loss (CE-CNN) as performance baseline.

A. Estimating the Apparent Age from Face Images

Across all datasets (Table 1), we found that both OR-CNN and CORAL-CNN outperform the standard cross-entropy loss (CE-CNN) on these ordinal regression tasks as expected. Similarly, as summarized in Table 1 and Figure 3, our CORAL method shows a substantial improvement over OR-CNN [1], 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 may conclude that guaranteed classifier consistency via CORAL has a substantial, positive effect on the predictive performance of an ordinal regression CNN (a more detailed analysis regarding the rank inconsistency by Niu et al's OR-CNN is provided in Section V-B).

Furthermore, it can be observed that for all methods, the overall predictive performance on the different datasets appears in the following order: MORPH-2 > AFAD > CACD > UTKFace (Table 1 and Figure 3). A possible explanation is that MORPH-2 has the best overall image quality and the photos were taking under relatively consistent lighting conditions and viewing angles. For instance, we found that AFAD includes some images of particularly low resolution (e.g., 20x20). While UTKFace and CACD also contain some

Table 1
Age prediction errors on the test sets *without* task importance weighting. All models are based on the ResNet-34 architecture.

Method	Random Seed	MORPH-2		AFAD		UTKFace		CACD	
		MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE
CE-CNN	0	3.40	4.88	3.98	5.55	6.57	9.16	6.18	8.86
	1	3.39	4.87	4.00	5.57	6.24	8.69	6.10	8.79
	2	3.37	4.87	3.96	5.50	6.29	8.78	6.13	8.87
	AVG $\pm$ SD	3.39 $\pm$ 0.02	4.89 $\pm$ 0.01	3.98 $\pm$ 0.02	5.54 $\pm$ 0.04	6.37 $\pm$ 0.18	8.88 $\pm$ 0.25	6.14 $\pm$ 0.04	8.84 $\pm$ 0.04
OR-CNN [1]	0	2.98	4.26	3.66	5.10	5.71	8.11	5.53	7.91
	1	2.98	4.26	3.69	5.13	5.80	8.12	5.53	7.98
	2	2.96	4.20	3.68	5.14	5.71	8.11	5.49	7.89
	AVG $\pm$ SD	2.97 $\pm$ 0.01	4.24 $\pm$ 0.03	3.68 $\pm$ 0.02	5.13 $\pm$ 0.02	5.74 $\pm$ 0.05	8.08 $\pm$ 0.06	5.52 $\pm$ 0.02	7.93 $\pm$ 0.05
CORAL-CNN (ours)	0	2.68	3.75	3.49	4.82	5.46	7.61	5.56	7.80
	1	2.63	3.66	3.46	4.83	5.46	7.63	5.37	7.64
	2	2.61	3.64	3.52	4.91	5.48	7.63	5.25	7.53
	AVG $\pm$ SD	**2.64 $\pm$ 0.04**	**3.68 $\pm$ 0.06**	**3.49 $\pm$ 0.03**	**4.85 $\pm$ 0.05**	**5.47 $\pm$ 0.01**	**7.62 $\pm$ 0.01**	**5.39 $\pm$ 0.16**	**7.66 $\pm$ 0.14**

lower-quality images, a possible reason why the methods perform worse on UTKFace compared to AFAD is that UTKFace is about ten times smaller than AFAD. Because CACD has approximately the same size as AFAD, the lower performance may be explained by the wider age range that needs to be considered (14-62 in CACD compared to 15-40 in AFAD).

B. *Inconsistencies Inccurred by OR-CNN*

This section analyzes the rank inconsistency issue of Niu et al.'s method [1] in more detail. Figure 4 shows an example of an inconsistent rank prediction for OR-CNN on a single image in the MORPH-2 test dataset.

Table 2 lists the average numbers of inconsistencies that were observed for the different test datasets predictions, where an inconsistency occurs if the predictions of the binary classification tasks are not rank-monotonic (as shown in the example in Figure 4). As expected, due to the theoretical guarantees, no rank inconsistencies were observed for CORAL-CNN. The average number of rank inconsistencies inccurred by Ordinal-CNN is between 2.32 and 5.56, depending on the dataset.

When comparing the average number of rank inconsistencies that occur among the test predictions that predict the age labels correctly (Table 2, penultimate column) to the number of inconsistencies among the incorrect predictions (Table 2, last column), it can be seen that the ordinal regression method by Niu et al [1] has a smaller number of rank inconsistencies if it predicts the age label correctly. This observation suggests that the rank inconsistencies are detrimental to the predictive performance, and the better performance of CORAL-CNN compared to Ordinal-CNN (Table and Figure) may be achieved by eliminating the rank inconsistency issue.

C. *Task Importance Weighting*

While all results described in the previous section are based on experiments without task importance weighting (i.e., $\forall k : \lambda^{(k)} = 1$), we repeated all experiments using our weighting scheme proposed in Section III-E, which takes label imbalances into account. Note that according to Theorem 1, CORAL still guarantees classifier consistency under any chosen task weighting scheme as long as weights are assigned positive values. From the results provided in Table 3, we find that by using a task weighting scheme that also takes label imbalances into account (Eq. 7), we can further improve the performance of the CORAL-CNN models across all four datasets.

To test the hypothesis that the improvements in predictive performance via the task importance weighting are due to addressing the label imbalance, we conducted additional experiments using balanced datasets (Figure 5B). To balance the MORPH-2 and AFAD datasets (Figure 5A), we selected a smaller age range (18-39 years for AFAD and 18-45 years for MORPH-2) and randomly removed samples from ages larger than the age with the smallest number of examples. The task importance weight distributions for the imbalanced and balanced datasets are shown in Figure 5.

As shown in Table 4, both Niu et al.'s ordinal regression method [1] and CORAL perform equally well with and without optional task importance weighting if the datasets are balanced. Considering this observation in the context of the predictive performance improvements measured on imbalanced datasets (Table 3), we conclude that the task importance weighting is an effective measure for working with imbalanced datasets.

VI. 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. Applied to four independent age estimation datasets, the results unequivocally showed that the CORAL framework substantially improved the predictive performance of CNNs for age estimation. 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.

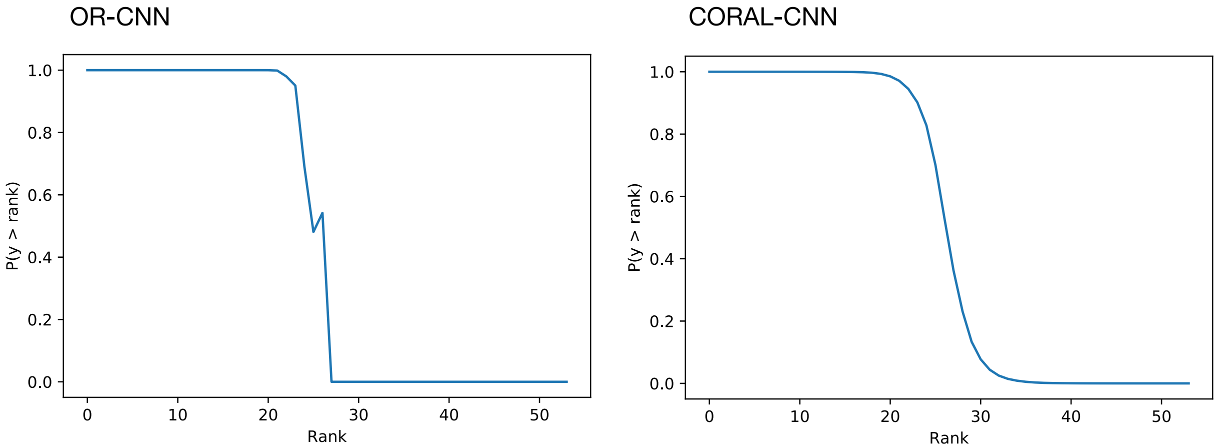

Figure 4. Plots show graphs of the predicted probabilities for each binary classifier task on one test data point in the MORPH-2 dataset by OR-CNN (left subpanel) and CORAL-CNN (right subpanel). In this example, the ordinal regression CNN has an inconsistency at rank 26. The CORAL-CNN does not suffer from inconsistencies such that the rank prediction is a cumulative distribution function.

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 focussing only on the correct and incorrect age predictions, respectively.

	CORAL-CNN All predictions	Ordinal-CNN [1] All predictions	Ordinal-CNN [1] Only correct predictions	Ordinal-CNN [1] Only incorrect predictions
Morph				
Seed 0	0	2.74	2.02	2.89
Seed 1	0	2.74	2.08	2.88
Seed 2	0	3.00	2.20	3.16
AFAD				
Seed 0	0	2.32	1.78	2.40
Seed 1	0	2.35	1.83	2.43
Seed 2	0	2.55	1.97	2.63
UTKFace				
Seed 0	0	4.79	3.64	4.92
Seed 1	0	5.73	4.05	5.95
Seed 2	0	5.07	3.84	5.21
CACD				
Seed 0	0	5.06	4.55	5.10
Seed 1	0	5.40	4.76	5.44
Seed 2	0	5.56	4.87	5.61

Table 3
Performance comparison after training with and without task importance weighting (Eq. 7). The performance values are reported as average MAE $\pm$ standard deviation from 3 independent runs each. All models are based on the ResNet-34 architecture.

Method	Importance Weight	MORPH-2	AFAD	UTKFace	CACD
OR-CNN [1]	NO	2.97 ± 0.01	3.68 ± 0.02	5.74 ± 0.05	5.52 ± 0.02
OR-CNN [1]	YES	2.91 ± 0.02	3.65 ± 0.03	5.76 ± 0.19	5.49 ± 0.02
CORAL-CNN (ours)	NO	2.64 ± 0.04	3.49 ± 0.03	5.47 ± 0.01	5.39 ± 0.16
CORAL-CNN (ours)	YES	$\mathbf{2.59 \pm 0.03}$	$\mathbf{3.48 \pm 0.03}$	$\mathbf{5.39 \pm 0.07}$	$\mathbf{5.35 \pm 0.09}$

Table 4
Age prediction errors on *balanced* MORPH-2 and AFAD test sets with and without task importance weighting. Each training run was repeated ten times, where each time, a different random seed (0-9) was chosen.

			MORPH		AFAD	
Methods	Importance Weight	Seed	MAE	RMSE	MAE	RMSE
OR-CNN [1]	No	0	2.95	4.12	3.82	5.20
	No	1	3.33	4.66	3.91	5.31
	No	2	2.98	4.17	3.83	5.19
	No	3	2.91	4.08	3.80	5.18
	No	4	3.09	4.40	3.83	5.24
	No	5	3.01	4.22	3.75	5.10
	No	6	2.87	4.08	3.78	5.13
	No	7	2.85	4.01	3.83	5.22
	No	8	2.93	4.12	3.76	5.08
	No	9	2.85	4.03	3.79	5.19
	No	AVG $\pm$ SD	2.98 $\pm$ 0.15	4.19 $\pm$ 0.20	3.81 $\pm$ 0.05	5.18 $\pm$ 0.07
OR-CNN [1]	Yes	0	2.86	4.02	3.77	5.13
	Yes	1	2.88	4.07	3.90	5.29
	Yes	2	2.94	4.14	3.77	5.08
	Yes	3	2.96	4.19	3.83	5.20
	Yes	4	3.06	4.28	3.83	5.19
	Yes	5	2.79	3.95	3.80	5.15
	Yes	6	2.81	3.96	3.90	5.27
	Yes	7	2.84	3.97	3.75	5.10
	Yes	8	3.46	4.75	3.74	5.07
	Yes	9	2.95	4.16	3.86	5.27
	Yes	AVG $\pm$ SD	2.96 $\pm$ 0.20	4.15 $\pm$ 0.24	3.82 $\pm$ 0.06	5.18 $\pm$ 0.08
CORAL-CNN (ours)	No	0	2.61	3.64	3.62	4.91
	No	1	2.58	3.57	3.60	4.98
	No	2	2.64	3.64	3.57	4.83
	No	3	2.68	3.70	3.63	4.90
	No	4	2.81	3.81	3.59	4.87
	No	5	2.58	3.57	3.59	4.86
	No	6	2.56	3.52	3.68	5.00
	No	7	2.67	3.70	3.61	4.88
	No	8	2.61	3.61	3.61	4.84
	No	9	2.64	3.67	3.56	4.83
	No	AVG $\pm$ SD	**2.64 $\pm$ 0.08**	**3.64 $\pm$ 0.09**	3.61 $\pm$ 0.04	4.89 $\pm$ 0.06
CORAL-CNN (ours)	Yes	0	2.62	3.61	3.59	4.86
	Yes	1	2.69	3.73	3.60	4.88
	Yes	2	2.62	3.58	3.54	4.80
	Yes	3	2.63	3.61	3.68	4.99
	Yes	4	2.62	3.62	3.60	4.87
	Yes	5	3.11	4.26	3.57	4.86
	Yes	6	2.60	3.61	3.60	4.90
	Yes	7	2.62	3.60	3.56	4.84
	Yes	8	2.66	3.68	3.63	4.92
	Yes	9	2.63	3.63	3.57	4.83
	Yes	AVG $\pm$ SD	2.68 $\pm$ 0.15	3.69 $\pm$ 0.20	**3.59 $\pm$ 0.04**	**4.88 $\pm$ 0.05**

VII. Acknowledgements

Support for this research was provided by the Office of the Vice Chancellor for Research and Graduate Education at the University of Wisconsin-Madison with funding from the Wisconsin Alumni Research Foundation. Also, we thank the NVIDIA Corporation for a generous donation via an NVIDIA GPU grant to support this study.

References

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

[2] O. M. Doyle, E. Westman, A. F. Marquand, P. Mecocci, B. Vellas, M. Tsolaki, I. Kłoszewska, H. Soininen, S. Lovestone, S. C. Williams *et al.*, "Predicting progression of Alzheimer's disease using ordinal regression," *PloS one*, vol. 9, no. 8, p. e105542, 2014.

[3] R. K. Weersma, P. C. Stokkers, A. A. van Bodegraven, R. A. van Hogezand, H. W. Verspaget, D. J. de Jong, C. Van Der Woude, B. Oldenburg, R. Linskens, and E. Festen, "Molecular prediction of disease risk and severity in a large dutch crohns disease cohort," *Gut*, vol. 58, no. 3, pp. 388–395, 2009.

[4] J. Y. Streifler, M. Eliasziw, O. R. Benavente, V. C. Hachinski, A. J. Fox, and H. Barnett, "Lack of relationship between leukoaraiosis and carotid artery disease," *Archives of neurology*, vol. 52, no. 1, pp. 21–24, 1995.

[5] M. K. Sigrist, M. W. Taal, P. Bungay, and C. W. McIntyre, "Progressive vascular calcification over 2 years is associated with arterial stiffening and increased mortality in patients with stages 4 and 5 chronic kidney disease," *Clinical Journal of the American Society of Nephrology*, vol. 2,

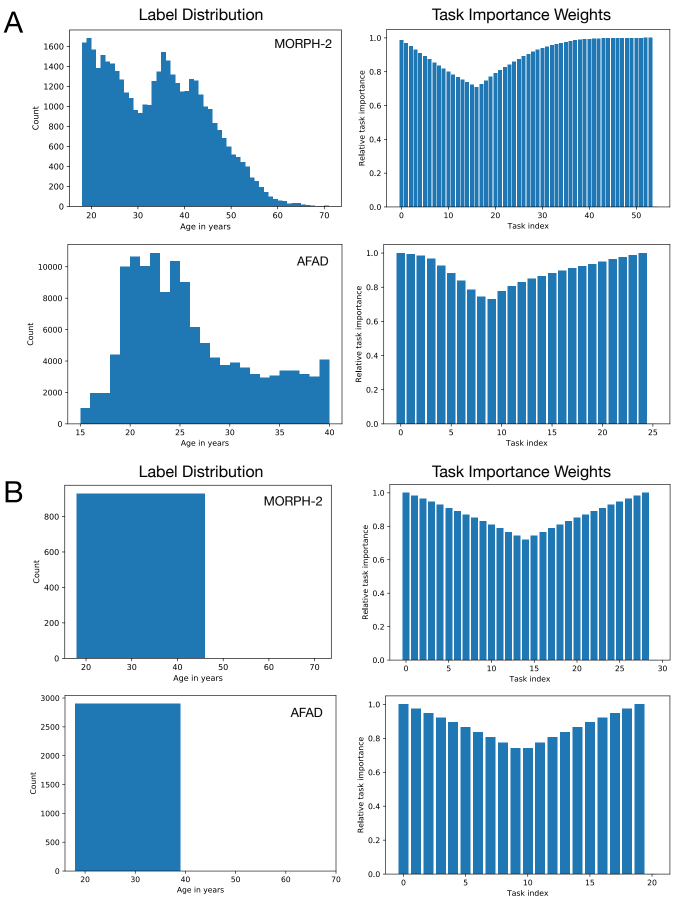

Figure 5. Age label distributions and task importance weights for the original, imbalanced datasets (A) and balanced datasets (B).

no. 6, pp. 1241–1248, 2007.

[6] R. Rettie, U. Grandcolas, and B. Deakins, "Text message advertising: Response rates and branding effects," *Journal of Targeting, Measurement and Analysis for Marketing*, vol. 13, no. 4, pp. 304–312, 2005.

[7] D. Parra, A. Karatzoglou, X. Amatriain, and I. Yavuz, "Implicit feedback recommendation via implicit-to-explicit ordinal logistic regression mapping," *Proceedings of the CARS Workshop of the Conference of Recommender Systems*, p. 5, 2011.

[8] L. Li and H.-T. Lin, "Ordinal regression by extended binary classification," in *Advances in Neural Information Processing Systems*, 2007, pp. 865–872.

[9] P. Yang, L. Zhong, and D. Metaxas, "Ranking model for facial age estimation," in *2010 International Conference on Pattern Recognition*. IEEE, 2010, pp. 3404–3407.

[10] K.-Y. Chang, C.-S. Chen, and Y.-P. Hung, "Ordinal hyperplanes ranker with cost sensitivities for age estimation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2011, pp. 585–592.

[11] D. Cao, Z. Lei, Z. Zhang, J. Feng, and S. Z. Li, "Human age estimation using ranking svm," in *Chinese Conference on Biometric Recognition*. Springer, 2012, pp. 324–331.

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

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

[14] K. Crammer and Y. Singer, "Pranking with ranking," in *Advances in Neural Information Processing Systems*, 2002, pp. 641–647.

[15] L. Shen and A. K. Joshi, "Ranking and reranking with perceptron," *Machine Learning*, vol. 60, no. 1-3, pp. 73–96, 2005.

[16] R. Herbrich, T. Graepel, and K. Obermayer, "Support vector learning for ordinal regression," in *Proceedings of the IET Conference on Artificial Neural Networks*, vol. 1, 1999, pp. 97–102.

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

[18] S. Rajaram, A. Garg, X. S. Zhou, and T. S. Huang, "Classification approach towards ranking and sorting problems," in *Proceedings of the European Conference on Machine Learning*. Springer, 2003, pp. 301–312.

[19] W. Chu and S. S. Keerthi, "New approaches to support vector ordinal regression," in *Proceedings of the International Conference on Machine Learning*. ACM, 2005, pp. 145–152.

[20] G. Levi and T. Hassner, "Age and gender classification using convolutional neural networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2015, pp. 34–42.

[21] R. Rothe, R. Timofte, and L. Van Gool, "DEX: Deep expectation of apparent age from a single image," in *Proceedings of the IEEE Conference on Computer Vision Workshops*, 2015, pp. 10–15.

[22] J.-C. Chen, A. Kumar, R. Ranjan, V. M. Patel, A. Alavi, and R. Chellappa, "A cascaded convolutional neural network for age estimation of unconstrained faces," in *Proceedings of the IEEE Conference on Biometrics Theory, Applications and Systems*, 2016, pp. 1–8.

[23] R. Ranjan, S. Sankaranarayanan, C. D. Castillo, and R. Chellappa, "An all-in-one convolutional neural network for face analysis," in *Proceedings of the IEEE Conference on Automatic Face & Gesture Recognition*, 2017, pp. 17–24.

[24] S. Chen, C. Zhang, M. Dong, J. Le, and M. Rao, "Using Ranking-CNN for age estimation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 5183–5192.

[25] E. Frank and M. Hall, "A simple approach to ordinal classification," in *Proceedings of the European Conference on Machine Learning*. Springer, 2001, pp. 145–156.

[26] L. Polania, D. Wang, and G. Fung, "Ordinal regression using noisy pairwise comparisons for body mass index range estimation," *arXiv preprint arXiv:1811.03268*, 2018.

[27] N. Ramanathan, R. Chellappa, and S. Biswas, "Age progression in human faces: A survey," *Journal of Visual Languages and Computing*, vol. 15, pp. 3349–3361, 2009.

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

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

[30] S. Raschka, "MLxtend: Providing machine learning and data science utilities and extensions to Python's scientific computing stack," *The Journal of Open Source Software*, vol. 3, no. 24, 2018.

[31] B.-C. Chen, C.-S. Chen, and W. H. Hsu, "Cross-age reference coding for age-invariant face recognition and retrieval," in *Proceedings of the European Conference on Computer Vision*. Springer, 2014, pp. 768–783.

[32] S. Y. Zhang, Zhifei and H. Qi, "Age progression/regression by conditional adversarial autoencoder," in *Proceddings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[33] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–778.

[34] D. P. Kingma and J. L. Ba, "Adam: A method for stochastic optimization," in *Proceedings of the Conference on Learning Representations*, 2015.

[35] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2818–2826.

[36] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[37] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, "Automatic differentiation in PyTorch," in *Neural Information Processing Systems Autodiff Workshop*, 2017.

VIII. Supplementary Materials

Ordinal Regression and CORAL CNNs with Other Common CNN Architectures

Initially, we developed the CORAL method in conjunction with the ResNet-34 architecture, as presented in the paper, since it is a popular architecture with a good trade-off between predictive and computational performance [1].

Since CORAL CNN is agnostic of the type of neural network architecture, we considered other commonly used convolutional neural network architectures, such as VGG16 [2] (Table S1), and Inception-v3 [3] (Table S2).

Previously, it was observed that CORAL-CNN outperforms the Ordinal-CNN by Niu et al. The same trend can be observed when the predictive performances of the CORAL approach is compared to Niu et al's method when using the VGG16 (Table S1) or Inception-v3 (Table S2) networks.

Moreover, when comparing the CORAL-CNN performances across architectures, the best predictive performance (lowest MAE) can be achieved when using the Inception-v3 architecture, which is consistent with the observation by Canziani et al. [4] that Inception-v3 outperforms ResNet-34 on classification tasks but is more expensive to train.

[1] Canziani, Alfredo, Adam Paszke, and Eugenio Culurciello. "An analysis of deep neural network models for practical applications." arXiv preprint arXiv:1605.07678 (2016).

[2] Simonyan, Karen, and Andrew Zisserman. "Very deep convolutional networks for large-scale image recognition." arXiv preprint arXiv:1409.1556 (2014).

[3] Szegedy, Christian, et al. "Rethinking the inception architecture for computer vision." Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.

[4] Canziani, Alfredo, Adam Paszke, and Eugenio Culurciello. "An analysis of deep neural network models for practical applications." arXiv preprint arXiv:1605.07678 (2016).

Table S1

Comparison of cross entropy (CE-CNN), Niu et. al's ordinal regression CNN (OR-CNN) and our CORAL-CNN based on the *VGG-16* architecture. The models were trained for 200 epochs using the ADAM optimizer with default settings as described in the main paper and a learning rate of $\alpha = 5 \times 10^{-05}$. For the CACD dataset, a learning rate of $\alpha = 1 \times 10^{-05}$ with 300 epochs was required for the models from all three methods to converge.

Method	Random Seed	MORPH-2		AFAD		UTKFace		CACD	
		MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE
CE-CNN	0	14.07	17.54	3.84	5.38	6.32	8.72	5.73	7.82
	1	14.07	17.54	3.87	5.39	6.26	8.55	5.66	7.68
	2	3.71	5.15	3.93	5.45	6.33	8.8	5.81	7.89
	AVG $\pm$ SD	10.62 $\pm$ 5.98	13.41 $\pm$ 7.15	3.88 $\pm$ 0.05	5.41 $\pm$ 0.04	6.30 $\pm$ 0.04	8.69 $\pm$ 0.13	5.73 $\pm$ 0.08	7.80 $\pm$ 0.11
OR-CNN [1]	0	2.75	3.82	3.53	4.95	6.42	8.60	5.31	7.47
	1	2.92	4.08	3.55	5.00	6.25	8.33	5.28	7,47
	2	2.95	4.14	3.72	5.23	6.50	8.81	5.39	7.52
	AVG $\pm$ SD	2.87 $\pm$ 0.11	4.01 $\pm$ 0.17	3.60 $\pm$ 0.10	5.06 $\pm$ 0.15	6.39 $\pm$ 0.13	8.58 $\pm$ 0.24	5.33 $\pm$ 0.06	7.49 $\pm$ 0.03
CORAL-CNN (ours)	0	2.76	3.73	3.45	4.78	5.95	8.28	5.25	7.49
	1	2.79	3.74	3.39	4.72	5.59	7.6	5.21	7.42
	2	2.87	3.94	3.4	4.75	5.96	8.22	5.28	7.48
	AVG $\pm$ SD	**2.81 $\pm$ 0.06**	**3.80 $\pm$ 0.12**	**3.41 $\pm$ 0.03**	**4.75 $\pm$ 0.03**	**5.83 $\pm$ 0.21**	**8.03 $\pm$ 0.38**	**5.25 $\pm$ 0.04**	**7.46 $\pm$ 0.04**

Table S2

Comparison of cross entropy (CE-CNN), Niu et. al's ordinal regression CNN (OR-CNN) and our CORAL-CNN based on the *Inception-v3* (i.e., Inception-v2 + auxiliary losses) architecture. All models were trained until convergence via 100 epochs using the ADAM optimizer with default settings as described in the main paper and a learning rate of $\alpha = 5 \times 10^{-04}$

Method	Random Seed	MORPH-2		AFAD		UTKFace		CACD	
		MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE
CE-CNN	0	3.07	4.39	3.78	5.30	6.73	9.47	5.52	8.18
	1	3.00	4.35	3.77	5.31	6.52	9.08	5.46	8.09
	2	3.00	4.36	3.79	5.33	6.81	9.4	5.44	8.04
	AVG $\pm$ SD	3.02 $\pm$ 0.04	4.37 $\pm$ 0.02	5.31 $\pm$ 0.02	3.78 $\pm$ 0.01	6.69 $\pm$ 0.15	9.32 $\pm$ 0.21	5.47 $\pm$ 0.04	8.10 $\pm$ 0.07
OR-CNN [1]	0	2.52	3.59	3.42	4.84	5.74	7.89	4.98	7.43
	1	2.57	3.69	3.45	4.87	5.49	7.58	4.93	7.37
	2	2.51	3.60	3.36	4.75	5.41	7.46	4.94	7.33
	AVG $\pm$ SD	2.53 $\pm$ 0.03	3.63 $\pm$ 0.06	3.41 $\pm$ 0.05	4.82 $\pm$ 0.06	5.55 $\pm$ 0.17	**7.64 $\pm$ 0.22**	4.95 $\pm$ 0.03	7.38 $\pm$ 0.05
CORAL-CNN (ours)	0	2.45	3.41	3.28	4.59	5.57	7.72	4.92	7.16
	1	2.41	3.36	3.32	4.63	5.26	7.3	4.91	7.21
	2	2.43	3.39	3.20	4.59	5.76	7.95	4.87	7.11
	AVG $\pm$ SD	**2.43 $\pm$ 0.02**	**3.39 $\pm$ 0.03**	**3.27 $\pm$ 0.06**	**4.60$\pm$ 0.02**	**5.53 $\pm$ 0.25**	7.66 $\pm$ 0.33	**4.90 $\pm$ 0.03**	**7.16 $\pm$ 0.05**