# Structured Output Ordinal Regression for Dynamic Facial Emotion Intensity Prediction

Minyoung Kim and Vladimir Pavlovic

Department of Computer Science, Rutgers University
110 Frelinghuysen Road, Piscataway, NJ 08854-8019, USA
{mikim,vladimir}@cs.rutgers.edu
http://seqam.rutgers.edu

**Abstract.** We consider the task of labeling facial emotion intensities in videos, where the emotion intensities to be predicted have ordinal scales (e.g., low, medium, and high) that change in time. A significant challenge is that the rates of increase and decrease differ substantially across subjects. Moreover, the actual absolute differences of intensity values carry little information, with their relative order being more important. To solve the intensity prediction problem we propose a new dynamic ranking model that models the signal intensity at each time as a label on an ordinal scale and links the temporally proximal labels using dynamic smoothness constraints. This new model extends the successful static ordinal regression to a structured (dynamic) setting by using an analogy with Conditional Random Field (CRF) models in structured classification. We show that, although non-convex, the new model can be accurately learned using efficient gradient search. The predictions resulting from this dynamic ranking model show significant improvements over the regular CRFs, which fail to consider ordinal relationships between predicted labels. We also observe substantial improvements over static ranking models that do not exploit temporal dependencies of ordinal predictions. We demonstrate the benefits of our algorithm on the Cohn-Kanade dataset for the dynamic facial emotion intensity prediction problem and illustrate its performance in a controlled synthetic setting.

**Keywords:** Video-based Facial Emotion Intensity Analysis, Ordinal Regression, Ranking, Structured Output Prediction.

## 1 Introduction

A typical task in analyzing video sequences of human emotions (e.g., facial expressions) or hand gestures is to divide the sequence into segments corresponding to different phases or intensities of the displayed artifact. For example, facial emotion signals typically follow envelope-like shapes in time: `neutral`, `increase`, `peak`, and `decrease`, beginning with low intensity, reaching a maximum, then tapering off. A significant challenge in modeling such an envelop is that the rates of increase and decrease differ substantially across subjects (e.g., different

K. Daniilidis, P. Maragos, N. Paragios (Eds.): ECCV 2010, Part III, LNCS 6313, pp. 649–662, 2010.

subjects express the same emotion with substantially different intensities). However, for subsequent recognition of different emotions across subject pools the absolute difference of intensity values is, to a large extent, less significant than the general envelope shape. Qualitative labeling and ranking is also preferred by human annotators, who can more easily judge coarse relative relationships instead of the absolute signal differences. In this work we propose to address these problems by modeling the shape of the emotion intensity envelope using a new structured ordinal regression approach, an extension of ranking to dynamic (structured) sequence domains.

The ordinal regression, often called the preference learning or ranking, is an emerging topic in the machine learning community [1] and has found applications in several traditional ranking problems, such as image classification and collaborative filtering [2,3], or image retrieval [4,5]. In the static setting, we want to predict the label $y$ of an item represented by feature vector $\mathbf{x} \in \mathbb{R}^p$ where the output bears particular meaning of preference or order (e.g., low, medium or high). The ordinal regression is fundamentally different from the standard regression in that the actual absolute difference of output values is nearly meaningless, but only their relative order matters (e.g., low $<$ medium $<$ high). The ordinal regression problems may not be optimally handled by the standard multi-class classification either because of classifier's ignorance of the ordinal scale and independent treatment of different output categories (e.g., low would be equally different from high as it would be from medium).

Despite success in static settings (i.e., vectorial input and a singleton output label), ranking problems are rarely explored in structured output prediction problems, such as the segmentation of emotion signal into regions of neutral, increasing or peak emotion. In this case the ranks or ordinal labels at different time instances should vary smoothly, with temporally proximal instances likely to have similar ranks. One may model this rank envelope by enforcing that the intensity rank at time $t-1$ has to be higher (or lower, depending on which part of the envelope one is on) than the next intensity at time $t$. Learning a static ranking model individually and independently for each time slice, however, fails to impose the same constraints during the decoding of test sequences.

In this work, we propose an intuitive but principled Conditional Random Field (CRF)-like model that can faithfully represent multiple ranking outputs correlated in a combinatorial structure (e.g., sequence or lattice). The binning modeling strategy adopted by recent static ranking approaches (see (2) in Sec. 2.1) is incorporated into our structured models through graph-based potential functions. This formulation leads to a family of log-nonlinear models that can still be estimated with high accuracy using general gradient-based search approaches. By considering the models that take into account the dynamically changing ranks of different emotion segments instead of their absolute intensity we are able to learn intensity-based segmentation of emotion sequences which is largely invariant to intra- and inter-subject variations.

We formally setup the problem and introduce basic notation below. We then briefly review the static ordinal regression in Sec. 2.1 and the CRF model in

Sec. 2.2, traditionally aimed at non-ordinal scale structured output classification. Our ordinal regression models are described in Sec. 3. After reviewing the related work in Sec. 4, we provide the experimental results in Sec. 5 where the superior prediction performance of the proposed structured output ranking model to the regular CRF model is demonstrated on both synthetic dataset and the real facial emotion intensity prediction problem on the Cohn-Kanade expression database.

### 1.1 Problem Setup and Notations

In the structured output prediction problems we deal with *multiple* output variables denoted by boldfaced $\mathbf{y}$ for distinction. $\mathbf{y}$ is composed of individual output variables $y_r$ (i.e., $\mathbf{y} = \{y_r\}$) where $r$ is the variable index. Each output variable is assumed to take one of $R$ different categories (i.e., $y_r \in \{1, \dots, R\}$) which are either nominal (regular classification) or ordinal (ranking or ordinal regression). Although it is fairly straightforward to extend our framework to arbitrary output structures, here we assume that the output variables $y_r$ in $\mathbf{y}$ are correlated in a 1-D temporal structure, with $r$ being the time index. The observation, denoted by $\mathbf{x} = \{\mathbf{x}_r\}$, is structured similarly as the output $\mathbf{y}$, and serves as input covariate for predicting $\mathbf{y}$.

Throughout the paper, we assume a supervised setting: we are given a training set of $n$ data pairs $\mathcal{D} = \{(\mathbf{x}^i, \mathbf{y}^i)\}_{i=1}^n$, which are i.i.d. samples from an underlying but unknown distribution $P_*(\mathbf{x}, \mathbf{y})$.

## 2 Static Ordinal Regression and Traditional Sequence Segmentation

### 2.1 Ordinal Regression

The goal of ordinal regression is to predict the label $y$ of an item represented by a feature vector[1] $\mathbf{x} \in \mathbb{R}^p$ where the output indicates the preference or order of this item. Formally, we let $y \in \{1, \dots, R\}$ for which $R$ is the number of preference grades, and $y$ takes an ordinal scale from the lowest preference $y = 1$ to the highest $y = R$, $y = 1 \prec y = 2 \prec \dots \prec y = R$.

The most critical aspect that differentiates the ordinal regression approaches from the multi-class classification methods is the modeling strategy. Assuming a linear modeling (straightforwardly extendible to a nonlinear version by kernel tricks), the multi-class classification typically (c.f. [6]) takes the form of

$$y = \arg\max_{c \in \{1,\dots,R\}} \mathbf{w}_c^\top \mathbf{x} + b_c. \tag{1}$$

For each class $c$, the hyperplane $(\mathbf{w}_c \in \mathbb{R}^p, b_c \in \mathbb{R})$ defines the confidence toward the class $c$. The class decision is made by selecting the one with the largest

[1] We use the notation $\mathbf{x}$ interchangeably for both a sequence observation $\mathbf{x} = \{\mathbf{x}_r\}$ and a vector, which is clearly distinguished by context.

confidence. The model parameters are $\{\{\mathbf{w}_c\}_{c=1}^R, \{b_c\}_{c=1}^R\}$. On the other hand, the recent ordinal regression approaches adopt the following modeling strategy:

$$y = c \;\text{ iff } \mathbf{w}^\top \mathbf{x} \in (b_{c-1}, b_c], \quad \text{where } -\infty = b_0 \leq b_1 \leq \cdots \leq b_R = +\infty. \tag{2}$$

The binning parameters $\{b_c\}_{c=0}^R$ form $R$ different bins, where their adjacent placement and the output deciding protocol of (2) naturally enforces the ordinal scale criteria. The parameters of the model become $\{\mathbf{w}, \{b_c\}_{c=0}^R\}$, far fewer than those of the classification models. The state-of-the-art Support Vector Ordinal Regression (SVOR) algorithms [2, 3] conform to this representation while they aim at maximizing margins at the nearby bins in the SVM formulation.

### 2.2 Conditional Random Fields for Sequence Segmentation

CRF [7, 8] is a log-linear model that represents the conditional distribution $P(\mathbf{y}|\mathbf{x})$ as the Gibbs form clamped on the observation $\mathbf{x}$:

$$P(\mathbf{y}|\mathbf{x}, \boldsymbol{\theta}) = Z(\mathbf{x}; \boldsymbol{\theta})^{-1} e^{s(\mathbf{x}, \mathbf{y}; \boldsymbol{\theta})}. \tag{3}$$

Here $Z(\mathbf{x}; \boldsymbol{\theta}) = \sum_{\mathbf{y} \in \mathcal{Y}} e^{s(\mathbf{x}, \mathbf{y}; \boldsymbol{\theta})}$ is the normalizing partition function ($\mathcal{Y}$ is a set of all possible output configurations), and $\boldsymbol{\theta}$ is the parameters[2] of the *score function* (or the negative energy) that can be written as:

$$s(\mathbf{x}, \mathbf{y}; \boldsymbol{\theta}) = \boldsymbol{\theta}^\top \boldsymbol{\Psi}(\mathbf{x}, \mathbf{y}), \tag{4}$$

where $\boldsymbol{\Psi}(\mathbf{x}, \mathbf{y})$ is the joint feature vector.

The choice of the output graph $G = (V, E)$ and the cliques critically affects model's representational capacity and the inference complexity. For the notational convenience, we further assume that we have either *node* cliques ($r \in V$) or *edge* cliques ($e = (r, s) \in E$). We denote the node features by $\boldsymbol{\Psi}_r^{(V)}(\mathbf{x}, y_r)$ and the edge features by $\boldsymbol{\Psi}_e^{(E)}(\mathbf{x}, y_r, y_s)$. Letting $\boldsymbol{\theta} = \{\mathbf{v}, \mathbf{u}\}$ be the parameters for node and edge features, respectively, the score function can be expressed as:

$$s(\mathbf{x}, \mathbf{y}; \boldsymbol{\theta}) = \sum_{r \in V} \mathbf{v}^\top \boldsymbol{\Psi}_r^{(V)}(\mathbf{x}, y_r) + \sum_{e=(r,s) \in E} \mathbf{u}^\top \boldsymbol{\Psi}_e^{(E)}(\mathbf{x}, y_r, y_s). \tag{5}$$

Although the representation in (5) is so general that it can subsume nearly arbitrary forms of features, in the conventional modeling practice, the node/edge features are often defined as the product of measurement features confined to cliques and the output class indicators. More specifically, denoting the measurement feature vector at node $r$ as $\boldsymbol{\phi}(\mathbf{x}_r)$, the node feature becomes:

$$\boldsymbol{\Psi}_r^{(V)}(\mathbf{x}, y_r) = \Big[ I(y_r = 1), \cdots, I(y_r = R) \Big]^\top \otimes \boldsymbol{\phi}(\mathbf{x}_r), \tag{6}$$

where $I(\cdot)$ is the indicator function that returns 1 (0) if the argument is true (false) and $\otimes$ denotes the Kronecker product. Hence the $k$-th block ($k = 1, \ldots, R$)

[2] For simplicity, we often drop the dependency on $\boldsymbol{\theta}$ in notations.

of $\boldsymbol{\Psi}_r^{(V)}(\mathbf{x}, y_r)$ is $\boldsymbol{\phi}(\mathbf{x}_r)$ if $y_r = k$, and the $\mathbf{0}$-vector otherwise. The edge feature is similarly defined where we typically employ the absolute difference between measurement features at adjoining nodes. Thus, $\boldsymbol{\Psi}_e^{(E)}(\mathbf{x}, y_r, y_s)$ is

$$\Big[I(y_r = k \ \wedge \ y_s = l)\Big]_{R\times R} \otimes \big|\boldsymbol{\phi}(\mathbf{x}_r) - \boldsymbol{\phi}(\mathbf{x}_s)\big|. \tag{7}$$

These feature forms are commonly used in CRFs with sequence [7] and lattice outputs [8, 9]. We call the product of parameters and the feature vectors on a clique the *(clique) potential*. For instance, $\mathbf{v}^\top \boldsymbol{\Psi}_r^{(V)}(\mathbf{x}, y_r)$ and $\mathbf{u}^\top \boldsymbol{\Psi}_e^{(E)}(\mathbf{x}, y_r, y_s)$ are the node potential and the edge potential, respectively. Hence the score function is the sum of the potentials over all cliques in the graph.

## 3 Structured Output Ordinal Regression Model

The above standard CRF modeling aims at *classification*, treating each output category nominally and equally different from all other categories. The consequence is that the model's node potential has a direct analogy to the static multi-class classification model of (1): For $y_r = c$, the node potential equals $\mathbf{v}_c^\top \boldsymbol{\phi}(\mathbf{x}_r)$ where $\mathbf{v}_c$ is the $c$-th block of $\mathbf{v}$, which corresponds to the $c$-th hyperplane $\mathbf{w}_c^\top \mathbf{x}_r + b_c$ in (1). The max can be replaced by the *softmax* function. To setup an exact equality, one can let $\boldsymbol{\phi}(\mathbf{x}_r) = [1, \mathbf{x}_r^\top]^\top$.

Conversely, the modeling strategy of the static ordinal regression methods such as (2) can be merged with the CRF through the node potentials to yield a structured output ranking model. The mechanism of doing so is not obvious because of the highly discontinuous nature of (2). We based our approach on the probabilistic model for ranking proposed by [10], which shares the notion of (2).

In [10], the noiseless probabilistic ranking likelihood is defined as

$$P_{ideal}(y = c|f(\mathbf{x})) = \begin{cases} 1 \text{ if } f(\mathbf{x}) \in (b_{c-1}, b_c] \\ 0 \text{ otherwise} \end{cases} \tag{8}$$

Here $f(\mathbf{x})$ is the model to be learned, which could be linear $f(\mathbf{x}) = \mathbf{w}^\top \mathbf{x}$. The effective ranking likelihood is constructed by contaminating the ideal model with noise. Under the Gaussian noise $\delta$ and after marginalization, one arrives at the ranking likelihood

$$P(y = c|f(\mathbf{x})) = \int_\delta P_{ideal}(y = c|f(\mathbf{x}) + \delta) \cdot \mathcal{N}(\delta; 0, \sigma^2) d\delta = \Phi\left(\frac{b_c - f}{\sigma}\right) - \Phi\left(\frac{b_{c-1} - f}{\sigma}\right), \tag{9}$$

where $\Phi(\cdot)$ is the standard normal cdf, and $\sigma$ is the parameter that controls the steepness of the likelihood function.

Now we set the node potential at node $r$ of the CRF to be the log-likelihood of (9), that is,

$$\mathbf{v}^\top \boldsymbol{\Psi}_r^{(V)}(\mathbf{x}, y_r) \longrightarrow \boldsymbol{\Gamma}_r^{(V)}(\mathbf{x}, y_r; \{\mathbf{a}, \mathbf{b}, \sigma\}), \quad \text{where}$$

$$\boldsymbol{\Gamma}_r^{(V)}(\mathbf{x}, y_r) := \sum_{c=1}^{R} I(y_r = c) \cdot \log\left(\Phi\left(\frac{b_c - \mathbf{a}^\top \boldsymbol{\phi}(\mathbf{x}_r)}{\sigma}\right) - \Phi\left(\frac{b_{c-1} - \mathbf{a}^\top \boldsymbol{\phi}(\mathbf{x}_r)}{\sigma}\right)\right). \tag{10}$$

Here, $\mathbf{a}$ (having the same dimension as $\boldsymbol{\phi}(\mathbf{x}_r)$), $\mathbf{b} = [-\infty = b_0, \ldots, b_R = +\infty]^\top$, and $\sigma$ are the new parameters, in contrast with the original CRF's node parameters $\mathbf{v}$. Substituting this expression into (5) leads to a new conditional model for structured ranking,

$$P(\mathbf{y}|\mathbf{x}, \boldsymbol{\Omega}) \propto \exp\left( \sum_{r \in V} \boldsymbol{\Gamma}_r^{(V)}(\mathbf{x}, y_r; \{\mathbf{a}, \mathbf{b}, \sigma\}) + \sum_{e=(r,s) \in E} \mathbf{u}^\top \boldsymbol{\Psi}_e^{(E)}(\mathbf{x}, y_r, y_s) \right). \quad (11)$$

We refer to this model as *CORF*, the Conditional Ordinal Random Field. The parameters of the CORF are denoted as $\boldsymbol{\Omega} = \{\mathbf{a}, \mathbf{b}, \sigma, \mathbf{u}\}$, with the ordering constraint $b_i < b_{i+1}, \forall i$. Note that the number of parameters is significantly fewer than that of the regular CRF.

Due to the nonlinear dependency of $\boldsymbol{\Gamma}$ on $\{\mathbf{a}, \mathbf{b}, \sigma\}$, (11) becomes a *log-nonlinear* model. It should first be noted that the new nonlinear modeling does not impose any additional complexity on the inference task. Since the graph topology remains the same, once the potentials are evaluated, the inference follows exactly the same procedures as that of the standard log-linear CRFs. Second, it is not difficult to see that the node potential $\boldsymbol{\Gamma}_r^{(V)}(\mathbf{x}, y_r)$, although non-linear, remains concave.

Unfortunately, the overall learning of CORF is non-convex because of the log-partition function (*log-sum-exp* of nonlinear concave functions). However, the log-likelihood objective is bounded above by 0, and the quasi-Newton or the stochastic gradient ascent [9] can be used to estimate the model parameters. We briefly describe the learning strategy. Initially, we set the edge parameters $\mathbf{u} = 0$ to form a static ranking model that treats each node independently. After learning the node parameters $\{\mathbf{a}, \mathbf{b}, \sigma\}$, we optimize the model w.r.t. $\mathbf{u}$ by gradient search while fixing the node parameters. The gradient of the log-likelihood w.r.t. $\mathbf{u}$ is (the same as the regular CRF):

$$\frac{\partial \log P(\mathbf{y}|\mathbf{x}, \boldsymbol{\Omega})}{\partial \mathbf{u}} = \sum_{e=(r,s) \in E} \left( \boldsymbol{\Psi}_e^{(E)}(\mathbf{x}, y_r, y_s) - \mathbb{E}_{P(y_r, y_s|\mathbf{x})} \left[ \boldsymbol{\Psi}_e^{(E)}(\mathbf{x}, y_r, y_s) \right] \right). \quad (12)$$

Once we obtain the initial $\boldsymbol{\Omega}$, we can start gradient search simultaneously for the whole parameters. The gradient of the log-likelihood w.r.t. $\mu = \{\mathbf{a}, \mathbf{b}, \sigma\}$ can be derived as:

$$\frac{\partial \log P(\mathbf{y}|\mathbf{x}, \boldsymbol{\Omega})}{\partial \mu} = \sum_{r \in V} \left( \frac{\partial \boldsymbol{\Gamma}_r^{(V)}(\mathbf{x}, y_r)}{\partial \mu} - \mathbb{E}_{P(y_r|\mathbf{x})} \left[ \frac{\partial \boldsymbol{\Gamma}_r^{(V)}(\mathbf{x}, y_r)}{\partial \mu} \right] \right), \quad (13)$$

where the gradient of the node potential can be computed analytically,

$$\frac{\partial \boldsymbol{\Gamma}_r^{(V)}(\mathbf{x}, y_r)}{\partial \mu} = \sum_{c=1}^{R} I(y_r = c) \cdot \frac{\mathcal{N}(z_0(r,c); 0, 1) \cdot \frac{\partial z_0(r,c)}{\partial \mu} - \mathcal{N}(z_1(r,c); 0, 1) \cdot \frac{\partial z_1(r,c)}{\partial \mu}}{\Phi(z_0(r,c)) - \Phi(z_1(r,c))},$$

$$\text{where } z_k(r,c) = \frac{b_{c-k} - \mathbf{a}^\top \boldsymbol{\phi}(\mathbf{x}_r)}{\sigma} \text{ for } k = 0, 1. \quad (14)$$

### 3.1 Model Reparameterization for Unconstrained Optimization

The gradient-based learning proposed above has to be accomplished while respecting two sets of constraints: (i) the order constraints on $\mathbf{b}$: $\{b_{j-1} \leq b_j$ for $j = 1, \ldots, R\}$, and (ii) the positive scale constraint on $\sigma$: $\{\sigma > 0\}$. Instead of general constrained optimization, we introduce a reparameterization that effectively reduces the problem to an unconstrained optimization task.

To deal with the order constraints in the parameters $\mathbf{b}$, we introduce the displacement variables $\delta_k$, where $b_j = b_1 + \sum_{k=1}^{j-1} \delta_k^2$ for $j = 2, \ldots, R-1$. So, $\mathbf{b}$ is replaced by the unconstrained parameters $\{b_1, \delta_1, \ldots, \delta_{R-2}\}$. The positiveness constraint for $\sigma$ is simply handled by introducing the free parameter $\sigma_0$ where $\sigma = \sigma_0^2$. Hence, the unconstrained node parameters are: $\{\mathbf{a}, b_1, \delta_1, \ldots, \delta_{R-2}, \sigma_0\}$. Then the gradients for $\frac{\partial z_k(r,c)}{\partial \mu}$ in (14) then become:

$$\frac{\partial z_k(r,c)}{\partial \mathbf{a}} = -\frac{1}{\sigma_0^2}\boldsymbol{\phi}(\mathbf{x}_r), \quad \frac{\partial z_k(r,c)}{\partial \sigma_0} = -\frac{2\big(b_{c-k} - \mathbf{a}^\top \boldsymbol{\phi}(\mathbf{x}_r)\big)}{\sigma_0^3}, \quad \text{for } k = 0, 1. \tag{15}$$

$$\frac{\partial z_0(r,c)}{\partial b_1} = \begin{cases} 0 & \text{if } c = R \\ \frac{1}{\sigma_0^2} & \text{otherwise} \end{cases}, \quad \frac{\partial z_1(r,c)}{\partial b_1} = \begin{cases} 0 & \text{if } c = 1 \\ \frac{1}{\sigma_0^2} & \text{otherwise} \end{cases}. \tag{16}$$

$$\frac{\partial z_0(r,c)}{\partial \delta_j} = \begin{cases} 0 & \text{if } c \in \{1, \ldots, j, R\} \\ \frac{2\delta_j}{\sigma_0^2} & \text{otherwise} \end{cases}, \quad \frac{\partial z_1(r,c)}{\partial \delta_j} = \begin{cases} 0 & \text{if } c \in \{1, \ldots, j+1\} \\ \frac{2\delta_j}{\sigma_0^2} & \text{otherwise} \end{cases}, \quad \text{for } j = 1, \ldots, R-2. \tag{17}$$

We additionally employ parameter regularization on the CORF model. For $\mathbf{a}$ and $\mathbf{u}$, we use the typical L2 regularizers $||\mathbf{a}||^2$ and $||\mathbf{u}||^2$. No specific regularization is necessary for the binning parameters $b_1$ and $\{\delta_j\}_{j=1}^{R-2}$ as they will be automatically adjusted according to the score $\mathbf{a}^\top \boldsymbol{\phi}(\mathbf{x}_r)$. For the scale parameter $\sigma_0$ we consider $(\log \sigma_0^2)^2$ as the regularizer, which essentially favors $\sigma_0 \approx 1$ and imposes quadratic penalty in log-scale.

## 4 Related Work

Developing sequence-based regressors is a recently emerging problem in computer vision. Some related work includes [11] where the problem of dynamic state estimation was tacked by the conditional state space model, an extension of the CRF to the continuous multi-variate output domain. The difficult density integrability constraints were effectively handled by the convex parameter learning. In [12], the joint task of localization and output prediction was considered, aiming at structured prediction and salient input selection at the same time. The approach can be particularly beneficial for un-segmented image/video data.

More closely related to our work, [13] proposes a ranking model based on relations between objects to be ranked, in an document retrieval setting. Unlike our CORF model the proposed continuous CRF model is a general regression model, and unable to impose the ordinal monotonicity constraints. [14] considered the sequence output prediction problem in which the outputs are partially orderable

sentiments in a document. Their model is a restricted subset of the CRF. To enforce the monotonicity constraints, they introduced a set of constraints on the CRF parameters based on the strong correlation between specific ordinal states and related binary features, also dependent on the positivity of the sentiment. Hence their approach may be restricted to discrete/binary observations/features and the particular application of the local sentiment flow estimation problem. Unlike these limitations, our approach is applicable to general features and applications since we impose the ordinal constraints on the potential functions.

## 5 Evaluations

We empirically demonstrate the performance of the proposed CORF model on the sequence labeling problem where each of the output states to be predicted has an ordinal scale. We first consider a synthetic setting, with sequences generated from a model with complex switching dynamics, where the ordinal output states are obtained by discretizing the true system states to emulate an ordinal preference scale. Next, we test the algorithm on the problem of predicting the emotion intensity from the facial image sequence. Each emotion state consists of three different intensity levels, `neutral` $<$ `increasing` $<$ `apex`, which naturally encode the total ordering typically exhibited in dynamic emotion sequences.

In these experiments, we focus on contrasting the performance of our CORF model with the standard CRF which treats the output categories as nominal classes. For both models, the optimization is accomplished using the quasi-Newton limited-BFGS method with a sufficiently large number of iterations to ensure the convergence in the regularized log-likelihood objective within a permissible precision. The balancing tradeoff between the regularization and the log-likelihood terms is estimated by grid search under cross validation. For both models the optimization starts from the zero-valued parameters with the exception of the displacement parameters $\delta_j = 1$ and the scale parameter $\sigma_0 = 1$ for the CORF.

### 5.1 Synthetic Sequences from Switching Linear Dynamical Systems

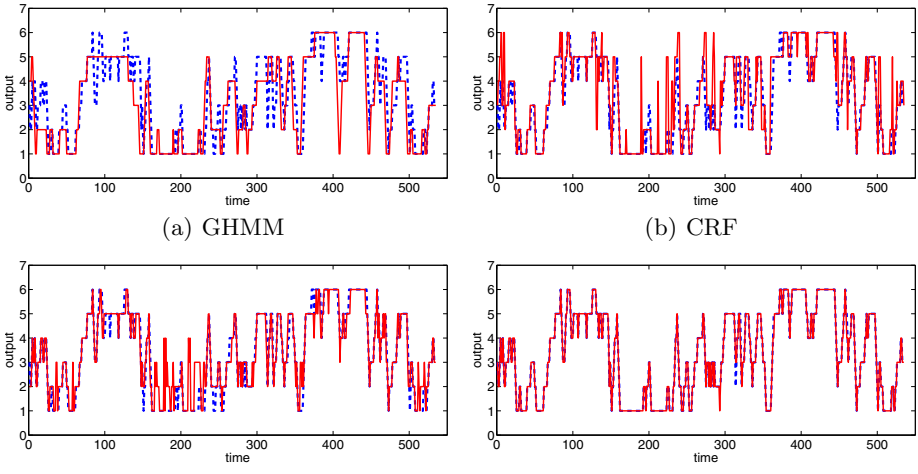
Ordinal scales can arise from observing and qualitatively quantizing the states of complex physical processes, while retaining mutual ordering of the quantized states. To simulate a complex physical dynamic processes, we consider the switching linear dynamical system (SLDS) [15]. We then quantize the states of the SLDS using an ordinal scale and seek to infer those dynamic ranks from observations.

In SLDS, the dynamical process can undergo transitions among different switching states over time, which are governed by different linear dynamic models. The overall system can be described using the state-space equations:

$$\mathbf{y}_t = \mathbf{A}(s_{t-1}) \cdot \mathbf{y}_{t-1} + \mathbf{v}_t(s_t), \quad \mathbf{x}_t = \mathbf{C}(s_t) \cdot \mathbf{y}_t + \mathbf{w}_t(s_t), \quad P(s_t = i | s_{t-1} = j) = \mathbf{Q}_{ij},$$

where $\mathbf{y}_t \in \mathbb{R}^d$ is the $d$-variate system state at time $t$, $\mathbf{x}_t \in \mathbb{R}^p$ is the $p$-variate observation features, and $s_t$ is the (discrete) switching state taking $K$ different states ($s_t = 1, \ldots, K$). The system parameters consist of $(\mathbf{A}(j), \mathbf{C}(j)) \in (\mathbb{R}^{d\times d}, \mathbb{R}^{p\times d})$ for each switching state $j = 1, \ldots, K$, and the $(K \times K)$ switching transition matrix $\mathbf{Q}$. The model takes into account the white noises $\mathbf{v}_t$ and $\mathbf{w}_t$.

We design the SLDS model with $K = 8$ switching states, $d = 1$-dim system state, and $p = 4$-dim observation features while properly choosing the model parameters and the (Gaussian) white noise variances (i.e., all system dynamics are stable, $|\mathbf{A}(j)| < 1$). We then generate 10 sequences from the SLDS model, where the sequence length $T$ varies as $T \sim \mathcal{N}(500, 30^2)$. For the generated sequences, we regard the system state $\mathbf{y}_t$ as the output to be predicted at time $t$ while $\mathbf{x}_t$ is the input feature vector at time $t$. To convert the real-valued $\mathbf{y}_t$ to ordinal-scale discrete-valued $y_t$, we discretize $\mathbf{y}_t$ into $R = 6$ categories, with each category being equally likely. We generate ten pairs of such sequences, one of which ($\mathbf{y}$) is illustrated in Fig. 1 as the blue dotted curve.




(a) GHMM (b) CRF

(c) (Static) SVOR (d) CORF

**Fig. 1.** Output prediction for synthetic SLDS sequences. The ground truth is depicted as blue dashed line while the predicted outputs are red solid lines.

For this set we perform leave-one-out validation. The average test errors (means and standard deviations) of the competing approaches are shown in Table 1. Here we present both the average 0/1 loss ($\frac{1}{T}\sum_t I(y_t \neq \overline{y}_t)$) and the absolute loss ($\frac{1}{T}\sum_t |y_t - \overline{y}_t|$), where $y_t$ and $\overline{y}_t$ are the predicted and the ground-truth label, respectively.

To see the baseline performance, we first test the Gaussian HMM (denoted by GHMM) where its hidden state at time $t$ represents the ordinal label $y_t$. Hence the joint likelihood maximization leads to a one-shot learning with no latent variables. The label prediction for a given test sequence can be accomplished using the well-known Viterbi decoding. The next model we contrast with is the regular

CRF. For the measurement features for the CRF, we use the quadratic expansion of $\mathbf{x}_t$ yielding 15-dim node features, which corresponds to the GHMM's Gaussian measurement modeling. The edge features are simply set to 1 to mimic GHMM's transition matrix. Not surprisingly, the CRF's discriminative modeling improves the prediction performance over the generative GHMM models.

We also compare our approach with the *static* ordinal regression approaches, which have been studied considerably in the machine learning community. These approaches are static and unable to handle structured outputs in a principled manner as they treat the time slices $\{(y_t, \mathbf{x}_t)\}_t$ as i.i.d. samples. Here we consider one of the most recent approaches[3], the support vector ordinal regression (SVOR) of [3], which optimizes multiple thresholds to define parallel discriminant hyperplanes for the ordinal scales. We use the method with explicit constraints. The features for the SVOR are the same as the node features of the CRFs. The SVM hyperparameters are selected by 5-fold cross validation.

Our CORF model again uses the same node/edge features as the CRF. As shown in the table, the CORF prediction is nearly perfect, outperforming other methods with strong statistical significance. Fig. 1 showing predicted and true ordinal ranks of a selected test sequence exemplifies this trend. Interestingly, the static ordinal regressor SVOR exhibits superior performance to the standard CRF learning, which can be attributed to the effective treatment of the ordinal-scale output variables, not present in the CRF model which treats all levels as equally different/similar. However, the SVOR exhibits non-smooth prediction as it fails to exploit the temporal dependency of predictions. The CORF, on the other hand, combines both the benefit of proper ordinal scales and the temporal smoothness, resulting in accurate predictions.

**Table 1.** Test errors in synthetic SLDS data set

| Methods | GHMM | CRF | SVOR | CORF |
|---|---|---|---|---|
| 0/1 Loss | 0.4687 ± 0.0567 | 0.2407 ± 0.0328 | 0.1847 ± 0.0493 | 0.0052 ± 0.0029 |
| Absolute Loss | 0.5894 ± 0.0605 | 0.3830 ± 0.0581 | 0.2028 ± 0.0678 | 0.0052 ± 0.0029 |

### 5.2 Dynamic Facial Emotion Intensity Prediction

The next task we consider is the facial emotion intensity prediction. We use the Cohn-Kanade facial expression database [16], which consists of six basic emotions (anger, disgust, fear, happiness, sadness, and surprise) performed by 100 students aged from 18 to 30 years old. In this experiment, we selected image sequences from 96 subjects. We randomly select 66 subjects as the training set, and the rest subjects as the testing set. After detecting faces by the cascaded face detector [17], we normalize them into $(64 \times 64)$ images which are aligned based on the eye locations similar to [18].

[3] We also tested the static approach [10], the Gaussian process ordinal regressor (GPOR). However, its test performance on this dataset was far worse than that of the SVOR.

Facial expression recognition is an active research area in computer vision [19, 20, 21, 22]. Unlike the traditional settings (e.g., [21]) where just the ending few peak frames are considered, we use the entire sequences that cover the onset of the expression all the way to the apex in order to conduct the task of dynamic emotion intensity labeling. The sequence lengths are about 20-frame long on average.

The frames in the sequences are manually labeled into three ordinal categories: `neutral` $<$ `increasing` $<$ `apex`. Overall the `increasing` state takes about $10 \sim 30\%$ of the frames in each sequence, while the other two states occupy the rest roughly equally on average. For the image features, we first extract the Haar-like features, following [22]. To reduce feature dimensionality, we apply PCA on the training frames for each emotion, which gives rise to $20 \sim 30$ dimensional feature vectors corresponding to 95% of the total energy. To normalize the sequence, we subtract the initial-frame feature vector from the subsequent frames, i.e., $\mathbf{x}_t \leftarrow \mathbf{x}_t - \mathbf{x}_1$.

The average per-frame test errors within each emotion class are shown in Table 2. Here we also contrasted with the static ordinal regression approach based on the probabilistic model, called the Gaussian process ordinal regressor (GPOR) [10]. Although the GPOR performs better than the SVOR in this problem, its independent treatment of the frames in sequences yields inferior performance to the dynamic models.

Our CORF consistently performs best for all emotions, exhibiting performance superior to the regular CRF that fails to consider ordering relationships between intensity levels. The static ordinal regressors (SVOR and GPOR) often result in highly biased predictions (e.g., either all `neutral` frames or all `apex`), which signifies the importance of capturing the smooth emotion dynamics in this problem. Interestingly, most approaches yield higher errors for "sadness" than other emotions, such as say "surprise". By visually inspecting the videos of these emotions, we have noticed that the intensity variations of "sadness" are far more subtle to discriminate than "surprise". For some selected test sequences, we also depict the decoded intensities by the CRF and the CORF in Fig. 2.

**Table 2.** Average test errors in facial emotion intensity prediction

(a) Anger

| Loss | GHMM | CRF | SVOR | GPOR | CORF |
|---|---|---|---|---|---|
| 0/1 | 0.4059 | 0.2890 | 0.6103 | 0.5735 | **0.1817** |
| Abs. | 0.4276 | 0.2951 | 0.8534 | 0.7977 | **0.2017** |

(b) Disgust

| Loss | GHMM | CRF | SVOR | GPOR | CORF |
|---|---|---|---|---|---|
| 0/1 | 0.1493 | 0.1154 | 0.5938 | 0.5187 | **0.0582** |
| Abs. | 0.1493 | 0.1154 | 0.9417 | 0.5662 | **0.0582** |

(c) Fear

| Loss | GHMM | CRF | SVOR | GPOR | CORF |
|---|---|---|---|---|---|
| 0/1 | 0.2941 | 0.2530 | 0.5733 | 0.4416 | **0.1689** |
| Abs. | 0.2971 | 0.2564 | 0.9051 | 0.6533 | **0.1689** |

(d) Happiness

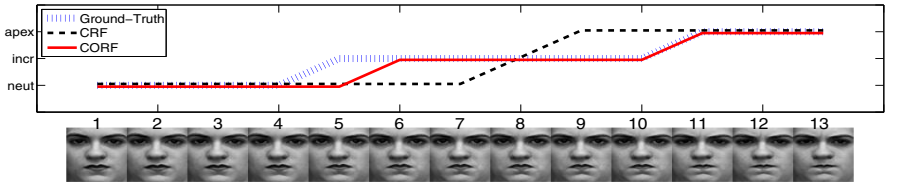
| Loss | GHMM | CRF | SVOR | GPOR | CORF |
|---|---|---|---|---|---|
| 0/1 | 0.2954 | 0.2341 | 0.4964 | 0.4216 | **0.1617** |
| Abs. | 0.3035 | 0.2341 | 0.7876 | 0.4515 | **0.1617** |

(e) Sadness

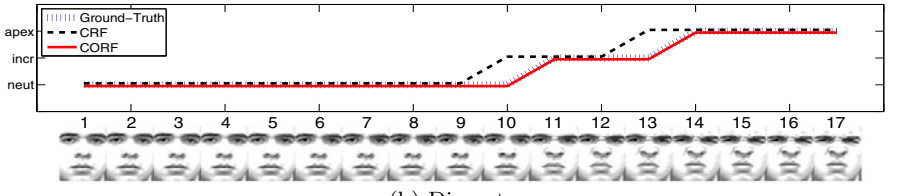
| Loss | GHMM | CRF | SVOR | GPOR | CORF |
|---|---|---|---|---|---|
| 0/1 | 0.4598 | 0.3538 | 0.6287 | 0.5561 | **0.2760** |
| Abs. | 0.5754 | 0.4388 | 0.9836 | 0.8993 | **0.3405** |

(f) Surprise

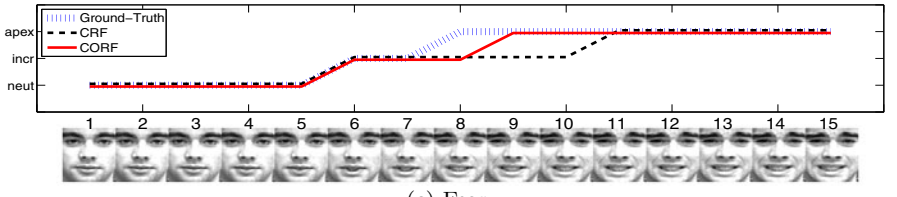
| Loss | GHMM | CRF | SVOR | GPOR | CORF |
|---|---|---|---|---|---|
| 0/1 | 0.1632 | 0.1397 | 0.5855 | 0.4065 | **0.0924** |
| Abs. | 0.1632 | 0.1397 | 0.9563 | 0.5984 | **0.0924** |



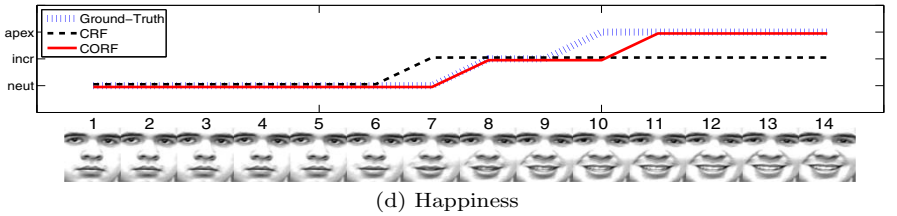

(a) Anger



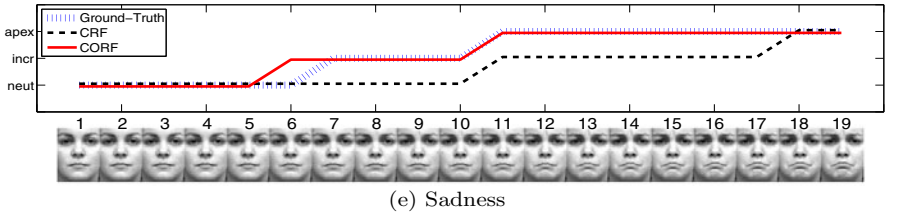

(b) Disgust



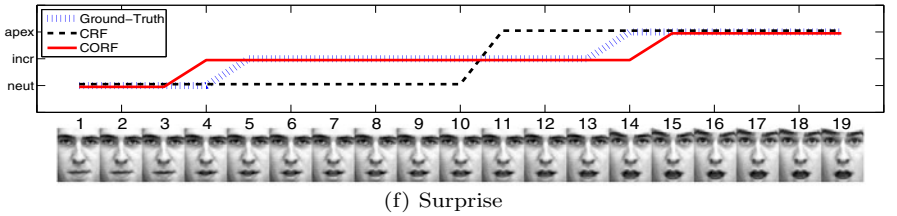

(c) Fear




(d) Happiness




(e) Sadness




(f) Surprise

**Fig. 2.** Facial emotion intensity prediction results for some test sequences. The ground truth is depicted as blue dotted line while the predicted outputs of the regular CRF and the CORF are in black dashed and red solid lines, respectively.

### 5.3 Discussion and Conclusions

Our approach, much like the similar modeling strategies in static ranking / ordinal regression settings [1, 2], treats the ordinal-scale states as intervals on a real line using a binning model. As a consequence, ordering and distinct interrelationship between different ranks/labels is preserved, which is of essence when modeling ordinal processes. As discussed in [1, 2], the multi-class SVM, and similarly the regular CRF in the dynamic setting, ignore the total ordering of the class labels. These classification-based models fail to model the correlation among the hyperplanes (or potentials in the CRFs) representing the classes, a task necessary for preserving the distinction of relationships among labels.

Another crucial aspect of our CORF model is its ability to preserve the transitivity and asymmetry of the ordinal scale states. As alluded to in [1], learning of preference relations may not be properly treated as a standard classification problem by considering pairs of objects since the properties of transitivity and asymmetry may be violated by traditional approaches due to the problem of stochastic transitivity.

The binning-based node potentials in our model also tend to yield smaller errors as they focus on closest neighboring intervals. That is, when the misclassification occurs, it is more likely to be close to the true class (interval) in the total ordering. On the other hand, in the regular CRF, the class-wise node potentials compete with one another "independently", failing to make use of proximity constraints. As a consequence, the misclassifications away from the true "label" incur higher cost in the ordinal regression compared to the label-distance agnostic classification setting. This all leads to more accurate predictions by CORF on classes of problems where ordinal scales are critical but have been commonly tackled as classification problems.

While this work focuses on the intensity estimation and segmentation of emotion signals as an example of this class of problems, similar approaches can be applied to other instances where dynamically changing ordinal scale is important. Modeling the envelope of hand gesture signals or dynamic qualitative characterizations of video events (e.g., low-to-high-to-low intensity of an explosion) can benefit from this setting.

**Acknowledgments.** We are grateful to Peng Yang and Dimitris N. Metaxas for their help and discussions throughout the course of this work. This material is based upon work supported by the National Science Foundation under Grant No. IIS-0916812.

## References

[1] Herbrich, R., Graepel, T., Obermayer, K.: Large margin rank boundaries for ordinal regression. In: Advances in Large Margin Classifiers, MIT Press, Cambridge (2000)

[2] Shashua, A., Levin, A.: Ranking with large margin principle: Two approaches. In: Neural Information Processing Systems (2003)

[3] Chu, W., Keerthi, S.S.: New approaches to support vector ordinal regression. In: International Conference on Machine Learning (2005)
[4] Hu, Y., Li, M., Yu, N.: Multiple-instance ranking: Learning to rank images for image retrieval. In: Computer Vision and Pattern Recognition (2008)
[5] Jing, Y., Baluja, S.: Pagerank for product image search. In: Proceeding of the 17th International Conference on World Wide Web (2008)
[6] Crammer, K., Singer, Y.: On the algorithmic implementation of multiclass kernel-based vector machines. Journal of Machine Learning Research 2, 265–292 (2001)
[7] Lafferty, J., McCallum, A., Pereira, F.: Conditional Random Fields: Probabilistic models for segmenting and labeling sequence data. In: International Conference on Machine Learning (2001)
[8] Kumar, S., Hebert, M.: Discriminative random fields. International Journal of Computer Vision 68, 179–201 (2006)
[9] Vishwanathan, S., Schraudolph, N., Schmidt, M., Murphy, K.: Accelerated training of conditional random fields with stochastic meta-descent. In: International Conference on Machine Learning (2006)
[10] Chu, W., Ghahramani, Z.: Gaussian processes for ordinal regression. Journal of Machine Learning Research 6, 1019–1041 (2005)
[11] Kim, M., Pavlovic, V.: Discriminative learning for dynamic state prediction. IEEE Transactions on Pattern Analysis and Machine Intelligence 31, 1847–1861 (2009)
[12] Ionescu, C., Bo, L., Sminchisescu, C.: Structural SVM for visual localization and continuous state estimation. In: International Conference on Computer Vision (2009)
[13] Qin, T., Liu, T.Y., Zhang, X.D., Wang, D.S., Li, H.: Global ranking using continuous conditional random fields. In: Neural Information Processing Systems (2008)
[14] Mao, Y., Lebanon, G.: Generalized isotonic conditional random fields. Machine Learning 77, 225–248 (2009)
[15] Pavlovic, V., Rehg, J.M., MacCormick, J.: Learning switching linear models of human motion. In: Neural Information Processing Systems (2000)
[16] Lien, J., Kanade, T., Cohn, J., Li, C.: Detection, tracking, and classification of action units in facial expression. J. Robotics and Autonomous Systems (1999)
[17] Viola, P., Jones, M.: Robust real-time object detection. International Journal of Computer Vision 57, 137–154 (2001)
[18] Tian, Y.: Evaluation of face resolution for expression analysis. In: Computer Vision and Pattern Recognition, Workshop on Face Processing in Video (2004)
[19] Lien, J.J., Cohn, J.F.: Automated facial expression recognition based on FACS action units. In: Int'l Conf. on Automatic Face and Gesture Recognition (1998)
[20] Cohen, I., Sebe, N., Garg, A., Chen, L.S., Huang, T.S.: Facial expression recognition from video sequences: Temporal and static modeling. Computer Vision and Image Understanding 91, 160–187 (2003)
[21] Shan, C., Gong, S., McOwan, P.W.: Conditional mutual information based boosting for facial expression recognition. In: British Machine Vision Conference (2005)
[22] Yang, P., Liu, Q., Metaxas, D.N.: Rankboost with l1 regularization for facial expression recognition and intensity estimation. In: International Conference on Computer Vision (2009)