Ordinal Regression / Ranking Learning
It can be considered an intermediate problem between regression and classification.

There are several approaches to tackle ordinal regression problems:

1. The naive idea is to transform the ordinal scales into numeric values, and then solve the problem as a standard regression problem.
2. Decompose the original ordinal regression problem into a set of binary classification tasks.
3. Formulate the original problem as a large augmented binary classification problem.

(Difficulty with approach 2 and 3 => the problem size of these formulations is a quadratic function of the training data size.)

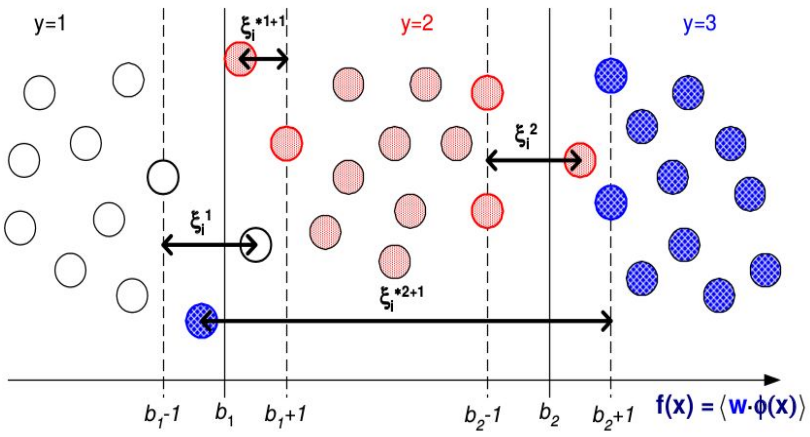
4. PRank algorithm

Chu and Keerthi proposed 2 new algos :

1. The first one takes only the adjacent ranks into account in determining the thresholds, and also introduces explicit constraints in the problem formulation that enforce the inequalities on the thresholds.
   $b1 \leq b2 \leq \ldots \leq br-1$
2. The second approach considers the training samples from all the ranks to determine each threshold. Interestingly in this second approach, the ordinal inequality constraints on the thresholds are automatically satisfied at the optimal solution though there are no explicit constraints on these thresholds.

For both approaches the size of the optimization problems is linear in the number of training samples.

Explicit Constraints on Thresholds



For binary classification (a special case of ordinal regression with r = 2), SVMs find an optimal direction that maps the feature vectors into function values on the real line, and a single optimized threshold is used to divide the real line into two regions for the two classes respectively.
For ordinal regression, the support vector formulation could attempt to find an optimal mapping direction w, and r − 1 thresholds, which define r − 1 parallel discriminant hyperplanes for the r ranks accordingly.

More exactly, each sample in the j-th category should have a function value that is less than the lower margin b j − 1, otherwise < w · φ(x^j_i)> − (b_j − 1) is the error (denoted as ξ^i_j); similarly, each sample from the (j +1)-th category should have a function value that is greater than the) upper margin b j + 1, otherwise (b_j + 1) − < w · φ(x^{j+1}_i) > is the error (denoted as $\xi^{*\,j+1}_1$). The superscript ∗ denotes that the error is associated with a sample in the adjacent upper category of the j-th threshold.

$$\min_{\boldsymbol{w},\boldsymbol{b},\boldsymbol{\xi},\boldsymbol{\xi}^*} \frac{1}{2}\langle \boldsymbol{w}\cdot\boldsymbol{w}\rangle + C\sum_{j=1}^{r-1}\left(\sum_{i=1}^{n^j}\xi_i^j + \sum_{i=1}^{n^{j+1}}\xi_i^{*j+1}\right)$$

Equation 3 ->

$$\begin{aligned}
&\text{subject to}\\
&\langle \boldsymbol{w}\cdot\phi(x_i^j)\rangle - b_j \le -1+\xi_i^j,\\
&\xi_i^j \ge 0, \qquad \text{for } i=1,\ldots,n^j;\\
&\langle \boldsymbol{w}\cdot\phi(x_i^{j+1})\rangle - b_j \ge +1-\xi_i^{*j+1},\\
&\xi_i^{*j+1} \ge 0, \qquad \text{for } i=1,\ldots,n^{j+1};\\
&\text{where } j \text{ runs over } 1,\ldots,r-1 \text{ and } C>0.
\end{aligned}$$

A problem with the above formulation is that the natural ordinal inequalities on the thresholds, i.e., $b_1 \leq b_2 \leq \ldots \leq b_{r-1}$ cannot be guaranteed to hold at the solution. To tackle this problem, we explicitly include the following constraints in (3):

$$b_{j-1} \leq b_j, \quad \text{for } j = 2, \ldots, r-1. \tag{4}$$

2.1. Primal and Dual Problems

By introducing two auxiliary variables $b_0 = -\infty$ and $b_r = +\infty$, the modified *primal* problem in (2)–(4) can be equivalently written as follows:

$$\min_{\boldsymbol{w},\boldsymbol{b},\boldsymbol{\xi},\boldsymbol{\xi}^*} \ \tfrac{1}{2}\langle \boldsymbol{w} \cdot \boldsymbol{w} \rangle + C \sum_{j=1}^{r} \sum_{i=1}^{n^j} \left(\xi_i^j + \xi_i^{*j} \right) \tag{5}$$

subject to

$$\begin{array}{ll} \langle \boldsymbol{w} \cdot \phi(x_i^j) \rangle - b_j \leq -1 + \xi_i^j, & \xi_i^j \geq 0, \ \forall i, j; \\ \langle \boldsymbol{w} \cdot \phi(x_i^j) \rangle - b_{j-1} \geq +1 - \xi_i^{*j}, & \xi_i^{*j} \geq 0, \ \forall i, j; \\ b_{j-1} \leq b_j, \ \forall j. & \end{array} \tag{6}$$

The *dual* problem can be derived by standard Lagrangian techniques. Let $\alpha_i^j \geq 0$, $\gamma_i^j \geq 0$, $\alpha_i^{*j} \geq 0$, $\gamma_i^{*j} \geq 0$ and $\mu^j \geq 0$ be the Lagrangian multipliers for the inequalities in (6). The Lagrangian for the *primal* problem is:

$$\begin{array}{l} \mathcal{L}_e = \tfrac{1}{2}\langle \boldsymbol{w} \cdot \boldsymbol{w} \rangle + C \sum_{j=1}^{r} \sum_{i=1}^{n^j} \left(\xi_i^j + \xi_i^{*j} \right) \\ - \sum_{j=1}^{r} \sum_{i=1}^{n^j} \alpha_i^j (-1 + \xi_i^j - \langle \boldsymbol{w} \cdot \phi(x_i^j) \rangle + b_j) \\ - \sum_{j=1}^{r} \sum_{i=1}^{n^j} \alpha_i^{*j} (-1 + \xi_i^{*j} + \langle \boldsymbol{w} \cdot \phi(x_i^{*j}) \rangle - b_{j-1}) \\ - \sum_{j=1}^{r} \gamma_i^j \xi_i^j - \sum_{j=1}^{r} \gamma_i^{*j} \xi_i^{*j} - \sum_{j=1}^{r} \mu^j (b_j - b_{j-1}). \end{array} \tag{7}$$

The KKT conditions for the *primal* problem require the following to hold:

$$\frac{\partial \mathcal{L}_e}{\partial \boldsymbol{w}} = \boldsymbol{w} - \sum_{j=1}^{r} \sum_{i=1}^{n^j} \left(\alpha_i^{*j} - \alpha_i^j \right) \phi(x_i^j) = 0; \tag{8}$$

$$\frac{\partial \mathcal{L}_e}{\partial \xi_i^j} = C - \alpha_i^j - \gamma_i^j = 0, \ \forall i, \ \forall j; \tag{9}$$

$$\frac{\partial \mathcal{L}_e}{\partial \xi_i^{*j}} = C - \alpha_i^{*j} - \gamma_i^{*j} = 0, \ \forall i, \ \forall j; \tag{10}$$

$$\frac{\partial \mathcal{L}_e}{\partial b_j} = \sum_{i=1}^{n^j} \left(\alpha_i^j + \mu^j \right) - \sum_{i=1}^{n^{j+1}} \left(\alpha_i^{*j+1} + \mu^{j+1} \right) = 0, \forall j.$$

Note that the dummy variables associated with b_0 and b_r, i.e. μ^1, μ^r, α_i^{*1}'s and α_i^r's, are always zero. The conditions (9) and (10) give rise to the constraints $0 \leq \alpha_i^j \leq C$ and $0 \leq \alpha_i^{*j} \leq C$ respectively. Let us now apply Wolfe duality theory to the *primal* problem. By introducing the KKT conditions (8)–(10) into the Lagrangian (7) and applying the kernel trick (1), the *dual* problem becomes a maximization problem involving the Lagrangian multipliers $\boldsymbol{\alpha}$, $\boldsymbol{\alpha}^*$ and $\boldsymbol{\mu}$:

$$\max_{\boldsymbol{\alpha},\boldsymbol{\alpha}^*,\boldsymbol{\mu}} \sum_{j,i} (\alpha_i^j + \alpha_i^{*j}) - \frac{1}{2} \sum_{j,i} \sum_{j',i'} (\alpha_i^{*j} - \alpha_i^j)(\alpha_{i'}^{*j'} - \alpha_{i'}^{j'}) \mathcal{K}(x_i^j, x_{i'}^{j'}) \tag{11}$$

subject to

$$\begin{array}{ll} 0 \leq \alpha_i^j \leq C, & \forall i, \ \forall j, \\ 0 \leq \alpha_i^{*j+1} \leq C, & \forall i, \ \forall j, \\ \sum_{i=1}^{n^j} \alpha_i^j + \mu^j = \sum_{i=1}^{n^{j+1}} \alpha_i^{*j+1} + \mu^{j+1}, & \forall j, \\ \mu^j \geq 0, & \forall j, \end{array} \tag{12}$$

where j runs over $1, \ldots, r-1$. Leaving the dummy variables out of account, the size of the optimization problem is $2n - n^1 - n^r$ ($\boldsymbol{\alpha}$ and $\boldsymbol{\alpha}^*$) plus $r-2$ (for $\boldsymbol{\mu}$).

The *dual* problem (11)–(12) is a convex quadratic programming problem. Once the $\boldsymbol{\alpha}$, $\boldsymbol{\alpha}^*$ and $\boldsymbol{\mu}$ are obtained by solving this problem, $\boldsymbol{w}$ is obtained from (8). The determination of the b_j's will be addressed in the next section. The discriminant function value for a new input vector x is

$$f(x) = \langle \boldsymbol{w} \cdot \phi(x) \rangle = \sum_{j,i} (\alpha_i^{*j} - \alpha_i^j) \mathcal{K}(x_i^j, x). \tag{13}$$

The predictive ordinal decision function is given by $\arg\min_i \{i : f(x) < b_i\}$.

2.2. Optimality Conditions for the Dual

To derive proper stopping conditions for algorithms that solve the *dual* problem and also determine the thresholds b_j's, it is important to write down the optimality conditions for the *dual*. Though the resulting conditions that are derived below look a bit clumsy because of the notations, the ideas behind them are very much similar to those for the binary SVM classifier case. The Lagrangian for the *dual* can be written down as follows:

$$\begin{array}{l} \mathcal{L}_d = \tfrac{1}{2} \sum_{j,i} \sum_{j',i'} (\alpha_i^{*j} - \alpha_i^j)(\alpha_{i'}^{*j'} - \alpha_{i'}^{j'}) \mathcal{K}(x_i^j, x_{i'}^{j'}) \\ + \sum_{j=1}^{r-1} \beta_j (\sum_{i=1}^{n^j} \alpha_i^j - \sum_{i=1}^{n^{j+1}} \alpha_i^{*j+1} + \mu^j - \mu^{j+1}) \\ - \sum_{j,i} (\eta_i^j \alpha_i^j + \eta_i^{*j} \alpha_i^{*j}) - \sum_{j,i} (\pi_i^j (C - \alpha_i^j) + \\ \pi_i^{*j} (C - \alpha_i^{*j})) - \sum_{j=2}^{r-1} \lambda^j \mu^j - \sum_{j,i} (\alpha_i^j + \alpha_i^{*j}) \end{array}$$

where the Lagrangian multipliers η_i^j, η_i^{*j}, π_i^j, π_i^{*j} and λ^j are non-negative, while β_j can take any value.

The KKT conditions associated with β_j can be given as follows:

$$\begin{array}{l} \frac{\partial \mathcal{L}_d}{\partial \alpha_i^j} = -f(x_i^j) - 1 - \eta_i^j + \pi_i^j + \beta_j = 0, \pi_i^j \geq 0, \\ \eta_i^j \geq 0, \pi_i^j (C - \alpha_i^j) = 0, \eta_i^j \alpha_i^j = 0, \ \text{for } i = 1, \ldots, n^j; \\ \frac{\partial \mathcal{L}_d}{\partial \alpha_i^{*j+1}} = f(x_i^{j+1}) - 1 - \eta_i^{*j+1} + \pi_i^{*j+1} - \beta_j = 0, \\ \pi_i^{*j+1} \geq 0, \eta_i^{*j+1} \geq 0, \pi_i^{*j+1} (C - \alpha_i^{*j+1}) = 0, \\ \eta_i^{*j+1} \alpha_i^{*j+1} = 0, \ \text{for } i = 1, \ldots, n^{j+1}; \end{array} \tag{14}$$

where $f(x)$ is defined as in (13), while the KKT conditions associated with the μ^j are

$$\beta_j - \beta_{j-1} - \lambda^j = 0, \lambda^j \mu^j = 0, \lambda^j \geq 0, \tag{15}$$

where $j = 2, \ldots, r-1$. The conditions in (14) can be re-grouped into the following six cases:

$$\begin{array}{lll} \text{case 1}: & \alpha_i^j = 0 & f(x_i^j) + 1 \le \beta_j \\ \text{case 2}: & 0 < \alpha_i^j < C & f(x_i^j) + 1 = \beta_j \\ \text{case 3}: & \alpha_i^j = C & f(x_i^j) + 1 \ge \beta_j \\ \text{case 4}: & \alpha_i^{*j+1} = 0 & f(x_i^{j+1}) - 1 \ge \beta_j \\ \text{case 5}: & 0 < \alpha_i^{*j+1} < C & f(x_i^{j+1}) - 1 = \beta_j \\ \text{case 6}: & \alpha_i^{*j+1} = C & f(x_i^{j+1}) - 1 \le \beta_j \end{array}$$

We can classify any variable into one of the following six sets:

$$\begin{array}{l} I_{0a}^j = \{i \in \{1, \ldots, n^j\} : 0 < \alpha_i^j < C\} \\ I_{0b}^j = \{i \in \{1, \ldots, n^{j+1}\} : 0 < \alpha_i^{*j+1} < C\} \\ I_1^j = \{i \in \{1, \ldots, n^{j+1}\} : \alpha_i^{*j+1} = 0\} \\ I_2^j = \{i \in \{1, \ldots, n^j\} : \alpha_i^j = 0\} \\ I_3^j = \{i \in \{1, \ldots, n^j\} : \alpha_i^j = C\} \\ I_4^j = \{i \in \{1, \ldots, n^{j+1}\} : \alpha_i^{*j+1} = C\} \end{array}$$

Let us denote $I_0^j = I_{0a}^j \cup I_{0b}^j$, $I_{up}^j = I_0^j \cup I_1^j \cup I_3^j$ and $I_{low}^j = I_0^j \cup I_2^j \cup I_4^j$. We further define $F_{up}^i(\beta_j)$ on the set I_{up}^j as

$$F_{up}^i(\beta_j) = \begin{cases} f(x_i^j) + 1 & \text{if} \quad i \in I_{0a}^j \cup I_3^j \\ f(x_i^{j+1}) - 1 & \text{if} \quad i \in I_{0b}^j \cup I_1^j \end{cases}$$

and $F_{low}^i(\beta_j)$ on the set I_{low}^j as

$$F_{low}^i(\beta_j) = \begin{cases} f(x_i^j) + 1 & \text{if} \quad i \in I_{0a}^j \cup I_2^j \\ f(x_i^{j+1}) - 1 & \text{if} \quad i \in I_{0b}^j \cup I_4^j \end{cases}$$

Then the conditions can be simplified as

$$\beta_j \le F_{up}^i(\beta_j) \, \forall i \in I_{up}^j \text{ and } \beta_j \ge F_{low}^i(\beta_j) \, \forall i \in I_{low}^j,$$

which can be compactly written as:

$$b_{low}^j \le \beta_j \le b_{up}^j \tag{16}$$

where $b_{up}^j = \min\{F_{up}^i(\beta_j) : i \in I_{up}^j\}$ and $b_{low}^j = \max\{F_{low}^i(\beta_j) : i \in I_{low}^j\}$.

The KKT conditions in (15) indicate that the condition, $\beta_{j-1} \le \beta_j$ always holds, and that $\beta_{j-1} = \beta_j$ if $\mu^j > 0$. To merge the conditions (15) and (16), let us define

$$\tilde{B}_{low}^j = \max\{b_{low}^k : k = 1, \ldots, j\}$$

and

$$\tilde{B}_{up}^j = \min\{b_{up}^k : k = j, \ldots, r-1\},$$

where $j = 1, \ldots, r-1$. The overall optimality conditions can be simply written as

$$B_{low}^j \le \beta_j \le B_{up}^j \quad \forall j$$

where

$$B_{low}^j = \begin{cases} \tilde{B}_{low}^{j+1} & \text{if } \mu^{j+1} > 0 \\ \tilde{B}_{low}^j & \text{otherwise} \end{cases}$$

and

$$B_{up}^j = \begin{cases} \tilde{B}_{up}^{j-1} & \text{if } \mu^j > 0 \\ \tilde{B}_{up}^j & \text{otherwise.} \end{cases}$$

Table 1. The basic framework of the SMO algorithm for support vector ordinal regression using explicit threshold constraints.

SMO	start at a valid point, $\boldsymbol{\alpha}$, $\boldsymbol{\alpha}^*$ and $\boldsymbol{\mu}$, that satisfy (12), find the current B_{up}^j and B_{low}^j $\forall j$
Loop	do 1. determine the *active threshold* J 2. optimize the pair of active variables and the set $\boldsymbol{\mu}_a$ 3. compute B_{up}^j and B_{low}^j $\forall j$ at the new point while the optimality condition (17) has not been satisfied
Exit	return $\boldsymbol{\alpha}$, $\boldsymbol{\alpha}^*$ and $\boldsymbol{b}$

We introduce a tolerance parameter $\tau > 0$, usually 0.001, to define approximate optimality conditions. The overall stopping condition becomes

$$\max\{B_{low}^j - B_{up}^j : j = 1, \ldots, r-1\} \le \tau. \tag{17}$$

From the conditions in (14) and (3), it is easy to see the close relationship between the b_j's in the *primal* problem and the multipliers β_j's. In particular, at the optimal solution, β_j and b_j are identical. Thus b_j can be taken to be any value from the interval, $[B_{low}^j, B_{up}^j]$. We can resolve any non-uniqueness by simply taking $b_j = \frac{1}{2}(B_{low}^j + B_{up}^j)$. Note that the KKT conditions in (15), coming from the additive constraints in (4) we introduced in Shashua and Levin's formulation, enforce $B_{low}^{j-1} \le B_{low}^j$ and $B_{up}^{j-1} \le B_{up}^j$ at the solution, which guarantee that the thresholds specified in these feasible regions will satisfy the inequality constraints $b_{j-1} \le b_j$; without the constraints in (4), the thresholds might be disordered at the solution!

2.3. SMO Algorithm

In this section we adapt the SMO algorithm (Platt, 1999; Keerthi et al., 2001) for the solution of (11)–(12). The key idea of SMO consists of starting with a valid initial point and optimizing only one pair of variables at a time while fixing all the other variables. The suboptimization problem of the two active variables can be solved analytically. Table 1 presents an outline of the SMO implementation for our optimization problem.

In order to determine the pair of *active variables* to optimize, we select the *active threshold* first. The index of the *active threshold* is defined as $J = \arg\max_j \{B_{low}^j - B_{up}^j\}$. Let us assume that B_{low}^J and B_{up}^J are actually defined by $b_{low}^{j_o}$ and $b_{up}^{j_u}$ respectively, and that the two multipliers associated with $b_{low}^{j_o}$ and $b_{up}^{j_u}$ are α_o and α_u. The pair of multipliers (α_o, α_u) is optimized from the current point $(\alpha_o^{old}, \alpha_u^{old})$ to reach the new point, $(\alpha_o^{new}, \alpha_u^{new})$.

It is possible that $j_o \ne j_u$. In this case, named as *cross update*, more than one equality constraint in (12) is involved in the optimization that may update the

variable set $\boldsymbol{\mu}_a = \{\mu^{\min\{j_o,j_u\}+1}, \ldots, \mu^{\max\{j_o,j_u\}}\}$, a subset of $\boldsymbol{\mu}$. In the case of $j_o = j_u$, named as *standard update*, only one equality constraint is involved and the variables of $\boldsymbol{\mu}$ are keep intact, i.e. $\boldsymbol{\mu}_a = \emptyset$. These suboptimization problems can be solved analytically, and the detailed formulas for updating can be found in our longer technical report (Chu & Keerthi, 2005).

3. Implicit Constraints on Thresholds

In this section we present a new approach to support vector ordinal regression. Instead of considering only the empirical errors from the samples of adjacent categories to determine a threshold, we allow the samples in all the categories to contribute errors for each threshold. A very nice property of this approach is that the ordinal inequalities on the thresholds are satisfied automatically at the optimal solution in spite of the fact that such constraints on the thresholds are not explicitly included in the new formulation.

Figure 2 explains the new definition of slack variables $\boldsymbol{\xi}$ and $\boldsymbol{\xi}^*$. For a threshold b_j, the function values of all the samples from all the lower categories, should be less than the lower margin $b_j - 1$; if that does not hold, then $\xi^j_{ki} = \langle \boldsymbol{w} \cdot \phi(x^k_i) \rangle - (b_j - 1)$ is taken as the error associated with the sample x^k_i for b_j, where $k \leq j$. Similarly, the function values of all the samples from the upper categories should be greater than the upper margin $b_j + 1$; otherwise $\xi^{*j}_{ki} = (b_j + 1) - \langle \boldsymbol{w} \cdot \phi(x^k_i) \rangle$ is the error associated with the sample x^k_i for b_j, where $k > j$. Here, the subscript ki denotes that the slack variable is associated with the i-th input sample in the k-th category; the superscript j denotes that the slack variable is associated with the lower categories of b_j; and the superscript $*j$ denotes that the slack variable is associated with the upper categories of b_j.

3.1. Primal Problem

By taking all the errors associated with all $r-1$ thresholds into account, the *primal* problem can be defined as follows:

$$\min_{\boldsymbol{w},\boldsymbol{b},\boldsymbol{\xi},\boldsymbol{\xi}^*} \frac{1}{2}\langle \boldsymbol{w} \cdot \boldsymbol{w} \rangle + C \sum_{j=1}^{r-1} \left(\sum_{k=1}^{j} \sum_{i=1}^{n^k} \xi^j_{ki} + \sum_{k=j+1}^{r} \sum_{i=1}^{n^k} \xi^{*j}_{ki} \right) \tag{18}$$

subject to

$$\begin{aligned} &\langle \boldsymbol{w} \cdot \phi(x^k_i) \rangle - b_j \leq -1 + \xi^j_{ki}, \quad \xi^j_{ki} \geq 0, \\ &\text{for } k = 1, \ldots, j \text{ and } i = 1, \ldots, n^k; \\ &\langle \boldsymbol{w} \cdot \phi(x^k_i) \rangle - b_j \geq +1 - \xi^{*j}_{ki}, \quad \xi^{*j}_{ki} \geq 0, \\ &\text{for } k = j+1, \ldots, r \text{ and } i = 1, \ldots, n^k; \end{aligned} \tag{19}$$

where j runs over $1, \ldots, r-1$. Note that there are $r-1$ inequality constraints for each sample x^k_i (one for each threshold).

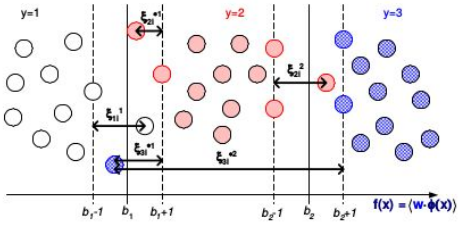


Figure 2. An illustration on the new definition of slack variables $\boldsymbol{\xi}$ and $\boldsymbol{\xi}^*$ that imposes implicit constraints on the thresholds. All the samples are mapped by $\langle \boldsymbol{w} \cdot \phi(x) \rangle$ onto the axis of function values. Note the term ξ^{*1}_{3i} in this graph.

To prove the inequalities on the thresholds at the optimal solution, let us consider the situation where $\boldsymbol{w}$ is fixed and only the b_j's are optimized. Note that the ξ^j_{ki} and ξ^{*j}_{ki} are automatically determined once the b_j are given. To eliminate these variables, let us define, for $1 \leq k \leq r$,

$$\begin{aligned} I^{\text{low}}_k(b) &= \{i \in \{1, \ldots, n^k\} : \langle \boldsymbol{w} \cdot \phi(x^k_i) \rangle - b \geq -1\}, \\ I^{\text{up}}_k(b) &= \{i \in \{1, \ldots, n^k\} : \langle \boldsymbol{w} \cdot \phi(x^k_i) \rangle - b \leq 1\}. \end{aligned}$$

It is easy to see that b_j is optimal iff it minimizes the function

$$\begin{aligned} e_j(b) = &\textstyle\sum_{k=1}^{j} \sum_{i \in I^{\text{low}}_k(b)} (\langle \boldsymbol{w} \cdot \phi(x^k_i) \rangle - b + 1) \\ &+ \textstyle\sum_{k=j+1}^{r} \sum_{i \in I^{\text{up}}_k(b)} (-\langle \boldsymbol{w} \cdot \phi(x^k_i) \rangle + b + 1) \end{aligned} \tag{20}$$

Let $B^\star_j$ denote the set of all minimizers of $e_j(b)$. By convexity, $B^\star_j$ is a closed interval. Given two intervals $B_1 = [c_1, d_1]$ and $B_2 = [c_2, d_2]$, we say $B_1 \leq B_2$ if $c_1 \leq c_2$ and $d_1 \leq d_2$.

Lemma 1. $B^\star_1 \leq B^\star_2 \leq \cdots \leq B^\star_{r-1}$

Proof. The "right side derivative" of e_j with respect to b is

$$g_j(b) = -\textstyle\sum_{k=1}^{j} |I^{\text{low}}_k(b)| + \sum_{k=j+1}^{r} |I^{\text{up}}_k(b)| \tag{21}$$

Take any one j and consider $B^\star_j = [c_j, d_j]$ and $B^\star_{j+1} = [c_{j+1}, d_{j+1}]$. Suppose $c_j > c_{j+1}$. Define $b^\star_j = c_j$ and $b^\star_{j+1} = c_{j+1}$. Since $b^\star_{j+1}$ is strictly to the left of the interval $B^\star_j$ that minimizes e_j, we have $g_j(b^\star_{j+1}) < 0$. Since $b^\star_{j+1}$ is a minimizer of e_{j+1} we also have $g_{j+1}(b^\star_{j+1}) \geq 0$. Thus we have $g_{j+1}(b^\star_{j+1}) - g_j(b^\star_{j+1}) > 0$; also, by (21) we get

$$0 < g_{j+1}(b^\star_{j+1}) - g_j(b^\star_{j+1}) = -|I^{\text{low}}_{j+1}(b^\star_{j+1})| - |I^{\text{up}}_{j+1}(b^\star_{j+1})|$$

which is impossible. In a similar way, $d_j > d_{j+1}$ is also not possible. This proves the lemma.

If the optimal b_j's are all unique,[2] then Lemma 1 implies that the b_j satisfy the natural ordinal ordering. Even when one or more b_j's are non-unique, Lemma 1 says that there exist choices for the b_j's that obey

[2] If, in the primal problem, we regularize the b_j's also (i.e., include the extra cost term $\sum_j b_j^2/2$) then the b_j are guaranteed to be unique. Lemma 1 still holds in this case.

the natural ordering. The fact that the order preservation comes about automatically is interesting and non-trivial, which differs from the PRank algorithm (Crammer & Singer, 2002) where the order preservation on the thresholds is easily brought in via their update rule.

It is also worth noting that Lemma 1 holds even for an extended problem formulation that allows the use of different costs (different C values) for different misclassifications (class k misclassified as class j can have a C_k^j). In applications such as collaborative filtering such a problem formulation can be very appropriate; for example, an A rated movie that is misrated as C may need to be penalized much more than if a B rated movie is misrated as C. Shashua and Levin's formulation and its extension given in section 2 of this paper do not precisely support such a differential cost structure. This is another good reason in support of the implicit problem formulation of the current section.

3.2. Dual Problem

Let $\alpha_{ki}^j \geq 0$, $\gamma_{ki}^j \geq 0$, $\alpha_{ki}^{*j} \geq 0$ and $\gamma_{ki}^{*j} \geq 0$ be the Lagrangian multipliers for the inequalities in (19). Using ideas parallel to those in section 2.1 we can show that the *dual* of (18)–(19) is the following maximization problem that involves only the multipliers $\boldsymbol{\alpha}$ and $\boldsymbol{\alpha}^*$:

$$\max_{\boldsymbol{\alpha},\boldsymbol{\alpha}^*} -\frac{1}{2}\sum_{k,i}\sum_{k',i'}\left(\sum_{j=1}^{k-1}\alpha_{ki}^{*j}-\sum_{j=k}^{r-1}\alpha_{ki}^{j}\right)\left(\sum_{j=1}^{k'-1}\alpha_{k'i'}^{*j} - \sum_{j=k'}^{r-1}\alpha_{k'i'}^{j}\right)\mathcal{K}(x_i^k, x_{i'}^{k'}) + \sum_{k,i}\left(\sum_{j=1}^{k-1}\alpha_{ki}^{*j}+\sum_{j=k}^{r-1}\alpha_{ki}^{j}\right) \tag{22}$$

subject to

$$\begin{aligned} &\sum_{k=1}^{j}\sum_{i=1}^{n^k}\alpha_{ki}^j = \sum_{k=j+1}^{r}\sum_{i=1}^{n^k}\alpha_{ki}^{*j} && \forall j \\ &0 \leq \alpha_{ki}^j \leq C && \forall j \text{ and } k \leq j \\ &0 \leq \alpha_{ki}^{*j} \leq C && \forall j \text{ and } k > j. \end{aligned} \tag{23}$$

The *dual* problem (22)–(23) is a convex quadratic programming problem. The size of the optimization problem is $(r-1)n$ where $n = \sum_{k=1}^{r} n^k$ is the total number of training samples. The discriminant function value for a new input vector x is

$$f(x) = \langle \boldsymbol{w} \cdot \phi(x) \rangle = \sum_{k,i}\left(\sum_{j=1}^{k-1}\alpha_{ki}^{*j}-\sum_{j=k}^{r-1}\alpha_{ki}^{j}\right)\mathcal{K}(x_i^k, x).$$

The predictive ordinal decision function is given by $\arg\min_i\{i : f(x) < b_i\}$.

The ideas for adapting SMO to (22)–(23) are similar to those in section 2.3. The resulting suboptimization problem is analogous to the case of *standard update* in section 2.3 where only one of the equality constraints from (23) is involved. Full details of the derivation of the dual problem as well as the SMO algorithm have been skipped for lack of space. These details are given in our longer technical report (Chu & Keerthi, 2005).

4. Numerical Experiments

We have implemented the two SMO algorithms for the ordinal regression formulations with explicit constraints (EXC) and implicit constraints (IMC),[3] along with the algorithm of Shashua and Levin (2003) for comparison purpose. The function caching technique and the double-loop scheme proposed by Keerthi et al. (2001) have been incorporated in the implementation for efficiency. We begin this section with a simple dataset to illustrate the typical behavior of the three algorithms, and then empirically study the scaling properties of our algorithms. Then we compare the generalization performance of our algorithms against standard support vector regression on eight benchmark datasets for ordinal regression. The following Gaussian kernel was used in these experiments:

$$\mathcal{K}(x, x') = \exp\left(-\tfrac{\kappa}{2}\textstyle\sum_{\varsigma=1}^{d}(x_\varsigma - x'_\varsigma)^2\right) \tag{24}$$

where x_ς denotes the ς-th element of the input vector x. The tolerance parameter τ was set to 0.001 for all the algorithms. We have utilized two evaluation metrics which quantify the accuracy of predicted ordinal scales $\{\hat{y}_1, \ldots, \hat{y}_t\}$ with respect to true targets $\{y_1, \ldots, y_t\}$: a)*Mean absolute error* is the average deviation of the prediction from the true target, i.e. $\frac{1}{t}\sum_{i=1}^{t}|\hat{y}_i - y_i|$, in which we treat the ordinal scales as consecutive integers; b)*Mean zero-one error* is simply the fraction of incorrect predictions.

4.1. Grading Dataset

The grading dataset was used in chapter 4 of Johnson and Albert (1999) as an example of the ordinal regression problem.[4] There are 30 samples of students' score. The "sat-math score" and "grade in prerequisite probability course" of these students are used as input features, and their final grades are taken as the targets. In our experiments, the six students with final grade A or E were not used, and the feature associated with the "grade in prerequisite probability course" was treated as a continuous variable though it had an ordinal scale. In Figure 3 we present the solution obtained by the

[3]The source code (written in ANSI C) of our implementation of the two algorithms can be found at http://www.gatsby.ucl.ac.uk/~chuwei/svor.htm.

[4]The grading dataset is available at http://www.mathworks.com/support/books/book1593.jsp.