The First International Symposium on Optimization and Systems Biology (OSB'07)
Beijing, China, August 8–10, 2007

Second Order Cone Programming Formulations for Robust Support Vector Ordinal Regression Machine*

Zhi-Xia Yang[1,2] Ying-Jie Tian[3] Nai-Yang Deng[1,†]

[1] College of Science, China Agricultural University, 100083, Beijing, China.
[2] College of Mathematics and System Science, Xinjiang University
[3] Chinese Academy of Sciences Research Center on Data Technology & Knowledge Economy

Abstract Support vector ordinal regression machine (SVORM) is an effective method for ordinal regression problem. Up to now, the SVORM implicitly assumes the training data to be known exactly. However, in practice, the training data subject to measurement noise. In this paper, we propose a robust version of SVORM. The robustness of the proposed method is validated by our preliminary numerical experiments.

1 Introduction

Ordinal regression [2, 7, 8] may be viewed as a problem bridging between the two standard machine learning tasks of classification and metric regression. Ordinal regression arises frequently in social science and information retrieval where human preferences play a major role. The training samples are labeled by ranks, which exhibits an ordering among the different categories. In contrast to metric regression problems, these ranks are of finite types and the metric distances between the ranks are not defined. These ranks are also different from the labels of multiple classes in classification problems due to the existence of the ordering information.

There are several approaches to deal with ordinal regression problems in the domain of machine learning, for example, in [1, 2, 7, 8, 9, 10, 11]. But in the mentioned methods, the parameters in the optimization problems are implicitly assumed to be known exactly. However, in real world applications, the parameters are not always known exactly; what have perturbations since they are estimated from the data subject to measurement and statistical errors [4]. Since the solutions to optimization problems are typically sensitive to parameter perturbations, errors in the input parameters tend to get amplified in the decision function, often resulting in far from optimal solutions [12, 13, 16]. So it will be useful to explore formulations that can yield robust discriminants to such estimation errors.

*This work is supported by the National Natural Science Foundation of China (No.10631070, No.70601033, No.10601064).
†Corresponding author. E-mail: dengnaiyang@vip.163.com

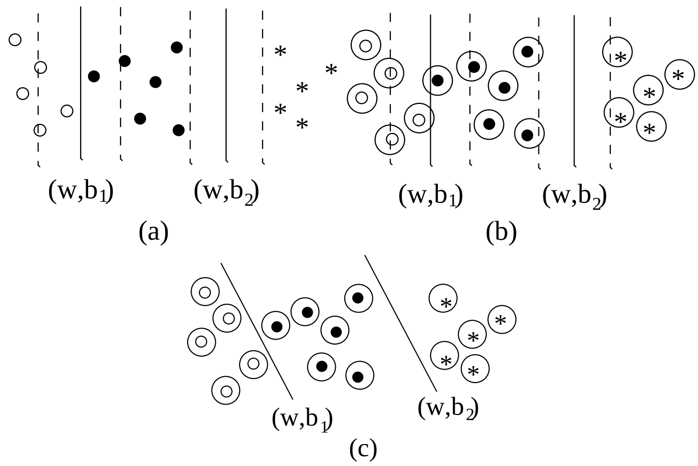

Figure 1: (a) The fixed margin version;(b) The effect of measurement noises;(c) The result of Robust version.

In this paper, we propose a robust formulation of support vector ordinal regression machine (SVORM) based on fixed margin version [1], which is represented as a second order cone program (SOCP) [5, 14].

2 Robust Counterpart of Linear SVORM

Support vector ordinal regression machine (SVORM) is studied in [1, 2]. Now we introduce its fixed margin version.

Suppose a training set is given by

$$T = \{x_i^j\}_{i=1,\cdots,l^j}^{j=1,\cdots,k}, \tag{1}$$

where $x_i^j \in R^n$ is the input, $j = 1,\cdots,k$ denotes the class number and $i = 1,\cdots,l^j$ is the index with each class. We need to find a real value function $g(x)$ and an orderly real sequence $b_1 \leq \cdots \leq b_{k-1}$ and construct a decision

$$f(x) = \min_{r\in\{1,\cdots,k\}} \{r : g(x) - b_r < 0\}, \tag{2}$$

where $b_k = +\infty$.

The training data $x_i^j,\ j = 1,\cdots,k, i = 1,\cdots,l^j$ used in Linear SVORM [1, 3] are implicitly assumed to be known exactly (see Fig.1 (a)). However, in real world applications, the data are corrupted by measurement and statistical errors [4]. Errors in the input space tend to get amplified in the decision function, which often results in misclassification. For example, suppose each training point in Fig.1 (a) is allowed to move in a sphere, the original decision function cannot separate the training set in this case (see Fig.1 (b)) properly. So we explore formulation which can yield robust discriminants to such estimation errors. Intuitively, it is expected to yield a separation lines as show in Fig.1 (c) which separate three classes of perturbed spheres.

Assume that

$$\bar{x}_i^j \in \{x_i^j + \rho_i^j u_i^j : \|u_i^j\| \le 1\},\ j = 1, \cdots, k,\ i = 1, \cdots, l^j, \tag{3}$$

where $\bar{x}_i^j$ is the true value of the training data and $\rho_i^j u_i^j$ is the measurement noise with $u_i^j \in R^n$, $\rho_i^j \ge 0$ being a given constant.

Now we are seeking for a decision function that minimizes the misclassification in the worst case, i.e. one that minimizes the maximum misclassification when samples are allowed to move within their corresponding confidence balls. In this case, the robust counterpart of linear SVORM is given by the following robust optimization problem

$$\min_{w,b,\xi^{(*)}} \quad \frac{1}{2}\|w\|^2 + C\sum_{j=1}^{k}\sum_{i=1}^{l^j}(\xi_i^j + \xi_i^{*j}), \tag{4}$$

$$\text{s.t.} \quad (w \cdot (x_i^j + \rho_i^j u_i^j)) - b_j \le -1 + \xi_i^j, \forall u_i^j \in \mathscr{U}, j = 1, \cdots, k,\ i = 1, \cdots, l^j, \tag{5}$$

$$(w \cdot (x_i^j + \rho_i^j u_i^j)) - b_{j-1} \ge 1 - \xi_i^{*j}, \forall u_i^j \in \mathscr{U}, j = 1, \cdots, k,\ i = 1, \cdots, l^j, \tag{6}$$

$$\xi_i^j \ge 0,\ \xi_i^{*j} \ge 0,\ j = 1, \cdots, k,\ i = 1, \cdots, l^j, \tag{7}$$

where $\mathscr{U} = \{u \in R^n : \|u\| \le 1\}$.

Since

$$\min\{\rho_i^j (w \cdot u_i^j) :\ u_i^j \in \mathscr{U}\} = -\rho_i^j \|w\|, \tag{8}$$

the problem (4)-(7) is equivalent to

$$\min_{w,b,\xi^{(*)}} \quad \frac{1}{2}t^2 + C\sum_{j=1}^{k}\sum_{i=1}^{l^j}(\xi_i^j + \xi_i^{*j}), \tag{9}$$

$$\text{s.t.} \quad -(w \cdot x_i^j) - \rho_i^j t + b_j + \xi_i^j \ge 1,\ j = 1, \cdots, k,\ i = 1, \cdots, l^j, \tag{10}$$

$$(w \cdot x_i^j) - \rho_i^j t - b_{j-1} + \xi_i^{*j} \ge 1,\ j = 1, \cdots, k,\ i = 1, \cdots, l^j, \tag{11}$$

$$\xi_i^j \ge 0,\ \xi_i^{*j} \ge 0,\ j = 1, \cdots, k,\ i = 1, \cdots, l^j, \tag{12}$$

$$\|w\| \le t. \tag{13}$$

By introducing new scalar variables u and v, replacing t^2 in the objective by $u - v$ and requiring that u and v satisfy the linear and second order cone constraints $u + v = 1$ and $\sqrt{t^2 + v^2} \le u$, since the latter imply that $t^2 \le u - v$, problem (9)-(13)

can be reformulated as a following second order cone program (SOCP):

$$\min_{w,b,\xi^{(*)},u,v,t} \quad \frac{1}{2}(u-v)+C\sum_{j=1}^{k}\sum_{i=1}^{l^j}(\xi_i^j+\xi_i^{*j}), \tag{14}$$

$$\text{s.t.} \quad -(w\cdot x_i^j)-\rho_i^j t+b_j+\xi_i^j\geq 1,\ j=1,\cdots,k,\ i=1,\cdots,l^j, \tag{15}$$

$$(w\cdot x_i^j)-\rho_i^j t-b_{j-1}+\xi_i^{*j}\geq 1,\ j=1,\cdots,k,\ i=1,\cdots,l^j, \tag{16}$$

$$\xi_i^j\geq 0,\ \xi_i^{*j}\geq 0,\ j=1,\cdots,k,\ i=1,\cdots,l^j, \tag{17}$$

$$\|w\|\leq t, \tag{18}$$

$$u+v=1, \tag{19}$$

$$\sqrt{t^2+v^2}\leq u, \tag{20}$$

where $b=(b_1,\cdots,b_{k-1})^T$, $b_0=-\infty$, $b_k=+\infty$, $\xi^{(*)}=(\xi_1^1,\cdots,\xi_{l^1}^1,\cdots,\xi_1^k,\cdots,\xi_{l^k}^k,$ $\xi_1^{*1},\cdots,\xi_{l^1}^{*1},\cdots,\xi_1^{*k},\cdots,\xi_{l^k}^{*k})$, the penalty parameter $C>0$.

3 Robust Linear SVORM Algorithm

In order to get the solution w,b of the SOCP problem (14)-(20), we usually solve its dual problem and express w,b by the solution of the dual problem.

Writing problem (14)-(20) into the standard form and using the definition of the dual of the standard form SOCP problem [5], the dual problem can be written as:

$$\max_{\alpha^{(*)},\beta,\gamma,z_u,z_v} \quad \sum_{j=1}^{k}\sum_{i=1}^{l^j}(\alpha_i^j+\alpha_i^{*j})+\beta, \tag{21}$$

$$\text{s.t.} \quad \gamma\leq\sum_{j,i}\rho_i^j(\alpha_i^j+\alpha_i^{*j})-\sqrt{\sum_{j,i}\sum_{j',i'}(\alpha_i^{*j}-\alpha_i^j)(\alpha_{i'}^{*j'}-\alpha_{i'}^{j'})(x_i^j\cdot x_{i'}^{j'})}, \tag{22}$$

$$\sqrt{\gamma^2+z_v^2}\leq z_u, \tag{23}$$

$$\beta+z_v=-\frac{1}{2},\beta+z_u=\frac{1}{2}, \tag{24}$$

$$\sum_{i=1}^{l^j}\alpha_i^j=\sum_{i=1}^{l^{j+1}}\alpha_i^{*j+1},\ j=1,2,\cdots,k-1, \tag{25}$$

$$0\leq\alpha_i^j,\alpha_i^{*j}\leq C, j=1,2,\cdots,k, i=1,2,\cdots,l^j, \tag{26}$$

where $\alpha^{(*)}=(\alpha_1^1,\cdots,\alpha_{l^1}^1,\cdots,\alpha_1^k,\cdots,\alpha_{l^k}^k,\alpha_1^{*1},\cdots,\alpha_{l^1}^{*1},\cdots,\alpha_1^{*k},\cdots,\alpha_{l^k}^{*k})^T$, $\alpha_i^{*1}=0$, $i=1,2,\cdots,n^1$, $\alpha_i^k=0$, $i=1,2,\cdots,n^k$.

Using the complementary equations at optimality about second order cone constraint, the solution w,b of the primal problem (14)-(20) can be expressed by the solution $\alpha^{(*)},\beta,\gamma,z_u,z_v$ of the dual problem (21)-(26). Because of the limitation of the space, the derivation is omitted and the final formulation is given in step 4 and step 5 in the following algorithm:

Algorithm 1: Robust Linear SVORM (R-LSVORM)

1. Given a training set (1);
2. Select $C > 0$;
3. Solve the dual problem (21)-(26) and get its solution $\alpha^{(*)} = (\alpha_1^1, \cdots, \alpha_{l^1}^1, \cdots, \alpha_1^k, \cdots, \alpha_{l^k}^k, \alpha_1^{*1}, \cdots, \alpha_{l^1}^{*1}, \cdots, \alpha_1^{*k}, \cdots, \alpha_{l^k}^{*k})^T, \beta, \gamma, z_u, z_v$;
4. Compute

$$g(x) = \frac{\gamma}{(z_v - z_u)(\sum\limits_{j=1}^{k}\sum\limits_{i=1}^{l^j} \rho_i^j(\alpha_i^j + \alpha_i^{*j}) - \gamma)} \sum_{j=1}^{k}\sum_{i=1}^{l^j}(\alpha_i^{*j} - \alpha_i^j)(x_i^j \cdot x); \tag{27}$$

5. For $j = 1, \cdots, k-1$, execute the following steps:
   5.1 Choose some component $\alpha_i^j \in (0, C)$ in $\alpha^{(*)}$. If we get such i, let

$$b_j = 1 + \sum_{j'=1}^{k}\sum_{i'=1}^{l^{j'}}(\alpha_{i'}^{*j'} - \alpha_{i'}^{j'})(x_{i'}^{j'} \cdot x_i^j) + \rho_i^j t, \tag{28}$$

   otherwise go to 5.2;
   5.2 Choose some component $\alpha_i^{*j+1} \in (0, C)$ in $\alpha^{(*)}$. If we get such i, let

$$b_j = \sum_{j'=1}^{k}\sum_{i'=1}^{l^{j'}}(\bar{\alpha}_{i'}^{*j'} - \bar{\alpha}_{i'}^{j'})(x_{i'}^{j'} \cdot x_i^{j+1}) - \rho_i^{j+1} - 1, \tag{29}$$

   otherwise go to 5.3;
   5.3 Let

$$b_j = \frac{1}{2}(b_j^{dn} + b_j^{up}), \tag{30}$$

   where

$$\begin{aligned} b_j^{dn} &= \max\{\max_{i \in I_1^j}(g(x_i^j) + \rho_i^j t + 1), \max_{i \in I_4^j}(g(x_i^{j+1}) - \rho_i^{j+1} t - 1)\}, \\ b_j^{up} &= \min\{\min_{i \in I_3^j}(g(x_i^j) + \rho_i^j t + 1), \min_{i \in I_2^j}(g(x_i^{j+1}) - \rho_i^{j+1} t - 1)\}, \end{aligned}$$

   where

$$\begin{aligned} &I_1^j = \{i \in \{1, \cdots, l^j\} | \alpha_i^j = 0\},\ I_2^j = \{i \in \{1, \cdots, l^{j+1}\} | \alpha_i^{*j+1} = 0\}, \\ &I_3^j = \{i \in \{1, \cdots, l^j\} | \alpha_i^j = C\},\ I_4^j = \{i \in \{1, \cdots, l^{j+1}\} | \alpha_i^{*j+1} = C\}; \end{aligned}$$

6. If there exists $j \in \{1, \cdots, k\}$ such that $b_j \leq b_{j-1}$, the algorithm stop or go to 2;
7. Define $b_k = +\infty$, construct the decision function

$$f(x) = \min_{r \in \{1, \cdots, k\}}\{r : g(x) - b_r < 0\}. \tag{31}$$

4 Robust Counterpart of Nonlinear SVORM

The above discussion is restricted in the linear case. In this section, we will analyze nonlinear SVORM by introducing Gaussian kernel function $K(x,x') = \exp(\frac{-\|x-x'\|^2}{2\sigma^2})$ with a real parameter σ.

Consider the training data (3) and assume that Φ is the transformation corresponding to the Gaussian kernels. It can be shown that

$$\Phi(\bar{x}_i^j) = \Phi(x_i^j) + \bar{\rho}_i^j \bar{u}_i^j,\ \bar{u}_i^j \in \bar{\mathscr{U}}, \tag{32}$$

where $\bar{\mathscr{U}}$ is a unit sphere in Hilbert space and $\bar{\rho}_i^j = (2 - 2\exp(-(\rho_i^j)^2/2\sigma^2))^{\frac{1}{2}}$.

This leads to the following algorithm

Algorithm 2: Robust SVORM (R-SVORM)

1. Given a training set (1);
2. Select $C > 0$ and a kernel parameter σ;
3. Solve the dual problem

$$\max_{\alpha^{(*)},\beta,\gamma,z_u,z_v} \quad \sum_{j=1}^{k}\sum_{i=1}^{l^j}(\alpha_i^j + \alpha_i^{*j}) + \beta, \tag{33}$$

$$\text{s.t.} \quad \gamma \le \sum_{j,i}\bar{\rho}_i^j(\alpha_i^j + \alpha_i^{*j}) - \sqrt{\sum_{j,i}\sum_{j',i'}(\alpha_i^{*j} - \alpha_i^j)(\alpha_{i'}^{*j'} - \alpha_{i'}^{j'})K(x_i^j, x_{i'}^{j'})}, \tag{34}$$

$$\sqrt{\gamma^2 + z_v^2} \le z_u, \tag{35}$$

$$\beta + z_v = -\frac{1}{2}, \beta + z_u = \frac{1}{2}, \tag{36}$$

$$\sum_{i=1}^{l^j}\alpha_i^j = \sum_{i=1}^{l^{j+1}}\alpha_i^{*j+1},\ j = 1,2,\cdots,k-1, \tag{37}$$

$$0 \le \alpha_i^j, \alpha_i^{*j} \le C, j = 1,2,\cdots,k, i = 1,2,\cdots,l^j, \tag{38}$$

   where $\alpha^{(*)} = (\alpha_1^1,\cdots,\alpha_{l^1}^1,\cdots,\alpha_1^k,\cdots,\alpha_{l^k}^k,\alpha_1^{*1},\cdots,\alpha_{l^1}^{*1},\cdots,\alpha_1^{*k},\cdots,\alpha_{l^k}^{*k})^T$, $\alpha_i^{*1} = 0,\ i = 1,2,\cdots,n^1$, $\alpha_i^k = 0,\ i = 1,2,\cdots,n^k$. and get its solution $\alpha^{(*)}, \beta$, γ, z_u, z_v;
4. Compute

$$g(x) = \frac{\gamma}{(z_v - z_u)(\sum_{j=1}^{k}\sum_{i=1}^{l^j}\bar{\rho}_i^j(\alpha_i^j + \alpha_i^{*j}) - \gamma)}\sum_{j=1}^{k}\sum_{i=1}^{l^j}(\alpha_i^{*j} - \alpha_i^j)K(x_i^j, x); \tag{39}$$

   where $\bar{\rho}_i^j = (2 - 2\exp(-(\rho_i^j)^2/2\sigma^2))^{\frac{1}{2}}$.
5. For $j = 1,\cdots,k-1$, execute the following steps:

   5.1 Choose some component $\alpha_i^j \in (0,C)$ in $\alpha^{(*)}$. If we get such i, let

$$b_j = 1 + \sum_{j'=1}^{k}\sum_{i'=1}^{l^{j'}}(\alpha_{i'}^{*j'} - \alpha_{i'}^{j'})K(x_{i'}^{j'}, x_i^j) + \bar{\rho}_i^j t, \tag{40}$$

otherwise go to 5.2;

5.2 Choose some component $\alpha_i^{*j+1} \in (0,C)$ in $\alpha^{(*)}$. If we get such i, let

$$b_j = \sum_{j'=1}^{k} \sum_{i'=1}^{l^{j'}} (\bar{\alpha}_{i'}^{*j'} - \bar{\alpha}_{i'}^{j'}) K(x_{i'}^{j'}, x_i^{j+1}) - \bar{\rho}_i^{j+1} - 1, \quad (41)$$

otherwise go to 5.3;

5.3 Let

$$b_j = \frac{1}{2}(b_j^{dn} + b_j^{up}), \quad (42)$$

where

$$b_j^{dn} = \max\{\max_{i \in I_1^j}(g(x_i^j) + \bar{\rho}_i^j t + 1), \max_{i \in I_4^j}(g(x_i^{j+1}) - \bar{\rho}_i^{j+1} t - 1)\},$$
$$b_j^{up} = \min\{\min_{i \in I_3^j}(g(x_i^j) + \bar{\rho}_i^j t + 1), \min_{i \in I_2^j}(g(x_i^{j+1}) - \bar{\rho}_i^{j+1} t - 1)\},$$

where

$$I_1^j = \{i \in \{1, \cdots, l^j\} | \alpha_i^j = 0\},\ I_2^j = \{i \in \{1, \cdots, l^{j+1}\} | \alpha_i^{*j+1} = 0\},$$
$$I_3^j = \{i \in \{1, \cdots, l^j\} | \alpha_i^j = C\},\ I_4^j = \{i \in \{1, \cdots, l^{j+1}\} | \alpha_i^{*j+1} = C\};$$

6. If there exists $j \in \{1, \cdots, k\}$ such that $b_j \leq b_{j-1}$, the algorithm stop or go to 2;
7. Define $b_k = +\infty$, construct the decision function

$$f(x) = \min_{r \in \{1, \cdots, k\}} \{r : g(x) - b_r < 0\}. \quad (43)$$

Note that we get the original SVORM algorithm by choosing $\rho_i^j = 0, j = 1, \cdots, k$, $i = 1, \cdots, l^j$.

5 Preliminary Numerical Results

Our numerical experiments follow the approach in [16]. In fact, following [15, 11], the ordinal regression problems are obtained from the regression problems in [15] by discretizing their output values. Due to the time consuming, only 4 regression problems are selected among the 29 ones in [15]. These 4 problems are the smallest ones according to the number of the training points.

In order to test our algorithms, the measurement error (3) in the training points is introduced, where, for simplicity, ρ_i^j is assumed to be a constant independent of i and j. On the other hand, for the test point, there is also a perturbation around the attribute x_i^j generated by $\bar{x}_i^j = x_i^j + \rho_i^j u_i^j$, where ρ is the same constant with that in the training point and the noise u_i^j is generated randomly from the normal distribution and scaled on the unit sphere.

The parameters in both Algorithm 1 and Algorithm 2 are chosen by ten-fold cross validation: in Algorithm 1, for data sets "Diabetes" and "Triazine" $C = 1000$,

Table 1: For Algorithm 1, the percentage of tenfold testing error for datasets with noise.

Dataset	Instances	Dimension	Model	ρ				
				0.1	0.2	0.3	0.4	0.5
Diabetes	43	2	R-LSVORM	0.4884	0.4651	0.4651	0.4651	0.4651
			LSVORM	0.4884	0.4884	0.4884	0.4884	0.4884
Pyrimidines	74	27	R-LSVORM	0.5811	0.5811	0.5811	0.5811	0.5811
			LSVORM	0.5946	0.5811	0.6081	0.6351	0.7027
Triazines	186	60	R-LSVORM	0.5215	0.5215	0.5215	0.5215	0.5215
			LSVORM	0.5376	0.6183	0.7366	0.6882	0.7258
Wisconsin	194	32	R-LSVORM	0.7320	0.7113	0.6495	0.6804	0.6959
			LSVORM	0.7320	0.7165	0.7423	0.7629	0.7216

Table 2: For Algorithm 2, the percentage of tenfold testing error for datasets with noise.

Dataset	Model	ρ				
		0.1	0.2	0.3	0.4	0.5
Diabetes	R-SVORM	0.4651	0.4651	0.4651	0.4651	0.4651
	SVORM	0.5349	0.5349	0.5581	0.4884	0.5116
Pyrimidines	R-SVORM	0.4865	0.5811	0.5811	0.5811	0.5946
	SVORM	0.4865	0.5811	0.6351	0.6892	0.6081
Triazines	R-SVORM	0.5215	0.5215	0.5215	0.5215	0.5215
	SVORM	0.5323	0.6613	0.7581	0.8280	0.8333
Wisconsin	R-SVORM	0.7216	0.7474	0.7216	0.7216	0.7216
	SVORM	0.7990	0.7784	0.7784	0.7784	0.7784

for data sets " Pyrimidines" $C = 10$, for data sets "Wisconsin" $C = 10000$; in Algorithm 2, for four data sets, the penalty parameters are the same $C = 1000$; the kernel parameters, for data sets "Diabetes" and "Triazines" $\sigma = 1$, for data sets " Pyrimidines" $\sigma = 4$, for data sets "Wisconsin" $\sigma = 0.25$. The numerical results for Algorithm 1 (R-LSVORM) and Algorithm 2(R-SVORM) are given in Table 1 and Table 2 respectively. What we are concerned here is the percentage of tenfold testing error, which are shown with different noise level ρ. In addition, for comparison, the results corresponding to the original LSVORM and SVORM are also listed in these tables. It can be seen that the performance of our Robust SVORM is better. All of the results on the percentages of tenfold testing error are comparable to that in [15].

References

[1] A. Shashua and A. Levin. Taxonomy of large margin principle algorithms for ordinal regression problems. Technical Report 2002-39. Leivniz Center for Research School of Computer Science and Eng., the hebrew University of Jerusaalem.

[2] R. Herbrich, T.Graepel, and K. Obermayer. Large margin rank boundaries for ordinal regression. Advances in Large Margin Classifiers, 2000, 115-132.

[3] Z.X. Yang, N.Y. Deng and Y.J. Tian. A multi-class classification algorithm based on ordinal regression machine. International Conference on CIMCA2005 & IAWTIC 2005, Vienna, Austria, 2005, 2, 810-814.

[4] D. Goldfarb, G. Iyengar. Robust portfolio selection problems. Math.Oper.Res. 28(1), 2003.

[5] F. Alizadeh, D. Goldfarb. Second order cone programming. Math.Program.,Ser.B 95:3-51, 2003.

[6] J. Sturm. Using SeDuMi, a matlab toolbox for optimization over symmetric cones. Department of Ecnometrics, Tilburg University, The Netherlands.

[7] W.W. Cohen, R.E. Schapire and Y. Singer. Learning to order things. Journal of Articial Intelligence Research (JAIR), 10:243-270, 1999.

[8] K. Crammer, Y. Singer. Pranking with ranking. In Proceedings of the Conference on Neural Information Processing Systems (NIPS), 2001.

[9] S. Har-Peled, D. Roth and D. Zimak. Constraint classification: A new approach to multiclass classification and ranking. In Advances in Neural Information Processing Systems 15, 2002.

[10] S. Kramer, G. Widmer, B. Pgahringer and M. DeGroeve. Prediction of ordinal classes using regression trees. Fundamenta Information, 47: 1-13, 2001.

[11] W. Chu, S.S. Keerthi. New approach to support vector ordinal regression. Appearing in Proceedings of the 22nd International Conference of on Machine Learning. Bonn. Germany. August 07-11. 2005. 145-152.

[12] A. Ben-Tal, A. Nemirovski. Robust convex optimization. Math. Oper. Res. 23(4), 769-805, 1998.

[13] V.K. Chopra, W.T. Ziemba. The effect of errors in means, variances and covariances on optimal protfolio choice. J. Portfolio Manag., Winter, 1993, 6-11.

[14] M.S. Lobo, L. Vandenberghe, S. boyd and H. Lébret. Applications of second-order cone programming. Linear Algebra and Applications, 284, 193-228.

[15] E.Frank, M. Hall. A smiple approach to ordinal classification. Lecture Notes in Computer Science, Springer, 2167, 145-156, 2001.

[16] P. Zhong, M. Fukushima. Second order cone programming formulations for robust multi-class classification. Neural Computation, 19(1): 258-282, 2007.