

Dimension Reduction for Object Ranking

Toshihiro Kamishima and Shotaro Akaho

National Institute of Advanced Industrial Science and Technology (AIST),
AIST Tsukuba Central 2, Umezono 1-1-1, Tsukuba, Ibaraki, 305-8568 Japan,
mail@kamishima.net(<http://www.kamishima.net/>), s.akaho@aist.go.jp

Abstract. Ordered lists of objects are widely used as representational forms. Such ordered objects include Web search results and best-seller lists. Techniques for processing such ordinal data are being developed, particularly methods for an object ranking task: i.e., learning functions used to sort objects from sample orders. In this article, we propose two dimension reduction methods specifically designed to improve prediction performance in an object ranking task.

1 Introduction

Orders are sequences of objects sorted according to some property and are widely used to represent data. For example, responses from Web search engines are lists of pages sorted according to their relevance to queries. Best-seller lists, which are item sequences sorted according to sales volume, are used on many E-commerce sites. Processing techniques for orders have immediate practical value, and so research concerning orders has become very active in recent years. In particular, several methods are being developed for learning functions used to sort objects represented by attribute vectors from example orders. We call this task *Object Ranking* [1, 2] and emphasize its usefulness for sensory tests¹[2, 3], information retrieval [4–8], and recommendation [9].

Several methods have been developed for the object ranking task. However, when the attribute vectors that describe objects are in very high dimensional space, these object ranking methods are degraded in prediction performance. The main reason for this is that the number of model parameters to be learned grows in accordance with the increase of dimensionality; thus, the acquired functions might not perform well when sorting unseen objects due to over-fitting.

Dimension reduction techniques are one obvious solution to the problems caused by high dimensionality. Dimension reduction is the task of mapping points originally in high dimensional space to a lower dimensional sub-space, while limiting the amount of lost information. Principal component analysis (PCA) is one of the typical techniques for dimension reduction. PCA is designed so that variations in original data are preserved as much as possible. It has been successfully used for other learning tasks but is less appropriate for an object ranking task. Since PCA is designed so as to preserve information regarding the objects

¹ measurement of respondents' sensations, feelings or impressions

themselves, useful information in terms of the target ordering might be lost by this approach. Therefore, in this paper, we propose *Rank Correlation Dimension Reduction (RCDR)* for dimension reduction in conjunction with object ranking. RCDR is designed to preserve information that is useful for mapping to the target ordering.

We propose our RCDR methods in Section 2. Experimental results are shown in Section 3. We discuss and summarize the results in Section 4.

2 Rank Correlation Dimension Reduction

In the previous section, we defined an object ranking task. Here, we show a dimension reduction technique specially designed for these object ranking methods. Note that the notations and the task of object ranking are described in our survey chapter of this book [1].

To obtain satisfactory results when using data mining or machine learning algorithms, it is important to apply pre-processing methods, such as feature selection, dealing with missing values, or dimension reduction. Appropriate pre-processing of data can improve prediction performance, and can occasionally reduce computational and/or memory costs. Some pre-processing techniques for mining or learning methods dealing with orders have been proposed. Bahamonde et al. [10] applied wrapper-type feature selection to an object ranking task. Slotta et al. [11] performed feature selection for classification of orders. In [12, 13], rank statistics were used for selecting informative genes from microarray data. To measure the similarities between orders, Kamishima and Akaho proposed a method to fill in missing objects in orders [14]. To our knowledge, however, dimension reduction techniques specially designed for an object ranking task have not yet been developed.

Similar to other types of learning tasks, such as classification or regression, dimension reduction techniques will be beneficial for object ranking tasks, in particular, if the number of attributes, K , is very large. With reduced dimensions, the generalization ability can be improved. Because the number of model parameters to be learned grows in accordance with K , the acquired functions might not perform well when sorting unseen objects due to over-fitting. In particular, if there are many non-informative attributes or if complex models are used, the problem of over-fitting will be alleviated by reducing dimensions.

To reduce the number of dimensions before performing object ranking, one might assume that reduction techniques used for other learning tasks can be used. However, this is not the case. Principal component analysis (PCA) is one of typical techniques for dimension reduction. PCA is designed so that information about data in original attribute vector space is preserved as much as possible. This approach is less appropriate for an object ranking task. Specifically, because an object ranking task must find a mapping from attribute vectors to the target ordering, it is not sufficient to preserve information only in source vectors. On the other hand, Diaconis' spectral analysis [15] for orders is another possibility. This is a technique to decompose distributions of orders into sub-components.

For example, first-order components represent the frequency that the object \mathbf{x}_j is l -th ranked, while second-order components represent the frequency that objects \mathbf{x}_j and \mathbf{x}_k are l -th and m -th ranked, respectively. However, our goal is not to find decomposition in an ordinal space, but to find a sub-space in an attribute vector space.

From the above discussion, it should be clear that we had to develop reduction techniques that preserve information about mappings from attribute vectors to the target ordering. This is analogous to Fisher’s discriminant analysis, which is a dimension reduction technique to preserve information about a mapping from an attribute vector to target classes.

Table 1. Computational complexities of object ranking algorithms

Cohen	RankBoost	SVOR	Order SVM	ERR
$N\bar{L}^2K$	$N\bar{L}^2K$	$N^2\bar{L}^4K$	$N^2\bar{L}^4K$	$N\bar{L}K^2$

NOTE: \bar{L} : the mean length of sample orders, N : the number of samples, and K : the dimension of attribute vectors. The number of ordered pairs and objects in S are approximated by $N\bar{L}^2$ and $N\bar{L}$, respectively. The SVM’s learning time is assumed to be quadratic in the number of training samples. The learning complexities of Cohen’s method or the RankBoost are as above if the number of iterations is constant. However, in practical use, because the number of iterations should be increased adaptively in accordance with the number of ordered pairs, their time complexities approach $N^2\bar{L}^4k$.

Additionally, the computational cost for reducing dimensions should not be much higher than that for object ranking methods. Computational complexities of object ranking methods in the learning stage are summarized in Table 1. These columns show the complexities for Cohen’s method [4], RankBoost [9], Order SVM [16], Support Vector Ordinal Regression (SVOR) [5] (also known as RankingSVM [6]), and Expected Rank Regression (ERR) [2]. You can find a summary of these methods in [1]. We assume that the number of ordered pairs and objects in S are approximated by $N\bar{L}^2$ and $N\bar{L}$, respectively (\bar{L} is the mean length of the sample orders). The SVM’s learning time is assumed to be quadratic in the number of training samples. The learning time of Cohen’s method or the RankBoost is linear in terms of $N\bar{L}^2$, if the number of iterations is constant. However, in practical use, the number of iterations should be increased adaptively. In the experiment in [9], the number of iterations was linearly increased in accordance with the number of ordered pairs, $N\bar{L}^2$. Therefore, their time complexities approach $N^2\bar{L}^4k$. When dimension reduction methods require much higher computational costs than those in Table 1, the reduction of dimensions greatly lessens scalability.

Taking into account what is mentioned above, our dimension reduction methods should satisfy two requirements.

1. It must be designed so as to preserve information about mappings from object attributes to targeting orders.
2. The computational complexity for dimension reduction should not be much larger than that for object ranking algorithms.

To fill these requirements, we propose *Rank Correlation Dimension Reduction (RCDR)*. Given a basis that consists of l vectors, the next $l+1$ vector is selected so as to preserve as much information about target ordering as possible. By repeating this procedure, we obtain the final sub-space.

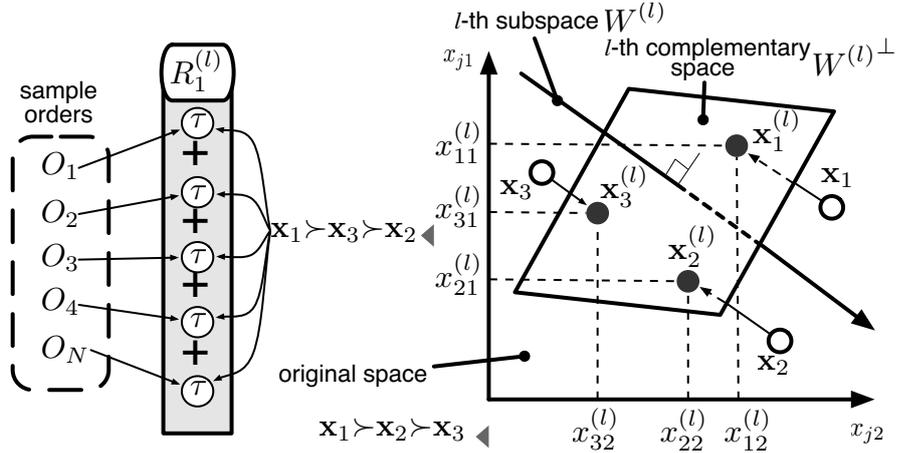


Fig. 1. An outline of rank correlation dimension reduction method

First, we outline our RCDR method. Let $\mathbf{w}^{(l)}$ be the l -th vector of a basis. The sub-space spanned by the basis, $\{\mathbf{w}^{(1)}, \dots, \mathbf{w}^{(l)}\}$, is called the l -th sub-space. We represent this sub-space by the matrix, $W^{(l)} = [\mathbf{w}^{(1)}, \dots, \mathbf{w}^{(l)}]$. Let $W^{(l)\perp}$ be the complementary space of the $W^{(l)}$, that is spanned by $(K-l)$ vectors which are orthogonal to all vectors in the basis, $\{\mathbf{w}^{(1)}, \dots, \mathbf{w}^{(l)}\}$. We are given sample orders S and attribute vectors, $\{\mathbf{x}_j\}$, and the basis of the l -th sub-space. This condition is depicted in Figure 1. The objects in the original K -dimensional spaces (marked by “o” in Figure 1) are projected to the complementary space $W^{(l)\perp}$ of the l -th sub-space. The projected objects (marked by “•” in Figure 1) are denoted by $\mathbf{x}_j^{(l)}$, and $\mathbf{x}_j^{(0)} \equiv \mathbf{x}_j$. By this projection, we can eliminate information about the target ordering contained in the sub-space $W^{(l)}$. For each $k = 1, \dots, K$, objects are sorted in descending order of the k -th attribute values of the objects projected to $W^{(l)\perp}$. In Figure 1, examples of those orders are $\mathbf{x}_1 \succ \mathbf{x}_3 \succ \mathbf{x}_2$ in the first attribute and $\mathbf{x}_1 \succ \mathbf{x}_2 \succ \mathbf{x}_3$ in the second attribute. The rank correlations between each of these orders and each sample order are calculated. Then, the sum of these rank correlations are denoted by $R_k^{(l)}$ (strict

definition will be given later). This $R_k^{(l)}$ represents the concordance between the target ordering and the k -th attribute values of the objects projected on the l -th complementary space. A new vector, $\mathbf{w}^{(l+1)}$, is chosen so that each element of this vector, $w_k^{(l+1)}$, is as proportional to the corresponding concordance, $R_k^{(l)}$, as possible.

Now, we formally describe our RCDR. Let $\mathbf{w}^{(l)} = [w_1^{(l)}, w_2^{(l)}, \dots, w_K^{(l)}]^\top$ be the l -th vector of a basis. These vectors are orthonormal to each other, i.e., $\mathbf{w}^{(l)\top} \mathbf{w}^{(m)} = 0, l \neq m$ and $\|\mathbf{w}^{(l)}\| = 1$. The dimension of the final sub-space is denoted by K' . We are given a set of sample orders $S = \{O_1, \dots, O_N\}$, the basis of the l -th sub-space, $W^{(l)}$, and the objects $\{\mathbf{x}_j | \mathbf{x}_j \in \mathcal{X}_S\}$, $\mathcal{X}_S \equiv \cup_{O_i \in S} \mathcal{X}_i$. From these, we derive the $(l+1)$ -th vector, $\mathbf{w}^{(l+1)}$, as follows. First, we define $R_1^{(l)}, \dots, R_K^{(l)}$ as the concordances between sample orders and the attribute values of the objects projected on the complementary space, $W^{(l)\perp}$. Let us focus on the sample order O_i and the k -th attribute values of objects. Because the goal of an object ranking task is to estimate the orders of objects, the relative ordering of attribute values is more important than the attribute values themselves. We therefore sort the k -th attribute values $x_{jk}^{(l)}$ of all objects $\mathbf{x}_j \in \mathcal{X}(O_i)$ in descending order, where $x_{jk}^{(l)}$ denotes the k -th attribute value of the object, $\mathbf{x}_j^{(l)}$ projected on the l -th complementary space. Note that the projected objects are represented as $[x_{j1}^{(l)}, \dots, x_{jK}^{(l)}]^\top$ on the coordinates of the original space. The resultant order is denoted by $O(\mathcal{X}_i, x_{jk}^{(l)})$. Because both this $O(\mathcal{X}_i, x_{jk}^{(l)})$ and the sample order O_i consist of the same set of objects, the concordance between these two orders can be measured by Kendall's τ . Such rank correlations are calculated between the k -th attribute values and each of sample orders in S , and these correlations are summed up:

$$R_k^{(l)} = \sum_{O_i \in S} \tau(O_i, O(\mathcal{X}_i, x_{jk}^{(l)})). \quad (1)$$

We use this sum as a measure of the concordance between the k -th attribute values of objects and the target ordering. Next, to fill the first requirement of the RCDR, the $(l+1)$ -th vector is chosen so that the above concordance is preserved as much as possible. Let us consider the vector,

$$\mathbf{R}^{(l)} = [R_1^{(l)}, \dots, R_K^{(l)}]^\top.$$

Because the elements of this vector are the concordances between attribute values and the target ordering, this vector would point in the direction that preserves information about the target ordering in the attribute space. Therefore, we choose the vector $\mathbf{w}^{(l+1)}$ so that it maximizes the cosine between $\mathbf{w}^{(l+1)}$ and $\mathbf{R}^{(l)}$ in the complementary space, $W^{(l)\perp}$. Further, the vector $\mathbf{R}^{(l)}$ is constant, and $\|\mathbf{w}^{(l+1)}\| = 1$; thus, the maximization of this cosine is equivalent to the $\mathbf{w}^{(l+1)}$.

This optimization problem is formalized as follows:

$$\begin{aligned} \mathbf{w}^{(l+1)} &= \arg \max_{\mathbf{w}} \mathbf{w}^\top \mathbf{R}^{(l)}, \\ \text{subject to: } & \|\mathbf{w}^{(l+1)}\|=1, \mathbf{w}^{(l+1)\top} \mathbf{w}^{(m)} = 0, m=1, \dots, l. \end{aligned} \quad (2)$$

Note that one might think that $\mathbf{w}^{(l)}$ becomes a zero vector, if $l \geq 2$, but this is not the case. When the performing standard regression and Pearson's correlation is maximized, $\mathbf{w}^{(l)}$ would be a zero vector for $l \geq 2$. This is because zero Pearson's correlation implies such orthogonality in the attribute space. However, because rank correlation doesn't imply orthogonality, $\mathbf{w}^{(l)}$ is generally a non-zero vector even if $l \geq 2$.

Inputs: $S = \{O_1, \dots, O_N\}$: a sample order set
 $\mathbf{x}_j \in \mathcal{X}_S \equiv \cup_{O_i \in S} \mathcal{X}_i$: attribute value vectors
 K' : the dimension of sub-space

- 1: $\mathbf{x}_j^{(0)} \equiv \mathbf{x}_j$
- 2: **for all** l **in** $0, \dots, (K' - 1)$ **do**
- 3: compute $\mathbf{R}^{(l)}$ s.t. $R_k^{(l)} = \sum_{O_i \in S} \tau(O_i, O(\mathcal{X}_i, x_{jk}^{(l)}))$
- 4: **if** $l > 0$ **then**
- 5: $W^{(l)} = [\mathbf{w}^{(1)}, \dots, \mathbf{w}^{(l)}]$
- 6: $\mathbf{R}^{(l)\perp} = (I - W^{(l)}W^{(l)\top})\mathbf{R}^{(l)}$
- 7: **else**
- 8: $\mathbf{R}^{(l)\perp} = \mathbf{R}^{(l)}$
- 9: **end if**
- 10: $\mathbf{w}^{(l+1)} = \mathbf{R}^{(l)\perp} / \|\mathbf{R}^{(l)\perp}\|$
- 11: **for all** \mathbf{x}_j **in** \mathcal{X}_S **do**
- 12: $\mathbf{x}_j^{(l+1)} = \mathbf{x}_j^{(l)} - \mathbf{w}^{(l+1)}\mathbf{w}^{(l+1)\top} \mathbf{x}_j^{(l)}$
- 13: **end for**
- 14: **end for**
- 15: **return** $W^{(K')} = [\mathbf{w}^{(1)}, \dots, \mathbf{w}^{(K')}]$

Fig. 2. Kendall rank correlation dimension reduction

Next, we solve Equation (2). The derivation of $\mathbf{w}^{(l+1)}$ can be easily shown by the following procedure: Calculate the vector of the correlations sums, $\mathbf{R}^{(l)}$, project this vector to the l -th complementary space, and normalize the projected vector. Once a new vector is derived, objects in the l -th complementary space, $\mathbf{x}^{(l)}$, are mapped to the new complementary space, and iteratively the next vector can be computed. This algorithm is shown in Figure 2. $\mathbf{R}^{(l)}$ is computed in line 3, projected to the current complementary space in lines 4–9, and normalized in line 10 so that its norm is one. In lines 11–13, the objects in the current complementary space are projected to the new complementary space. Because the concordance is measured by Kendall's τ , we call this method *Kendall RCDR*. The computational complexities of lines, 3, 4–9, 10, and 11–13 are $O(N\bar{L}^2K)$, $O(KK')$, $O(K)$ and $O(N\bar{L}K)$, respectively; thus, the complexity

per one iteration is $O(N\bar{L}^2K)$ (generally $N\bar{L}^2 \gg K'$), and the total complexity is $O(N\bar{L}^2KK')$. As noted before, because the complexity of Cohen’s method and RankBoost practically approaches $O(N^2\bar{L}^4K)$, our Kendall RCDR is strictly faster than object ranking methods except for ERR (see Table 1). In practical use, RCDR is not so slow than ERR, because \bar{L} is generally small. To further save time complexity, we replace Kendall’s τ in line 3 of the algorithm by Spearman’s ρ , because ρ and τ are highly correlated. We call this method *Spearman RCDR*. Because its time complexity is $O(NKK'\bar{L}\log\bar{L})$, this method becomes faster than the ERR method if $K'\log\bar{L} < K$. Therefore, our RCDR methods satisfy the second requirement. Note that the Kendall RCDR is faster than the Spearman RCDR in the special case: $L_i = 2, O_i \in S$. Joachims et al. proposed a method to implicitly collect sample orders whose lengths are two [6]. The Kendall RCDR is useful in such cases.

3 Experiments

After showing a simple example of our RCDR methods, we describe the experimental results for real data sets.

3.1 A Preliminary Experiment

Table 2. Vectors of a Basis derived by our RCDRs and the PCA

the first vector						
method	1	2	3	4	5	
KRCDR	0.70	0.64	0.31	-0.06	-0.06	0.146
SRCDR	0.70	0.64	0.32	-0.06	-0.06	0.173
PCA	0.02	-0.74	0.54	-0.39	0.00	0.393
the second vector						
method	1	2	3	4	5	
KRCDR	-0.27	-0.17	0.93	-0.13	-0.13	0.007
SRCDR	-0.30	-0.15	0.94	-0.05	-0.05	0.007
PCA	-0.06	-0.18	0.39	0.90	0.00	0.213

NOTE: The first to fifth columns of each table show the components of vectors, $\mathbf{w}^{(1)}$ and $\mathbf{w}^{(2)}$. In the last columns, values of $\|\mathbf{R}^{(l)}\|/N$ are shown for the RCDRs and the contribution ratio is shown for the PCA.

To show what is produced by our two RCDR methods, we present a simple example using artificial data. We give the ideal weight vector by $\mathbf{w}^* = [1, 1, 0.5, 0, 0]$, and set the dimensions of the original space as $K = 5$ and the

number of objects as $|\mathcal{X}^*| = 1000$. For each object $\mathbf{x}_j \in \mathcal{X}^*$, the first to the fourth attribute values are randomly generated according to the normal distribution, $N(0, 1)$, while the fifth value is equal to the fourth. We generated 300 sample orders as follows: Five objects were selected uniformly at random from \mathcal{X}^* ; then these objects were sorted in descending order of $\mathbf{w}^{*\top} \mathbf{x}_j$. We applied Kendall RCDR, Spearman RCDR, and PCA to this data set. The first and second vectors are shown in the upper and lower parts of Table 2, respectively. In each row, we show vectors derived by Kendall RCDR, Spearman RCDR, and PCA. The first to the fifth columns show the elements of vectors. In the last column, the norm lengths of the sum vector of rank correlations per sample order, $\|\mathbf{R}^{(l)}\|/N$, are shown for the RCDR cases, and the contribution ratios are shown for the PCA cases.

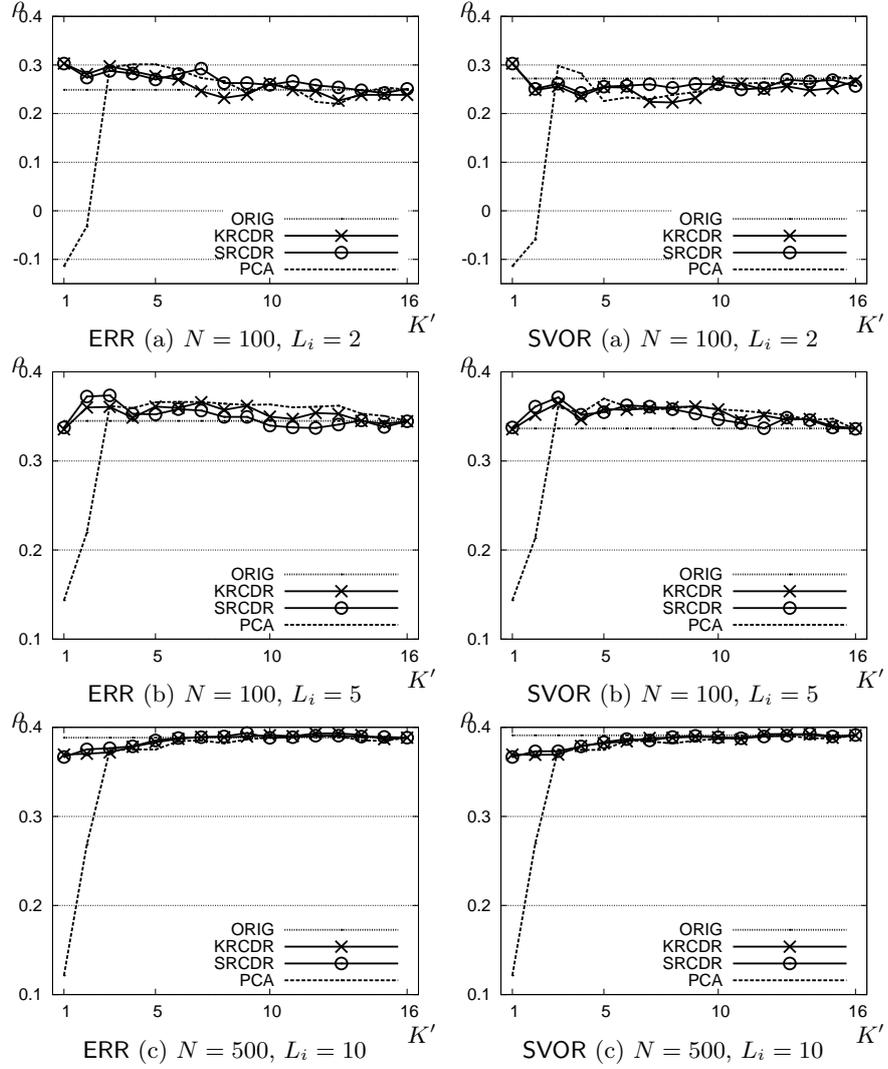
Let’s look at the first vector. The vectors derived by the two RCDR methods show resemblance. This indicates that one can use the faster RCDR method; concretely, Spearman RCDR is better except for the case $L_i = 2$. Because the fourth and the fifth elements of the \mathbf{w}^* are zero, no information useful for the target ordering is represented in these axes. In our RCDR cases, the fourth and the fifth weights of vectors are almost zero; thus, these useless axes can be ignored. In the PCA case, the fourth weight is far from zero, because no information about the target ordering is taken into account. The PCA merely ignores axes that are correlated in attribute space, such as in the fifth element. Further, because variances in all dimensions are equal, the contribution ratio is not so large, even if the target ordering is decided by a linear function.

We turn to the second component. In the RCDR cases, the correlation vector size $\|\mathbf{R}^{(2)}\|/N$ is much smaller than $\|\mathbf{R}^{(1)}\|/N$; this means that the second vector is far less informative than the first, because the target ordering is generated by a linear function in this example. In the PCA case, the contribution ratio indicates that useful information still remains in this vector. Note that it is not guaranteed that the $\|\mathbf{R}^{(l)}\|/N$ decreases in accordance with the increase of l , and vectors with bigger $\|\mathbf{R}^{(l)}\|/N$ don’t always contribute to predicting the target ordering. However, we empirically observed that if $\|\mathbf{R}^{(l)}\|/N$ is very small, the corresponding vector is not informative. We believe that $\|\mathbf{R}^{(l)}\|/N$ can be used as an index for the importance of vectors.

3.2 Experiments on Real Data Sets

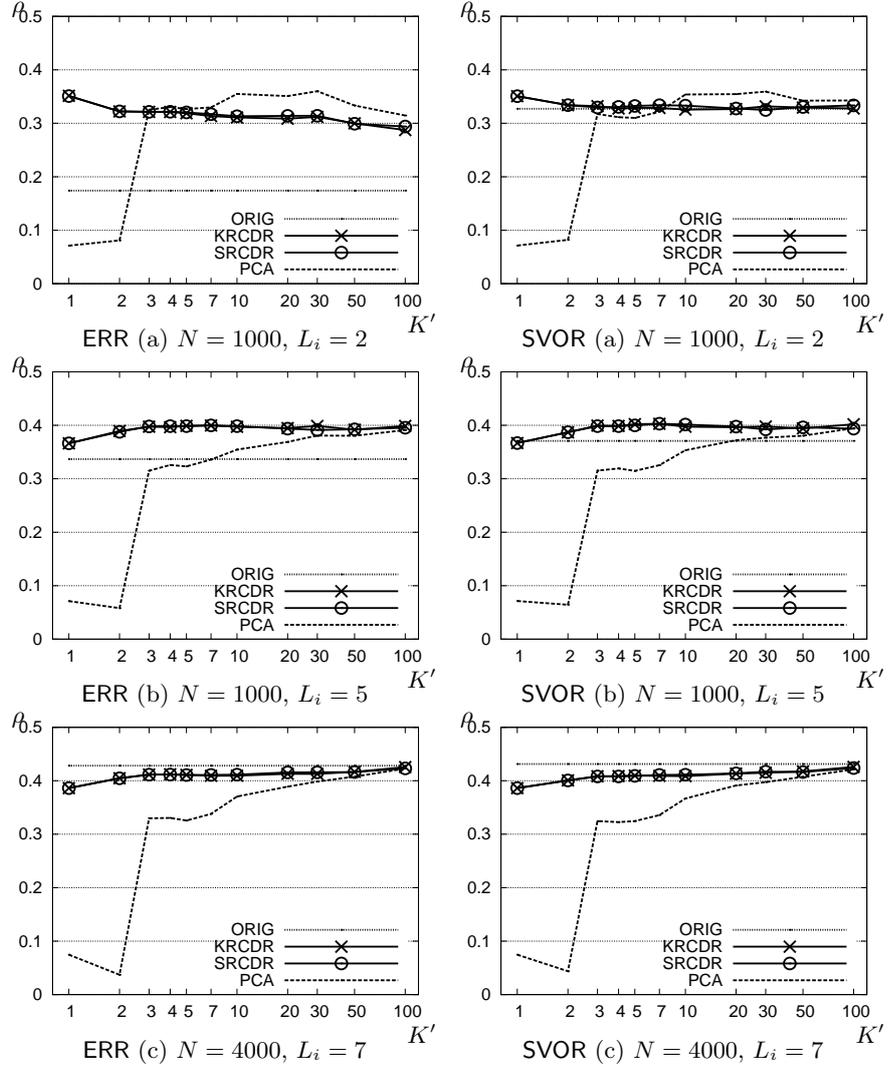
We applied the methods described in Section 2 to real data from questionnaire surveys². The first data set was a survey of preferences in sushi (Japanese food), and is denoted by SUSHI. In this data set, $N = 500$, $L_i = 10$, and $|\mathcal{X}^*| = 100$. Objects are represented by 12 binary and 4 numerical attributes. The second data set was a questionnaire survey of news article titles sorted according to their significance, and is denoted by NEWS. These news articles were obtained from “CD-Mainichi-Newspapers 2003.” In this data set, $N = 4000$, $L_i = 7$, and $|\mathcal{X}^*| = 11872$. The variance among sample orders was slightly broader than the

² <http://www.kamishima.net/sushi/>



Note: The concordances between sample orders and estimated orders were measured by Spearman’s ρ . These charts show variation of ρ in accordance with the number of dimensions K' . The charts that labeled, ERR and SVOR, show the results derived by the expected rank regression and the support vector ordinal regression, respectively. N is the size of a data set and L_i is the length of sample orders. The curves labeled “ORIG” show the result derived without application of dimension reduction. The curves labeled “KRCDR”, “SRCDR”, and “PCA” show the estimation results after reducing dimensions by the corresponding method.

Fig. 3. Comparison of dimension reduction methods on SUSHI data sets



Note: Note: The concordances between sample orders and estimated orders were measured by Spearman's ρ . These charts show variation of ρ in accordance with the number of dimensions K' . The charts that labeled, ERR and SVOR, show the results derived by the expected rank regression and the support vector ordinal regression, respectively. N is the size of a data set and L_i is the length of sample orders. The curves labeled "ORIG" show the result derived without application of dimension reduction. The curves labeled "KRCDR", "SRCDR", and "PCA" show the estimation results after reducing dimensions by the corresponding method.

Fig. 4. Comparison of dimension reduction methods on NEWS data sets

SUSHI data. Titles were represented by 0-1 vectors indicating whether a specified keyword appears in the title. Among 18381 keywords, we selected 595 keywords that were observed 30 or more times. Additionally, we used 8 binary attributes to represent article categories; thus, the number of attributes was 603 in total.

To evaluate the usefulness of our dimension reduction methods, we applied the Support Vector Ordinal Regression (SVOR) [5] (also known as RankingSVM [6]), and Expected Rank Regression (ERR) [1, 2], to these two data sets. We used our original implementation for the ERR and the SVM^{light} ³ for the SVOR. We chose these two methods, because they differently behaved in the survey [2]. The SVOR was robust for the noise in attribute values, but not for the perturbations in sample orders. Contrarily, the ERR could be resistant to ordinal noises, but not to the variation in attribute values. As a family of fitting functions, no kernel was used and a linear model was adopted.

Sample order sets were partitioned into testing and training sets. The ranking function was learned from training a sample order set with original attributes or reduced attributes. After learning, prediction performance was measured by the mean of ρ between an order in a testing set, O_t , and the corresponding estimated order, \hat{O}_t . The larger ρ was, the better the prediction performance was. The number of folds in cross-validation was ten for SUSHI and five for NEWS. Note that we reduced the number of folds in the NEWS experiment, because the size of the NEWS data set was larger and we had to save required computational time. In the left and right parts of the Figure 3 and 4, we show the variation of mean ρ in accordance with the dimensions of the reduced space, K' , for SUSHI and NEWS, respectively. For both data sets, N or L_i was varied by eliminating sample orders or objects; the results for these sets are shown in each sub-figure. The charts that labeled, ERR and SVOR, show the results derived by the expected rank regression and the support vector ordinal regression, respectively. N and/or L_i increased from the sub-figure (a) to (c); thus, orders became the most difficult to estimate in the sub-figure (a) case. The curves labeled by “KRCDR”, “SRCDR”, and “PCA” show the mean ρ derived by ERR after applying Kendall RCDR, Spearman RCDR, and PCA, respectively. The label “ORIG” indicates that no reduction method was used, and original attribute vectors were adopted.

From these figures, the following conclusions can be drawn. First, in terms of the variation according to the increase of the dimensions, both the ERR and SVOR behaved very similarly. This showed that our RCDR technique is independent from the learning algorithms. Second, the two RCDR methods show resemblance; thus, the faster method can be used for dimension reduction. Third, both RCDRs performed better in prediction than PCA. The difference was particularly clear when the number of dimensions K' was small. This means that RCDR successfully preserved information useful for estimating target orders. Therefore, we can say that RCDR is more effective than PCA when carrying out an object ranking task. Fourth, our RCDR technique could improve the prediction performance. The curves labeled “SRCDR”/“KRCDR” were compared with those labeled “ORIG.” The reduced vectors could lead to better predic-

³ <http://svmlight.joachims.org/>

tion than the original vectors. We think that this is because the models used for ordering were simplified while useful information was preserved. This can be confirmed by the fact that the improvements were prominent when N and/or L_i were small. The simpler model could produce better generalization ability for a limited number of samples. Therefore, our reduction technique is useful for improving prediction performance.

Finally, we can exploit the components of vectors for qualitative analysis. We obtained the first vector, $\mathbf{w}^{(1)}$, derived from the SUSHI, $N=500$, $L_i=10$ data set by applying our Kendall RCDR method. The components of the vectors, $w_1^{(1)}, \dots, w_K^{(K)}$, were sorted in descending order of their absolute values, $|w_k^{(1)}|$. The top 5 components were as follows:

$$\begin{aligned} w_{13}^{(1)} &= 0.5951 && \text{the frequency the user eats} \\ w_{15}^{(1)} &= 0.4278 && \text{how many restaurants supply the sushi} \\ w_1^{(1)} &= 0.4237 && \text{red fish (e.g., fatty tuna)} \\ w_{14}^{(1)} &= 0.2822 && \text{inexpensiveness} \\ w_{12}^{(1)} &= -0.2317 && \text{lightness or non-oiliness in tasting} \end{aligned}$$

From these components, we can say that “users primarily prefer sushi that they frequently eat and that is supplied in many sushi restaurants.”

4 Discussion and Conclusion

In this paper, we proposed a dimension reduction technique specialized for an object ranking task. The method was designed so as to preserve information about a relation from object attribute vectors to the target ordering. For this purpose, we developed Kendall RCDR and Spearman RCDR. We then applied these methods to real data sets. From the experimental results, we arrived at the following conclusions. First, the RCDR methods outperform PCA when carrying out an object ranking task. Second, by using the RCDR technique, performance in prediction can be improved, especially when training samples are not adequate. Finally, our two RCDR methods are comparable in prediction performance. Therefore, the faster method should be used; concretely, Spearman RCDR is better except for the condition where $L_i = 2$.

Intuitively speaking, in the l -th iteration of the RCDR, the algorithm finds the vector that is most relevant to target ordering. After that, by mapping attribute vectors to the new sub-space, components in attributes related to this vector are subtracted. At this time, it might be effective to subtract the explained component in the target ordering from sample orders. We will try such improvement by using a technique like Diaconis’ spectral analysis [15].

Acknowledgments: This work is supported by the grants-in-aid 14658106 and 16700157 of the Japan society for the promotion of science. Thanks are due to the Mainichi Newspapers for permission to use the articles. We would also like to thank Thorsten Joachims for providing the SVM^{light} software.

References

1. Kamishima, T., Kazawa, H., Akaho, S.: A survey and empirical comparison of object ranking methods. In: *Preference Learning*. Springer (2010)
2. Kamishima, T., Kazawa, H., Akaho, S.: Supervised ordering — an empirical survey. In: *Proc. of The 5th IEEE Int'l Conf. on Data Mining*. (2005) 673–676
3. Luaces, O., Bayón, G.F., Quevedo, J.R., Díez, J., del Coz, J.J., Bahamonde, A.: Analyzing sensory data using non-linear preference learning with feature subset selection. In: *Proc. of the 15th European Conf. on Machine Learning*. (2004) 286–297 [LNAI 3201].
4. Cohen, W.W., Schapire, R.E., Singer, Y.: Learning to order things. *Journal of Artificial Intelligence Research* **10** (1999) 243–270
5. Herbrich, R., Graepel, T., Bollmann-Sdorra, P., Obermayer, K.: Learning preference relations for information retrieval. In: *ICML-98 Workshop: Text Categorization and Machine Learning*. (1998) 80–84
6. Joachims, T.: Optimizing search engines using clickthrough data. In: *Proc. of The 8th Int'l Conf. on Knowledge Discovery and Data Mining*. (2002) 133–142
7. Radlinski, F., Joachims, T.: Query chains: Learning to rank from implicit feedback. In: *Proc. of The 11th Int'l Conf. on Knowledge Discovery and Data Mining*. (2005) 239–248
8. Yu, H.: SVM selective sampling for ranking with application to data retrieval. In: *Proc. of The 11th Int'l Conf. on Knowledge Discovery and Data Mining*. (2005) 354–363
9. Freund, Y., Iyer, R., Schapire, R.E., Singer, Y.: An efficient boosting algorithm for combining preferences. *Journal of Machine Learning Research* **4** (2003) 933–969
10. Bahamonde, A., Bayón, G.F., Quevedo, J.D.J.R., Luaces, O., del Coz, J.J., Alonso, J., Goyache, F.: Feature subset selection for learning preferences: A case study. In: *Proc. of The 21st Int'l Conf. on Machine Learning*. (2004) 49–56
11. Slotta, D.J., Vergara, J.P., Ramakrishnan, N., Heath, L.S.: Algorithms for feature selection in rank-order spaces. Technical Report TR-05-08, Computer Science, Virginia Tech. (2005)
12. Dettling, M., Bühlmann, P.: Supervised clustering of genes. *Genome Biology* **3**(12) (2002) research0069.1–0069.15
13. Deng, L., Pei, J., Ma, J., Lee, D.L.: A rank sum test method for informative gene discovery. In: *Proc. of The 10th Int'l Conf. on Knowledge Discovery and Data Mining*. (2004) 410–419
14. Kamishima, T., Akaho, S.: Filling-in missing objects in orders. In: *Proc. of The 4th IEEE Int'l Conf. on Data Mining*. (2004) 423–426
15. Diaconis, P.: A generalization of spectral analysis with application to ranked data. *The Annals of Statistics* **17**(3) (1989) 949–979
16. Kazawa, H., Hirao, T., Maeda, E.: Order SVM: a kernel method for order learning based on generalized order statistics. *Systems and Computers in Japan* **36**(1) (2005) 35–43

Index

rank correlation dimension reduction, 2 RCDR, *see* rank correlation dimension reduction