Supervised ordering by regression combined with Thurstone's model

Toshihiro Kamishima · Shotaro Akaho

Abstract   In this paper, we advocate a learning task that deals with the orders of objects, which we call the *Supervised Ordering* task. The term *order* means a sequence of objects sorted according to a specific property, such as preference, size, cost. The aim of this task is to acquire the rule that is used for estimating an appropriate order of a given unordered object set. The rule is acquired from sample orders consisting of objects represented by attribute vectors. Developing solution methods for accomplishing this task would be useful, for example, in carrying out a questionnaire survey to predict one's preferences. We develop a solution method based on a regression technique imposing a Thurstone's model and evaluate the performance and characteristics of these methods based on the experimental results of tests using both artificial data and real data.

Keywords   Order · Rank · Supervised learning · Regression analysis · Thurstone's law of comparative judgment · Sensory survey

1 Introduction

In this paper, we advocate a new learning task that deals with the orders of objects, which we call *Supervised Ordering*. Here, the term *order* means a sequence of objects sorted according to a specific property. Suppose there are three kinds of meals, x_a, x_b, and x_c. Then one example of order is the sequence $x_c \succ x_a \succ x_b$, in which the sorting is performed according to Taro's preference. This order represents that he likes meal x_c the most, and meal x_b the least. The goal of a supervised ordering task is to acquire the rule that is used for

T. Kamishima (✉) · S. Akaho
National Institute of Advanced Industrial Science and Technology (AIST), Tsukuba Central 2, Umezono 1–1–1, Tsukuba, Ibaraki, Japan
e-mail: mail@kamishima.net
URL: http://www.kamishima.net/

S. Akaho
e-mail: s.akaho@aist.go.jp

estimating an appropriate order for any object set. The rule is learned from training sample orders consisting of objects represented by attribute vectors.

An example of performing a supervised ordering task would be completing a questionnaire survey of preferences in meals. The surveyor presents several kinds of meals to each respondent and requests that he/she sort the meals according to his/her preference. By applying a supervised ordering method, the surveyor will be able to estimate the respondents' preference order of meals, and to determine, for example, the most preferred meal, or the factors affecting preference. For such a survey, it is typical to use the Semantic Differential method (Osgood et al. 1957). By this method, the respondents' preferences are measured on a scale, the extremes of which are expressed by antonyms:

<p align="center">prefer   5   4   3   2   1   don't prefer</p>

A proper interpretation of responses using this scale would be, for example, that respondents prefer objects categorized as "5" to objects categorized as "4". However, to analyze the acquired ratings as interval values, that are numerical values the scale and origin of which are free, the following two assumptions must be introduced: "The total lengths of all the scales are equal" and "All the intervals of one scale are equal" (Nakamori 2000). However, it is commonly difficult for respondents to quantify their own sensation so as to satisfy these assumptions. One way to avoid these assumptions is by introducing a ranking method. In this method, the respondents' sensations or preferences are measured by the orders sorted according to the target to measure. But this measuring method has not frequently been used due to the lack of an analysis technique. We therefore undertook to develop a learning technique targeting orders.

In this paper, we formally specify the supervised ordering task, then propose its solution method based on a regression technique incorporating Thurstone's model. We also show Cohen's method, which was designed for a task close to that of supervised ordering, and its variants. We apply these methods to both artificial and real data, respectively. Analysis of the experimental results revealed the characteristic results of these methods.

We proceed as follows. In Sect. 2, we present the formal specifications of a supervised ordering task. We propose several supervised ordering methods in Sect. 3, and show experimental results in Sect. 4. Finally, Sect. 5 summarizes our conclusions.

1.1 Related work

Though research in analysis techniques for orders has not been widely conducted, several previous studies exist. A pioneering work is Thurstone's law of comparative judgment (Thurstone 1927). Thurstone proposed a method of constructing a real-value scale from a set of ordered object pairs. The ordinal regression is a regression method adopting models in which dependent variables are ordered categorical type (Agresti 1996; Anderson and Philips 1981; McCullagh 1980). Some probabilistic models for orders have been developed (Babington Smith 1950; Fligner and Verducci 1990; Mallows 1957; Plackett 1975). Recently, there has been active research in the processing of orders. Mannila and Meek (2000) tried to establish the structure expressed by partial orders among a given set of orders. Sai et al. (2001) proposed association rules between order variables. Kamishima and Fujiki studied the k-o'means algorithm for clustering incomplete orders (Kamishima and Fujiki 2003). Kamishima developed collaborative filtering techniques based on order responses (Kamishima 2003). Kamishima and Akaho proposed a technique for filling-in missing objects in orders (Kamishima and Akaho 2004). Lebanon and Lafferty discussed the combination method for orders using conditional probability models (Lebanon and Lafferty 2002).

Table 1 Examples of preference orders in sushi

Respondent	Preference order
Taro	$O_{\text{Taro}} = x_F \succ x_S \succ x_E \succ x_C$
Jiro	$O_{\text{Jiro}} = x_F \succ x_S \succ x_E$
Hanako	$O_{\text{Hanako}} = x_E \succ x_F \succ x_C$

Objects: x_C (Cucumber roll), x_E (Egg), x_F (Fatty tuna), x_S (Squid), x_T (Tuna)

The task studied by Cohen et al. (1999), Herbrich et al. (1998), and Joachims (2002) is a special case of our task. We later describe these in Sect. 3.1.

2 Supervised ordering

2.1 Basic notations

This section formally states a supervised ordering task.

An object x_j corresponds to an object, entity, or substance to be sorted. The universal object set, X^*, consists of all possible objects. Each object x_j is represented by the attribute value vector, $x_j = (x_{j1}, x_{j2}, \ldots, x_{jk})$, where k is the number of attributes. In this paper, we focus on the case in which all the attributes are categorical. The l-th attribute can take one of the $|v_l|$ values, $\{v_{l1}, \ldots, v_{l|v_l|}\}$.

The order is denoted by $O = x_1 \succ x_2 \succ \cdots \succ x_3$. The transitivity is preserved in the same order; i.e., if $x_1 \succ x_2$ and $x_2 \succ x_3$, then $x_1 \succ x_3$. To express the order of two objects, $x_1 \succ x_2$, we use the sentence "x_1 precedes x_2." $X_i \subseteq X^*$ denotes the object set that is composed of all the objects that appear in the order O_i. Let $|A|$ be the size of the set A, then $|X_i|$ is equal to the length of the order O_i. If $X_i = X^*$, an order O_i is called a complete order, otherwise it is called an incomplete order. A rank, $r(O_i, x_j)$, is the cardinal number that indicates the position of the object x_j in the order O_i.

For two orders, O_1 and O_2, consider an object pair x_a and x_b, such that $x_a, x_b \in X_1 \cap X_2, x_a \neq x_b$. We say that the orders O_1 and O_2 are concordant w.r.t. x_a and x_b, if two objects are placed in the same order, i.e.,

$$(r(O_1, x_a) - r(O_1, x_b))(r(O_2, x_a) - r(O_2, x_b)) \geq 0;$$

otherwise, these two orders are discordant. O_1 and O_2 are concordant if O_1 and O_2 are concordant w.r.t. all object pairs such that $x_a, x_b \in X_1 \cap X_2, x_a \neq x_b$.

Table 1 shows three example orders, each of which is sorted according to one respondent's preference in *sushi* (a Japanese meal). In this case, each object corresponds to a kind of sushi (see note in Table 1), and is represented by the six attributes in Table 8. A universal set is composed of five objects, $X^* = \{x_C, x_E, x_F, x_S, x_T\}$. For Taro's preference order, $X_{\text{Taro}} = \{x_C, x_E, x_F, x_S\}$, $|X_{\text{Taro}}| = 4$, and $r(O_{\text{Taro}}, x_E) = 3$. O_{Taro} is incomplete due to the fact $X_{\text{Taro}} \subset X^*$. Further, O_{Taro} and O_{Hanako} are concordant w.r.t. x_F and x_C, but are discordant w.r.t. x_F and x_E, thus O_{Taro} and O_{Hanako} are discordant. O_{Taro} and O_{Jiro} are concordant.

The pair set Pair(O_i) is composed of all the object pairs $x_a \succ x_b$, such that x_a precedes x_b in the order O_i. For example, O_{Jiro} is decomposed into three pairs, $x_F \succ x_S$, $x_F \succ x_E$, and $x_S \succ x_E$. Given a set of orders, S, Pair(S) is composed of all pairs in Pair(O_i), $O_i \in S$.

Note that $\mathrm{Pair}(S) \neq \bigcup_{O_i \in S} \mathrm{Pair}(O_i)$, because pairs of the same form are multiply contained in $\mathrm{Pair}(S)$. For example, for $S = \{O_{Taro}, O_{Jiro}, O_{Hanako}\}$, $\mathrm{Pair}(S)$ consists of 12 pairs including two $x_F \succ x_S$ originated from O_{Taro} and O_{Jiro}.

2.2 Formalization of a supervised ordering task

The supervised ordering task is composed of two major stages: a learning stage and a sorting stage. In the learning stage, given a set of sample orders, $S = \{O_1, O_2, \ldots, O_N\}$, the learner acquires a sorting function, $\mathrm{sort}(X_u)$, that is the function of an unordered object set $X_u \subseteq X^*$. In the sorting stage, based on the acquired function, $\mathrm{sort}(X_u)$, the constituent objects in X_u are sorted in an appropriate order. We denote the estimated order of X_u by $\hat{O}_u$. For example, suppose that a sorting function is learned from $S = \{O_{Taro}, O_{Jiro}, O_{Hanako}\}$ in Table 1. Given an unordered object set $X_u = \{x_C, x_E, x_F, x_T\}$ as an argument of the sorting function, these objects are sorted into an estimated order, say, $\hat{O}_u = x_F \succ x_T \succ x_E \succ x_C$. This estimated order should satisfy the following two conditions: First, the estimated order should be maximally concordant with sample orders in S. The degree of concordance is measured by the similarities between orders as shown later. The above $\hat{O}_u$ is perfectly concordant with O_{Taro} or O_{Jiro}, and almost concordant with O_{Hanako}. Second, the objects not appearing in S, e.g., x_T in Table 1, have to be sorted by using the sorting function. In an estimated order, the ranks of such unseen objects should be close to those of the objects that are neighbors of the unseen objects in attribute space. For example, the x_T (Tuna) is a neighbor of the x_F (Fatty Tuna) in an attribute space, thus it would be appropriate that the x_T was ranked next to the x_F.

We next show a concordance or similarity measure between two orders. In this investigation, we adopted the well known statistics *Spearman's Rank Correlation* ρ (Kendall and Gibbons 1990), defined as the correlation between ranks of objects. The ρ between two orders, O_1 and O_2, consisting of the same objects (i.e., $X_1 = X_2 (\equiv X)$) is defined as:

$$\rho(O_1, O_2) = \frac{\sum_{x \in X} \left(r(O_1, x) - \bar{r}_1\right)\left(r(O_2, x) - \bar{r}_2\right)}{\sqrt{\sum_{x \in X} \left(r(O_1, x) - \bar{r}_1\right)^2} \sqrt{\sum_{x \in X} \left(r(O_2, x) - \bar{r}_2\right)^2}},$$

where $\bar{r}$ denotes the mean rank, i.e., $(1/|X|) \sum_{x \in X} r(O_i, x)$. If no tie in rank is allowed, this can be calculated by the simple formula:

$$\rho(O_1, O_2) = 1 - \frac{6 \times \sum_{x \in X} \left(r(O_1, x) - r(O_2, x)\right)^2}{|X|^3 - |X|}.$$

The ρ becomes 1 if the two orders are perfectly concordant, and -1 if one order is a reverse of the other order. It is known that for two random orders having lengths of $|X|$, the t-value,

$$t = \rho \sqrt{\frac{|X| - 2}{1 - \rho^2}}, \tag{1}$$

asymptotically follows the Student t-distribution with degree of freedom $|X| - 2$ (Kendall and Gibbons 1990).

It is worthwhile to point out that this supervised ordering task is connected to the notion of the center of orders (Marden 1995). A central order is the order to maximize the mean similarity among given sample orders. For finding such central orders, many methods and models have been developed; for example, Thurstonian type models (Thurstone 1927), a Babington Smith model (Babington Smith 1950), a Mallows model (Mallows 1957), or a Plackett-Luce

model (Plackett 1975). Unlike those in a supervised ordering task, these sample orders are the sequences of objects that are represented by unique identifiers, not by attribute vectors. Therefore, methods for finding central orders cannot be used for ordering objects unseen in sample orders. However, the sorting function of a supervised ordering case can be used for sorting unseen objects based on information of their attributes. A supervised ordering task can be considered as an extension of this task so as to deal with objects represented by attribute vectors.

3 Methods

In this section, we describe the two types of solution methods for a supervised ordering task. One is based on a classification technique and was proposed by Cohen et al. The other is our new method based on a regression technique.

3.1 Cohen classification (CC)

Cohen et al. (1999), Herbrich et al. (1998), and Joachims (2002) studied the task that is the special case of a supervised ordering task. Thus, we first describe this. The main difference in formalization between their task and a supervised ordering task is the length of sample orders. Samples of their task are ordered pairs of objects, i.e., the length of orders are exactly two, whereas any length of samples are allowed in a supervised ordering task. Formally, the learner is given samples of ordered pairs, $S_p = \{P_1, \ldots, P_N\}$, where P_i denotes an order the length of which is two. As in the case of a supervised ordering task, the aim of this task is to acquire the function $\mathrm{sort}(X_u)$. Note that the size of the unordered object set X_u is not limited to two. Objects are again sorted into an order so that they are concordant with given sample pairs, and objects neighboring in attribute space are ranked proximately in the order.

Cohen et al. proposed a sorting function defined by the following procedure. First, the conditional probability function, $\Pr[x_a \succ x_b | x_a, x_b]$, of the event that x_a precedes x_b given the attribute values of x_a and x_b is derived from samples. Cohen et al. (1999) used their original Hedge algorithm to learn the probability function. We consider their main contribution to be the sorting procedure using this function, which can be replaced with any algorithm to learn the probability. In addition, their method required adoption of an on-line learning algorithm to perform the application task, whereas our application problem can be solved by using off-line algorithms. We therefore adopted a basic and well studied algorithm, the naive Bayes classifier (Mitchell 1997).

$$\Pr[x_a \succ x_b | x_a, x_b] \propto \Pr[x_a, x_b | x_a \succ x_b] \Pr[x_a \succ x_b],$$

$$\Pr[x_a, x_b | x_a \succ x_b] \approx \prod_{l=1}^{k} \Pr[x_{al}, x_{bl} | x_a \succ x_b].$$

As the probability $\Pr[x_{al}, x_{bl} | x_a \succ x_b]$, we adopt the following Bayesian estimator with the Dirichlet prior so that the probability is non-zero:

$$\Pr[x_{al}, x_{bl} | x_a \succ x_b] = \frac{|x_{al} \succ x_{bl}| + |v_l|^{-2}}{|S_p| + 1},$$

Table 2 Cohen's Greedy-order Algorithm

INPUT: X_u

1) $\hat{O}_u \leftarrow \emptyset$

2) **foreach** $x \in X_u$ **do** score$(x) \leftarrow \sum_{x' \in X_u} \Pr[x \succ x'|x, x']$

3) $x_{top} \leftarrow \arg\max_{x \in X_u}$ score(x)

4) $\hat{O}_u \leftarrow \hat{O}_u \succ x_{top}, X_u \leftarrow X_u - \{x_{top}\}$,

5) **foreach** $x \in X_u$ **do** score$(x) \leftarrow$ score$(x) - \Pr[x \succ x_{top}|x, x_{top}]$

6) **if** $X_u \neq \emptyset$ **then goto** step 3

7) **output** $\hat{O}_u$

---

where $|x_{al} \succ x_{bl}|$ is the total number of the object pairs $x_i \succ x_j$ such that $x_{il} = x_{al}$ and $x_{jl} = x_{bl}$ are members of the S_p. $\Pr[x_a \succ x_b]$ can be calculated by

$$\Pr[x_a \succ x_b] = \frac{|x_a \succ x_b| + 0.5}{|x_a \succ x_b| + |x_b \succ x_a| + 1},$$

where $|x_a \succ x_b|$ is the number of the object pairs, $x_a \succ x_b$, in the S_p.

After having learned such a probability function, an estimated order, $\hat{O}_u$, is derived so as to maximize the agreement:

$$\sum_{x_a \succ x_b \in \text{Pair}(\hat{O}_u)} \Pr[x_a \succ x_b|x_a, x_b], \tag{2}$$

where $\text{Pair}(\hat{O}_u)$ is a pair set of $\hat{O}_u$ in Sect. 2.1. To derive $\hat{O}_u$, the permutation of X_u that maximizes Eq. 2 is searched for among all possible permutations. However, because finding the optimal order is not tractable if $|X_u|$ is large, the greedy algorithm in Table 2 is used. Simply speaking, this algorithm sequentially chooses the most-preceding object according to the function score(x).

To apply Cohen's method for solving a supervised ordering task, we convert the given sample orders into the form of ordered pairs. Specifically, a sample set of a supervised ordering task, S, is converted into the pair set, Pair(S), in Sect. 2.1, and the Pair(S) is considered as sample pairs of Cohen's task, S_p. From the converted set of ordered pairs, the probability function $\Pr[x_a \succ x_b|x_a, x_b]$ can be derived. By applying the above Cohen's procedure, objects in any unordered set X_u will be sorted appropriately. We refer to this method as *Cohen Classification* (CC).

It is worthwhile to discuss the difference between Cohen's task and a supervised ordering task. These two tasks are distinct in terms of the length of sample orders; it must be two in a Cohen's, whereas a longer length is allowed in a supervised ordering task. This distinction seems slight, but affects the following two features: concordance among orders and distributional properties of samples. We first discuss what is an order maximally concordant with sample orders. Due to the statistical generation nature of sample orders, these orders may conflict with each other. For example, how should three orders, $x_a \succ x_b$, $x_b \succ x_c$, and $x_c \succ x_a$, be combined? To resolve such conflicts, preferential conditions must be introduced to induce combined orders. Such conditions have been studied since the 18th century. For example, in the literature of social choice, de Condorcet pointed out that there are disagreements among orders derived by ordering schemes grounded in plausible different policies (de Condorcet 1995). Arrow proved the Arrow's impossibility theorem (Arrow 1963); there

is no method to combine orders so as to simultaneously satisfy four plausible conditions called *U*, *P*, *I*, and *ND*. Among such conditions, we here focus on Arrow's Condition I: independence from irrelevant alternatives. Specifically, if a sorting function is fixed, either $x_a \succ x_b$ or $x_b \succ x_a$ is concordant with $\hat{O}_u$ for any X_u such that $x_a, x_b \in X_u$. In other words, the order with respect to two objects is determined depending on only these two objects. For example, once a sorting function outputted $x_a \succ x_b \succ x_c$, the function must sort $\{x_a, x_b, x_d\}$ into orders that are concordant with $x_a \succ x_b$, i.e., $x_d \succ x_a \succ x_b$, $x_a \succ x_d \succ x_b$, or $x_a \succ x_b \succ x_d$. In the statistical case, this can be formulated by

$$\Pr[r(\hat{O}_u, x_a) > r(\hat{O}_u, x_b)|X_u] = \Pr[r(\hat{O}_u, x_a) > r(\hat{O}_u, x_b)|\{x_a, x_b\}].$$

If the lengths of sample orders are two, i.e., Cohen's case, a sorting function learned from these samples is implicitly assumed to output orders that satisfy this condition because there is no information about the dependency of determination of the order between two objects on the other objects, due to the fact that third objects are not given in sample orders. In a supervised ordering case, in which there are sample orders composed of more than two objects, such information will be available. Therefore, learner cannot learn sorting functions not satisfying Arrow's Condition I, even if needed. Note that the supervised ordering methods in this paper assume this condition, based on the fact that the condition is preferable in many applications. For example, in the case of the sushi preference survey in Table 1, *Fatty Tuna* would be preferred to *Cucumber Roll*, even if any other types of sushi were considered together with these types of sushi.

Moreover, the difference in the length of orders, especially between the orders of the length two and the orders the length of which are longer than two, affects the dependence of objects in sample orders. If sample orders are given in a form of ordered pairs as in Cohen's case, the independence among pairs would be high. On the other hand, pairs in Pair(*S*) are not highly independent, because if $x_a \succ x_b$ and $x_b \succ x_c$ are in Pair(*S*), there is a high probability that $x_a \succ x_c$ originated from the decomposition of $x_a \succ x_b \succ x_c$ would also be in Pair(*S*). Therefore, between task formulations of Cohen's and of a supervised ordering, the distribution of ordered pairs differs considerably.

3.2 Babington Smith classification (BSC)

As noted in Sect. 2.2, a supervised ordering method can be considered as an extension of a method for finding central orders. That is to say, supervised ordering methods consist of two components: one is an *attribute component* for mapping attributes to internal representation, the other an *ordering component* for sorting objects based on the representation. As an attribute component of CC, traditional classification modeling is used. However, a model for its ordering part is not found in any traditional models, so its properties are unknown. Hence, we analyzed the properties of CC based on the behavior of its variant method employing an existing ordering model that is very close to that of CC. Below, we show this variant method, which is introduced only for the purpose of analyzing the properties of CC.

By replacing the summation in the Cohen's criterion (2) with the production, we obtain

$$\prod_{x_a \succ x_b \in \text{Pair}(\hat{O}_u)} \Pr[x_a \succ x_b | x_a, x_b]. \tag{3}$$

This coincides with the *Babington Smith Model* (Babington Smith 1950; Marden 1995), that is, the simple and early developed probability model for orders, up to the normalization term. This model represents the probability of orders under the independence assumption among

ordered pairs. For example, the order $x_a \succ x_b \succ x_c$ is observed under the assumption that the events $x_a \succ x_b$, $x_a \succ x_c$, and $x_b \succ x_c$ are independent.

This algorithm is the same as the algorithm in Table 2, except in terms of the calculation steps of scores:

2)   **foreach** $x \in X_u$ **do** score$(x) \leftarrow \prod_{x' \in X_u} \Pr[x \succ x' | x, x']$
5)   **foreach** $x \in X_u$ **do** score$(x) \leftarrow$ score$(x)/\Pr[x \succ x_{top} | x, x_{top}]$

We refer to this method as *Babington Smith Classification* (BSC). The mathematical formulation of BSC is not only similar to that of CC, but also the experimental results in Sect. 4 will show that these methods lead to similar estimated orders.

3.3 Thurstone regression (TR)

In this section, we advocate an alternative solution method based on the regression technique combined with Thurstone's model. We call this method the *Thurstone Regression* (TR). We describe this new method, then show why it allows appropriate orders to be estimated.

In the sorting stage of TR, a linear score function of attribute value vector, x_j, is used. The score $\hat{r}(x_j) = w \cdot x_j$ is the inner product of the weight vector w and the x_j. The estimated order is derived by sorting the objects $x_j \in X_u$ according to the score $\hat{r}(x_j)$. This function is acquired in the learning stage. The learning procedure is composed of two stages. First, the given sample orders are combined into one order based on Thurstone's model. Second, applying a regression technique derives a linear function $\hat{r}(x_j)$ to estimate the ranks in this combined order.

We show a derivation process of the combined order. All the object sets in the training example set are collected into one object set, X_S, i.e., $\bigcup_{X_i \in S} X_i$. Next, a central order, that is an order consisting of X_S and being maximally concordant with sample orders, is found. To derive this central order, O_S, one creates the set of object pairs $\mathrm{Pair}(S)$ and calculates the probabilities for every pair in $\mathrm{Pair}(S)$:

$$\Pr[x_a \succ x_b] = \frac{|x_a \succ x_b| + 0.5}{|x_a \succ x_b| + |x_b \succ x_a| + 1},$$

where $|x_a \succ x_b|$ is the number of the object pairs, $x_a \succ x_b$, in the $\mathrm{Pair}(S)$. These probabilities are applied to Thurstone's law of comparative judgment (Thurstone 1927); that is, to the generative model of orders. This model assumes that the score is assigned to each object x_j. The orders are derived by sorting according to the scores. Scores follow normal distribution; i.e., $N(\mu_j, \sigma)$, where μ_j is the mean score of the object x_j, and σ is a common constant standard deviation. By applying the least square method to this model (Mosteller 1951), the μ'_j, that is the linear transformation of the μ_j, is derived as

$$\mu'_j = \frac{1}{|X_u|} \sum_{x \in X_u} \Phi^{-1}\big(\Pr[x_j \succ x]\big), \qquad (4)$$

where $\Phi(\cdot)$ is the normal cumulative distribution function. The μ'_j is derived for each object in X_S, and the combined central order O_S can be derived by sorting according to these μ'_j.

After the calculation of the order O_S, we then acquire the score function, $\hat{r}(x_j)$, that measures the degree to which the object x_j precedes the other objects. The function $\hat{r}(x_j)$ is derived by the regression technique for categorical attributes using dummy variables; this is also known as the *Type 1 Quantification method* (Hayashi 1952). First, the attribute value vector of each object, $x_j \in X_S$, is transformed into the vector $d(x_j)$. The first step of this

transformation is to represent x_j by the 0–1 dummy vector. For example, consider the transformation of the attribute value vector $x_j = (v_{12}, v_{21}, v_{31})$, and the 1st, 2nd, and 3rd attribute can take $\{v_{11}, v_{12}\}$, $\{v_{21}, v_{22}, v_{23}\}$, and $\{v_{31}, v_{32}\}$, respectively. This vector is converted into

$$x_j \Rightarrow (\; \underbrace{\underset{v_{11}}{0} \; \underset{v_{12}}{1}}_{x_{j1}} \; \underbrace{\underset{v_{21}}{1} \; \underset{v_{22}}{0} \; \underset{v_{23}}{0}}_{x_{j2}} \; \underbrace{\underset{v_{31}}{1} \; \underset{v_{32}}{0}}_{x_{j3}} \;).$$

The $d(x_j)$ is then derived by eliminating the 0–1 entries corresponding to the first values except in terms of the first attribute. Namely, the entries $v_{21}, v_{31}, \ldots, v_{k1}$ are deleted. In the case of the above example, the entries v_{21} and v_{31} are deleted, and we obtain

$$d(x_j) = (\; \underbrace{\underset{v_{11}}{0} \; \underset{v_{12}}{1}}_{x_{j1}} \; \underbrace{\underset{v_{22}}{0} \; \underset{v_{23}}{0}}_{x_{j2}} \; \underbrace{\underset{v_{32}}{0}}_{x_{j3}} \;).$$

Such $d(x_j)$ are derived for each element x_j in X_S, and the examples, $(r(O_S, x_j), d(x_j))$, are formed. By applying the standard regression method to this example set,

$$\{(r(O_S, x_1), d(x_1)), (r(O_S, x_2), d(x_2)), \ldots, (r(O_S, x_{|X_S|}), d(x_{|X_S|}))\},$$

we obtain the weight vector, w. The dependent and independent variables of this regression model are $r(O_S, x_j)$ and $d(x_j)$, respectively. Entries for the w represent the weights of corresponding respective elements of the transformed attribute value vector $d(x)$.

Below, we describe a rationale for why this TR works. In this method, objects in X_u are sorted according to the scores $\hat{r}(x_j) = w \cdot x_j$. This score estimated by TR can be considered as an estimator of the true scores $r^*(x_j)$ given the sample orders. Suppose that there is an universal order O^* of all the objects in X^*, and this order is generated from Thurstone's model, i.e., from $r^*(x_j) \sim \mu_j^* + N(0, \sigma^*)$. Assume that for any two adjacent objects x_j and x_k in O^*, the probabilities of permutation of these two objects are equal, then $|\mu_j^* - \mu_k^*|$ becomes constant. An order derived from $r^*(x_j)$ is invariant in terms of any monotonic transformation, thus μ_j^* can be replaced with a rank $r(O^*, x_j)$ without loss of generality. Accordingly, sample orders are generated from the model:

$$r^*(x_j) \sim r(O^*, x_j) + N(0, c\sigma^*),$$

where c is an arbitrary constant. We next model $r(O^*, x_j)$ by the linear function $w \cdot x_j$, and acquire this function from given sample orders. The weights w can be estimated by using the above type 1 quantification method, which is a regression for categorical variables, by considering $r(O^*, x_j)$ as the dependent variables. However, $r(O^*, x_j)$ are unknown; thus, these first have to be estimated. From the sample orders, an order O_S of objects in X_S can be estimated as the least square estimator. $r(O_S, x_j)$ may not be equal to $r(O^*, x_j)$ provided that $X_S \subseteq X^*$. According to a result in the order statistics literature (Arnold et al 1992), if X_S are uniformly sampled from X^* and $|X^*|$ is finite, an an expectation of $r(O^*, x_j)$ is

$$E[r(O^*, x_j)] = \frac{|X^*| + 1}{|X_S| + 1} r(O_S, x_j).$$

Instead of a linear function that leads to $r(O^*, x_j)$, we try to acquire those that lead to the expectation $E[r(O^*, x_j)]$. Though $|X^*|$ is unknown, because $|X^*|$ and $|X_S|$ are constant, the linear transformation of the $w \cdot x_j$ can be estimated by learning the function targeting $r(O_S, x_j)$ instead of $E[r(O^*, x_j)]$. This transformed function can be used to estimate an

order that is concordant with O^*. It follows from what has been said that the function $\hat{r}(x_j)$ acquired by the TR method can be used for estimation of an order $\hat{O}_u$.

Note that the TR method satisfies Arrow's Condition I, because the score of x_j is independent of any other objects.

3.4 The computational complexity of algorithms

Finally, we remark the computational complexity. In the learning stage, classification-based CC and BSC methods require counting up the number of objects in the training example set; therefore, the order of the complexity becomes $O(|\mathrm{Pair}(S)|) \approx O(N|\bar{X}|^2)$, where $|\bar{X}|$ is the mean length of sample orders. In cases of the TR, the learning time is $O(\max(N|\bar{X}|^2, |X_S|^2))$; that is, the sum of times $O(N|\bar{X}|^2)$ for $\Pr[x_a \succ x_b]$, $O(|X_S|^2)$ for all the μ'_j, $O(|X_S|\log(|X_S|))$ for sorting, and $O(N)$ for regression. Consequently, the regression-based TR is more time consuming than either of the classification-based CC and BSC methods.

In the sorting stage, classification-based methods take time $O(|X_u|^2)$ to sort based on the preference functions; the TR method takes time $O(|X_u|\log(|X_u|))$ for sorting according to the $\hat{r}(x_j)$ values. In short, the TR is faster than the classification-based methods in terms of sorting.

4 Experiments

We apply the supervised ordering methods described in the previous section to both artificial and real data, respectively.

4.1 Experiments on artificial data

To analyze the characteristics of these supervised ordering methods, we first applied these methods to artificial data.

4.1.1 Experimental conditions on artificial data

We prepared nine types of universal object sets, X^*, denoted in the form of $(k, |v|)$. k and $|v|(=|v_1|=\cdots=|v_k|)$ specify the number of attributes and the number of attribute values, respectively. k is set to 3, 5, or 7, and $|v|$ is set to 3, 4, or 5; thus, a total of 9 types of sets are generated. For example, for the $(3, 4)$ type, the universal object set consists of $4^3{=}64$ objects.

The universal orders, O^*, for these universal object sets are decided based on the linear score function. For example, when object x_7 is described by the attribute value vector $x_7 = (v_{11}, v_{23}, v_{31})$, the score becomes $w_1^a w_{11}^v + w_2^a w_{23}^v + w_3^a w_{31}^v$, where w_j^a and w_{jl}^v are weights assigned to the j-th attribute and the l-th attribute value of the j-th attribute, respectively. The universal order is a sequence of objects sorted according to their scores. For each type of attribute vector, we randomly generated 10 different sets of weights between 0 and 1. We consequently generated 90 tuples of a universal object set and its corresponding universal order.

Further, from each of these tuples, we generated 9 example sets: $|X_i|$ (length of sample orders) is 3, 5, or 10, and the N (the number of examples) is 10, 30, or 50. In total, 810 example sets were generated. $|X_i|$ of objects are sampled from X^*, sorted so as to be concordant with O^*. The resultant orders are referred to as true orders O_i^*. In *noiseless* cases,

Table 3 The means of ρ on the noiseless data sets

	All	$|X_i|$			N		
		3	5	10	10	30	50
CC	0.811	0.668	0.829	0.937	0.717	0.838	0.879
BSC	0.811	0.668	0.828	0.937	0.717	0.838	0.879
TR	0.818	0.637	0.856	0.961	0.725	0.852	0.877

sample orders are these true orders themselves. Using these example sets, we tested the three methods outlined in Sect. 3: CC, BSC, and TR.

As the testing procedure, we adopted a leave-one-out (LVO) test; that is, the N-fold cross-validation test. As an error measure, we adopted a true error, that is a ρ between the estimated order, $\hat{O}_i$, and its true order, O_i^*.

4.1.2 Experimental results on noiseless artificial data

Goals of experiments We first examined noiseless data. The main goal of this series of experiments was to compare the Cohen classification (CC) method and our Thurstone regression (TR) method. We observed the characteristics of CC and TR methods when varying the number of samples, N, or the length of samples, $|X_i|$. Based on these observations, we attempted to clarify the differences between these two methods. Specifically, we discuss the relationship of these two methods to Arrow's Condition I and the independence assumption of sample pairs described in Sect. 3.1.

We preparatorily examined the BSC. As described in Sect. 3.2, by comparing the results obtained by applying BSC with those obtained by applying CC, we found that the ordering component of CC is very close to that of BSC in terms of not only mathematical formulation, but performance in estimation.

Experimental results We first calculated the means of ρ for each example set, and the means of these means are shown in Table 3. The first column, labeled "all", shows the means of the ρ over all 810 example sets. The 810 example sets are then divided into three groups according to the length of the sample orders. Note that each group contains 270 example sets. The second, the third, and the fourth columns show the means over the example sets composed of sample orders with lengths of 3, 5, and 10, respectively. The following three columns show the means over the example sets containing 10, 30, and 50 training examples, respectively. For the CC and the TR, we also show the matrix of the ρ means, each of which entry shows the result over 90 example sets having the same order length and the same sample size as shown in Table 4. Even though no noises were added to orders or attributes, all methods failed to learn the rule that can perfectly recover the true orders, largely due to the fact that the sample orders are incomplete.

Overall, in accordance with increase in the length of sample orders or in the number of examples, the ρ grows bigger and more proper orders are estimated. It is a natural consequence that more accurate estimation is achieved when more information is provided. To check the accuracies of estimated orders, we then used the property of Eq. 1. When $|X_i|=10$, the probability that the ρ between a random order and the true order becomes larger than 0.715 is 1%. Based on the finding that the means in the column of $|X_i|=10$ were much larger than this value, we were able to conclude that all methods could produce the proper orders. In

Table 4 The means of ρ by CC and TR methods

		$|X_i|$	
	3	5	10
(a) Cohen classification (CC)			
10	0.529	0.723	0.900
N 30	0.699	0.864	0.949
50	0.776	0.899	0.961
(b) Thurstone regression (TR)			
10	0.459	0.776	0.940
N 30	0.702	0.886	0.969
50	0.750	0.907	0.975

Table 5 Difference between the mean ρ derived by the CC method and that derived by the other methods

	all	$|X_i|$			N		
		3	5	10	10	30	50
BSC	+0.348	+0.242	−0.347	$\overline{\overline{+3.073}}$	−0.147	+1.132	+0.619
TR	$\overline{\overline{+1.784}}$	$\underline{\underline{-3.091}}$	$\overline{\overline{+5.480}}$	$\overline{\overline{+13.312}}$	+0.738	$\overline{\overline{+3.221}}$	−0.414

the case of $|X_i|$=5, the probability that such a ρ becomes larger than 0.862 is 3%. Thus, it can be said that although estimation accuracy is degraded from the $|X_i|$=10 case, fully precise orders are estimated. However, in the $|X_i|$=3 case, the probability that ρ for random orders will become larger than 0.650 is about 27%. For such short orders, it could not to confirm whether fully appropriate orders could be estimated or not. In short, all three methods work well when the sample orders were sufficiently long.

We next examined paired t-tests to check for differences among these methods. Table 5 shows the t-values for the difference between means of ρ derived by the target CC method and those derived by the other methods. The positive t-values indicate the inferiority of the CC. The single-overline and double-overline (single-underline and double-underline) indicate that the compared methods are superior (inferior), and that the difference was statistically significant at the levels of 5% and 1%, respectively. In summary, the CC and the BSC were comparable, and the TR was superior to the other two. We next analyze these results more carefully.

Discussion on experimental results We first observed the relationship between the CC and BSC methods. In Sect. 3.2, these two methods are almost equivalent in terms of their mathematical formulation except in terms of their objective functions, Eqs. 2 and 3; the former is the sum, while the later the product of $\Pr[x_a \succ x_b | x_a, x_b]$. According to the results in Table 3, the performances on estimation were approximately equal. Further, according to the statistical test (the first row of Table 5), the significant differences are not observed in almost all the cases. We therefore would be able to analyze the properties of Cohen's ordering model by referring those of the Babington Smith (BS) model.

We here note two known properties of the BS model. First, the orders derived from this model violate Arrow's Condition I. For example, suppose three objects, x_a, x_b, and x_c, with

probabilities $\Pr[x_a \succ x_b] = 0.6$, $\Pr[x_a \succ x_c] = 0.1$, and $\Pr[x_b \succ x_c] = 0.9$, respectively. When sorting objects $\{x_a, x_b\}$, the BS model leads to the order $x_a \succ x_b$ because $Pr[x_a \succ x_b] = 0.6$ is the maximum among possible orders. However, for $\{x_a, x_b, x_c\}$, the order $x_b \succ x_c \succ x_a$ is selected because the probability is maximized at $Pr[x_b \succ x_c \succ x_a] = 0.324/Z$, where Z is a normalization constant. Accordingly, the order of x_a and x_b depends on the other object, x_c. This phenomenon is the case in terms of Cohen's ordering model. In the case of a Thurstone model, Arrow's Condition I is not violated; all the objects in X^* are sorted according to $\hat{r}(x_j)$, which depends only on x_j.

Second, in the BS model, it is assumed that ordered pairs of objects are independently generated. Based on the BS model, an order for given X_u is generated as follows: For every object pair x_a and x_b in X_u, either $x_a \succ x_b$ or $x_b \succ x_a$ is *independently* chosen with the respective probabilities $\Pr[x_a \succ x_b]$ and $\Pr[x_b \succ x_a]$. If the generated ordered pairs do not violate transitivity and can be integrated into one consistent order, the integrated order is generated; otherwise these ordered pairs are all abandoned and re-generated. This process is repeated until a consistent order is generated. As described above, this model strongly assumes independence among the ordered pairs. This independence would be assumed in Cohen's model, too. On the other hand, in a Thurstone model, scores of each object are assumed to be independently generated; thus, the independence assumption among object pairs is less influential.

We then turn to our main goal of these experiments, the comparison of the CC and TR methods. According to Tables 3 and 4, the TR method is generally more effective than the CC. We consider that the advantages of the TR method can be explained as follows. First, as described in Sect. 3.1, the task and the method of Cohen et al. postulate Arrow's Condition I, whereas their ordering model violates this condition. Such contradiction is not seen in our TR method. We consider this difference evidence of the overall advantage of the TR.

The other cause originates from the fact that an ordering model of the CC more strongly assumes independence among the ordered pairs. This hypothesis would be supported by the experimental results in Table 5. The CC method is superior when the length of the sample orders, $|X_i|$, is short. However, in accordance with the increase of $|X_i|$, the TR method becomes superior. This result is particularly contrasted with the fact that no clear difference was observed between the two methods in terms of changes in the number of examples, N. When $|X_i|$ is short, the number of pairs satisfying the transitivity property are fewer, so the independence assumption among pairs is relatively satisfied. However, in correspondence with the increase of $|X_i|$, the independence assumption is more frequently violated. The longer the length of orders becomes, the more advantageous the TR method is.

Finally, we show an additional experimental result. The CC method adopted a greedy search for finding the order that maximizes the criteria (Eq. 2), so the optimal solution may not be acquired. Thus, we investigated the difference between the result derived by the greedy search in Table 2 and the optimal result derived by the exhaustive search. The t-value of the ρ derived by the optimal search minus the one derived by the greedy search is -2.273. Surprisingly, the ρ derived by the greedy search is significantly better at the level of about 1%. This result indicates that finding the order so that winners in pairwise comparisons win in the total order is not equivalent to finding the orders totally concordant with sample orders. This result enhances the difference between the goal of a Cohen's task and that of a supervised ordering task. Note that we cannot clearly explain the above advantage of the greedy search, but can present one hypothesis. With regard to the sequential selection of the most preceding objects, the ordering model of the greedy search is not a pure BS model. This sequential process can be seen in a Plackett–Luce model (Plackett 1975, Marden 1995), and this model does not violate Arrow's Condition I. We consider that sequential selection resulted in comparatively fewer violations of Arrow's Condition I, and made the greedy search more advantageous.

Table 6 The means of ρ for the example sets with order noise

	0%	1%	3%	5%	10%
CC	0.811	0.809 (0.998)	0.805 (0.993)	0.802 (0.989)	0.792 (0.976)
BSC	0.811	0.810 (0.998)	0.805 (0.992)	0.803 (0.989)	0.792 (0.976)
TR	0.818	0.817 (0.999)	0.814 (0.995)	0.810 (0.989)	0.797 (0.974)

Table 7 The means of ρ for the example sets with attribute noise

	0%	1%	3%	5%	10%
CC	0.811	0.805 (0.992)	0.798 (0.984)	0.795 (0.980)	0.778 (0.960)
BSC	0.811	0.806 (0.993)	0.798 (0.984)	0.795 (0.980)	0.779 (0.960)
TR	0.818	0.815 (0.996)	0.811 (0.992)	0.800 (0.978)	0.776 (0.948)

4.1.3 Experimental results on artificial data with noise

We next tested the robustness of our method against the two types of noise. One is the noise caused when two adjacent objects are randomly permuted. In Table 6, we show the means of ρ in the example with a 0% to 10% noise level. The noise level is the ratio of the number of permutated adjacent pairs to the total number of adjacent pairs. The other is the noise that attribute values flip to another value. Table 7 shows the means of ρ in the example sets with a 0% to 10% noise level. The ratio of the ρ to the 0% case is shown in parentheses. Overall, in terms of robustness against noise, the performance in the case of all the methods was degraded to the same degree relative to the increasing level of noise.

4.2 Experiments on real data

To test the performance of supervised ordering methods on real data, we applied the CC and the TR to real data from the questionnaire survey of preferences in sushi, which were used for the tasks of collaborative filtering (Kamishima 2003) and clustering (Kamishima and Fujiki 2003).

We asked each respondent to sort 10 objects (i.e., types of sushi) according to his/her preference. Such a sensory survey is a very suitable topic for analysis based on orders. The objects were randomly selected from 100 objects according to their probability distribution, based on menu data from 25 sushi restaurants found on the WWW. For each respondent, the objects were randomly permutated to cancel the effect of the display order on their responses. The total number of respondents was 1039. We eliminated the data obtained within a response time that was either too short or too long. Consequently, data for 1025 respondents was extracted. In Table 8, we show the attributes of the objects. The first and second attributes are categorical, the others numerical. These numerical values are discretized by dividing the ranges between the maximum and the minimum into five bins of equal width. In contrast to the case of artificial data, for this data set we could not know true orders. We therefore calculated the empirical error; that is, the ρ between a sample order, O_i, and an estimated one, $\hat{O}_i$.

To select the proper attribute sets, we examined all combinations of attributes, and the best result was obtained when eliminating the 6-th attribute. We thus show the result using the 1-st through the 5-th attributes. The means of ρ by performing the LVO test were 0.415 for

Table 8 Attributes of objects (sushi)

No.	Descriptions	$|v_l|$
1	Style of the sushi: *hand-shaped* or *rolled*	2
2	Category of the sushi (e.g., blue fish, vegetables, ...)	12
3	Heaviness or oiliness of the taste of sushi	5
4	Frequency with which the respondents eat the sushi	5
5	Normalized prices of the sushi	5
6	Frequency with which sushi restaurants supply the sushi	5

Table 9 Weights of the model learned by the `TR` method

Attribute	Low		Med		High
Attr No.3: oiliness	–2.88	–1.55	0.54	–7.30	0.00
Attr No.4: eating frequency	0.00	–15.66	–42.14	–62.48	–70.64
Attr No.5: price	0.00	2.52	0.94	–6.92	–3.80

the `CC` and 0.409 for the `TR`. These performances appear degraded in comparison with the results on noiseless artificial data, because the performances are measured by using empirical errors that include variations of samples in addition to true errors. According to Eq. 1, the probability that the ρ between the random and the sample orders becomes larger than 0.409 is about 12.1%. Thus, the estimated orders are clearly better than the baseline results; i.e., better than the random orders. We consider this level of performance to be sufficient for use in the analysis of preference data. The t-value for the differences between the ρ is +1.532, which indicates that the `CC` appeared to outperform, though the difference was not significant even at the level of 5%. We observed an advantage of the `TR` method in the results on artificial data when the length of sample orders was sufficiently long, but no such advantage was found in this result. We suppose that the comparison between empirical errors tended to be affected by a higher level of noise, and the difference became less clear.

Finally, further analysis can be performed by observing the estimated orders and other related data. Table 9 summarizes the weights, w, associated with each of the attribute values of the third, fourth, and fifth attributes. These weights of the regression models are learned by the `TR`. The smaller the weight of the attribute value, the more preferred the sushi. Based on these weights, rather oily, though not the most oily, types of sushi are preferred. The respondents preferred the types of sushi they ate most frequently. Expensive, but not the most expensive, types of sushi tended to be preferred. Another example of the result of such analysis is this: because the ρ of my (the first author Kamishima's) preference order is 0.564, it can be concluded that my preference in sushi is the moderately typical type.

5 Conclusion

In this paper, we describe a new learning task and present its solution method: Thurstone Regression. We applied this method to both artificial and real data, respectively, and analyzed the experimental results. The results revealed that all the methods can result in placing objects

in the proper order, provided that the sample orders are sufficiently long. The advantages of the TR over the CC become clearer in correlation with increase in the length of sample orders. We articulate the hypothesis that this difference is brought about by the independence assumption among object pairs and/or the violation of Arrow's Condition I, and showed some rationales for this hypothesis.

We plan to extend the TR method to deal with numerical attributes, and/or to adapt it to non-linear models. We found that the factor $|X_S|/|X^*|$ affects the estimation performance, and intended to investigate this aspect further.

References

Agresti A (1996) Categorical data analysis. Wiley

Anderson JA, Philips PR (1981) Regression, discrimination and measurement models for ordered categorical variables. J R Stat Soc (C) 30(1):22–31

Arnold BC, Balakrishnan N, Nagaraja HN (1992) A first course in order statistics. Wiley

Arrow KJ (1963) Social choice and individual values, 2nd edn. Yale University Press

Babington Smith B (1950) Discussion on professor ross's paper. J R Stat Soc (B) 12:53–56, (A. S. C. Ross, Philological Probability Problems, pp 19–41)

Cohen WW, Schapire RE, Singer Y (1999) Learning to order things. J Artif Intel Res10:243–270

de Condorcet M (1995) On constitution and the functions of provincial assemblies (1788). In: McLean I, Urken AB (eds) Classics of social choice, The University of Michigan Press, chap 7, pp 113–144

Fligner MA, Verducci JS (1990) Posterior probabilities for a consensus ordering. Psychometrika 55(1):53–63

Hayashi C (1952) On the prediction of phenomena from qualitative data and the quantification of qualitative data from the mathematico-statistical point of view. Ann Inst Stat Math 3:69–98

Herbrich R, Graepel T, Bollmann-Sdorra P, Obermayer K (1998) Learning preference relations for information retrieval. In: ICML-98 Workshop: text categorization and machine learning, pp 80–84

Joachims T (2002) Optimizing search engines using clickthrough data. In: Proceedings of the 8th international conference on knowledge discovery and data mining, pp 133–142

Kamishima T (2003) Nantonac collaborative filtering: Recommendation based on order responses. In: Proceedings of the 9th International conference on knowledge discovery and data mining, pp 583–588

Kamishima T, Akaho S (2004) Filling-in missing objects in orders. In: Proceedings of the IEEE International conference on data mining

Kamishima T, Fujiki J (2003) Clustering orders. In: Proceedings of the 6th International conference on discovery science, pp 194–207, [LNAI 2843]

Kendall M, Gibbons JD (1990) Rank correlation methods, 5th edn. Oxford University Press

Lebanon G, Lafferty J (2002) Crankng: combining rankings using conditional probability models on permutations. In: Proceedings of the 19th International conference on machine learning, pp 363–370

Mallows CL (1957) Non-null ranking models. I Biometrika 44:114–130

Mannila H, Meek C (2000) Global partial orders from sequential data. In: Proceedings of the 6th International conference on knowledge discovery and data mining, pp 161–168

Marden JI (1995) Analyzing and modeling rank data, monographs on statistics and applied probability, vol 64. Chapman & Hall

McCullagh P (1980) Regression models for ordinal data. J R Stat Soc (B) 42(2):109–142

Mitchell TM (1997) Machine learning. The McGraw-Hill

Mosteller F (1951) Remarks on the method of paired comparisons: I—the least squares solution assuming equal standard deviations and equal correlations. Psychometrika 16(1):3–9

Nakamori Y (2000) Kansei data Kaiseki. Morikita Shuppan Co., Ltd., (in Japanese)

Osgood CE, Suci GJ, Tannenbaum PH (1957) The Measurement of meaning. University of Illinois Press

Plackett RL (1975) The analysis of permutations. J R Stat Soc (C) 24(2):193–202

Sai Y, Yao YY, Zhong N (2001) Data analysis and mining in ordered information tables. In: Proceedings of the IEEE International conference on data mining, pp 497–504

Thurstone LL (1927) A law of comparative judgment. Psychol Rev 34:273–286