Noname manuscript No.
(will be inserted by the editor)

Median constrained bucket order rank aggregation

**Antonio D'Ambrosio · Carmela Iorio ·
Michele Staiano · Roberta Siciliano**

Received: date / Accepted: date

Abstract The rank aggregation problem can be summarized as the problem of aggregating individual preferences expressed by a set of judges to obtain a ranking that represents the best synthesis of their choices. Several approaches for handling this problem have been proposed and are generally linked with either axiomatic frameworks or alternative strategies. In this paper, we present a new definition of median ranking and frame it within the Kemeny's axiomatic framework. Moreover, we show the usefulness of our approach in a practical case about triage prioritization.

Keywords Tied rankings · Median ranking · Kemeny distance · Triage prioritization

1 Introduction

Preference rankings are data that express an individual's preferences in terms of a set of available alternatives. These data can be expressed either through order vectors (or orderings; when alternatives are placed in order from best to worst) or rank vectors (or rankings; when alternatives are fixed in any prespecified order and preferences are expressed by using integers to indicate the rank of each alternative) (Marden, 1996). As the meaning is the same, we will use both terms as synonymous with rankings. Suppose a set of n items must be ranked by m judges. When a judge gives a complete and strict precedence ranking of the items, thus

This work has been partially supported by the H2020-EU.3.5.4. project 'Moving Towards Adaptive Governance in Complexity: Informing Nexus Security (MAGIC)', grant agreement number 689669.

Antonio D'Ambrosio
Department of Economics and Statistics
University of Naples Federico II, Naples, Italy
E-mail: antdambr@unina.it

Carmela Iorio
Department of Industrial Engineering
University of Naples Federico II, Naples, Italy
E-mail: carmela.iorio@unina.it

Michele Staiano
Department of Industrial Engineering
University of Naples Federico II, Naples, Italy
E-mail: michele.staiano@unina.it

Roberta Siciliano
Department of Industrial Engineering
University of Naples Federico II, Naples, Italy
E-mail: roberta@unina.it

producing a permutation of the first n integers, the resulting ranking is called *complete* (or full). Individuals could assign the same integer to two or more items. In this case, the resulting ordering is called *tied* (or weak). When judges are asked to rank only a subset of the entire set of objects (or when the rank associated with some items is missing), the resulting ordering is called *partial* ranking (Heiser and D'Ambrosio, 2013). Sometimes tied rankings are called *bucket orders* (Gionis et al., 2006; Ukkonen et al., 2009; Kenkre et al., 2011), when a set of items is tied for a given location. Many statistical analyses can be performed with preference rankings and paired comparison rankings; examples include inference on top-k lists (Hall and Schimek, 2012; Sampath and Verducci, 2013), cluster analysis and related techniques (Murphy and Martin, 2003; Busse et al., 2007; Heiser and D'Ambrosio, 2013; Brentari et al., 2016), supervised classification methods (D'Ambrosio, 2008; Lee and Yu, 2010; D'Ambrosio and Heiser, 2016; Plaia and Sciandra, 2017), multivariate analysis (Cohen and Mallows, 1980; Busing et al., 2005; Busing, 2006; Yu et al., 2013), and probability models (Bradley and Terry, 1952; Fienberg and Larntz, 1976; Dittrich et al., 2000; Yu et al., 2016). For a detailed overview of statistical methods and models for preference rankings, refer to Marden (1996).
One of the problems when dealing with preference rankings is the identification of the so-called consensus ranking, i.e., the ranking that best synthesizes the consensus opinion. Depending on the reference framework, this problem is known as a social choice problem, a rank aggregation problem, a median ranking problem, a central ranking detection, or a Kemeny problem (Kemeny and Snell, 1962; Marden, 1996; Fürnkranz and Hüllermeier, 2010; Amodio et al., 2016). This (NP-hard) problem has received increasing importance over the years, both as a main research task and as an essential starting point for other types of analysis (Corain and Salmaso, 2007; Fields et al., 2013; Heiser and D'Ambrosio, 2013; Desarkar et al., 2016; D'Ambrosio and Heiser, 2016; Telcs et al., 2016; Sciandra et al., 2017; Dopazo and Martínez-Céspedes, 2017; Švendová and Schimek, 2017). The rank aggregation problem has been approached using several different approaches, some of which work only when complete rankings are used as the input and produce an output that is a full consensus ranking (Meila et al., 2007; Aledo et al., 2013; D'Ambrosio et al., 2015). Other proposals work with complete, tied and partial rankings and produce an output solution that either can contain ties (Emond and Mason, 2002; Gionis et al., 2006; Lin and Ding, 2009; Ukkonen et al., 2009; Lin, 2010; Amodio et al., 2016; D'Ambrosio et al., 2017; Aledo et al., 2017b) or cannot contain ties (Aledo et al., 2017a; Badal and Das, 2018). By following the classification made by Cook (2006), there are two broad classes of approaches to consensus ranking: the so-called *ad hoc* methods, which are generally based on counting such as Borda or Condorcet-like tools, and the distance-based approaches, for which the detection of the consensus ranking is based on the minimization of a distance measure that is suitably defined for preference rankings. Recently, 'distance-free' methods have been introduced in the literature (Hall and Schimek, 2012; Švendová and Schimek, 2017). Within the category of distance-based techniques, several axiomatic approaches have been proposed (Kemeny and Snell, 1962; Cook et al., 1997; Gionis et al., 2006; Biernacki and Jacques, 2013), including both MCMC and Bayesian methods (Dwork et al., 2001; Deng et al., 2014; Asfaw et al., 2017). We apply in the Kemeny's axiomatic framework (Kemeny and Snell, 1962) and assume that the geometrical space of preference rankings is the permutation polytope (Thompson, 1993). When dealing with tied rankings, our reference geometrical space is the generalized permutation polytope (Heiser and D'Ambrosio, 2013; D'Ambrosio et al., 2017), for which the natural distance measure is the Kemeny distance. We assume that the consensus ranking is the median ranking defined as that ranking (or those rankings) that minimizes the sum of the Kemeny distance between itself and each of the rankings expressed by a set of m judges. For an extensive discussion on this approach, refer to (Emond

and Mason, 2002; Heiser and D'Ambrosio, 2013; Amodio et al., 2016; D'Ambrosio et al., 2017).
In this paper, we define a novel problem of rank aggregation: finding the median ranking under the constraint that the final solution must contain a prespecified number of buckets. Our proposal has been motivated by the search for a solution to a real problem concerning the choices made by a set of nurses within the so-called triage prioritization.
It is worth noting that we are introducing a special situation of rank aggregation that, as far we know, has never been approached in the literature. Tied rankings (or bucket orders) were considered as indifferent declarations until a few years ago. For a long time, ranking data have been synonymous with permutations. Currently, dealing with tied rankings is the rule rather than an exception in many real situations. We show that one may be interested in constrained solutions in certain situations. For instance, according to the Bordeaux Official Wine Classification, wines are ranked in quality from their first growth to their fifth (Premier Cru,..., Cinquieme Cru). In that wine tasting experiment, the final solution is requested to be constrained into five buckets. We propose a solution to the stated problem by modifying some tools that were originally designed to solve the rank aggregation problem without any limitation in terms of the nature of the solution, in the sense that the solution can be either a full ranking or a tied ranking.
The paper is organized as follows: Section 2 introduces the median ranking problem, and Section 3 describes the modified algorithms to cope with the prespecified bucket orders. In Section 4 we show the potential of our proposal both on some well-known datasets and on the triage dataset. We conclude the paper with some critical discussions in Section 5.

2 Kemeny distance and median ranking

Cardinality of the universe of rankings with n items is equal to

$$\mathcal{Z}^n = \sum_{b=1}^{n} b! \left\{ n \atop b \right\}, \tag{1}$$

where $\left\{ n \atop b \right\}$ indicates the Stirling number of the second kind, which corresponds to the number of ways to partition a set of n objects into b nonempty subsets. These b nonempty subsets correspond to the buckets. Table 1 shows the cardinality of the universe of rankings from $n = 1$ to 5 items.

Table 1 Cardinality of the universe of rankings that contain ties for $n = 1, 2, \ldots, 5$ (last column). The columns that indicate the buckets (b) show the cardinality of the rankings of n items constrained in b buckets

n \b	1	2	3	4	5	...	$\mathcal{Z}^n$
1	1	-	-	-	-	-	1
2	1	2	-	-	-	-	3
3	1	6	6	-	-	-	13
4	1	14	36	24	-	-	75
5	1	30	150	240	120	-	541
...	...	...	...	...	...	...	

Let Y and X be two rankings of n items. The Kemeny distance can be computed as follows: build two 1 by $n \times (n-1)$ vectors $\mathbf{y}$ and $\mathbf{x}$ in which each column represents one pair of objects. After the computation of $y_{[KL]} = sign(y_K - y_L)$ and

$x_{[KL]} = sign(x_K - x_L)$, $K > L = 1, \dots, n$, the Kemeny distance is

$$d(Y,X) = \sum_{[KL]=1}^{\frac{1}{2}n(n-1)} \left| y_{[KL]} - x_{[KL]} \right|. \tag{2}$$

Kemeny and Snell (1962) proved that this measure is the unique distance that satisfies a set of axioms[1] that a distance metric defined on preference rankings should satisfy. Heiser and D'Ambrosio (2013) pointed out that, when ties are not allowed, the Kemeny distance and the Kendall's τ distance are equivalent. Let $\mathbf{X}$ be a $m \times n$ data matrix containing m rankings of n objects. The median ranking $\hat{Y}$ is that ranking for which:

$$\hat{Y} = \underset{Y \in \mathcal{Z}^n}{\arg\min} \sum_{k=1}^{m} d(X_k, Y). \tag{3}$$

Emond and Mason (2000, 2002) defined a rank correlation coefficient called τ_X that can deal with tied rankings and proved that it is naturally linked to the Kemeny distance in this way:

$$\tau_X(X,Y) = 1 - 2\frac{d(X,Y)}{(n(n-1))}.$$

They conceived a branch-and-bound algorithm that can find the median ranking by maximizing the τ_X instead of minimizing the Kemeny distance. Their starting point was the definition of a matrix representation of a ranking called *score matrix* by considering each possible pairwise comparison of the n items to be ranked. A score matrix was already defined by both Kendall (1938) and Kemeny and Snell (1962), but Emond and Mason (2002) slightly modified the coding of the pairs of items in a tie. Let Y be a ranking and $\mathbf{Y}$ be the corresponding $n \times n$ score matrix with elements y_{ij}, $i, j = 1, \dots, n$. Emond and Mason assigned the score 1 if the i-th object is either ranked ahead of or in a tie with the j-th item and -1 otherwise. In this way, they defined the rank correlation coefficient between two rankings X and Y as

$$\tau_X(X,Y) = \frac{\sum_{i,j=1}^{n} x_{ij} y_{ij}}{n(n-1)}. \tag{4}$$

Let $X^{(1)}, \dots, X^{(h)}$ be a set of rankings that are not necessarily distinct, each with an associated weight w_h, with $\sum_{h=1}^{k} w_h = m$. Emond and Mason proposed their branch-and-bound algorithm by first defining the *combined input matrix* $\mathbf{C}$ with elements c_{ij}, i.e., a score matrix that aggregates the individual score matrices of the m judges:

$$\mathbf{C} = \sum_{h=1}^{k} x_{ij}^{(h)} w_h. \tag{5}$$

Then, they stated the rank aggregation problem as the identification of that ranking (or those rankings) $\hat{Y}$ for which:

$$\hat{Y} = \underset{Y \in \mathcal{Z}^n}{\arg\max} \frac{\sum_{i,j=1}^{n} c_{ij} y_{ij}}{\sum_{h=1}^{k} w_h \left(n\left(n-1\right)\right)}. \tag{6}$$

Thus, the right-hand side of Equation 6 corresponds to the average of the τ_X rank correlation coefficient computed between the median ranking $\hat{Y}$ and all the rankings stated by the m judges.

[1] The first axiom states that the distance measure must be a metric. The second axiom is about the invariance of the distance under a random permutations of the items. The third axiom is about consistency in measurement: the distance between two rankings does not change after deleting a set of items that agrees in rank for both rankings. The last axiom is the statement of a measurement unit: the minimum distance is equal to one.

3 Median constrained bucket orders

The so-called *optimal bucket order problem* (Gionis et al., 2006; Ukkonen et al., 2009; Kenkre et al., 2011; Aledo et al., 2017b), i.e., dealing with rank aggregation while allowing ties in the solution, is a recent description of the problem originally stated by Kemeny and Snell (1962) when defined the median ranking. Within the Kemeny's axiomatic approach, both exact (Emond and Mason, 2002) and heuristic algorithms (Amodio et al., 2016; D'Ambrosio et al., 2017) have been proposed. These algorithms, no matter the nature of the rankings input, search for the best solution in $\mathcal{Z}^n$. Here, we introduce a modification of both the Emond and Mason's branch-and-bound algorithm and the differential evolution algorithm of D'Ambrosio et al. to produce a consensus ranking by searching for the solution only in a subset of $\mathcal{Z}^n$ that contains *exactly* b buckets. We call this solution the *median constrained bucket order*. The problem can be formalized as follow: let $X^{(1)}, \ldots, X^{(k)}$ be a set of rankings of n items, each bearing a weight w_h, with $\sum_{h=1}^{k} w_h = m$. The constrained median bucket order is that ranking (or those rankings) $\hat{Y}$ for which

$$\hat{Y} = \underset{Y \in \mathcal{Z}^{n \setminus b}}{\arg\min} \sum_{h=1}^{k} w_h d(X^{(h)}, Y) = \underset{Y \in \mathcal{Z}^{n \setminus b}}{\arg\max} \frac{\sum_{i,j=1}^{n} c_{ij} y_{ij}}{m\,(n\,(n-1))}, \tag{7}$$

where $\mathcal{Z}^{n \setminus b}$ is the subset of $\mathcal{Z}^n$ in which there are *exactly* b buckets. Note that, theoretically, there are no restrictions on the nature of the input rankings, in the sense that input rankings do not necessarily have to be bucket orders.

3.1 Modification of branch-and-bound and BB-like algorithms

Emond and Mason (2000, 2002) conceived a branch-and-bound (BB) algorithm to maximize Equation 6 by defining an upper limit on the value of the dot product of the right side of the equation. The combined input matrix, as defined in Equation 5, plays a key role in the BB algorithm because it contains all the information that a sample of judges can provide in the rank aggregation problem. Regardless of the number of judges, the dimensionality of the problem is governed by the number of items and the form of the combined input matrix. The original Emond and Mason's algorithm can be summarized as follows: starting from an initial solution Y, the algorithm starts by creating three mutually exclusive branches based on the relative position of the first two objects, with either one preferred to the other or in a tie. An initial penalty is then computed as $V - B$, with $V = \sum_{i,j}^{n} |c_{ij}|$ and $B = \sum_{i,j=1}^{n} y_{ij} c_{ij}$ relative to the initial solution. Then an incremental penalty δP for each of the branches is computed by considering the corresponding elements c_{ij} and c_{ji} of $\mathbf{C}$ as summarized by the following schema:

- object i is preferred to object j (Branch 1):
  if $c_{ij} > 0$ *and* $c_{ji} < 0$, then $\delta P = 0$
  if $c_{ij} > 0$ *and* $c_{ji} > 0$, then $\delta P = c_{ji}$
  if $c_{ij} < 0$ *and* $c_{ji} > 0$, then $\delta P = c_{ji} - c_{ij}$
- object i is tied with object j (Branch 2):
  if $c_{ij} > 0$ *and* $c_{ji} < 0$, then $\delta P = -c_{ji}$
  if $c_{ij} > 0$ *and* $c_{ji} > 0$, then $\delta P = 0$
  if $c_{ij} < 0$ *and* $c_{ji} > 0$, then $\delta P = -c_{ij}$
- object j is preferred to object i (Branch 3):
  if $c_{ij} > 0$ *and* $c_{ji} < 0$, then $\delta P = c_{ij} - c_{ji}$
  if $c_{ij} > 0$ *and* $c_{ji} > 0$, then $\delta P = c_{ij}$
  if $c_{ij} < 0$ *and* $c_{ji} > 0$, then $\delta P = 0$.

Each branch with a penalty is greater than (or equal to) the initial penalty, is skipped and not considered. The remaining branches are processed by placing the third highest ranked object in all possible positions relative to the objects already considered. This process continues until all remaining objects are evaluated. The penalty is iteratively updated according to either the concordance or the discordance of the values of the score matrix of the candidate median ranking with the ones of the combined input matrix. To find the solution in the restricted ranking universe that is constrained to be in a prespecified number of buckets, we restrict the searching space in the ranking universe to n items by cutting all branches that contain a number of groups of ties larger than b, thereby supposing a number b of buckets. The algorithm then continues as in the original formulation. The result is the median ranking, i.e., the ranking (or rankings) with the highest average τ_X. For a detailed discussion of the BB algorithm, refer to Emond and Mason (2000, 2002). As highlighted by Brusco and Stahl (2006), BB algorithms guarantee to find the optimal solution, but in the worst case the complexity is as high as that of exhaustive search.
Amodio et al. (2016) proposed a modification of Emond and Mason's BB algorithm that achieves remarkable savings in terms of computational burden. Given an input ranking X and the combined input matrix $\mathbf{C}$, the algorithm, called Quick, evaluates the penalty by considering all items in the input ranking; meanwhile, in the original BB formulation, the penalty is computed by considering only the elements of the combined input matrix associated with the processed items. The input ordering X is given as a random rank vector of length n that contains only the first b integers, i.e., a random candidate median constrained bucket order. Then, every time an object is added in the iterative process, the algorithm stores the ranking, which is constrained to b buckets, removes all other rankings in the processed branch, and processes only the bucket order associated with the minimum current penalty. Algorithm 1 shows the pseudocode of the Quick algorithm for constrained bucket order detection.

Algorithm 1: QUICK algorithm for constrained bucket order detection

input : $\mathbf{C}$,X

1 **initialize:** fix the rank of the first ranked object in X;
2 consider the next ranked object in X;
3 evaluate the penalty for all the rankings obtained by placing that object in all possible positions with respect to the fixed ranked objects;
4 store only the ranking *constrained into b buckets* associated with minimum penalty;
5 fix the rank of the processed object and return to step 2 until all objects in X are processed;
6 obtain the update ranking Y, and repeat all previous steps by replacing X with Y (once);

output : $\hat{Y}$ = median constrained bucket order

The output of the Quick algorithm is sensitive to the random choice of the initial ranking. Moreover, it can get stuck in local optima. Thus, the algorithm is repeated several times and returns the best solution(s) among the iterations. For an extensive discussion of the behavior of the Quick algorithm and comparisons in terms of its performance with respect to the BB algorithm, refer to Amodio et al. (2016).

3.2 Differential evolution algorithm

The differential evolution algorithm for median ranking detection (DECoR) (D'Ambrosio et al., 2017) can be used for problems with a (very) large number of items to be ranked. The BB algorithm has an important limitation due to the number of the items to be ranked: if there are more than approximately 15 or 20 objects, the algorithm can run for hours or days (Emond and Mason, 2000; Amodio et al., 2016).

The Quick algorithm is much faster than BB and works for problems up to 50 or even 100 objects. Recently, D'Ambrosio et al. (2017) proposed a discretization step for the differential evolution algorithms to allow for a discrete optimization problem such as the detection of the median ranking. Differential evolution algorithms consist of the following steps: initialization, mutation, crossover and selection. The DECoR algorithm starts by creating a population of rankings that provides random permutations of the first n integers. The adopted mutation strategy is the so-called */rand/1/bin*, where *rand* means that the vector to be permuted is randomly chosen, 1 is the number of difference vectors considered for perturbation and *bin* means that the type of crossover step is set as binomial (Storn and Price, 1995). For discretization step, the hierarchical approach (which considers the ranks of the values) has been preferred to the closest integer approach because the latter, considering the integer value closest to the real obtained value, can produce inconsistent values and further steps are required to fix this problem (Onwubolu and Davendra, 2009; D'Ambrosio et al., 2017). Equation 8 shows the cost function that is minimized by the DECoR algorithm, which can be directly computed by exploiting the properties of the combined input matrix using Equations 4 and 7:

$$cost(Y) = \sum_{h=1}^{k} w_h \frac{n(n-1)}{2} \left(1 - \tau_X(\mathbf{C}, \mathbf{Y})\right), \tag{8}$$

where $\tau_X(\mathbf{C}, \mathbf{Y})$ represents the averaged τ_X associated with the ranking Y.
To allow the user to obtain the best median ranking with a prespecified number of buckets, we slightly modified the original DECoR algorithm. As an initialization step, we create a population of dimension P by randomly sampling vectors with n columns from a discrete uniform distribution with support $1, 2, \ldots, b$, with b indicating the number of buckets. Then, we use the closest integer approach as a discretization step. The algorithm then continues as in the original formulation. The result is the ranking (or rankings) associated with the highest average τ_X. As the DECoR algorithm is sensitive to the random initial population, the obtained solutions can be local optima. For this reason, the algorithm is run several times and returns the best solution among the iterations. For a detailed discussion on the DECoR algorithm and a comparison of DECoR and a large variety of proposals dealing with the rank aggregation problem within Kemeny's axiomatic framework, refer to D'Ambrosio et al. (2017).

4 Median constrained bucket order in practice

We evaluated our proposal using both simulation and applications on real data. Note that our proposal can be appreciated if there is motivation to restrict the search of the median ranking in a restricted universe. The simulation study is devoted to checking the behavior of the algorithms in terms of both computing time and accuracy. It is worth stressing that the main goal of this work is to introduce the concept of median constrained bucket order. For this reason, many algorithms and methods already known in the literature can be adapted to find the solution in the constrained restricted space of bucket orders, provided that they can deal with tied rankings. We show the behavior of the BB algorithm when possible, as well as that of both Quick and DECoR. A comparison with other proposals is not in the scope of this work because modifying the search space does not modify the already highlighted results (see, for example, the study conducted by D'Ambrosio et al. (2017) and Badal and Das (2018)).
The first experiment on real data only shows the practical use of our approach without any theoretical reason for the choice of the number of buckets. The second

experiment is an example of a correct choice of the number of buckets and shows a decision support strategy aimed to improve the performance of a given service.
Analysis were performed on a laptop with an Intel Core i7 5500U processor with 8.00 GB of RAM. Algorithms have been implemented in both MATLAB and R environments. The version of R will be soon available as an extension of the R package *ConsRank* (D'Ambrosio et al., 2016). The version of MATLAB can be downloaded from `http://wpage.unina.it/antdambr/Software/Matlab/BucketConsRank`.

4.1 Simulation study

We generated datasets by considering a number of objects n equal to 10 and 50. When $n = 10$ we considered two levels of buckets: $b = 3$ and $b = 5$. When $n = 50$, we considered two levels of buckets: $b = 5$ and $b = 7$. The number of judges m was randomly chosen between $m = 15$ and $m = 75$. We considered theoretical situations in which the sample of rankings are all generated with the same prespecified number of buckets to try to reproduce an experiment in which a set of m judges is asked to rank n items in exactly b buckets. We designed three levels of noise, i.e., high, moderate and low, according to these rules:

- $n = 10$;
  - generate ranking universe $\mathcal{Z}^{10/b}$ of 10 elements constrained into b buckets;
  - randomly select a ranking Y from the universe and compute the Kemeny distance between it and all the rankings in the universe;
  - compute the probability of each ranking to be sampled according to the exponential function $P(X_i) = exp(-\lambda d(Y, X_i))/\sum_{X_i \in \mathcal{Z}^{n/b}} exp(-\lambda d(C, X_i))$, where X_i is the i-th ranking in the universe. Similar to the Mallows model (Mallows, 1957), the parameter λ governs the spread of the rankings around the theoretical center Y. We set the values of λ equal to 0.1, 0.4 and 0.8 for the cases of high, moderate and low noise, respectively;
  - generate the dataset by sampling from $\mathcal{Z}^{10/b}$ according to the distribution P.
- $n = 50$;
  - randomly generate a rank vector Y of length 50 containing only the first b integers;
  - iteratively generate a random rank vector X of length 50 containing only the first b integers and compute the Kemeny distance between Y and X. If $d(X, Y) \leq k$ store the ranking X, otherwise continue. Stop the procedure when the dataset counts m distinct rankings. The value of k governs the level of noise. As the maximum Kemeny distance is equal to $md = n(n-1)$, we set the values of k equal to $0.5md$, $0.25md$ and $0.15md$ for the cases of high, moderate and low noise, respectively.

For each level of noise, each value n and each level of b, we generated 10 datasets, for a total of 120 datasets. The BB algorithm was run only in the case of 10 items. In this case, the branch-and-bound algorithm serves as a benchmark for the other algorithms. Table 2 shows the average of both the computing time and the τ_X rank correlation coefficient associated with the median constrained bucket order. As expected, the lower the noise, the larger the τ_X, indicating that rankings are more concentrated around the true median constrained bucket order. Regarding the heuristic algorithms, DECoR outperforms the Quick algorithm because it always finds the same solution as BB. Both DECoR and Quick were run 100 times.
Table 3 shows the performances of DECoR and Quick in the simulations with $n = 50$ items. In this case, DECoR works better because, on average, the τ_X rank correlation coefficient associated with the solution has results that are equal or greater than

Table 2 Simulation setting with $n = 10$ items: performance of BB, Quick and DECoR. $\mathcal{Z}^{n/b}$H, $\mathcal{Z}^{n/b}$M and $\mathcal{Z}^{n/b}$L indicate the conditions with n items, b buckets and situations with high (H), moderate (M) and low (L) noise.

Setting		$\mathcal{Z}^{10/3}$ H	$\mathcal{Z}^{10/3}$ M	$\mathcal{Z}^{10/3}$ L	$\mathcal{Z}^{10/5}$ H	$\mathcal{Z}^{10/5}$ M	$\mathcal{Z}^{10/5}$ L
Computing time in seconds	BB	0.26	0.29	0.14	0.52	1.26	0.67
	Quick	3.84	3.78	3.79	5.14	5.16	5.20
	DECoR	14.40	14.54	14.47	15.22	15.47	15.71
Averaged τ_X	BB	0.37	0.78	0.97	0.30	0.64	0.91
	Quick	0.36	0.78	0.97	0.29	0.64	0.91
	DECoR	0.37	0.78	0.97	0.30	0.64	0.91

those recorded for Quick. In terms of computing time, the Quick algorithm is faster than DECoR. This result is already known: as highlighted in the study conducted by D'Ambrosio et al. (2017), the Quick algorithm should be preferred to DECoR for problems up to 50 items. Here, we observe that the algorithms are not equivalent in terms of accuracy unless we consider a 'standard' rank aggregation problem (see D'Ambrosio et al., 2017).

Table 3 Simulation setting with $n = 50$ items: performance of Quick and DECoR. $\mathcal{Z}^{n/b}$H, $\mathcal{Z}^{n/b}$M and $\mathcal{Z}^{n/b}$L indicate the conditions with n items, b buckets and situations with high (H), moderate (M) and low (L) noise.

Setting		$\mathcal{Z}^{50/5}$ H	$\mathcal{Z}^{50/5}$ M	$\mathcal{Z}^{50/5}$ L	$\mathcal{Z}^{50/7}$ H	$\mathcal{Z}^{50/7}$ M	$\mathcal{Z}^{50/7}$ L
Computing time in seconds	Quick	75.79	73.67	73.54	94.22	96.32	95.67
	DECoR	115.55	125.49	124.33	152.31	147.27	142.36
Averaged τ_X	Quick	0.21	0.61	0.68	0.31	0.66	0.75
	DECoR	0.21	0.69	0.76	0.31	0.75	0.83

4.2 Sushi dataset

The *PrefLib* repository (Mattei and Walsh, 2013) is a reference library of preference data containing over 3,000 freely available datasets. We test the performance of DECoR with the 'ED-00014-00000003.toi' dataset, also known as sushi dataset. The extension 'toi' means 'order with ties - incomplete list'. There are 5,000 judges and 100 items (kind of sushi) to be ranked. Each judge ranks 10 sushi items. We used the DECoR algorithm by setting 5 buckets and 100 replications. The solution reported in Table 4 was found in 673.995 seconds with $\tau_X = 0.0032$. Note that a rank correlation coefficient close to zero means that, on average, the median ranking is uncorrelated with the orderings expressed by the judges (i.e., there is no consensus and thus no bucket order).

4.3 Application to triage prioritization data of hospital admissions

The usefulness of this approach can be appreciated when there are experimental situations in which a number n of items must be ranked by some judges in such a way that their preferences have to be compressed into a number b of buckets, with $1 < b < n$. As a real example, we report the results attained by applied our novel procedure to an experiment conducted in the Emergency Department (ED) of two popular hospitals in Naples regarding triage, i.e., the admission phase of the ED. A sample of 18 nurses from Hospital α and a sample of 35 nurses from Hospital β had to place in order $n = 25$ cases, according to severity, into $b = 4$ ordered codes: red (R), yellow (Y), green (G) and white (W), with $R < Y < G < W$. This experiment is equivalent to asking a set of m judges to rank n items and only b different ranks,

Table 4 Sushi dataset: median constrained bucket order. Number of buckets: 5

Bucket 1	Bucket 2	Bucket 3	Bucket 4	Bucket 5
negi-toro roll	ebi	sayori	aka-gai	tori-gai
	anago	inari-zushi	kazunoko	aoyagi
	maguro	hamaguri	shako	ba-sashi
	ika	ume-roll	saba	tobiuo
	uni	zuke	kohada	karasumi
	tako	kaiware	miru-gai	namako
	ikura	kujira	kappa-roll	
	tamago	kyabia	geso	
	toro	uni-kurage	iwashi	
	ama-ebi		hokki-gai	
	hotate-gai		kani-miso	
	tai		takuwan-roll	
	hamachi		tobiko	
	awabi		ume&shiso roll	
	sake		komochi-konbu	
	chu-toro		sazae	
	hirame		hamo	
	aji		nasu	
	kani		nattou	
	unagi		ankimo	
	tekka-maki		kanpyo-maki	
	kanpachi		gyu-sashi	
	katsuo		tsubu-gai	
	shimaaji		ana-kyu-maki	
	engawa		hira-gai-tairagi	
	negi-toro		okura	
	nattou-maki		shiso-maki	
	botan-ebi		ika-nattou	
	mentaiko		himo-clam	
	sarada		mekabu	
	suzuki		kue	
	taraba-gani		sawara	
	tarako		sasami	
	toro-salmon		kamo	
	sanma		himo-cuc	
	shirauo		hoya	
	fugu		battera	
	sarada-maki		karei	
	mentaiko-maki		shishamo	
	buri		kaki	
	kuruma-ebi			
	ishigaki-dai			
	mamakari			
	hiramasa			

with $1 < b < n$. We used the DECoR algorithm with 100 repetitions. The median constrained bucket order for Hospital α is

$$[3\ 24]\ [1\ 5\ 6\ 7\ 10\ 15\ 16\ 20]\ [8\ 9\ 11\ 12\ 14\ 17\ 19\ 21\ 22\ 25]\ [2\ 4\ 13\ 18\ 23].$$

The median constrained bucket order for Hospital β is

[3 24] [1 5 7 10 16 21] [2 6 8 9 11 12 14 15 17 19 20 22 25] [4 13 18 23].

The cases ranked in a tie (the buckets) are in brackets. The buckets represent the cases coded R, Y, G and W, respectively. To these solutions, a weighted τ_X rank correlation coefficient is associated that equals 0.6865 and 0.6903, respectively. These averaged values indicate a good degree of internal agreement of the nurses within their assignment process.
After the experiment, a supervisor revealed the 'true' coding of each case, which is:

[3 24] [1 5 6 7 10 12 15 16 20] [8 9 11 14 17 19 21 22 25] [2 4 13 18 23].

The agreement between the true bucket order and the median constrained bucket orders is clear for Hospital α, as the τ_X rank correlation coefficient between these orderings equals 0.917, which shows the good decision process of the nurses. The same measure for Hospital β is equal to 0.697, which shows a global decision process that is not as good as that of Hospital α. We can check the equality of the median constrained bucket orders by using the R^2 statistic, as described in (Marden, 1996, Chapter 4, p. 102):

$$R^2 = 1 - \frac{\sum_{l=1}^{L} \sum_{i=1}^{m^{(l)}} d(X^{(li)} \hat{Y}^{(l)})}{\sum_{l=1}^{L} \sum_{i=1}^{m^{(l)}} d(X^{(li)} \hat{Y})}, \tag{9}$$

where L and $m^{(l)}$ are the groups and the sample size within each group, $X^{(li)}$ is the i-th ranking of the l-th group, and $\hat{Y}^{(l)}$ and $\hat{Y}$ are the constrained median bucket orders for the l-th group and for the entire sample respectively. If the bucket orders in the two samples are equal, then $R^2 = 0$, which constitutes the null hypothesis of the test. As highlighted by Marden (1996), even if the theoretical maximum value of R^2 equals one, it often achieves values that are practically close to zero when the sample size is quite large. In our case $R^2 = 0.0477$. The test has been performed by computing a randomized p-value with 1,000 replications (Feigin and Alvo, 1986; Marden, 1996), which resulted in a value less than 0.001. We can conclude that nurses at Hospital β need a more 'general' training phase than the ones working at Hospital α.
This example shows the usefulness of both the τ_X rank correlation coefficient as a measure of general agreement and the novel approach of constraining the median ranking to be expressed with a prespecified number of buckets.

5 Concluding remarks

The paper deals with a special kind of consensus ranking problem, which is called median constrained bucket order. This problem is extended to address cases in which the solution must be restricted to a subset of the ranking universe that contains a prespecified number of buckets to form an *a priori* constraint of the setting. This kind of solution can be useful in practice when there is a good reason to find such a constrained solution. One possible application is when a group of judges is asked to place in order a set of n items by allowing exactly b buckets, and the solution to the problem must be a tied ranking with exactly b buckets. The Bordeaux Official Wine Classification example is a typical application. Sometimes, this experimental situation can be implicitly recognized, as in the case of the triage prioritization problem.
Among the various axiomatic distance-based approaches proposed in the framework of the rank aggregation problem, we defined our approach under the Kemeny's axiomatic framework. Of course, other approaches have been introduced (see, for instance, Cook et al., 1997) and other ways to detect the median ranking have been proposed (see, for instance, Hall and Schimek, 2012; Asfaw et al., 2017; Aledo et al.,

2018); hence, our approach should be viewed within these frameworks.
Both branch-and-bound and differential evolution algorithms have been specifically redesigned to cope with the bucket constraint, and a simulation comparative study showed that both can be effectively applied to a range of real problems.

Acknowledgments

The authors would like to thank Prof. Dr. Giuseppe Zollo and Dr. Lorella Cannavacciuolo of the University of Naples Federico II for kindly providing us the triage dataset.
The authors would also like to thank both the Editor and the two anonymous reviewers, whose comments highly contributed to improving the quality of the manuscript.

References

Aledo JA, Gámez JA, Molina D (2013) Tackling the rank aggregation problem with evolutionary algorithms. Applied Mathematics and Computation 222:632–644
Aledo JA, Gámez JA, Rosete A (2017a) Partial evaluation in rank aggregation problems. Computers & Operations Research 78:299–304
Aledo JA, Gámez JA, Rosete A (2017b) Utopia in the solution of the bucket order problem. Decision Support Systems 97:69–80
Aledo JA, Gámez JA, Rosete A (2018) Approaching rank aggregation problems by using evolution strategies: The case of the optimal bucket order problem. European Journal of Operational Research 270(3):982–998
Amodio S, D'Ambrosio A, Siciliano R (2016) Accurate algorithms for identifying the median ranking when dealing with weak and partial rankings under the Kemeny axiomatic approach. European Journal of Operational Research 249(2):667–676
Asfaw D, Vitelli V, Sørensen Ø, Arjas E, Frigessi A (2017) Time-varying rankings with the Bayesian Mallows model. Stat 6(1):14–30
Badal PS, Das A (2018) Efficient algorithms using subiterative convergence for Kemeny ranking problem. Computers & Operations Research 98:198 – 210
Biernacki C, Jacques J (2013) A generative model for rank data based on insertion sort algorithm. Computational Statistics & Data Analysis 58:162–176
Bradley RA, Terry ME (1952) Rank analysis of incomplete block designs: I. the method of paired comparisons. Biometrika 39(3/4):324–345
Brentari E, Dancelli L, Manisera M (2016) Clustering ranking data in market segmentation: a case study on the Italian McDonald's customers' preferences. Journal of Applied Statistics 43(11):1959–1976
Brusco MJ, Stahl S (2006) Branch-and-bound applications in combinatorial data analysis. Springer Science & Business Media, New York, NY
Busing FM (2006) Avoiding degeneracy in metric unfolding by penalizing the intercept. British Journal of Mathematical and Statistical Psychology 59(2):419–427
Busing FM, Groenen PJ, Heiser WJ (2005) Avoiding degeneracy in multidimensional unfolding by penalizing on the coefficient of variation. Psychometrika 70(1):71–98
Busse LM, Orbanz P, Buhmann JM (2007) Cluster analysis of heterogeneous rank data. In: Proceedings of the 24th international conference on Machine learning, ACM, New York, NY, pp 113–120
Cohen A, Mallows C (1980) Analysis of ranking data. Tech. rep., AT&T Bell Laboratories Memorandum, Murray Hill, NJ
Cook WD (2006) Distance-based and ad hoc consensus models in ordinal preference ranking. European Journal of operational research 172(2):369–385

Cook WD, Kress M, Seiford LM (1997) A general framework for distance-based consensus in ordinal ranking models. European Journal of Operational Research 96(2):392–397

Corain L, Salmaso L (2007) A non-parametric method for defining a global preference ranking of industrial products. Journal of Applied Statistics 34(2):203–216

D'Ambrosio A (2008) Tree based methods for data editing and preference rankings. PhD thesis, Doctoral dissertation, Università degli Studi di Napoli Federico II, DOI http://www.fedoa.unina.it/2746/

D'Ambrosio A, Heiser WJ (2016) A recursive partitioning method for the prediction of preference rankings based upon Kemeny distances. Psychometrika 81(3):774–794

D'Ambrosio A, Amodio S, Iorio C (2015) Two algorithms for finding optimal solutions of the Kemeny rank aggregation problem for full rankings. Electronic Journal of Applied Statistical Analysis 8(2):198–213

D'Ambrosio A, Amodio S, Mazzeo G (2016) ConsRank, Compute the Median Ranking(s) According to the Kemeny's Axiomatic Approach. R package version 2.0.0. https://cran.r-project.org/web/packages/ConsRank/index.html/

D'Ambrosio A, Mazzeo G, Iorio C, Siciliano R (2017) A differential evolution algorithm for finding the median ranking under the kemeny axiomatic approach. Computers & Operations Research 82:126–138

Deng K, Han S, Li KJ, Liu JS (2014) Bayesian aggregation of order-based rank data. Journal of the American Statistical Association 109(507):1023–1039

Desarkar MS, Sarkar S, Mitra P (2016) Preference relations based unsupervised rank aggregation for metasearch. Expert Systems with Applications 49:86–98

Dittrich R, Katzenbeisser W, Reisinger H (2000) The analysis of rank ordered preference data based on Bradley-Terry type models. OR Spectrum 22(1):117–134

Dopazo E, Martínez-Céspedes ML (2017) Rank aggregation methods dealing with ordinal uncertain preferences. Expert Systems With Applications 78:103–109

Dwork C, Kumar R, Naor M, Sivakumar D (2001) Rank aggregation methods for the web. In: Proceedings of the 10th international conference on World Wide Web, ACM, New York, NY, pp 613–622

Emond EJ, Mason DW (2000) A new technique for high level decision support. Department of National Defence Canada, Operational Research Division, Directorate of Operational Research (Corporate, Air & Maritime), URL http://cradpdf.drdc-rddc.gc.ca/PDFS/zbc84/p513851.pdf

Emond EJ, Mason DW (2002) A new rank correlation coefficient with application to the consensus ranking problem. Journal of Multi-Criteria Decision Analysis 11(1):17–28

Feigin PD, Alvo M (1986) Intergroup diversity and concordance for ranking data: An approach via metrics for permutations. The Annals of Statistics 4(2):691–707

Fields EB, Okudan GE, Ashour OM (2013) Rank aggregation methods comparison: A case for triage prioritization. Expert Systems with Applications 40(4):1305–1311

Fienberg SE, Larntz K (1976) Log linear representation for paired and multiple comparisons models. Biometrika 63(2):245–254

Fürnkranz J, Hüllermeier E (2010) Preference learning: An introduction. Springer, Berlin, Germany

Gionis A, Mannila H, Puolamäki K, Ukkonen A (2006) Algorithms for discovering bucket orders from data. In: Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining, ACM, New York, NY, pp 561–566

Hall P, Schimek MG (2012) Moderate-deviation-based inference for random degeneration in paired rank lists. Journal of the American Statistical Association 107(498):661–672

Heiser WJ, D'Ambrosio A (2013) Clustering and prediction of rankings within a Kemeny distance framework. In: Lausen B, Van den Poel D, Ultsch A (eds) Algorithms from and for Nature and Life, Springer, Cham, Switzerland, pp 19–31

Kemeny JG, Snell JL (1962) Mathematical models in the social sciences. Blaisdell Publishing Company, New York, NY

Kendall MG (1938) A new measure of rank correlation. Biometrika 30(1/2):81–93

Kenkre S, Khan A, Pandit V (2011) On discovering bucket orders from preference data. In: Proceedings of the 2011 SIAM International Conference on Data Mining, SIAM, pp 872–883

Lee PH, Yu PL (2010) Distance-based tree models for ranking data. Computational Statistics & Data Analysis 54(6):1672–1682

Lin S (2010) Space oriented rank-based data integration. Statistical Applications in Genetics and Molecular Biology 9(1):1544–6115

Lin S, Ding J (2009) Integration of ranked lists via cross entropy Monte Carlo with applications to mRNA and microRNA studies. Biometrics 65(1):9–18

Mallows CL (1957) Non-null ranking models. I. Biometrika 44(1/2):114–130

Marden JI (1996) Analyzing and modeling rank data. CRC Press, London, UK

Mattei N, Walsh T (2013) Preflib: A library of preference data HTTP://PREFLIB.ORG. In: Proceedings of the 3rd International Conference on Algorithmic Decision Theory (ADT 2013), Springer, Berlin, Germany, Lecture Notes in Artificial Intelligence, pp 259–270

Meila M, Phadnis K, Patterson A, Bilmes J (2007) Consensus ranking under the exponential model. In: Proceedings of the Twenty-Third Conference Annual Conference on Uncertainty in Artificial Intelligence (UAI-07), AUAI Press, Corvallis, Oregon, pp 285–294

Murphy TB, Martin D (2003) Mixtures of distance-based models for ranking data. Computational statistics & data analysis 41(3):645–655

Onwubolu G, Davendra D (2009) Differential evolution: a handbook for global permutation-based combinatorial optimization. Springer, Berlin, Germany

Plaia A, Sciandra M (2017) Weighted distance-based trees for ranking data. Advances in Data Analysis and Classification DOI https://doi.org/10.1007/s11634-017-0306-x

Sampath S, Verducci JS (2013) Detecting the end of agreement between two long ranked lists. Statistical Analysis and Data Mining: The ASA Data Science Journal 6(6):458–471

Sciandra M, Plaia A, Capursi V (2017) Classification trees for multivariate ordinal response: an application to student evaluation teaching. Quality & Quantity 51(2):641–655

Storn R, Price K (1995) Differential evolution-a simple and efficient adaptive scheme for global optimization over continuous spaces. Tech. Rep. TR-95-012, ICSI Berkeley, URL `http://www.icsi.berkeley.edu/ftp/global/pub/techreports/1995/tr-95-012.pdf`

Švendová V, Schimek MG (2017) A novel method for estimating the common signals for consensus across multiple ranked lists. Computational Statistics & Data Analysis 115:122–135

Telcs A, Kosztyán ZT, Török Á (2016) Unbiased one-dimensional university ranking–application-based preference ordering. Journal of Applied Statistics 43(1):212–228

Thompson GL (1993) Generalized permutation polytopes and exploratory graphical methods for ranked data. The Annals of Statistics 21(3):1401–1430

Ukkonen A, Puolamäki K, Gionis A, Mannila H (2009) A randomized approximation algorithm for computing bucket orders. Information Processing Letters 109(7):356–359

Yu PL, Lee PH, Wan W (2013) Factor analysis for paired ranked data with application on parent–child value orientation preference data. Computational Statistics 28(5):1915–1945

Yu PL, Lee PH, Cheung S, Lau EY, Mok DS, Hui HC (2016) Logit tree models for discrete choice data with application to advice-seeking preferences among chinese christians. Computational Statistics 31(2):799–827