Comparing and Aggregating Rankings with Ties

Ronald Fagin* Ravi Kumar* Mohammad Mahdian†

D. Sivakumar*‡ Erik Vee§

Abstract

Rank aggregation has recently been proposed as a useful abstraction that has several applications, including meta-search, synthesizing rank functions from multiple indices, similarity search, and classification. In database applications (catalog searches, fielded searches, parametric searches, etc.), the rankings are produced by sorting an underlying database according to various fields. Typically, there are a number of fields that each have very few distinct values, and hence the corresponding rankings have many ties in them. Known methods for rank aggregation are poorly suited to this context, and the difficulties can be traced back to the fact that we do not have sound mathematical principles to compare two *partial rankings*, that is, rankings that allow ties.

In this work, we provide a comprehensive picture of how to compare partial rankings. We propose several metrics to compare partial rankings, present algorithms that efficiently compute them, and prove that they are within constant multiples of each other. Based on these concepts, we formulate aggregation problems for partial rankings, and develop a highly efficient algorithm to compute the top few elements of a near-optimal aggregation of multiple partial rankings. In a model of access that is suitable for databases, our algorithm reads essentially as few elements of each partial ranking as are necessary to determine the winner(s).

Proc. 2004 ACM Symposium on Principles of Database Systems (PODS '04), pp. 47–58

*IBM Almaden Research Center, 650 Harry Road, San Jose, CA 95120. Email: {fagin, ravi, siva}@almaden.ibm.com

†MIT Laboratory for Computer Science, 200 Technology Square, Cambridge, MA 02139. Part of this work was done while the author was visiting IBM Almaden Research Center. Email: mahdian@theory.lcs.mit.edu

‡Contact author. Fax: (408) 927 3030

§Computer Science and Engineering, University of Washington, Seattle, WA 98195. Part of this work was done while the author was visiting IBM Almaden Research Center. Email: env@cs.washington.edu

1 Introduction

Rank aggregation. Rank aggregation is the problem of combining several ranked lists of objects in a robust way to produce a single ranking of the objects. This problem has a long and interesting history that goes back at least two centuries. While the philosophical aspects of rank aggregation have been debated extensively during this period, the mathematics of rank aggregation has gained more attention in the last eighty years, and the computational aspects are still within the purview of active research.

In computer science, rank aggregation has proved to be a useful and powerful paradigm in several applications including meta-search [8, 21, 19, 18, 1, 17], combining experts [4], synthesizing rank functions from multiple indices [9], biological databases [20], similarity search [11], and classification [17, 11]. An important contribution of the work of [8] is to adopt and highlight the merits of a proposal of Kemeny's for performing rank aggregation: namely given multiple rankings, find a ranking whose total *Kendall tau distance*[1] to the given rankings is minimized. While this formulation is mathematically crisp, applications usually demand additional flexibility from the aggregation algorithm. For example, in aggregating search results, we are faced with the problem that we have access only to the top few elements of the rankings. In [8], this issue was addressed by suitably modifying the heuristics for full rank aggregation, but without providing any mathematical justification. This situation was later remedied in [10], where notions of "near metrics" were introduced, and a robust, unified class of near metrics was identified to compare "top k lists". This allowed the formulation of appropriate rank aggregation problems for top k lists, and the design of efficient (approximation) algorithms for these problems.

Challenges for rank aggregation in databases. While the extensive work in economics and computer science provide a mathematical basis for aggregation of full/top k rankings, the context of database-centric applications poses two formidable challenges for rank aggregation. We outline these next.

(1) In many applications of rank aggregation, there is an underlying database of records that are first ranked in several ways; typically, each ranked list is produced when the user specifies some criterion to rank (and/or filter) the records according to one of the attributes in the schema. Once the records are sorted in different ways, an aggregation algorithm combines the ranked lists to produce the final output. Common examples are catalog searches, fielded/parametric searches, and "advanced search" options.

For example, in online commerce, users often state their preferences for products according to various criteria. In a database of restaurants (e.g., `www.dine.com`), it is common to rank the restaurants based on the user's preferences for cuisine, driving distance, price, star ratings, etc; in airline reservations (e.g., `www.travelocity.com`), it is common to rank flight plans by price, airline preference, number of connections, flight times, etc. Other examples include searching for an article in scientific bibliography databases (e.g., `www.ams.org/mathscinet`) using preference criteria on attributes such as title, year of publication, number of citations, etc.; searching for a protein in biological databases (e.g., `www.rcsb.org/pdb`) based on attributes like chain type, compound information, experimental technique, resolution, etc; and searching for NSF awards (`www.nsf.gov/verity/srchawdf.htm`) based on attributes such as award amount, start date, etc.

While many database attributes are usually numeric, there are attributes that are inherently non-numeric. For instance, in the restaurant selection example above, "type of cuisine" is a non-numeric attribute. The number of distinct values in such non-numeric attributes is often very small. Therefore, when one sorts according to values this attribute can take, the resulting rank ordering of the objects is not a permutation any more; it is an ordering with ties, also known as a *partial ranking*. Notice that partial rankings could result even for numeric attributes. For example, in travel reservations, the field "number of connections" is

[1] The Kendall tau distance between two rankings (permutations) is defined as the number of pairwise disagreements between the two rankings; it is easy to see that it is a metric on the space of permutations.

a numeric attribute, but usually has no more than four values. Furthermore, the user may not be interested in using the complete range of values for a numeric attribute even if the database might permit it. For instance, in the restaurant selection example, even though distance is numeric, the user might wish to treat any distance up to ten miles to be the same in his/her preference.

Thus, the first main feature of rank aggregation in database applications is that, due to preference criteria on few-valued attributes, we need to deal with partial rankings rather than full rankings. While it is possible to treat this issue heuristically by arbitrarily ordering the tied elements to produce a full ranking, we seek ways that are mathematically more well-founded.

(2) In database-centric applications, we are often interested in only the top few answers of the aggregation. Certainly, this is the case with all the above examples. This feature leads to the quest for algorithms that quickly obtain the top result(s) of aggregation, perhaps in sub-linear time, without even having to read each ranking in its entirety. This issue was addressed in [11], where an aggregation heuristic based on median rank values was studied[2]. This median rank aggregation has the nice property that it admits an *instance-optimal algorithm* in the sense of [12] under a model of access that is relevant for databases. This feature is shared neither by the more sophisticated heuristics in [8] based on matchings and Markov chains, nor by the most natural heuristic based on average ranks. Furthermore, median is clearly robust, since it mitigates the effect of outliers.

Since the applications we focus on in this paper are database-centric, it is tempting to try to adapt the median-based algorithm for aggregating partial rankings. However, there are two obstacles to such an attempt. First of all, though the median rank aggregation algorithm was argued to be heuristically good as an aggregation algorithm, nothing provable was known about its efficacy. In particular, it was not known if median rank aggregation produced an approximately optimal aggregation with respect to the Kendall distance. Secondly, the median rank aggregation algorithm was proposed in [8, 11] assuming that the inputs are permutations. Consequently, it is not clear if the algorithm would perform well, even in a heuristic sense, when the inputs are partial rankings.

To summarize, the aggregation of partial rankings is an important problem in the context of many database applications and it is useful to develop algorithms that quickly obtain the top few results of the aggregation. The single main obstacle is that we do not have sound mathematical principles to compare two partial rankings; this is precisely what we study in this paper. Our main contribution is a comprehensive solution to comparing and aggregating partial rankings.

Summary of our contributions. We define four metrics between partial rankings. These are obtained by suitably generalizing the Kendall tau distance and the Spearman footrule distance on permutations (cf. [6]) in two different ways. In both approaches, to compare two partial rankings, we compare the two sets of full rankings obtained from the partial rankings by breaking ties in all possible ways. A classical way (cf. [5]) to compare two sets in a metric space is the well-known method of using the Hausdorff distance between the sets[3]. The drawback of using the Hausdorff extensions of Kendall tau and Spearman footrule is that they are less intuitive, and the impossibility of pathological cases is not obvious. Our second method to compare the two sets avoids this pitfall, and is based on succinctly summarizing the two sets by compact vectors—their "profiles"—and applying the L_1 distance between the profile vectors. By definition, these metrics admit efficient computation, and furthermore, they are extremely intuitive and quite natural. These metrics are defined and discussed in Section 3.

[2]In fact, rank aggregation based on median rank, along with complicated tie-breaking rules, is used in judging Olympic figure skating [3].

[3]The Hausdorff distance between two point sets A and B in a metric space with metric $d(\cdot, \cdot)$ is defined as $\max\{\max_{\gamma_1 \in A} \min_{\gamma_2 \in B} d(\gamma_1, \gamma_2), \max_{\gamma_2 \in B} \min_{\gamma_1 \in A} d(\gamma_1, \gamma_2)\}$.

While the metrics obtained through profiles can be efficiently computed, the Hausdorff metrics are max-min over exponentially large sets and it is not at all obvious a priori if they can be computed efficiently as well. We solve this problem by first obtaining a complete characterization of how the Hausdorff distance is achieved between two partial rankings (for both Kendall tau and Spearman footrule versions). Namely, we show how to efficiently construct full rankings from partial rankings so that computing the underlying metric (Kendall/Spearman) on the full rankings allows us to compute the Hausdorff distances. These characterizations enable us to compute the Hausdorff distances efficiently; furthermore, while the proofs of the characterizations are technically quite intricate, the resulting algorithms are extremely simple. The computational aspects of the metrics are discussed in Section 4.

Having four metrics on partial rankings is good news, but exactly which one should a practitioner use to compare partial rankings? Furthermore, which one is best suited to formulating an aggregation problem for partial rankings? Our summary answer to these questions is that the exact choice doesn't matter much. Namely, we show, following the lead of [7, 10], that these metrics are all within constant multiples of each other. Diaconis and Graham [7] showed that the Kendall tau distance and the Spearman footrule distance are "equivalent," in the sense that they are within a factor of two from each other. We show a similar relation for the Hausdorff versions of these metrics fairly easily; the relationship between the Hausdorff and the profile versions of the Kendall tau metric also turns out to be fairly simple to establish. Proving an analog of the Diaconis–Graham inequalities for the profile metrics turns out to be rather tricky, and requires us to uncover considerable structure inside partial rankings. We present these equivalence results in Section 5.

Finally, we turn to algorithms to aggregate partial rankings. Here we fully reap the benefit of having defined four distinct metrics on partial rankings and having established their equivalence with much technical maneuvering. Namely, if we care primarily about aggregations that are approximately optimal with respect to a metric, we now have four viewpoints from which to attack the problem! Thus, a constant factor approximation algorithm for aggregation with respect to one metric is automatically a constant factor approximation algorithm for aggregation with respect to all the other metrics. It turns out that an algorithm that is based on the median rank algorithm [8, 11] lends itself naturally to efficient aggregation with respect to the profile version of the Spearman footrule metric. We show the algorithm derived from median ranks is a constant factor approximation algorithm with respect to this metric. Also, as mentioned before, being a median-based algorithm, our algorithm reads only as few elements of each partial ranking as possible in order to determine the winner(s) of the aggregation—in this aspect, the algorithm is extremely database-friendly and practical.

By the equivalence outlined above, it follows that the median rank algorithm is an approximation algorithm for rank aggregation with respect to all our metrics. In fact, since partial rankings generalize full rankings as well as top k lists, the median rank algorithm yields a very efficient solution to the aggregation problems for these objects[4]. It also vindicates the use of median in earlier work [8, 11]. These results are presented in Section 6.

Related work. Kendall [16] defined two variations of the Kendall tau distance for partial rankings of which one is a normalized version of the Kendall tau distance through profiles. Baggerly [2] defined two versions of the Spearman footrule distance for partial rankings of which one is similar to our Spearman footrule metric through profiles. However, neither work proceeds significantly beyond simply providing the definition. For top k lists, which are special case of partial rankings, Critchlow [5] defined Hausdorff versions of Kendall tau and Spearman footrule distances and Fagin et al. [10] studied further properties of these metrics. Goodman and Kruskal [13] proposed an approach for comparing partial rankings, which was

[4]Median rank aggregation offers a good solution to the following loosely-stated problem as well: find a constant-factor approximation algorithm for the Kendall tau rank aggregation problem for full rankings that is both "non-trivial" (since one of the input rankings always achieves a factor two approximation) and computationally simple (unlike computing an optimal solution to the Spearman footrule aggregation problem, which requires the computation of a minimum-cost perfect matching).

recently utilized [14] for evaluating strategies for similarity search on the Web. A serious disadvantage of Goodman and Kruskal's approach is that it is not always defined (this problem did not arise in the application of [14]).

2 Preliminaries

Bucket orders. A *bucket order* is, intuitively, a linear order with ties. More formally, a bucket order is a transitive binary relation $\lhd$ for which there are sets $\mathcal{B}_1, \ldots, \mathcal{B}_t$ (the *buckets*) that form a partition of the domain such that $x \lhd y$ if and only if there are i, j with $i < j$ such $x \in \mathcal{B}_i$ and $y \in \mathcal{B}_j$. If $x \in \mathcal{B}_i$, we may refer to $\mathcal{B}_i$ as the *bucket of* x. We may say that bucket $\mathcal{B}_i$ *precedes* bucket $\mathcal{B}_j$ if $i < j$. Thus, $x \lhd y$ if and only if the bucket of x precedes the bucket of y. We think of the members of a given bucket as "tied". A linear order is a bucket order where every bucket is of size 1. We now define the *position* of bucket $\mathcal{B}$, denoted $pos(\mathcal{B})$. Let $\mathcal{B}_1, \ldots, \mathcal{B}_t$ be the buckets in order (so that bucket $\mathcal{B}_i$ precedes bucket $\mathcal{B}_j$ when $i < j$). Then $pos(\mathcal{B}_i) = (\sum_{j<i} |\mathcal{B}_j|) + (|\mathcal{B}_i| + 1)/2$. Intuitively, $pos(\mathcal{B}_i)$ is the average location within bucket $\mathcal{B}_i$.

Partial ranking. Just as we can associate a ranking with a linear order (i.e., permutation), we associate a *partial ranking* $\boldsymbol{\sigma}$ with each bucket order, by letting $\boldsymbol{\sigma}(x) = pos(\mathcal{B})$ when $\mathcal{B}$ is the bucket of x. We refer to a partial ranking associated with a linear order as a *full ranking*. When it is not otherwise specified, we assume that all partial rankings have the same domain, denoted D. We say that x *is ahead of* y *in* $\boldsymbol{\sigma}$ if $\boldsymbol{\sigma}(x) < \boldsymbol{\sigma}(y)$. We say that x *and* y *are tied in* $\boldsymbol{\sigma}$ if $\boldsymbol{\sigma}(x) = \boldsymbol{\sigma}(y)$.

We define a *top* k *list* to be a partial ranking where the top k buckets are singletons, representing the top k elements, and the bottom bucket contains all other members of the domain. Note that in [10] there is no bottom bucket in a top k list. This is because in [10] each top k list has its own domain of size k, unlike our scenario where there is a fixed domain.

Given a partial ranking $\boldsymbol{\sigma}$ with domain D, we define its *reverse*, denoted $\boldsymbol{\sigma}^{\mathrm{R}}$, in the expected way. That is, for all $d \in D$, let $\boldsymbol{\sigma}^{\mathrm{R}}(d) = |D| + 1 - \boldsymbol{\sigma}(d)$.

Refinements of partial rankings. Given two partial rankings $\boldsymbol{\sigma}$ and $\boldsymbol{\tau}$, both with domain D, we say that $\boldsymbol{\sigma}$ is a *refinement* of $\boldsymbol{\tau}$ and write $\boldsymbol{\sigma} \succeq \boldsymbol{\tau}$ if the following holds: for all $i, j \in D$, we have $\boldsymbol{\sigma}(i) < \boldsymbol{\sigma}(j)$ whenever $\boldsymbol{\tau}(i) < \boldsymbol{\tau}(j)$. Notice that when $\boldsymbol{\tau}(i) = \boldsymbol{\tau}(j)$, there is no order forced on $\boldsymbol{\sigma}$. When $\boldsymbol{\sigma}$ is a full ranking, we say that $\boldsymbol{\sigma}$ is a *full refinement* of $\boldsymbol{\tau}$. Given two partial rankings, $\boldsymbol{\sigma}$ and $\boldsymbol{\tau}$ both with domain D, we frequently make use of a particular refinement of $\boldsymbol{\sigma}$ in which ties are broken according to $\boldsymbol{\tau}$. Define $\boldsymbol{\tau}$*-refinement of* $\boldsymbol{\sigma}$, denoted $\boldsymbol{\tau} * \boldsymbol{\sigma}$, to be the refinement of $\boldsymbol{\sigma}$ with the following properties. For all $i, j \in D$, if $\boldsymbol{\sigma}(i) = \boldsymbol{\sigma}(j)$ and $\boldsymbol{\tau}(i) < \boldsymbol{\tau}(j)$, then $\boldsymbol{\tau} * \boldsymbol{\sigma}(i) < \boldsymbol{\tau} * \boldsymbol{\sigma}(j)$. If $\boldsymbol{\sigma}(i) = \boldsymbol{\sigma}(j)$ and $\boldsymbol{\tau}(i) = \boldsymbol{\tau}(j)$, then $\boldsymbol{\tau} * \boldsymbol{\sigma}(i) = \boldsymbol{\tau} * \boldsymbol{\sigma}(j)$. Notice that when $\boldsymbol{\tau}$ is in fact a full ranking, then $\boldsymbol{\tau} * \boldsymbol{\sigma}$ is also a full ranking. Also note that $*$ is an associative operation, so that if $\boldsymbol{\rho}$ is a partial ranking with domain D, it makes sense to talk about $\boldsymbol{\rho} * \boldsymbol{\tau} * \boldsymbol{\sigma}$.

Notation. When f and g are functions with the same domain D, we denote the L_1 distance between f and g by $L_1(f, g)$. Thus, $L_1(f, g) = \sum_{i \in D} |f(i) - g(i)|$.

2.1 Metrics, near metrics, and equivalence classes

A binary function d is called *symmetric* if $d(x, y) = d(y, x)$ for all x, y in the domain, and is called *regular* if $d(x, y) = 0$ if and only if $x = y$. A *distance measure* is a nonnegative, symmetric, regular binary function. A *metric* is a distance measure d that satisfies the *triangle inequality* $d(x, z) \leq d(x, y) + d(y, z)$ for all x, y, z in the domain.

The definitions and results in this section were derived in [10], in the context of comparing top k lists. Two seemingly different notions of a "near metric" are were defined in [10]: their first notion of near metric is based on "relaxing" the polygonal inequality that a metric is supposed to satisfy.

Definition 1 (Near metric) *A distance measure on partial rankings with domain D is a* near metric *if there is a constant c, independent of the size of D, such that the distance measure satisfies the relaxed polygonal inequality: $d(x,z) \leq c(d(x,x_1)+d(x_1,x_2)+\cdots+d(x_{n-1},z))$ for all $n > 1$ and $x, z, x_1, \ldots, x_{n-1} \in D$.*[5]

The other notion of near metric given in [10] is based on bounding the distance measure above and below by positive constant multiples of a metric. It was shown that both the notions of near metrics coincide. This theorem inspired to define what it means for a distance measure to be "almost" a metric, and a robust notion of "similar" or "equivalent" distance measures. We modify the definitions in [10] slightly to fit our scenario, where there is a fixed domain D.

Definition 2 (Equivalent distance measures) *Two distance measures d and d' between partial rankings with domain D are* equivalent *if there are positive constants c_1 and c_2 such that $c_1 d'(\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2) \leq d(\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2) \leq c_2 d'(\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2)$, for every pair $\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2$ of partial rankings.*[6]

It is clear that the above definition leads to an equivalence relation (i.e., reflexive, symmetric, and transitive). It follows from [10] that a distance measure is equivalent to a metric if and only if it is a near metric.

2.2 Metrics on full rankings

The study of metrics on full rankings is classical (cf. [15, 6]). We now review two well-known notions of metrics on full rankings, namely the Kendall tau distance and the Spearman footrule distance.

Let $\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2$ be two full rankings with domain D. The *Spearman footrule distance* is simply the L_1 distance $L_1(\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2)$. The definition of the Kendall tau distance requires a little more work.

Let $\mathcal{P} = \{\{i,j\} \mid i \neq j \text{ and } i,j \in D\}$ be the set of unordered pairs of distinct elements. The *Kendall tau distance* between full rankings is defined as follows. For each pair $\{i,j\} \in \mathcal{P}$ of distinct members of D, if i and j are in the same order in $\boldsymbol{\sigma}_1$ and $\boldsymbol{\sigma}_2$, then let the penalty $\bar{K}_{i,j}(\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2) = 0$; and if i and j are in the opposite order (such as i being ahead of j in $\boldsymbol{\sigma}_1$ and j being ahead of i in $\boldsymbol{\sigma}_2$), then let $\bar{K}_{i,j}(\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2) = 1$. The Kendall tau distance is given by $K(\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2) = \sum_{\{i,j\} \in \mathcal{P}} \bar{K}_{i,j}(\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2)$. The Kendall tau distance turns out to be equal to the number of exchanges needed in a bubble sort to convert one full ranking to the other.

Diaconis and Graham [7] proved a classical result, which states that for every two full rankings σ_1, σ_2,

$$K(\sigma_1, \sigma_2) \leq F(\sigma_1, \sigma_2) \leq 2K(\sigma_1, \sigma_2). \tag{1}$$

In other words, Kendall tau and Spearman footrule are equivalent metrics for full rankings.

3 Metrics for comparing partial rankings

In this section we define the distance between partial rankings. The first set of metrics is based on profile vectors (Section 3.1). and the second set is based on the Hausdorff distance (Section 3.2). Appendix A.3 compares these metrics (when the partial rankings are top k lists) with the distance measures for top k lists that are developed in [10].

[5] It makes sense to say that the constant c is independent of the size of D when, as in [10], each of the distance measures considered is actually a family, parameterized by D. We need to make an assumption that c is independent of the size of D, since otherwise we are simply considering distance measures over finite domains, where there is always such a constant c.

[6] As before, the constants c_1 and c_2 are assumed to be independent of the size of D.

3.1 Metrics based on profiles

Let $\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2$ be two partial rankings with domain D. We now define a family of generalizations of the Kendall tau distance to partial rankings. These are based on a generalization [10] of the Kendall tau distance to top k lists.

Let p be a fixed parameter, $0 \leq p \leq 1$. Similar to our definition of $\bar{K}_{i,j}(\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2)$ for full rankings $\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2$, we define a penalty $\bar{K}^{(p)}_{i,j}(\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2)$ for partial rankings $\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2$ for $\{i, j\} \in \mathcal{P}$. There are three cases.

Case 1: i and j are in different buckets in both $\boldsymbol{\sigma}_1$ and $\boldsymbol{\sigma}_2$. If i and j are in the same order in $\boldsymbol{\sigma}_1$ and $\boldsymbol{\sigma}_2$ (such as $\boldsymbol{\sigma}_1(i) > \boldsymbol{\sigma}_1(j)$ and $\boldsymbol{\sigma}_2(i) > \boldsymbol{\sigma}_2(j)$) then let $\bar{K}^{(p)}_{i,j}(\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2) = 0$; this corresponds to "no penalty" for $\{i, j\}$. If i and j are in the opposite order in $\boldsymbol{\sigma}_1$ and $\boldsymbol{\sigma}_2$ (such as $\boldsymbol{\sigma}_1(i) > \boldsymbol{\sigma}_1(j)$ and $\boldsymbol{\sigma}_2(i) < \boldsymbol{\sigma}_2(j)$) then let the penalty $\bar{K}^{(p)}_{i,j}(\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2) = 1$.

Case 2: i and j are in the same bucket in both $\boldsymbol{\sigma}_1$ and $\boldsymbol{\sigma}_2$. We then let the penalty $\bar{K}^{(p)}_{i,j}(\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2) = 0$. Intuitively, both partial rankings agree that i and j are tied.

Case 3: i and j are in the same bucket in one of the partial rankings $\boldsymbol{\sigma}_1$ and $\boldsymbol{\sigma}_2$, but in different buckets in the other partial ranking. In this case, we let the penalty $\bar{K}^{(p)}_{i,j}(\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2) = p$.

Based on these cases, define $K^{(p)}$, the *Kendall distance with penalty parameter p*, as follows:

$$K^{(p)}(\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2) = \sum_{\{i,j\} \in \mathcal{P}} \bar{K}^{(p)}_{i,j}(\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2).$$

It is easy to show that if the penalty value in Case 2 were strictly positive then, the resulting quantity is not even a distance measure. Also, $K^{(p)}$ is a metric for $p \in [1/2, 1]$, is a near metric for $p \in (0, 1/2)$, and is not even a distance measure for $p = 0$. (See Appendix A.2 for a proof.) For the rest of the paper, we focus on the natural case $p = 1/2$, since it corresponds to an "average" penalty for two elements i and j that are tied in one partial ranking but not in the other partial ranking. We denote $K^{(1/2)}$ by K_{prof}, since, as we now show, there is an alternative but equivalent definition in terms of a "profile".

Let $\mathcal{O} = \{(i, j) : i \neq j \text{ and } i, j \in D\}$ be the set of ordered pairs of distinct elements in the domain D. Let $\boldsymbol{\sigma}$ be a partial ranking (as usual, with domain D). For $(i, j) \in \mathcal{O}$, define p_{ij} to be $1/4$ if $\boldsymbol{\sigma}(i) < \boldsymbol{\sigma}(j)$, to be 0 if $\boldsymbol{\sigma}(i) = \boldsymbol{\sigma}(j)$, and to be $-1/4$ if $\boldsymbol{\sigma}(i) > \boldsymbol{\sigma}(j)$. Define the *$K$-profile* of $\boldsymbol{\sigma}$ to be the vector $\langle p_{ij} : (i, j) \in \mathcal{O} \rangle$. It is straightforward to verify that $K_{\mathrm{prof}}(\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2)$ is simply the L_1 distance between the K-profiles of $\boldsymbol{\sigma}_1$ and $\boldsymbol{\sigma}_2$.[7]

It is clear how to generalize the Spearman footrule distance to partial rankings—we simply take it to be $L_1(\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2)$, just as before. We refer to this value as $F_{\mathrm{prof}}(\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2)$, for reasons we now explain. Let us define the *F-profile* of a partial ranking $\boldsymbol{\sigma}$ to be the vector of values $\boldsymbol{\sigma}(i)$. So the F-profile is indexed by D, whereas the K-profile is indexed by $\mathcal{O}$. Just as the K_{prof} value of two partial rankings (or of the corresponding bucket orders) is the L_1 distance between their K-profiles, the F_{prof} value of two partial rankings (or of the corresponding bucket orders) is the L_1 distance between their F-profiles. Since F_{prof} and K_{prof} are L_1 distances, they are automatically metrics.

3.2 The Hausdorff metrics

Let A and B be finite sets of objects and let d be a metric of distances between objects. The *Hausdorff distance* between A and B is given by

$$d_{\mathrm{Haus}}(A, B) = \max \left\{ \max_{\gamma_1 \in A} \min_{\gamma_2 \in B} d(\gamma_1, \gamma_2), \max_{\gamma_2 \in B} \min_{\gamma_1 \in A} d(\gamma_1, \gamma_2) \right\}. \tag{2}$$

[7] Each pair $\{i, j\}$ with $i \neq j$ is counted twice, once as (i, j) and once as (j, i). This is why the values of p_{ij} are $1/4$, 0, and $-1/4$ rather than $1/2$, 0, and $-1/2$.

Although this looks fairly nonintuitive, it is actually quite natural, as we now explain. The quantity $\min_{\gamma_2 \in B} d(\gamma_1, \gamma_2)$ is the distance between γ_1 and the set B. Therefore, the quantity $\max_{\gamma_1 \in A} \min_{\gamma_2 \in B} d(\gamma_1, \gamma_2)$ is the maximal distance of a member of A from the set B. Similarly, the quantity $\max_{\gamma_2 \in B} \min_{\gamma_1 \in A} d(\gamma_1, \gamma_2)$ is the maximal distance of a member of B from the set A. Therefore, the Hausdorff distance between A and B is the maximal distance of a member of A or B from the other set. Thus, A and B are within Hausdorff distance s of each other precisely if every member of A and B is within distance s of some member of the other set. The Hausdorff distance is well known to be a metric.

Critchlow [5] used the Hausdorff distance to define a metric between top k lists. We generalize his construction to give a metric between partial rankings. Given a metric d that gives the distance $d(\gamma_1, \gamma_2)$ between full rankings γ_1 and γ_2, define the distance between partial rankings $\boldsymbol{\sigma}_1$ and $\boldsymbol{\sigma}_2$ to be

$$\max \left\{ \max_{\gamma_1 \succeq \boldsymbol{\sigma}_1} \min_{\gamma_2 \succeq \boldsymbol{\sigma}_2} d(\gamma_1, \gamma_2), \max_{\gamma_2 \succeq \boldsymbol{\sigma}_2} \min_{\gamma_1 \succeq \boldsymbol{\sigma}_1} d(\gamma_1, \gamma_2) \right\}, \tag{3}$$

where γ_1 and γ_2 are full rankings. In particular, when d is the footrule distance, this gives us the metric F_{Haus} between partial rankings, and when d is the Kendall distance, this gives us the metric K_{Haus} between partial rankings. Both F_{Haus} and K_{Haus} are indeed metrics, since they are special cases of the Hausdorff distance.

4 Computing the metrics

It is clear from the definition that both K_{prof} and F_{prof} can be computed in polynomial time. In this section we show how to compute the Hausdorff metrics K_{Haus} and F_{Haus} in polynomial time. We make use of these results later to prove that all of our metrics are in the same equivalence class, and in particular that K_{Haus} and K_{prof} are in the same equivalence class. (Note that once we show in Section 5 that all the metrics are equivalent, then it follows that both the Hausdorff metrics can be approximated in polynomial time by computing the profile metrics.)

First, we prove that $\min_\tau F(\sigma, \tau)$, where τ ranges over all full refinements of $\boldsymbol{\tau}$, is attained at $\tau = \sigma * \boldsymbol{\tau}$, and similarly for $\min_\tau K(\sigma, \tau)$. This shows that the minimum occurs when we take the σ-refinement of $\boldsymbol{\tau}$.

Lemma 3 *Let σ be a full ranking, and let $\boldsymbol{\tau}$ be a partial ranking. Then the quantity $F(\sigma, \tau)$, taken over all full refinements $\tau \succeq \boldsymbol{\tau}$, is minimized for $\tau = \sigma * \boldsymbol{\tau}$. Similarly, the quantity $K(\sigma, \tau)$, taken over all full refinements $\tau \succeq \boldsymbol{\tau}$, is minimized for $\tau = \sigma * \boldsymbol{\tau}$.*

The next lemma states that the maximum of minimum occurs when we take the $(\rho * \boldsymbol{\tau}^{\mathrm{R}})$-refinement of $\boldsymbol{\sigma}$, for an arbitrary full ranking ρ.

Lemma 4 *Let $\boldsymbol{\sigma}$ and $\boldsymbol{\tau}$ be partial rankings, and let ρ be any full ranking. Then the quantity $F(\sigma, \sigma * \boldsymbol{\tau})$, taken over all full refinements $\sigma \succeq \boldsymbol{\sigma}$, is maximized when $\sigma = \rho * \boldsymbol{\tau}^{\mathrm{R}} * \boldsymbol{\sigma}$. Similarly, the quantity $K(\sigma, \sigma * \boldsymbol{\tau})$, taken over all full refinements $\sigma \succeq \boldsymbol{\sigma}$, is maximized when $\sigma = \rho * \boldsymbol{\tau}^{\mathrm{R}} * \boldsymbol{\sigma}$.*

Combining the previous two lemmas, we obtain a complete characterization of the Hausdorff distance.

Theorem 5 *Let $\boldsymbol{\sigma}$ and $\boldsymbol{\tau}$ be partial rankings, let $\boldsymbol{\sigma}^{\mathrm{R}}$ be the reverse of $\boldsymbol{\sigma}$, and let $\boldsymbol{\tau}^{\mathrm{R}}$ be the reverse of $\boldsymbol{\tau}$. Let ρ be any full ranking. Then*

$$\begin{aligned} F_{\mathrm{Haus}}(\boldsymbol{\sigma}, \boldsymbol{\tau}) &= \max\left\{F(\rho * \boldsymbol{\tau}^{\mathrm{R}} * \boldsymbol{\sigma}, \rho * \boldsymbol{\sigma} * \boldsymbol{\tau}), F(\rho * \boldsymbol{\tau} * \boldsymbol{\sigma}, \rho * \boldsymbol{\sigma}^{\mathrm{R}} * \boldsymbol{\tau})\right\} \\ K_{\mathrm{Haus}}(\boldsymbol{\sigma}, \boldsymbol{\tau}) &= \max\left\{K(\rho * \boldsymbol{\tau}^{\mathrm{R}} * \boldsymbol{\sigma}, \rho * \boldsymbol{\sigma} * \boldsymbol{\tau}), K(\rho * \boldsymbol{\tau} * \boldsymbol{\sigma}, \rho * \boldsymbol{\sigma}^{\mathrm{R}} * \boldsymbol{\tau})\right\} \end{aligned}$$

Let $\boldsymbol{\sigma}$ and $\boldsymbol{\tau}$ be partial rankings. Theorem 5 gives us a simple algorithm for computing $F_{\mathrm{Haus}}(\boldsymbol{\sigma}, \boldsymbol{\tau})$ and $K_{\mathrm{Haus}}(\boldsymbol{\sigma}, \boldsymbol{\tau})$: we simply pick an arbitrary full ranking ρ and do the computations given in Theorem 5. Let $\sigma_1 = \rho * \boldsymbol{\tau}^{\mathrm{R}} * \boldsymbol{\sigma}$, let $\tau_1 = \rho * \boldsymbol{\sigma} * \boldsymbol{\tau}$, let $\sigma_2 = \rho * \boldsymbol{\tau} * \boldsymbol{\sigma}$, and let $\tau_2 = \rho * \boldsymbol{\sigma}^{\mathrm{R}} * \boldsymbol{\tau}$. Theorem 5 tells us that $F_{\mathrm{Haus}}(\boldsymbol{\sigma}, \boldsymbol{\tau}) = \max\{F(\sigma_1, \tau_1), F(\sigma_2, \tau_2)\}$ and $K_{\mathrm{Haus}}(\boldsymbol{\sigma}, \boldsymbol{\tau}) = \max\{K(\sigma_1, \tau_1), K(\sigma_2, \tau_2)\}$. It is interesting that the same pairs, namely (σ_1, τ_1) and (σ_2, τ_2) are the candidates for exhibiting the Hausdorff distance for both F and K. Note that the only role that the arbitrary full ranking ρ plays is to arbitrarily break ties (in the same way for $\boldsymbol{\sigma}$ and $\boldsymbol{\tau}$) for pairs (i, j) of distinct elements that are in the same bucket in both $\boldsymbol{\sigma}$ and $\boldsymbol{\tau}$. A way to describe the pair (σ_1, τ_1) intuitively is: break the ties in $\boldsymbol{\sigma}$ based on the reverse of the ordering in $\boldsymbol{\tau}$, break the ties in $\boldsymbol{\tau}$ based on the the ordering in $\boldsymbol{\sigma}$, and break any remaining ties arbitrarily (but in the same way in both). A similar description applies to the pair (σ_2, τ_2).

The algorithm we have described for computing $F_{\mathrm{Haus}}(\boldsymbol{\sigma}, \boldsymbol{\tau})$ and $K_{\mathrm{Haus}}(\boldsymbol{\sigma}, \boldsymbol{\tau})$ is based on creating pairs (σ_1, τ_1) and (σ_2, τ_2), one of which must exhibit the Hausdorff distance. The next proposition gives a direct algorithm for computing $K_{\mathrm{Haus}}(\boldsymbol{\sigma}, \boldsymbol{\tau})$, that we make use of later.

Proposition 6 *Let $\boldsymbol{\sigma}$ and $\boldsymbol{\tau}$ be partial rankings. Let S be the set of pairs $\{i, j\}$ of distinct elements such that i and j appear in the same bucket of $\boldsymbol{\sigma}$ but in different buckets of $\boldsymbol{\tau}$, let T be the set of pairs $\{i, j\}$ of distinct elements such that i and j appear in the same bucket of $\boldsymbol{\tau}$ but in different buckets of $\boldsymbol{\sigma}$, and let U be the set of pairs $\{i, j\}$ of distinct elements that are in different buckets of both $\boldsymbol{\sigma}$ and $\boldsymbol{\tau}$ and are in a different order in $\boldsymbol{\sigma}$ and $\boldsymbol{\tau}$. Then $K_{\mathrm{Haus}}(\boldsymbol{\sigma}, \boldsymbol{\tau}) = |U| + \max\{|S|, |T|\}$.*

5 Relationships between the metrics

In this section we show that all our metrics are in the same equivalence class.

Theorem 7 *The metrics F_{prof}, K_{prof}, F_{Haus}, and K_{Haus} are all in the same equivalence class.*

Proof. First, we show

$$K_{\mathrm{Haus}}(\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2) \leq F_{\mathrm{Haus}}(\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2) \leq 2K_{\mathrm{Haus}}(\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2). \tag{4}$$

The proof of this equivalence between F_{Haus} and K_{Haus} uses the robustness of the Hausdorff definition with respect to equivalent metrics. It is fairly easy, and is given in Appendix A.5.1.

Next, we show

$$K_{\mathrm{prof}}(\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2) \leq F_{\mathrm{prof}}(\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2) \leq 2K_{\mathrm{prof}}(\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2). \tag{5}$$

We note that (5) is much more complicated to prove than (4). The proof involves two main concepts: "reflecting" each partial ranking so that every element has a mirror image and using the notion of "nesting", which means that the interval spanned by an element and its image in one partial ranking sits inside the interval spanned by the same element and its image in the other partial ranking. The proof is presented in Appendix A.5.2.

We note that the equivalences given by (4) and (5) are interesting in their own right.

Finally, we show that

$$K_{\mathrm{prof}}(\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2) \leq K_{\mathrm{Haus}}(\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2) \leq 2K_{\mathrm{prof}}(\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2). \tag{6}$$

The above equivalence is proved using Proposition 6.

Using (4), (5), and (6), the proof is complete. □

As we discussed earlier, the above theorem shows that our metrics are quite robust. The equivalence will come in handy when we design aggregation algorithms for partial rankings in Section 6.

6 Aggregation of partial rankings

In this section we show how to employ aggregation algorithms based on the median to achieve constant factor approximation algorithms for partial rankings. We achieve a factor of three with respect to the F_{prof} metric. By Theorem 7, we therefore have constant factor approximation algorithms with respect to the other metrics $F_{\mathrm{Haus}}, K_{\mathrm{Haus}}$, and K_{prof}. When the inputs themselves are full rankings, the median based algorithm achieves a (better) factor of two with respect to the Spearman footrule distance. This answers an open question from [8, 11].

Given a list $a_1, a_2, \ldots, a_m$ of numbers, we define $\mathrm{median}(a_1, \ldots, a_m)$ to be the set of values that would typically be taken as the median of the list (note that if m is odd, it is a set containing just one number). More precisely, suppose that the a_i's are relabeled so that $a_1 \leq a_2 \leq \cdots \leq a_m$. Then $\mathrm{median}(a_1, \ldots, a_m)$ is the set $\left\{a_{\frac{m+1}{2}}\right\}$ when m is odd, and the set $\{a_{\frac{m}{2}}, a_{\frac{m}{2}+1}, (a_{\frac{m}{2}} + a_{\frac{m}{2}+1})/2\}$ when m is even. Given a list $f_1, \ldots, f_m$ of functions, each mapping $D \to \mathcal{R}$, we abuse notation slightly and define $\mathrm{median}(f_1, \ldots, f_m)$ to be the set of valid median functions. More precisely, we define $\mathrm{median}(f_1, \ldots, f_m)$ to be

$$\{f : D \to \mathcal{R} \mid f(d) \in \mathrm{median}(f_1(d), \ldots, f_m(d)), \text{ for every } d \in D\}$$

The following lemma, previously noted in [11], shows the importance of the median function for rank aggregation. Basically, it says that median is the best function for minimizing L_1-norm quantities.

Lemma 8 ([11]) *Let $f_1, f_2, \ldots, f_m$ be functions mapping $D \to \mathcal{R}$. Assume $f \in \mathrm{median}(f_1, \ldots, f_m)$. Then for every function $g : D \to \mathcal{R}$,*

$$\sum_{i=1}^{m} L_1(f, f_i) \leq \sum_{i=1}^{m} L_1(g, f_i).$$

Using Lemma 8, we show the following theorem, which says that the median aggregation algorithm can be used to produce a top k list that is within a factor of three of the optimum top k list (in fact, we need to run the median aggregation algorithm only long enough to output the first k objects). The proof, in a generalized version, appears in the appendix. Noting that a full ranking is actually a top-$|D|$ list, we see that the theorem also implies that the median aggregation algorithm produces a near-optimal full ranking.

Theorem 9 *Let $\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2, \ldots, \boldsymbol{\sigma}_m$ be partial rankings. Assume $f \in \mathrm{median}(\boldsymbol{\sigma}_1, \ldots, \boldsymbol{\sigma}_m)$. Suppose $\boldsymbol{\sigma}$ is a top k list whose first k objects are the same as the first k objects of f and are ordered according to f, with ties among the top k broken arbitrarily. Then for every top k list $\boldsymbol{\tau}$,*

$$\sum_{i=1}^{m} L_1(\boldsymbol{\sigma}, \boldsymbol{\sigma}_i) \leq 3 \sum_{i=1}^{m} L_1(\boldsymbol{\tau}, \boldsymbol{\sigma}_i).$$

We now recall the merits of median as an aggregation operator in the context of databases, as discussed in [11]. In [11], the median rank aggregation algorithm was implemented by using two cursors for each attribute to implicitly rank the database objects with respect to the query without having to sort for every query. This ensures that the data is accessed in a localized and pre-defined order, without any random access or extra storage, thereby permitting extremely efficient implementations. In fact, this algorithm was shown to be instance-optimal [12]—among the class of algorithms that access the input rankings in sequential order, this algorithm is the best possible algorithm (to within a constant factor) on every input instance. By the above theorem, we automatically inherit all the benefits of the median rank aggregation algorithm even for partial rankings.

To see the simplicity of the whole algorithm, here is an instantiation to obtain the top element: access each of the partial rankings, one element at a time, until some database object is seem in more than $m/2$ (i.e., more than half the number of the inputs) times; output this object as the top result of the aggregation. Theorem 9 guarantees that the top k list output by the algorithm is nearly as good as any other top k list. We note that the output satisfies an even stronger notion of optimality; this is discussed further in Appendix A.6.3.

In the above discussion, we assumed that the final goal of aggregation is to produce a full ranking (or top k ranking) that is good when compared against other full rankings. In some applications, it may be desirable (and sufficient) for the aggregation to obtain a partial ranking, but then the partial ranking should compare well against all partial rankings (and not just full rankings or top k lists). We consider this and show that it is possible to get a good approximation even in this case. Unfortunately, the algorithm we use cannot be branded database-friendly, as it is based on dynamic programming.

Theorem 10 *Let $\boldsymbol{\sigma}_1, \ldots, \boldsymbol{\sigma}_m$ be partial rankings, and assume $f \in \mathrm{median}(\boldsymbol{\sigma}_1, \ldots, \boldsymbol{\sigma}_m)$. Suppose that $f^\dagger$ is a partial ranking such that for all partial rankings $\boldsymbol{\tau}$, we have $L_1(f^\dagger, f) \leq L_1(\boldsymbol{\tau}, f)$. Then for every partial ranking $\boldsymbol{\sigma}$, we have*

$$\sum_{i=1}^{m} L_1(f^\dagger, \boldsymbol{\sigma}_i) \leq 2 \sum_{i=1}^{m} L_1(\boldsymbol{\sigma}, \boldsymbol{\sigma}_i).$$

Furthermore, an $f^\dagger$ that satisfies $L_1(f^\dagger, f) \leq L_1(\boldsymbol{\tau}, f)$ for all $\boldsymbol{\tau}$ can be computed in $O(|D|^2)$ time by dynamic programming.

When the inputs themselves are full rankings and the output is required to be a full ranking, we obtain a stronger result—we show that the median aggregation algorithm achieves an approximation factor of two with respect to Spearman footrule distance. Note that this was an open problem emerging from the work of [8, 11]; note also that we achieve a stronger approximation factor of two rather than three. Given a function $f : D \to \mathcal{R}$, it naturally defines a partial ranking, denoted $\hat{f}$, as follows: for all $i, j \in D$, if $f(i) < f(j)$, then set $\hat{f}(i) < \hat{f}(j)$; if $f(i) = f(j)$, then set $\hat{f}(i) = \hat{f}(j)$. We show:

Theorem 11 *Let $\sigma_1, \sigma_2, \ldots, \sigma_m$ be full rankings with domain D. Assume $f \in \mathrm{median}(\sigma_1, \ldots, \sigma_m)$, and let $\boldsymbol{\sigma}$ be a refinement of $\hat{f}$ where ties are broken arbitrarily. Then $\sum_{i=1}^{m} L_1(\boldsymbol{\sigma}, \sigma_i) \leq 2 \sum_{i=1}^{m} L_1(\boldsymbol{\tau}, \sigma_i)$ for every partial ranking $\boldsymbol{\tau}$.*

In particular, if $\boldsymbol{\sigma}$ and $\boldsymbol{\tau}$ are taken to be full rankings in Theorem 11, then this shows that $\boldsymbol{\sigma}$ is a near-optimal choice for full rankings that aggregate $\sigma_1, \sigma_2, \ldots, \sigma_m$.

7 Conclusions

In this paper we consider metrics between partial rankings, motivated by need for such metrics in various database applications. We define four intuitive and natural metrics between partial rankings. We obtain efficient polynomial time algorithms to compute these metrics. We also show that these metrics are all within constant multiples of each other. Armed with this, we obtain a constant factor approximation algorithm for aggregation with respect to each of the metrics by obtaining a constant factor approximation algorithm with respect to just one of them. Our algorithm is based on median rank and admits very efficient database-friendly implementations.

References

[1] J. A. Aslam and M. Montague. Models for metasearch. In *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 276–284, 2001.

[2] K. A. Baggerly. *Visual Estimation of Structure in Ranked Data*. PhD thesis, Rice University, 1995.

[3] G. W. Bassett, Jr. and J. Persky. Rating skating. *Journal of the American Statistical Association*, 89(427):1075–1079, 1994.

[4] W. W. Cohen, R. E. Schapire, and Y. Singer. Learning to order things. *Journal of Artificial Intelligence Research*, 10:243–270, 1999.

[5] D. E. Critchlow. *Metric Methods for Analyzing Partially Ranked Data*. Number 34 in Lecture Notes in Statistics. Springer-Verlag, 1980.

[6] P. Diaconis. *Group Representation in Probability and Statistics*. Number 11 in IMS Lecture Series. Institute of Mathematical Statistics, 1988.

[7] P. Diaconis and R. Graham. Spearman's footrule as a measure of disarray. *Journal of the Royal Statistical Society, Series B*, 39(2):262–268, 1977.

[8] C. Dwork, R. Kumar, M. Naor, and D. Sivakumar. Rank aggregation methods for the web. In *Proceedings of the 10th International World Wide Web Conference*, pages 613–622, 2001.

[9] R. Fagin, R. Kumar, K. McCurley, J. Novak, D. Sivakumar, J. Tomlin, and D. Williamson. Searching the workplace web. In *Proceedings of the 12th International World Wide Web Conference*, pages 366–375, 2003.

[10] R. Fagin, R. Kumar, and D. Sivakumar. Comparing top k lists. In *Proceedings of the 14th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 28–36, 2003. Full version in *SIAM Journal on Discrete Mathematics*, 17(1):134–160, 2003.

[11] R. Fagin, R. Kumar, and D. Sivakumar. Efficient similarity search and classification via rank aggregation. In *Proceedings of the 2003 ACM SIGMOD International Conference on Management of Data*, pages 301–312, 2003.

[12] R. Fagin, A. Lotem, and M. Naor. Optimal aggregation algorithms for middleware. In *Proceedings of the 20th ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems*, pages 102–113, 2001. Full version in *Journal of Computer and System Sciences*, 66(4):614–656, 2003.

[13] L. A. Goodman and W. H. Kruskal. Measures of association for cross classification. *Journal of the American Statistical Association*, 49:732–764, 1954.

[14] T. H Haveliwala, A. Gionis, D. Klein, and P. Indyk. Evaluating strategies for similarity search on the web. In *Proceedings of the 11th International World Wide Web Conference*, pages 432–442, 2002.

[15] M. Kendall and J. D. Gibbons. *Rank Correlation Methods*. Edward Arnold, 1990.

[16] M. G. Kendall. The treatment of ties in ranking problems. *Biometrika*, 33(3):239–251, 1945.

[17] G. Lebanon and J. D. Lafferty. Cranking: Combining rankings using conditional probability models on permutations. In *Proceedings of the 19th International Conference on Machine Learning*, pages 363–370, 2002.

[18] M. Montague and J. A. Aslam. Condorcet fusion for improved retrieval. In *Proceedings of the 11th International Conference on Information and Knowledge Management*, pages 538–548, 2002.

[19] M. E. Renda and U. Straccia. Web metasearch: Rank vs. score based rank aggregation methods. In *Proceedings of the 18th Annual Symposium on Applied Computing*, pages 841–846, 2003.

[20] J. Sese and S. Morishita. Rank aggregation method for biological databases. *Genome Informatics*, 12:506–507, 2001.

[21] R. R. Yager and V. Kreinovich. On how to merge sorted lists coming from different web search tools. *Soft Computing Research Journal*, 3:83–88, 1999.

A Appendix

A.1 Preliminaries for the proofs

Types. When we speak of the buckets of a partial ranking, we are referring to the buckets of the corresponding bucket order. Let $\mathcal{B}_1, \ldots, \mathcal{B}_t$ be the buckets of the partial ranking $\boldsymbol{\sigma}$ in order (thus, $pos(\mathcal{B}_i) < pos(\mathcal{B}_j)$ when $i < j$). We define the *type* of $\boldsymbol{\sigma}$ to be the sequence $|\mathcal{B}_1|, |\mathcal{B}_2|, \ldots, |\mathcal{B}_t|$, and denote it by $type(\boldsymbol{\sigma})$. For example, if σ is a full ranking, then $type(\sigma)$ is the sequence $1, 1, \ldots, 1$ with the number 1 appearing $|D|$ times. We define a *top k list* to be a partial ranking $\boldsymbol{\sigma}$ where $type(\boldsymbol{\sigma})$ is the sequence $1, 1, \ldots, 1, |D| - k$, with the number 1 appearing before $|D| - k$ a total of k times.

We also define the notion of *swapping* in the normal way. If $a, b \in D$, then *swapping a and b in $\boldsymbol{\sigma}$* produces a new order $\boldsymbol{\sigma}'$ where $\boldsymbol{\sigma}'(a) = \boldsymbol{\sigma}(b)$, $\boldsymbol{\sigma}'(b) = \boldsymbol{\sigma}(a)$, and $\boldsymbol{\sigma}'(d) = \boldsymbol{\sigma}(d)$ for all $d \in D - \{a, b\}$.

Finally, we state a fact that we use often.

Lemma 12 *Suppose $a \leq b$ and $c \leq d$. Then $|a - c| + |b - d| \leq |a - d| + |b - c|$.*

Proof. To see this, first note that by symmetry, we can assume, without loss of generality, that $a \leq c$. Now there are three cases: $a \leq b \leq c \leq d$, $a \leq c \leq b \leq d$, and $a \leq c \leq d \leq b$. In each case, it is easy to check that the above inequality holds. □

A.2 Choice of penalty values for $K^{(p)}$

We now discuss our choice of penalty in Cases 2 and 3. In Case 2, where i and j are in the same bucket in both $\boldsymbol{\sigma}_1$ and $\boldsymbol{\sigma}_2$, what if we had defined there to be a positive penalty $\bar{K}^{(p)}_{i,j}(\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2) = q > 0$? Then if $\boldsymbol{\sigma}$ were an arbitrary partial ranking that has some bucket of size at least 2, we would have $K^{(p)}(\boldsymbol{\sigma}, \boldsymbol{\sigma}) \geq q > 0$. So $K^{(p)}$ would not have been a metric, or even a distance measure, since we would have lost the property that $K^{(p)}(\boldsymbol{\sigma}, \boldsymbol{\sigma}) = 0$.

What about the choice of penalty p in Case 3? We show the following:

Proposition 13 *$K^{(p)}$ is a metric for $p \in [1/2, 1]$, is a near metric for $p \in (0, 1/2)$ and is not a distance measure for $p = 0$.*

Proof. Let us first consider the case $p = 0$. We now show that $K^{(0)}$ is not even a distance measure. Let the domain D have exactly two elements a and b. Let $\boldsymbol{\tau}_1$ be the full ranking where a precedes b, let $\boldsymbol{\tau}_2$ be the partial ranking where a and b are in the same bucket, and let $\boldsymbol{\tau}_3$ be the full ranking where b precedes a. Then $K^{(0)}(\boldsymbol{\tau}_1, \boldsymbol{\tau}_2) = 0$ even though $\boldsymbol{\tau}_1 \neq \boldsymbol{\tau}_2$. So indeed, $K^{(0)}$ is not a distance measure. Note also that the

near triangle inequality is violated badly in this example, since $K^{(0)}(\boldsymbol{\tau}_1, \boldsymbol{\tau}_2) = 0$ and $K^{(0)}(\boldsymbol{\tau}_2, \boldsymbol{\tau}_3) = 0$, but $K^{(0)}(\boldsymbol{\tau}_1, \boldsymbol{\tau}_3) = 1$.

It is easy to see that $K^{(p)}$ is a distance measure for every p with $0 < p \leq 1$. We now show that $K^{(p)}$ does not satisfy the triangle inequality when $0 < p < 1/2$ and satisfies the triangle inequality when $1/2 \leq p \leq 1$. Let $\boldsymbol{\tau}_1$, $\boldsymbol{\tau}_2$, and $\boldsymbol{\tau}_3$ be as in our previous example. Then $K^{(p)}(\boldsymbol{\tau}_1, \boldsymbol{\tau}_2) = p$, $K^{(p)}(\boldsymbol{\tau}_2, \boldsymbol{\tau}_3) = p$, and $K^{(p)}(\boldsymbol{\tau}_1, \boldsymbol{\tau}_3) = 1$. So the triangle inequality fails for $0 < p < 1/2$, since $K^{(p)}(\boldsymbol{\tau}_1, \boldsymbol{\tau}_3) > K^{(p)}(\boldsymbol{\tau}_1, \boldsymbol{\tau}_2) + K^{(p)}(\boldsymbol{\tau}_2, \boldsymbol{\tau}_3)$. On the other hand, the triangle inequality holds for $1/2 \leq p \leq 1$, since then it is easy to verify that $\bar{K}^{(p)}_{i,j}(\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_3) \leq \bar{K}^{(p)}_{i,j}(\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2) + \bar{K}^{(p)}_{i,j}(\boldsymbol{\sigma}_2, \boldsymbol{\sigma}_3)$, and so $K^{(p)}(\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_3) \leq K^{(p)}(\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2) + K^{(p)}(\boldsymbol{\sigma}_2, \boldsymbol{\sigma}_3)$.

We now show that $K^{(p)}$ is a near metric for the remaining values of p, where $0 < p < 1/2$. It is easy to verify that if $0 < p < p' \leq 1$, then $K^{(p)}(\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2) \leq K^{(p')}(\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2) \leq (p'/p)K^{(p)}(\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2)$. Hence, all of the distance measures $K^{(p)}$ are in the same equivalence class whenever $0 < p$. As noted earlier, it follows from [10] that a distance measure is in the same equivalence class as a metric if and only if it is a near metric. Since $K^{(p)}$ is in the same equivalence class as the metric $K^{(1/2)}$ when $0 < p < 1/2$, we conclude that in this case, $K^{(p)}$ is a near metric.

□

A.3 Metrics in this paper for top k lists vs. distance measures defined in [10]

We compare our metrics, when restricted to top k lists, with the the distance measures on top k lists introduced in [10]. Recall that for us, a top k list is a partial ranking consisting of k singleton buckets, followed by a bottom bucket of size $|D| - k$. However, in [10], a top k list is a bijection of a domain onto $\{1, ..., k\}$. Let $\boldsymbol{\sigma}$ and $\boldsymbol{\tau}$ be top k lists (of our form), which may have different domains. Define the *active domain for* $\boldsymbol{\sigma}$, $\boldsymbol{\tau}$ to be the union of the elements in the top k buckets of $\boldsymbol{\sigma}$ and the elements in the top k buckets of $\boldsymbol{\tau}$. In order to make our scenario compatible with the scenario of [10], we assume during our comparison that the domain D equals the active domain for $\boldsymbol{\sigma}$, $\boldsymbol{\tau}$. Our definitions of $K^{(p)}$, F_{Haus}, and K_{Haus} are then exactly the same in the two scenarios. (Unlike earlier, even the case $p = 0$ gives a distance metric, since the unpleasant situation where $K^{(0)}(\tau_1, \tau_2) = 0$ even though $\tau_1 \neq \tau_2$ does not arise for top k lists τ_1 and τ_2.) In spite of this, $K^{(p)}$, F_{Haus}, and K_{Haus} are only near metrics in [10] in spite of being metrics for us. This is because in [10], the active domain varies, depending on which pair of top k lists is being compared.

Our definition of $K_{\mathrm{prof}}(\boldsymbol{\sigma}, \boldsymbol{\tau})$ is equivalent to the definition of $K_{\mathrm{avg}}(\boldsymbol{\sigma}, \boldsymbol{\tau})$ in [10], namely the average value of $K(\sigma, \tau)$ over all full rankings σ, τ where $\sigma \succeq \boldsymbol{\sigma}$ and $\tau \succeq \boldsymbol{\tau}$. It is interesting to note that if $\boldsymbol{\sigma}$ and $\boldsymbol{\tau}$ were not top k lists but arbitrary partial rankings, then K_{avg} would not be a distance measure, since $K_{\mathrm{avg}}(\boldsymbol{\sigma}, \boldsymbol{\sigma})$ can be strictly positive if $\boldsymbol{\sigma}$ is an arbitrary partial ranking.

Let ℓ be a real number greater than k. The *footrule distance with location parameter* ℓ, denoted $F^{(\ell)}$, is defined in [10] to be obtained, intuitively, by treating each element that is not among the top k elements as if it were in position ℓ, and then taking the L_1 distance. More formally, let $\boldsymbol{\sigma}$ and $\boldsymbol{\tau}$ be top k lists (of our form). Define the function $f_{\boldsymbol{\sigma}}$ with domain D by letting $f_{\boldsymbol{\sigma}}(i) = \boldsymbol{\sigma}(i)$ if $1 \leq \boldsymbol{\sigma}(i) \leq k$, and $f_{\boldsymbol{\sigma}}(i) = \ell$ otherwise. Similarly, define the function $f_{\boldsymbol{\tau}}$ with domain D by letting $f_{\boldsymbol{\tau}}(i) = \boldsymbol{\tau}(i)$ if $1 \leq \boldsymbol{\tau}(i) \leq k$, and $f_{\boldsymbol{\tau}}(i) = \ell$ otherwise. Then $F^{(\ell)}(\boldsymbol{\sigma}, \boldsymbol{\tau})$ is defined to be $L_1(f_{\boldsymbol{\tau}}, f_{\boldsymbol{\sigma}})$. It is straightforward to verify that $F_{\mathrm{prof}}(\boldsymbol{\sigma}, \boldsymbol{\tau}) = F^{(\ell)}(\boldsymbol{\sigma}, \boldsymbol{\tau})$ for $\ell = (|D| + k + 1)/2$.

A.4 Proofs for Section 4

In this section, we prove the results stated in Section 4. We begin with a lemma.

Lemma 14 *Let π be a full ranking, and let $\boldsymbol{\sigma}$ be a partial ranking. Suppose that $\pi \neq \boldsymbol{\sigma}$. Then there exist i, j such that $\pi(j) = \pi(i) + 1$ while $\boldsymbol{\sigma}(j) \leq \boldsymbol{\sigma}(i)$. If $\boldsymbol{\sigma}$ is in fact a full ranking, then $\boldsymbol{\sigma}(j) < \boldsymbol{\sigma}(i)$.*

Proof. Order the elements of the domain $D = \{d_1, d_2, \ldots, d_{|D|}\}$ so that $\pi(d_1) < \pi(d_2) < \cdots < \pi(d_{|D|})$. If $\boldsymbol{\sigma}(d_\ell) < \boldsymbol{\sigma}(d_{\ell+1})$ for all ℓ, then we would have $K_{\text{prof}}(\boldsymbol{\sigma}, \pi) = 0$, contradicting the fact that $\pi \neq \boldsymbol{\sigma}$. Hence, there must be some ℓ for which $\boldsymbol{\sigma}(d_{\ell+1}) \leq \boldsymbol{\sigma}(d_\ell)$. Setting $i = d_\ell$, $j = d_{\ell+1}$ gives us the lemma.

If $\boldsymbol{\sigma}$ is a full ranking, then $\boldsymbol{\sigma}(j) \neq \boldsymbol{\sigma}(i)$, showing $\boldsymbol{\sigma}(j) < \boldsymbol{\sigma}(i)$. □

Lemma 15 (Lemma 3 restated) *Let σ be a full ranking, and let $\boldsymbol{\tau}$ be a partial ranking. Then the quantity $F(\sigma, \tau)$, taken over all full refinements $\tau \succeq \boldsymbol{\tau}$, is minimized for $\tau = \sigma * \boldsymbol{\tau}$. Similarly, the quantity $K(\sigma, \tau)$, taken over all full refinements $\tau \succeq \boldsymbol{\tau}$, is minimized for $\tau = \sigma * \boldsymbol{\tau}$.*

Proof. First, note that if $\tau \succeq \boldsymbol{\tau}$ then there is a full ranking π such that $\tau = \pi * \boldsymbol{\tau}$. We show that $F(\sigma, \sigma * \boldsymbol{\tau}) \leq F(\sigma, \pi * \boldsymbol{\tau})$ and $K(\sigma, \sigma * \boldsymbol{\tau}) \leq K(\sigma, \pi * \boldsymbol{\tau})$ for every full ranking π. The lemma will then follow.

Let

$$U = \{\text{full } \pi \mid F(\sigma, \sigma * \boldsymbol{\tau}) > F(\sigma, \pi * \boldsymbol{\tau})\},$$
$$V = \{\text{full } \pi \mid K(\sigma, \sigma * \boldsymbol{\tau}) > K(\sigma, \pi * \boldsymbol{\tau})\},$$

and let $S = U \cup V$. If S is empty, then we are done. So suppose not. Over all full rankings $\pi \in S$, choose π to be the full ranking that minimizes $K(\sigma, \pi)$.

Since $\pi \neq \sigma$, Lemma 14 guarantees that we can find a pair i, j such that $\pi(j) = \pi(i) + 1$, but $\sigma(j) < \sigma(i)$. Produce π' by swapping i and j in π. Clearly, π' has one fewer inversion with respect to σ than π does. Hence, $K(\sigma, \pi') < K(\sigma, \pi)$. We show that $\pi' \in S$, thus giving a contradiction.

If i and j are in different buckets for $\boldsymbol{\tau}$, then $\pi' * \boldsymbol{\tau} = \pi * \boldsymbol{\tau}$. Hence, $F(\sigma, \pi' * \boldsymbol{\tau}) = F(\sigma, \pi * \boldsymbol{\tau})$ and $K(\sigma, \pi' * \boldsymbol{\tau}) = K(\sigma, \pi * \boldsymbol{\tau})$. So if $\pi \in U$, then $\pi' \in U$ as well. Similarly, if $\pi \in V$, then $\pi' \in V$. In either case, $\pi' \in S$.

On the other hand, assume that i and j are in the same bucket for $\boldsymbol{\tau}$. Then $\pi' * \boldsymbol{\tau}(i) = \pi * \boldsymbol{\tau}(j)$ and $\pi' * \boldsymbol{\tau}(j) = \pi * \boldsymbol{\tau}(i)$. Furthermore, since $\pi(i) < \pi(j)$ and i and j are in the same bucket, we have $\pi * \boldsymbol{\tau}(i) < \pi * \boldsymbol{\tau}(j)$, while $\sigma(j) < \sigma(i)$.

Either $\pi \in U$ or $\pi \in V$. First, consider the case where $\pi \in U$. Substituting $a = \pi * \boldsymbol{\tau}(i)$, $b = \pi * \boldsymbol{\tau}(j)$, $c = \sigma(j)$, $d = \sigma(i)$ in Lemma 12, we have

$$\begin{aligned} |\pi' * \boldsymbol{\tau}(j) - \sigma(j)| + |\pi' * \boldsymbol{\tau}(i) - \sigma(i)| &= |\pi * \boldsymbol{\tau}(i) - \sigma(j)| + |\pi * \boldsymbol{\tau}(j) - \sigma(i)| \\ &\leq |\pi * \boldsymbol{\tau}(i) - \sigma(i)| + |\pi * \boldsymbol{\tau}(j) - \sigma(j)| \end{aligned}$$

We also have $|\pi' * \boldsymbol{\tau}(d) - \sigma(d)| = |\pi * \boldsymbol{\tau}(d) - \sigma(d)|$ for all $d \in D - \{i, j\}$ since $\pi' * \boldsymbol{\tau}$ and $\pi * \boldsymbol{\tau}$ agree everywhere but at i and j. Summing, we have $F(\sigma, \pi' * \boldsymbol{\tau}) \leq F(\sigma, \pi * \boldsymbol{\tau})$. Since $\pi \in U$, then $F(\sigma, \pi * \boldsymbol{\tau}) < F(\sigma, \sigma * \boldsymbol{\tau})$. So $\pi' \in U$ by transitivity.

Now consider the case where $\pi \in V$. By our choice, $\pi(j) = \pi(i) + 1$. Hence, $\pi * \boldsymbol{\tau}(j) = \pi * \boldsymbol{\tau}(i) + 1$ since i and j are in the same bucket of $\boldsymbol{\tau}$. Similarly, $\pi' * \boldsymbol{\tau}(i) = \pi' * \boldsymbol{\tau}(j) + 1$. And as we noted earlier, $\pi * \boldsymbol{\tau}$ and $\pi' * \boldsymbol{\tau}$ agree everywhere except at i and j. In other words, $\pi' * \boldsymbol{\tau}$ is just $\pi' * \boldsymbol{\tau}$, with the *adjacent* elements i and j swapped. Since $\sigma(i) > \sigma(j)$ we see that $\pi' * \boldsymbol{\tau}$ has exactly one fewer inversion with respect to σ than $\pi * \boldsymbol{\tau}$ does. That is, $K(\sigma, \pi' * \boldsymbol{\tau}) < K(\sigma, \pi * \boldsymbol{\tau})$. Since $\pi \in V$, we have $K(\sigma, \pi * \boldsymbol{\tau}) < K(\sigma, \sigma * \boldsymbol{\tau})$. So π' must be in V as well, by transitivity.

In either case, we have produced a $\pi' \in S$ such that $K(\sigma, \pi') < K(\sigma, \pi)$, contradicting the minimality of π. Hence, S must have been empty, as we wanted. □

Lemma 16 (Lemma 4 restated) *Let $\boldsymbol{\sigma}$ and $\boldsymbol{\tau}$ be partial rankings, and let ρ be any full ranking. Then the quantity $F(\sigma, \sigma * \boldsymbol{\tau})$, taken over all full refinements $\sigma \succeq \boldsymbol{\sigma}$, is maximized when $\sigma = \rho * \boldsymbol{\tau}^{\mathrm{R}} * \boldsymbol{\sigma}$. Similarly, the quantity $K(\sigma, \sigma * \boldsymbol{\tau})$, taken over all full refinements $\sigma \succeq \boldsymbol{\sigma}$, is maximized when $\sigma = \rho * \boldsymbol{\tau}^{\mathrm{R}} * \boldsymbol{\sigma}$.*

Proof. First, note that for any full refinement $\sigma \succeq \boldsymbol{\sigma}$, there is some full ranking, π, such that $\sigma = \pi * \boldsymbol{\sigma}$. We show that for all full rankings π that

$$\begin{aligned} F(\rho * \boldsymbol{\tau}^{\mathrm{R}} * \boldsymbol{\sigma}, \rho * \boldsymbol{\tau}^{\mathrm{R}} * \boldsymbol{\sigma} * \boldsymbol{\tau}) &\geq F(\pi * \boldsymbol{\sigma}, \pi * \boldsymbol{\sigma} * \boldsymbol{\tau}) \\ \text{and } K(\rho * \boldsymbol{\tau}^{\mathrm{R}} * \boldsymbol{\sigma}, \rho * \boldsymbol{\tau}^{\mathrm{R}} * \boldsymbol{\sigma} * \boldsymbol{\tau}) &\geq K(\pi * \boldsymbol{\sigma}, \pi * \boldsymbol{\sigma} * \boldsymbol{\tau}) \end{aligned}$$

The lemma will then follow.

Let $U = \left\{\text{full } \pi \mid F(\rho * \boldsymbol{\tau}^{\mathrm{R}} * \boldsymbol{\sigma}, \rho * \boldsymbol{\tau}^{\mathrm{R}} * \boldsymbol{\sigma} * \boldsymbol{\tau}) < F(\pi * \boldsymbol{\sigma}, \pi * \boldsymbol{\sigma} * \boldsymbol{\tau})\right\}$, let $V = \left\{\text{full } \pi : K(\rho * \boldsymbol{\tau}^{\mathrm{R}} * \boldsymbol{\sigma}, \rho * \boldsymbol{\tau}^{\mathrm{R}} * \boldsymbol{\sigma} * \boldsymbol{\tau}) < K(\pi * \boldsymbol{\sigma}, \pi * \boldsymbol{\sigma} * \boldsymbol{\tau})\right\}$, and let $S = U \cup V$. If S is empty, then we are done. So suppose not. Over all full rankings $\pi \in S$, choose π to be the full ranking that minimizes $K(\rho * \boldsymbol{\tau}^{\mathrm{R}}, \pi)$.

Since $\pi \neq \rho * \boldsymbol{\tau}^{\mathrm{R}}$, Lemma 14 guarantees that we can find a pair i, j such that $\pi(j) = \pi(i) + 1$, but $\rho * \boldsymbol{\tau}^{\mathrm{R}}(j) < \rho * \boldsymbol{\tau}^{\mathrm{R}}(i)$. Produce π' by swapping i and j. Clearly, π' has one fewer inversion with respect to $\rho * \boldsymbol{\tau}^{\mathrm{R}}$ than π does. That is, $K(\rho * \boldsymbol{\tau}^{\mathrm{R}}, \pi') < K(\rho * \boldsymbol{\tau}^{\mathrm{R}}, \pi)$. We now show that $\pi' \in S$, producing a contradiction.

If i and j are in different buckets for $\boldsymbol{\sigma}$, then $\pi' * \boldsymbol{\sigma} = \pi * \boldsymbol{\sigma}$. Hence, $F(\pi' * \boldsymbol{\sigma}, \pi' * \boldsymbol{\sigma} * \boldsymbol{\tau}) = F(\pi * \boldsymbol{\sigma}, \pi * \boldsymbol{\sigma} * \boldsymbol{\tau})$ and $K(\pi' * \boldsymbol{\sigma}, \pi' * \boldsymbol{\sigma} * \boldsymbol{\tau}) = K(\pi * \boldsymbol{\sigma}, \pi * \boldsymbol{\sigma} * \boldsymbol{\tau})$. So if $\pi \in U$, then $\pi' \in U$. Similarly, if $\pi \in V$, then $\pi' \in V$. Hence, π must be in S.

Likewise, if i and j are in the same bucket for both $\boldsymbol{\sigma}$ and $\boldsymbol{\tau}$, then swapping i and j in π swaps their positions in both $\pi * \boldsymbol{\sigma} * \boldsymbol{\tau}$ and $\pi * \boldsymbol{\sigma}$. So again, we see $F(\pi' * \boldsymbol{\sigma}, \pi' * \boldsymbol{\sigma} * \boldsymbol{\tau}) = F(\pi * \boldsymbol{\sigma}, \pi * \boldsymbol{\sigma} * \boldsymbol{\tau})$ and $K(\pi' * \boldsymbol{\sigma}, \pi' * \boldsymbol{\sigma} * \boldsymbol{\tau}) = K(\pi * \boldsymbol{\sigma}, \pi * \boldsymbol{\sigma} * \boldsymbol{\tau})$. As before, $\pi' \in S$.

Now, consider the case when i and j are in the same bucket for $\boldsymbol{\sigma}$, but in different buckets for $\boldsymbol{\tau}$. First of all, $\pi' * \boldsymbol{\sigma}$ is just $\pi * \boldsymbol{\sigma}$ with i and j swapped since i and j are in the same bucket for $\boldsymbol{\sigma}$; further, notice that i and j are adjacent in $\pi * \boldsymbol{\sigma}$. Second, $\pi' * \boldsymbol{\sigma} * \boldsymbol{\tau} = \pi * \boldsymbol{\sigma} * \boldsymbol{\tau}$ since i and j are in different buckets for $\boldsymbol{\tau}$.

Since $\pi(i) < \pi(j)$, we have $\pi * \boldsymbol{\sigma}(i) < \pi * \boldsymbol{\sigma}(j)$. Further, $\boldsymbol{\tau}(i) < \boldsymbol{\tau}(j)$ since $\rho * \boldsymbol{\tau}^{\mathrm{R}}(j) < \rho * \boldsymbol{\tau}^{\mathrm{R}}(i)$ and $\rho * \boldsymbol{\tau}^{\mathrm{R}}$ is a refinement of the reverse of $\boldsymbol{\tau}$. Hence, $\pi * \boldsymbol{\sigma} * \boldsymbol{\tau}(i) < \pi * \boldsymbol{\sigma} * \boldsymbol{\tau}(j)$. We have two cases to consider. Either $\pi \in U$ or $\pi \in V$.

Let us first examine the case that $\pi \in U$. Substituting $a = \pi * \boldsymbol{\sigma}(i)$, $b = \pi * \boldsymbol{\sigma}(j)$, $c = \pi * \boldsymbol{\sigma} * \boldsymbol{\tau}(i)$, $d = \pi * \boldsymbol{\sigma} * \boldsymbol{\tau}(j)$, in Lemma 12 gives us

$$\begin{aligned} |\pi * \boldsymbol{\sigma}(i) - \pi * \boldsymbol{\sigma} * \boldsymbol{\tau}(i)| &+ |\pi * \boldsymbol{\sigma}(j) - \pi * \boldsymbol{\sigma} * \boldsymbol{\tau}(j)| \\ \leq |\pi * \boldsymbol{\sigma}(i) - \pi * \boldsymbol{\sigma} * \boldsymbol{\tau}(j)| &+ |\pi * \boldsymbol{\sigma}(j) - \pi * \boldsymbol{\sigma} * \boldsymbol{\tau}(i)| \\ = |\pi' * \boldsymbol{\sigma}(j) - \pi' * \boldsymbol{\sigma} * \boldsymbol{\tau}(j)| &+ |\pi' * \boldsymbol{\sigma}(i) - \pi' * \boldsymbol{\sigma} * \boldsymbol{\tau}(i)| \end{aligned}$$

We also have that $|\pi' * \boldsymbol{\sigma}(d) - \pi' * \boldsymbol{\sigma} * \boldsymbol{\tau}(d)| = |\pi * \boldsymbol{\sigma}(d) - \pi * \boldsymbol{\sigma} * \boldsymbol{\tau}(d)|$ for all $d \in D - \{i, j\}$. Summing over all d, we obtain $F(\pi * \boldsymbol{\sigma}, \pi * \boldsymbol{\sigma} * \boldsymbol{\tau}) \leq F(\pi' * \boldsymbol{\sigma}, \pi' * \boldsymbol{\sigma} * \boldsymbol{\tau})$. Since $\pi \in U$, we have that $F(\rho * \boldsymbol{\tau}^{\mathrm{R}} * \boldsymbol{\sigma}, \rho * \boldsymbol{\tau}^{\mathrm{R}} * \boldsymbol{\sigma} * \boldsymbol{\tau}) < F(\pi * \boldsymbol{\sigma}, \pi * \boldsymbol{\sigma} * \boldsymbol{\tau})$. Hence, $\pi' \in U$ by transitivity.

We now examine the case that $\pi \in V$. From above, we see that $\pi' * \boldsymbol{\sigma} * \boldsymbol{\tau} = \pi * \boldsymbol{\sigma} * \boldsymbol{\tau}$, while $\pi' * \boldsymbol{\sigma}$ and $\pi * \boldsymbol{\sigma}$ differ only by swapping the adjacent elements i and j. Since $\pi' * \boldsymbol{\sigma}(i) > \pi' * \boldsymbol{\sigma}(j)$ while $\pi' * \boldsymbol{\sigma} * \boldsymbol{\tau}(i) < \pi' * \boldsymbol{\sigma} * \boldsymbol{\tau}(j)$, we see that there is exactly one more inversion between $\pi' * \boldsymbol{\sigma}$ and $\pi' * \boldsymbol{\sigma} * \boldsymbol{\tau}$ than between $\pi * \boldsymbol{\sigma}$ and $\pi * \boldsymbol{\sigma} * \boldsymbol{\tau}$. That is, $K(\pi * \boldsymbol{\sigma}, \pi * \boldsymbol{\sigma} * \boldsymbol{\tau}) < K(\pi' * \boldsymbol{\sigma}, \pi' * \boldsymbol{\sigma} * \boldsymbol{\tau})$. By our assumption, $\pi \in V$, hence $K(\rho * \boldsymbol{\tau}^{\mathrm{R}} * \boldsymbol{\sigma}, \rho * \boldsymbol{\tau}^{\mathrm{R}} * \boldsymbol{\sigma} * \boldsymbol{\tau}) < K(\pi * \boldsymbol{\sigma}, \pi * \boldsymbol{\sigma} * \boldsymbol{\tau})$. It follows that $\pi' \in V$.

So in each case, we have produced a $\pi' \in S$ such that $K(\rho * \boldsymbol{\tau}^{\mathrm{R}}, \pi') < K(\rho * \boldsymbol{\tau}^{\mathrm{R}}, \pi)$, contradicting the minimality of π. Hence, S must have been empty, as we wanted. □

Theorem 17 (Theorem 5 restated) *Let $\boldsymbol{\sigma}$ and $\boldsymbol{\tau}$ be partial rankings, let $\boldsymbol{\sigma}^{\mathrm{R}}$ be the reverse of $\boldsymbol{\sigma}$, and let $\boldsymbol{\tau}^{\mathrm{R}}$ be the reverse of $\boldsymbol{\tau}$. Let ρ be any full ranking. Then*

$$F_{\mathrm{Haus}}(\boldsymbol{\sigma}, \boldsymbol{\tau}) = \max\left\{F(\rho * \boldsymbol{\tau}^{\mathrm{R}} * \boldsymbol{\sigma}, \rho * \boldsymbol{\sigma} * \boldsymbol{\tau}), F(\rho * \boldsymbol{\tau} * \boldsymbol{\sigma}, \rho * \boldsymbol{\sigma}^{\mathrm{R}} * \boldsymbol{\tau})\right\}$$

$$K_{\mathrm{Haus}}(\boldsymbol{\sigma}, \boldsymbol{\tau}) \;=\; \max\left\{K(\rho * \boldsymbol{\tau}^{\mathrm{R}} * \boldsymbol{\sigma}, \rho * \boldsymbol{\sigma} * \boldsymbol{\tau}), K(\rho * \boldsymbol{\tau} * \boldsymbol{\sigma}, \rho * \boldsymbol{\sigma}^{\mathrm{R}} * \boldsymbol{\tau})\right\}$$

Proof. We prove it for F_{Haus}. The proof for K_{Haus} is analogous. Recall that

$$F_{\mathrm{Haus}}(\boldsymbol{\sigma}, \boldsymbol{\tau}) = \max\left\{\max_{\sigma}\min_{\tau} F(\sigma, \tau), \max_{\tau}\min_{\sigma} F(\sigma, \tau)\right\}$$

where throughout this proof, σ and τ range through all full refinements of $\boldsymbol{\sigma}$ and $\boldsymbol{\tau}$, respectively. We show $\max_\sigma \min_\tau F(\sigma, \tau) = F(\rho * \boldsymbol{\tau}^{\mathrm{R}} * \boldsymbol{\sigma}, \rho * \boldsymbol{\sigma} * \boldsymbol{\tau})$. The fact that $\max_\tau \min_\sigma F(\sigma, \tau) = F(\rho * \boldsymbol{\tau} * \boldsymbol{\sigma}, \rho * \boldsymbol{\sigma}^{\mathrm{R}} * \boldsymbol{\tau})$ follows similarly.

Think of $\sigma \succeq \boldsymbol{\sigma}$ as fixed. Then by Lemma 3, the quantity $F(\sigma, \tau)$, where τ ranges over all full refinements of $\boldsymbol{\tau}$, is minimized when $\tau = \sigma * \boldsymbol{\tau}$. That is, $\min_\tau F(\sigma, \tau) = F(\sigma, \sigma * \boldsymbol{\tau})$.

By Lemma 4, the quantity $F(\sigma, \sigma * \boldsymbol{\tau})$, where σ ranges over all full refinements of $\boldsymbol{\sigma}$, is maximized when $\sigma = \rho * \boldsymbol{\tau}^{\mathrm{R}} * \boldsymbol{\sigma}$. Hence, $\max_\sigma \min_\tau F(\sigma, \tau) = F(\rho * \boldsymbol{\tau}^{\mathrm{R}} * \boldsymbol{\sigma}, \rho * \boldsymbol{\tau}^{\mathrm{R}} * \boldsymbol{\sigma} * \boldsymbol{\tau})$. Since $\rho * \boldsymbol{\tau}^{\mathrm{R}} * \boldsymbol{\sigma} * \boldsymbol{\tau} = \rho * \boldsymbol{\sigma} * \boldsymbol{\tau}$, we have $\max_\sigma \min_\tau F(\sigma, \tau) = F(\rho * \boldsymbol{\tau}^{\mathrm{R}} * \boldsymbol{\sigma}, \rho * \boldsymbol{\sigma} * \boldsymbol{\tau})$, as we wanted. □

Proposition 18 (Proposition 6 restated) *Let $\boldsymbol{\sigma}$ and $\boldsymbol{\tau}$ be partial rankings. Let S be the set of pairs $\{i, j\}$ of distinct elements such that i and j appear in the same bucket of $\boldsymbol{\sigma}$ but in different buckets of $\boldsymbol{\tau}$, let T be the set of pairs $\{i, j\}$ of distinct elements such that i and j appear in the same bucket of $\boldsymbol{\tau}$ but in different buckets of $\boldsymbol{\sigma}$, and let U be the set of pairs $\{i, j\}$ of distinct elements that are in different buckets of both $\boldsymbol{\sigma}$ and $\boldsymbol{\tau}$ and are in a different order in $\boldsymbol{\sigma}$ and $\boldsymbol{\tau}$. Then $K_{\mathrm{Haus}}(\boldsymbol{\sigma}, \boldsymbol{\tau}) = |U| + \max\{|S|, |T|\}$.*

Proof. As before, let $\sigma_1 = \rho * \boldsymbol{\tau}^{\mathrm{R}} * \boldsymbol{\sigma}$, let $\tau_1 = \rho * \boldsymbol{\sigma} * \boldsymbol{\tau}$, let $\sigma_2 = \rho * \boldsymbol{\tau} * \boldsymbol{\sigma}$, and let $\tau_2 = \rho * \boldsymbol{\sigma}^{\mathrm{R}} * \boldsymbol{\tau}$. It is straightforward to see that the set of pairs $\{i, j\}$ of distinct elements that are in a different order in σ_1 and τ_1 is exactly the union of the disjoint sets U and S. Therefore, $K(\sigma_1, \tau_1) = |U| + |S|$. Identically, the set of pairs $\{i, j\}$ of distinct elements that are in a different order in σ_2 and τ_2 is exactly the union of the disjoint sets U and T, and hence $K(\sigma_2, \tau_2) = |U| + |T|$. But by Theorem 5, we know that $K_{\mathrm{Haus}}(\boldsymbol{\sigma}, \boldsymbol{\tau}) = \max\{K(\sigma_1, \tau_1), K(\sigma_2, \tau_2)\} = \max|U| + |S|, |U| + |T|$. The result follows immediately. □

A.5 Proofs for Section 5

In this section we prove the equivalence of all our metrics.

A.5.1 Equivalence of F_{Haus} and K_{Haus}

In this section, we prove the simple result that the Diaconis–Graham inequalities (1) extend to F_{Haus} and K_{Haus}. We begin with a lemma. In this lemma, for metric d, we define d_{Haus} as in (2), and similarly for metric d'.

Lemma 19 *Assume that d and d' are metrics where there is a constant c such that $d \le c \cdot d'$. Then $d_{\mathrm{Haus}} \le c \cdot d'_{\mathrm{Haus}}$.*

Proof. Let A and B be as in (2). Assume without loss of generality that $d_{\mathrm{Haus}}(A, B) = \max_{\gamma_1 \in A}\min_{\gamma_2 \in B} d(\gamma_1, \gamma_2)$. Find γ_1 in A that maximizes $\min_{\gamma_2 \in B} d(\gamma_1, \gamma_2)$, and γ_2 in B that minimizes $d(\gamma_1, \gamma_2)$. Therefore, $d_{\mathrm{Haus}}(A, B) = d(\gamma_1, \gamma_2)$. Find γ_2' in B that minimizes $d'(\gamma_1, \gamma_2')$. (There is such an γ_2' since by assumption on the definition of Hausdorff distance, A and B are finite sets.) Then $d_{\mathrm{Haus}}(A, B) = d(\gamma_1, \gamma_2) \le d(\gamma_1, \gamma_2')$, since γ_2 minimizes $d(\gamma_1, \gamma_2)$. Also $d(\gamma_1, \gamma_2') \le c \cdot d'(\gamma_1, \gamma_2')$, by assumption on d and d'. Finally $c \cdot d'(\gamma_1, \gamma_2') \le c \cdot d'_{\mathrm{Haus}}(A, B)$, by definition of d'_{Haus} and the fact that γ_2' minimizes $d'(\gamma_1, \gamma_2')$. Putting these inequalities together, we obtain $d_{\mathrm{Haus}}(A, B) \le c \cdot d'_{\mathrm{Haus}}(A, B)$, which completes the proof. □

We can now show that the Diaconis–Graham inequalities (1) extend to F_{Haus} and K_{Haus}.

Theorem 20 *Let $\boldsymbol{\sigma}_1$ and $\boldsymbol{\sigma}_2$ be partial rankings. Then $K_{\mathrm{Haus}}(\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2) \leq F_{\mathrm{Haus}}(\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2) \leq 2K_{\mathrm{Haus}}(\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2)$.*

Proof. The first inequality $K_{\mathrm{Haus}}(\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2) \leq F_{\mathrm{Haus}}(\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2)$ follows from the first Diaconis–Graham inequality $K(\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2) \leq F(\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2)$ and Lemma 19, where we let the roles of d, d' and c be played by K, F, and 1 respectively. The second inequality $F_{\mathrm{Haus}}(\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2) \leq 2K_{\mathrm{Haus}}(\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2)$ follows from the second Diaconis–Graham inequality $F(\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2) \leq 2K(\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2)$ and Lemma 19, where we let the roles of d, d' and c be played by F, K, and 2 respectively. □

A.5.2 Equivalence of F_{prof} and K_{prof}

In order to generalize the Diaconis–Graham inequalities to F_{prof} and K_{prof}, we convert a pair of partial rankings into full rankings in such a way that both the F_{prof} and K_{prof} distances between the partial rankings is precisely $\frac{1}{4}$ times the F and K distances between the full rankings, respectively. Given a partial ranking, $\boldsymbol{\sigma}$, with domain D, produce a duplicate set, $D^\sharp = \{i^\sharp : i \in D\}$. Further, produce a new partial ranking, $\boldsymbol{\sigma}^\sharp$, with domain $D \cup D^\sharp$ defined by $\boldsymbol{\sigma}^\sharp(i) = \boldsymbol{\sigma}^\sharp(i^\sharp) = 2\boldsymbol{\sigma}(i) - 1/2$ for all $i \in D$.

It is easy to see that $\boldsymbol{\sigma}^\sharp$ is a well-defined partial ranking. Further, it is not hard to check that for any partial ranking $\boldsymbol{\tau}$,

$$\begin{aligned} F_{\mathrm{prof}}(\boldsymbol{\sigma}^\sharp, \boldsymbol{\tau}^\sharp) &= 4F_{\mathrm{prof}}(\boldsymbol{\sigma}, \boldsymbol{\tau}) \\ K_{\mathrm{prof}}(\boldsymbol{\sigma}^\sharp, \boldsymbol{\tau}^\sharp) &= 4K_{\mathrm{prof}}(\boldsymbol{\sigma}, \boldsymbol{\tau}) \end{aligned}$$

In order for us to prove our theorem, we still need to convert $\boldsymbol{\sigma}^\sharp$ from a partial ranking to a full ranking. For any full ranking π with domain D, define a full ranking $\pi^\natural$ with domain $D \cup D^\sharp$ as follows:

$$\begin{aligned} \pi^\natural(d) &= \pi(d) \text{ for all } d \in D \\ \pi^\natural(d^\sharp) &= 2|D| + 1 - \pi(d) \text{ for all } d \in D \end{aligned}$$

so that $\pi^\natural$ ranks elements of D in the same order as π, elements of $D^\sharp$ in the reverse order of π, and all elements of D before all elements of $D^\sharp$.

We define $\boldsymbol{\sigma}_\pi = \pi^\natural * (\boldsymbol{\sigma}^\sharp)$. For instance, suppose $\mathcal{B}$ is a bucket of $\boldsymbol{\sigma}^\sharp$ containing the items $a, b, c, a^\sharp, b^\sharp, c^\sharp$, and suppose that π orders the items $\pi(a) < \pi(b) < \pi(c)$. Then $\boldsymbol{\sigma}_\pi$ will contain the sequence $a, b, c, c^\sharp, b^\sharp, a^\sharp$. Also notice that $\frac{1}{2}(\boldsymbol{\sigma}_\pi(a) + \boldsymbol{\sigma}_\pi(a^\sharp)) = \frac{1}{2}(\boldsymbol{\sigma}_\pi(b) + \boldsymbol{\sigma}_\pi(b^\sharp)) = \frac{1}{2}(\boldsymbol{\sigma}_\pi(c) + \boldsymbol{\sigma}_\pi(c^\sharp)) = pos(\mathcal{B})$. In fact, because of this "reflected-duplicate" property, we see that in general, for any $d \in D$,

$$\frac{1}{2}(\boldsymbol{\sigma}_\pi(d) + \boldsymbol{\sigma}_\pi(d^\sharp)) = \boldsymbol{\sigma}^\sharp(d) = \boldsymbol{\sigma}^\sharp(d^\sharp) = 2\boldsymbol{\sigma}(d) - 1/2 \tag{7}$$

The following lemma shows that no matter what order π we choose, the Kendall distance between $\boldsymbol{\sigma}_\pi$ and $\boldsymbol{\tau}_\pi$ is exactly 4 times the K_{prof} distance between $\boldsymbol{\sigma}$ and $\boldsymbol{\tau}$.

Lemma 21 *Let $\boldsymbol{\sigma}, \boldsymbol{\tau}$ be partial rankings, and let π be any full ranking on the same domain. Then $K(\boldsymbol{\sigma}_\pi, \boldsymbol{\tau}_\pi) = 4K_{\mathrm{prof}}(\boldsymbol{\sigma}, \boldsymbol{\tau})$.*

Proof. By cases. □

Notice that Lemma 21 holds for any choice of π. The analogous statement is not true for F_{prof}. In that case, we need to choose π specifically for the pair of partial rankings we are given. In particular, we need to avoid a property we call "nesting."

Given fixed $\boldsymbol{\sigma}, \boldsymbol{\tau}$, we say that an element $d \in D$ is *nested* with respect to π if either

$$\begin{aligned} [\boldsymbol{\sigma}_\pi(d), \boldsymbol{\sigma}_\pi(d^\sharp)] &\sqsubset [\boldsymbol{\tau}_\pi(d), \boldsymbol{\tau}_\pi(d^\sharp)] \\ \text{or } [\boldsymbol{\tau}_\pi(d), \boldsymbol{\tau}_\pi(d^\sharp)] &\sqsubset [\boldsymbol{\sigma}_\pi(d), \boldsymbol{\sigma}_\pi(d^\sharp)] \end{aligned}$$

where the notation $[s,t] \sqsubset [u,v]$ for integers s,t,u,v means that $[s,t] \subseteq [u,v]$ and $s \neq u$, $t \neq v$. It is sometimes convenient to write $[u,v] \sqsupset [s,t]$ for $[s,t] \sqsubset [u,v]$.

The following lemma shows us why we want to avoid nesting.

Lemma 22 *Given partial rankings $\boldsymbol{\sigma}, \boldsymbol{\tau}$ and full ranking π, suppose that there are no elements that are nested with respect to π. Then $F(\boldsymbol{\sigma}_\pi, \boldsymbol{\tau}_\pi) = 4F_{\mathrm{prof}}(\boldsymbol{\sigma}, \boldsymbol{\tau})$.*

Proof. Let $d \in D$. Since d is not nested with respect to π, either

$$\begin{aligned} &\boldsymbol{\sigma}_\pi(d) \leq \boldsymbol{\tau}_\pi(d) \text{ and } \boldsymbol{\sigma}_\pi(d^\sharp) \leq \boldsymbol{\tau}_\pi(d^\sharp) \\ \text{or} \quad &\boldsymbol{\sigma}_\pi(d) \geq \boldsymbol{\tau}_\pi(d) \text{ and } \boldsymbol{\sigma}_\pi(d^\sharp) \geq \boldsymbol{\tau}_\pi(d^\sharp) \end{aligned}$$

In either case, we see

$$|\boldsymbol{\sigma}_\pi(d) - \boldsymbol{\tau}_\pi(d)| + |\boldsymbol{\sigma}_\pi(d^\sharp) - \boldsymbol{\tau}_\pi(d^\sharp)| = |\boldsymbol{\sigma}_\pi(d) - \boldsymbol{\tau}_\pi(d) + \boldsymbol{\sigma}_\pi(d^\sharp) - \boldsymbol{\tau}_\pi(d^\sharp)|$$

But recall that $\frac{1}{2}(\boldsymbol{\sigma}_\pi(d) + \boldsymbol{\sigma}_\pi(d^\sharp)) = 2\boldsymbol{\sigma}(d) - 1/2$ and similarly for $\boldsymbol{\tau}_\pi$. Substituting gives us

$$|\boldsymbol{\sigma}_\pi(d) - \boldsymbol{\tau}_\pi(d)| + |\boldsymbol{\sigma}_\pi(d^\sharp) - \boldsymbol{\tau}_\pi(d^\sharp)| = 4|\boldsymbol{\sigma}(d) - \boldsymbol{\tau}(d)|$$

Hence,

$$\begin{aligned} F(\boldsymbol{\sigma}_\pi, \boldsymbol{\tau}_\pi) &= \sum_{d \in D} (|\boldsymbol{\sigma}_\pi(d) - \boldsymbol{\tau}_\pi(d)| + |\boldsymbol{\sigma}_\pi(d^\sharp) - \boldsymbol{\tau}_\pi(d^\sharp)|) \\ &= \sum_{d \in D} 4|\boldsymbol{\sigma}(d) - \boldsymbol{\tau}(d)| \\ &= 4F_{\mathrm{prof}}(\boldsymbol{\sigma}, \boldsymbol{\tau}) \end{aligned}$$

□

In the proof of the following lemma, we show that in fact, there is always a full ranking π with no nested elements.

Lemma 23 *Let $\boldsymbol{\sigma}, \boldsymbol{\tau}$ be partial rankings. Then there exists a full ranking π on the same domain such that $F(\boldsymbol{\sigma}_\pi, \boldsymbol{\tau}_\pi) = 4F_{\mathrm{prof}}(\boldsymbol{\sigma}, \boldsymbol{\tau})$.*

Proof. We produce a full ranking π that has no nested elements. For any full ranking ρ, we say its *first nest* is $\min_d \rho(d)$, where d is allowed to range over all nested elements of ρ; we say its first nest is ∞ if ρ has no nests. Choose π so that its first nest is as large as possible.

If π has no nested elements, then we are done. Otherwise, let a be the element such that $\pi(a)$ is the first nest of π. By definition, a is nested. Without loss of generality, assume that $[\boldsymbol{\sigma}_\pi(a), \boldsymbol{\sigma}_\pi(a^\sharp)] \sqsupset [\boldsymbol{\tau}_\pi(a), \boldsymbol{\tau}_\pi(a^\sharp)]$. We find $b \in D$ so that $\pi(a) < \pi(b)$, and swapping a and b in π will leave b unnested. To this end, let

$$\begin{aligned} S_1 &= \left\{ d \in D - \{a\} \;\middle|\; [\boldsymbol{\sigma}_\pi(a), \boldsymbol{\sigma}_\pi(a^\sharp)] \sqsupset [\boldsymbol{\sigma}_\pi(d), \boldsymbol{\sigma}_\pi(d^\sharp)] \right\} \\ S_2 &= \left\{ d \in D - \{a\} \;\middle|\; [\boldsymbol{\sigma}_\pi(a), \boldsymbol{\sigma}_\pi(a^\sharp)] \sqsupset [\boldsymbol{\tau}_\pi(d), \boldsymbol{\tau}_\pi(d^\sharp)] \right\} \end{aligned}$$

Choose $b \in S_1 - S_2$. To see such a b exists, note that $|S_1| = \frac{1}{2}|[\boldsymbol{\sigma}_\pi(a), \boldsymbol{\sigma}_\pi(a^\sharp)]| - 1$, while $|S_2| \leq \frac{1}{2}|[\boldsymbol{\sigma}_\pi(a), \boldsymbol{\sigma}_\pi(a^\sharp)]| - 2$, since $[\boldsymbol{\sigma}_\pi(a), \boldsymbol{\sigma}_\pi(a^\sharp)] \sqsupset [\boldsymbol{\tau}_\pi(a), \boldsymbol{\tau}_\pi(a^\sharp)]$ but a is not counted in S_2. Note that $b \in S_1$ implies a and b are in the same bucket for $\boldsymbol{\sigma}$. It further implies that $\pi(a) < \pi(b)$.

Furthermore, a and b are in different buckets for $\boldsymbol{\tau}$. To see this, suppose that a and b were in the same bucket for $\boldsymbol{\tau}$. Then since $\pi(a) < \pi(b)$, we would have $\boldsymbol{\tau}_\pi(a) < \boldsymbol{\tau}_\pi(b)$ and $\boldsymbol{\tau}_\pi(a^\sharp) > \boldsymbol{\tau}_\pi(b^\sharp)$. That is, $[\boldsymbol{\tau}_\pi(a), \boldsymbol{\tau}_\pi(a^\sharp)] \sqsupset [\boldsymbol{\tau}_\pi(b), \boldsymbol{\tau}_\pi(b^\sharp)]$. But a is nested, so by our assumption, $[\boldsymbol{\sigma}_\pi(a), \boldsymbol{\sigma}_\pi(a^\sharp)] \sqsupset [\boldsymbol{\tau}_\pi(a), \boldsymbol{\tau}_\pi(a^\sharp)] \sqsupset [\boldsymbol{\tau}_\pi(b), \boldsymbol{\tau}_\pi(b^\sharp)]$. This contradicts the fact that $b \notin S_2$. Hence, a and b must be in different buckets for $\boldsymbol{\tau}$.

Now, produce π' by swapping a and b in π. Since $\pi(a) < \pi(b)$, we see $\pi'(b) = \pi(a) < \pi(b) = \pi'(a)$. We wish to prove that the first nest for π' is larger than the first nest for π, giving a contradiction. We do so by showing that b is unnested for π' and further, that d is unnested for π' for all $d \in D$ such that $\pi'(d) < \pi'(b) = \pi(a)$. In order to prove this, we need to examine the affect of swapping a and b in π.

To this end, consider a bucket $\mathcal{B}$ of $\boldsymbol{\sigma}$. Let $\pi|_{\mathcal{B}}$ denote the order that π induces on $\mathcal{B}$. Since $\pi'(d) = \pi(d)$ for all d such that $\pi(d) < \pi(a)$, we see that $\pi'|_{\mathcal{B}}(d) = \pi|_{\mathcal{B}}(d)$ for all such d. Hence, $\boldsymbol{\sigma}_{\pi'}(d) = \boldsymbol{\sigma}_\pi(d)$ and $\boldsymbol{\sigma}_{\pi'}(d^\sharp) = \boldsymbol{\sigma}_\pi(d^\sharp)$ for all such d. Therefore, for all d such that $\pi(d) < \pi(a)$

$$[\boldsymbol{\sigma}_{\pi'}(d), \boldsymbol{\sigma}_{\pi'}(d^\sharp)] \quad = \quad [\boldsymbol{\sigma}_\pi(d), \boldsymbol{\sigma}_\pi(d^\sharp)] \tag{8}$$

Let $\mathcal{B}$ be the bucket of $\boldsymbol{\sigma}$ that contains a and b. Then $\pi'|_{\mathcal{B}}$ is just $\pi|_{\mathcal{B}}$ with a and b swapped. So $\pi'|_{\mathcal{B}}(b) = \pi|_{\mathcal{B}}(a)$. Hence, $\boldsymbol{\sigma}_{\pi'}(b) = \boldsymbol{\sigma}_\pi(a)$ and $\boldsymbol{\sigma}_{\pi'}(b^\sharp) = \boldsymbol{\sigma}_\pi(a^\sharp)$. That is,

$$[\boldsymbol{\sigma}_{\pi'}(b), \boldsymbol{\sigma}_{\pi'}(b^\sharp)] \quad = \quad [\boldsymbol{\sigma}_\pi(a), \boldsymbol{\sigma}_\pi(a^\sharp)] \tag{9}$$

We now consider a bucket $\mathcal{B}$ of $\boldsymbol{\tau}$. Arguing as we did for buckets of $\boldsymbol{\sigma}$, we have that for all d such that $\pi(d) < \pi(a)$,

$$[\boldsymbol{\tau}_{\pi'}(d), \boldsymbol{\tau}_{\pi'}(d^\sharp)] \quad = \quad [\boldsymbol{\tau}_\pi(d), \boldsymbol{\tau}_\pi(d^\sharp)] \tag{10}$$

Now, let $\mathcal{B}$ be the bucket of $\boldsymbol{\tau}$ that contains a. Since π and π' differ only by swapping a and b, and $\pi'(a) > \pi(a)$, we see that $\pi'|_{\mathcal{B}}(a) \geq \pi|_{\mathcal{B}}(a)$. Hence, $\boldsymbol{\tau}_{\pi'}(a) \geq \boldsymbol{\tau}_\pi(a)$ and $\boldsymbol{\tau}_{\pi'}(a^\sharp) \leq \boldsymbol{\tau}_\pi(a^\sharp)$. That is,

$$[\boldsymbol{\tau}_{\pi'}(a), \boldsymbol{\tau}_{\pi'}(a^\sharp)] \quad \subseteq \quad [\boldsymbol{\tau}_\pi(a), \boldsymbol{\tau}_\pi(a^\sharp)] \tag{11}$$

Finally, let $\mathcal{B}$ be the bucket of $\boldsymbol{\tau}$ that contains b. Since π and π' differ only by swapping a and b, and $\pi'(b) < \pi(b)$, we see that $\pi'|_{\mathcal{B}}(b) \leq \pi|_{\mathcal{B}}(b)$. Hence, $\boldsymbol{\tau}_{\pi'}(b) \leq \boldsymbol{\tau}_\pi(b)$ and $\boldsymbol{\tau}_{\pi'}(b^\sharp) \geq \boldsymbol{\tau}_\pi(b^\sharp)$. That is,

$$[\boldsymbol{\tau}_{\pi'}(b), \boldsymbol{\tau}_{\pi'}(b^\sharp)] \quad \supseteq \quad [\boldsymbol{\tau}_\pi(b), \boldsymbol{\tau}_\pi(b^\sharp)] \tag{12}$$

We are now ready to prove the lemma. From equations (8) and (10), we see that d remains unnested for all d such that $\pi'(d) < \pi(a) = \pi'(b)$. So we only need to show that b is unnested for π' to finish the proof.

If b were nested for π', then either $[\boldsymbol{\sigma}_{\pi'}(b), \boldsymbol{\sigma}_{\pi'}(b^\sharp)] \sqsupset [\boldsymbol{\tau}_{\pi'}(b), \boldsymbol{\tau}_{\pi'}(b^\sharp)]$ or $[\boldsymbol{\tau}_{\pi'}(b), \boldsymbol{\tau}_{\pi'}(b^\sharp)] \sqsupset [\boldsymbol{\sigma}_{\pi'}(b), \boldsymbol{\sigma}_{\pi'}(b^\sharp)]$. First, suppose that $[\boldsymbol{\sigma}_{\pi'}(b), \boldsymbol{\sigma}_{\pi'}(b^\sharp)] \sqsupset [\boldsymbol{\tau}_{\pi'}(b), \boldsymbol{\tau}_{\pi'}(b^\sharp)]$. Then

$$\begin{aligned}
[\boldsymbol{\sigma}_\pi(a), \boldsymbol{\sigma}_\pi(a^\sharp)] &= [\boldsymbol{\sigma}_{\pi'}(b), \boldsymbol{\sigma}_{\pi'}(b^\sharp)] \text{ from equation (9)} \\
&\sqsupset [\boldsymbol{\tau}_{\pi'}(b), \boldsymbol{\tau}_{\pi'}(b^\sharp)] \text{ by supposition} \\
&\supseteq [\boldsymbol{\tau}_\pi(b), \boldsymbol{\tau}_\pi(b^\sharp)] \text{ from (12)}
\end{aligned}$$

But this contradicts the fact that $b \notin S_2$. Now, suppose that $[\boldsymbol{\tau}_{\pi'}(b), \boldsymbol{\tau}_{\pi'}(b^\sharp)] \sqsupset [\boldsymbol{\sigma}_{\pi'}(b), \boldsymbol{\sigma}_{\pi'}(b^\sharp)]$. Then

$$\begin{aligned}
[\boldsymbol{\tau}_{\pi'}(b), \boldsymbol{\tau}_{\pi'}(b^\sharp)] &\sqsupset [\boldsymbol{\sigma}_{\pi'}(b), \boldsymbol{\sigma}_{\pi'}(b^\sharp)] \text{ by supposition} \\
&= [\boldsymbol{\sigma}_\pi(a), \boldsymbol{\sigma}_\pi(a^\sharp)] \text{ from equation (9)} \\
&\sqsupset [\boldsymbol{\tau}_\pi(a), \boldsymbol{\tau}_\pi(a^\sharp)] \text{ since } a \text{ is nested, by assumption} \\
&\supseteq [\boldsymbol{\tau}_{\pi'}(a), \boldsymbol{\tau}_{\pi'}(a^\sharp)] \text{ from (11)}
\end{aligned}$$

But this implies that a and b are in the same bucket for $\boldsymbol{\tau}$, a contradiction. Hence, b must not be nested for π'.

Hence, if any element d is nested for π', it must be the case that $\pi'(d) > \pi'(b) = \pi(a)$. That is, the first nest for π' is larger than the first nest for π, contradicting our choice of π. Therefore, π must have had no nested elements. By Lemma 22, $F(\boldsymbol{\sigma}_\pi, \boldsymbol{\tau}_\pi) = 4F_{\mathrm{prof}}(\boldsymbol{\sigma}, \boldsymbol{\tau})$, as we wanted. □

Putting these two lemmas together, we conclude the following.

Theorem 24 *Let $\boldsymbol{\sigma}$ and $\boldsymbol{\tau}$ be partial rankings. Then $K_{\mathrm{prof}}(\boldsymbol{\sigma}, \boldsymbol{\tau}) \leq F_{\mathrm{prof}}(\boldsymbol{\sigma}, \boldsymbol{\tau}) \leq 2K_{\mathrm{prof}}(\boldsymbol{\sigma}, \boldsymbol{\tau})$.*

Proof. Given $\boldsymbol{\sigma}$ and $\boldsymbol{\tau}$, let π be the full ranking guaranteed in Lemma 23. Then we have

$$\begin{aligned} K_{\mathrm{prof}}(\boldsymbol{\sigma}, \boldsymbol{\tau}) &= 4K(\boldsymbol{\sigma}_\pi, \boldsymbol{\tau}_\pi) \text{ by Lemma 21} \\ &\leq 4F(\boldsymbol{\sigma}_\pi, \boldsymbol{\tau}_\pi) \text{ from Diaconis–Graham} \\ &= F_{\mathrm{prof}}(\boldsymbol{\sigma}, \boldsymbol{\tau}) \text{ by Lemma 23} \end{aligned}$$

And similarly,

$$\begin{aligned} F_{\mathrm{prof}}(\boldsymbol{\sigma}, \boldsymbol{\tau}) &= 4F(\boldsymbol{\sigma}_\pi, \boldsymbol{\tau}_\pi) \text{ by Lemma 23} \\ &\leq 8K(\boldsymbol{\sigma}_\pi, \boldsymbol{\tau}_\pi) \text{ from Diaconis–Graham} \\ &= 2K_{\mathrm{prof}}(\boldsymbol{\sigma}, \boldsymbol{\tau}) \text{ by Lemma 21} \end{aligned}$$

□

A.5.3 Equivalence of K_{Haus} and K_{prof}

We show that K_{Haus} and K_{prof} are in the same equivalence class, thereby proving (6).

Lemma 25 *Let $\boldsymbol{\sigma}_1$ and $\boldsymbol{\sigma}_2$ be partial rankings. Then $K_{\mathrm{prof}}(\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2) \leq K_{\mathrm{Haus}}(\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2) \leq 2K_{\mathrm{prof}}(\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2)$.*

Proof. As in Proposition 6 (but where we let $\boldsymbol{\sigma}_1$ play the role of $\boldsymbol{\sigma}$ and $\boldsymbol{\sigma}_2$ play the role of $\boldsymbol{\tau}$), let S be the set of pairs $\{i, j\}$ of distinct elements such that i and j appear in the same bucket of $\boldsymbol{\sigma}_1$ but in different buckets of $\boldsymbol{\sigma}_2$, let T be the set of pairs $\{i, j\}$ of distinct elements such that i and j appear in the same bucket of $\boldsymbol{\sigma}_2$ but in different buckets of $\boldsymbol{\sigma}_1$, and let U be the set of pairs $\{i, j\}$ of distinct elements that are in different buckets of both $\boldsymbol{\sigma}_1$ and $\boldsymbol{\sigma}_2$ and are in a different order in $\boldsymbol{\sigma}_1$ and $\boldsymbol{\sigma}_2$. By Proposition 6, we know that $K_{\mathrm{Haus}}(\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2) = |U| + \max\{|S|, |T|\}$. It follows from the definition of K_{prof} that $K_{\mathrm{prof}}(\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2) = |U| + \frac{1}{2}|S| + \frac{1}{2}|T|$. The theorem now follows from the straightforward inequalities $|U| + \frac{1}{2}|S| + \frac{1}{2}|T| \leq |U| + \max\{|S|, |T|\} \leq 2(|U| + \frac{1}{2}|S| + \frac{1}{2}|T|)$. □

A.6 Proofs for Section 6

A.6.1 Basic machinery

In this section we develop the basic machinery needed to prove the theorems in Section 6.

The following lemma appears to be folklore; for completeness, we include a proof here. Note that Lemma 12 is, in fact, a special case of this lemma.

Lemma 26 *If A and B are two multisets of numbers of the same size, and the cost of matching $a \in A$ to $b \in B$ is $|a - b|$, then the order-preserving perfect matching (i.e., the matching that matches the i-th largest element of A to the i-th largest element of B) is a minimum cost perfect matching between A and B.*

Proof. Let us denote the elements of A and B by $a_1, a_2, \ldots, a_n$ and $b_1, b_2, \ldots, b_n$, respectively. Consider a minimum cost perfect matching μ between A and B that matches the element a_i with $b_{\mu(i)}$. Assume, without loss of generality, that $a_1 \leq a_2 \leq \cdots \leq a_n$. If $b_{\mu(1)} \leq b_{\mu(2)} \leq \cdots \leq b_{\mu(n)}$, then we are done. Otherwise, there exists i such that $b_{\mu(i)} > b_{\mu(i+1)}$. Using Lemma 12, where the roles of a, b, c, d are played by $a_i, b_i, b_{\mu(i+1)}, b_{\mu(i)}$ respectively, we obtain $|a_i - b_{\mu(i+1)}| + |a_{i+1} - b_{\mu(i)}| \leq |a_i - b_{\mu(i)}| + |a_{i+1} - b_{\mu(i+1)}|$. Therefore, the matching μ' defined by

$$\mu'(j) = \begin{cases} \mu(i+1) & j = i \\ \mu(i) & j = i+1 \\ \mu(j) & \text{otherwise} \end{cases}$$

has a cost not greater than the cost of μ. Thus, μ' is a minimum cost perfect matching between A and B. Furthermore, the number of j's such that $b_{\mu'(j)} > b_{\mu'(j+1)}$ is strictly smaller than the corresponding number for μ. Therefore, by repeating the above procedure we eventually get a minimum cost perfect matching that is order preserving. □

Given functions $f : D \to \mathcal{R}$ and $g : D \to \mathcal{R}$, we say that f and g are *consistent* with each other if there is no pair $i, j \in D$ such that $f(i) < f(j)$ and $g(i) > g(j)$. We now show that this notion is symmetric in the role of f and g. Assume that f and g are consistent with each other, and there is a pair $i, j \in D$ such that $g(i) < g(j)$ and $f(i) > f(j)$. By reversing the roles of i and j, we obtain a contradiction to the fact that f and g are consistent with each other. Although, as we just showed, this relationship is symmetric, it is not transitive, since the constant function is consistent with all other functions. We define $\langle f \rangle$ to be the set of all *partial rankings* that are consistent with f.

Recall that if $\boldsymbol{\sigma}$ is a partial ranking consisting of buckets $\mathcal{B}_1, \mathcal{B}_2, \ldots, \mathcal{B}_t$ with $pos(\mathcal{B}_1) < pos(\mathcal{B}_2) < \cdots < pos(\mathcal{B}_t)$, the *type* of $\boldsymbol{\sigma}$, denoted $type(\boldsymbol{\sigma})$, is the sequence $|\mathcal{B}_1|, |\mathcal{B}_2|, \ldots, |\mathcal{B}_t|$. Given a type α, define $\langle f \rangle_\alpha$ to be the subset of $\langle f \rangle$ consisting of partial rankings with type α.

Lemma 27 *Let $f : D \to \mathcal{R}$, let α be a type, and suppose $\boldsymbol{\sigma} \in \langle f \rangle_\alpha$. Then $L_1(\boldsymbol{\sigma}, f) \leq L_1(\boldsymbol{\tau}, f)$ for all partial rankings $\boldsymbol{\tau}$ such that $type(\boldsymbol{\tau}) = \alpha$.*

Proof. Consider the multisets $A = \{\!\{\boldsymbol{\sigma}(x) : x \in D\}\!\}$ and $B = \{\!\{f(x) : x \in D\}\!\}$. It is clear from the definition of partial rankings and types that every partial ranking $\boldsymbol{\tau}$ of type α corresponds to a perfect matching between D and A. Since there is a one-to-one correspondence between D and B, every such $\boldsymbol{\tau}$ also corresponds to a perfect matching between A and B. Furthermore, the cost of this perfect matching (assuming that the cost of matching $a \in A$ with $b \in B$ is $|a - b|$) is precisely $L_1(\boldsymbol{\tau}, f)$. Thus, by Lemma 26, the minimum value of $L_1(\boldsymbol{\tau}, f)$ is achieved when $\boldsymbol{\tau}$ is consistent with f, that is, when it belongs to $\langle f \rangle_\alpha$. Hence, $L_1(\boldsymbol{\sigma}, f) \leq L_1(\boldsymbol{\tau}, f)$. □

Lemma 28 *Let $f : D \to \mathcal{R}$, and let $\hat{f}$ be the partial ranking associated with it. Let $\boldsymbol{\sigma}$ be a refinement of $\hat{f}$. Then for every full ranking τ, we have $L_1(\boldsymbol{\sigma}, f) \leq L_1(\tau, f)$.*

Proof. Let $\boldsymbol{\sigma} \succeq \hat{f}$, and let σ be a full ranking that is a refinement of $\boldsymbol{\sigma}$. We show $L_1(\boldsymbol{\sigma}, f) \leq L_1(\sigma, f)$, hence $L_1(\boldsymbol{\sigma}, f) \leq L_1(\tau, f)$ by Lemma 27 (both have the same type, namely $1, 2, \ldots, |D|$).

To this end, let $\mathcal{B}$ be a bucket of $\boldsymbol{\sigma}$. Since $\boldsymbol{\sigma}$ is a refinement of $\hat{f}$, we see that f is constant over all $i \in \mathcal{B}$; call this value $f_{\mathcal{B}}$. Since σ is a refinement of $\boldsymbol{\sigma}$, we see that $\sum_{i \in \mathcal{B}} \sigma(i) = |\mathcal{B}| \cdot pos(\mathcal{B})$. So we have

$$\sum_{i \in \mathcal{B}} |\sigma(i) - f_{\mathcal{B}}| \geq \left| \sum_{i \in \mathcal{B}} (\sigma(i) - f_{\mathcal{B}}) \right| = |\mathcal{B}| \cdot |pos(\mathcal{B}) - f_{\mathcal{B}}| = \sum_{i \in \mathcal{B}} |\boldsymbol{\sigma}(i) - f_{\mathcal{B}}| \,.$$

Summing the above over all buckets of $\boldsymbol{\sigma}$, we see that $L_1(\sigma, f) \geq L_1(\boldsymbol{\sigma}, f)$, as we wanted. □

A.6.2 Proofs

Theorems 9,10, and 11 are special cases of the following theorem, as we shall show.

Theorem 29 *Let $f_1, \ldots, f_m$ be functions mapping $D \to \mathcal{R}$, and assume $f \in \text{median}(f_1, \ldots, f_m)$. Also, let S be a set of functions (for instance, the set of top k lists, or the set of partial rankings). Suppose that f' is a function such that for all functions $g \in S$, we have $L_1(f', f) \leq L_1(g, f)$. Then for all functions $g \in S$, we have*

$$\sum_{i=1}^{m} L_1(f', f_i) \leq 3 \sum_{i=1}^{m} L_1(g, f_i).$$

If the functions $f_1, \ldots, f_m \in S$, then we have for all functions h,

$$\sum_{i=1}^{m} L_1(f', f_i) \leq 2 \sum_{i=1}^{m} L_1(h, f_i).$$

Proof.

$$\begin{aligned}
\sum_{i=1}^{m} L_1(f', f_i) &\leq \sum_{i=1}^{m} (L_1(f', f) + L_1(f, f_i)) \text{ by the triangle inequality} \\
&\leq \sum_{i=1}^{m} (L_1(g, f) + L_1(f, f_i)) \text{ by assumption} \\
&\leq \sum_{i=1}^{m} (L_1(g, f_i) + L_1(f_i, f) + L_1(f, f_i)) \text{ by the triangle inequality} \\
&\leq 3 \sum_{i=1}^{m} L_1(g, f_i) \text{ by Lemma 8}
\end{aligned}$$

As for the second part,

$$\begin{aligned}
\sum_{i=1}^{m} L_1(f', f_i) &\leq \sum_{i=1}^{m} (L_1(f', f) + L_1(f, f_i)) \text{ by the triangle inequality} \\
&\leq \sum_{i=1}^{m} 2L_1(f, f_i) \text{ by assumption, since each } f_i \in S \\
&\leq 2 \sum_{i=1}^{m} L_1(h, f_i) \text{ by Lemma 8}
\end{aligned}$$

□

Recall that Theorem 9 considered the case when the output of the aggregation is forced to be a top k list. We now show a more general form of this theorem when the output can be specified to be any fixed type α and where we aggregate not just partial rankings but arbitrary funciton.

Corollary 30 (Generalized form of Theorem 9) *Let $f_1, f_2, \ldots, f_m$ be functions mapping $D \to \mathcal{R}$. Assume $f \in \text{median}(f_1, \ldots, f_m)$. Let α be a type, and assume $\boldsymbol{\sigma} \in \langle f \rangle_\alpha$. Then for every partial ranking $\boldsymbol{\tau}$ such that $type(\boldsymbol{\tau}) = \alpha$,*

$$\sum_{i=1}^{m} L_1(\boldsymbol{\sigma}, f_i) \leq 3 \sum_{i=1}^{m} L_1(\boldsymbol{\tau}, f_i)$$

Furthermore, if all of the f_i's are partial rankings with $type(f_1) = type(f_2) = \cdots = type(f_m) = \alpha$, then for all functions g,

$$\sum_{i=1}^{m} L_1(\boldsymbol{\sigma}, f_i) \leq 2 \sum_{i=1}^{m} L_1(g, f_i)$$

Proof. Let S be the set of partial rankings with type α. Then combining Lemma 27 and Theorem 29 gives us the result. □

Corollary 31 (Generalized form of Theorem 10) *Let $f_1, \ldots, f_m$ be functions mapping $D \to \mathcal{R}$, and assume $f \in \text{median}(f_1, \ldots, f_m)$. Suppose that $f^\dagger$ is a partial ranking such that for all partial rankings $\boldsymbol{\tau}$, we have $L_1(f^\dagger, f) \leq L_1(\boldsymbol{\tau}, f)$. Then for all partial rankings $\boldsymbol{\sigma}$, we have*

$$\sum_{i=1}^{m} L_1(f^\dagger, f_i) \leq 3 \sum_{i=1}^{m} L_1(\boldsymbol{\sigma}, f_i).$$

If the functions $f_1, \ldots, f_m$ are in fact partial rankings then we have for all functions g,

$$\sum_{i=1}^{m} L_1(f^\dagger, f_i) \leq 2 \sum_{i=1}^{m} L_1(g, f_i).$$

Furthermore, in this second case where the functions $f_1, \ldots, f_m$ are partial rankings, an $f^\dagger$ that satisfies $L_1(f^\dagger, f) \leq L_1(\boldsymbol{\tau}, f)$ for all partial orders $\boldsymbol{\tau}$ can be computed in $O(|D|^2)$ time by dynamic programming.

Proof. Setting S to be the set of all partial rankings in Theorem 29, the result is immediate once we are given $f^\dagger$. The $O(|D|^2)$ time dynamic programming algorithm to calculate an appropriate $f^\dagger$ is presented in Section A.6.4. □

Corollary 32 (Generalized form of Theorem 11) *Let $\sigma_1, \sigma_2, \ldots, \sigma_m$ be full rankings with domain D. Let $f \in \text{median}(\sigma_1, \ldots, \sigma_m)$, and let $\boldsymbol{\sigma}$ be any refinement of $\hat{f}$. Then $\sum_{i=1}^{m} L_1(\boldsymbol{\sigma}, \sigma_i) \leq 2 \sum_{i=1}^{m} L_1(g, \sigma_i)$ for every function g.*

Proof. Setting S to be the set of all full rankings in Theorem 29, and using Lemma 28, the result follows. □

A.6.3 Stronger notions of optimality

Theorem 9 tells us that the median aggregation algorithm allows us to produce a top k list that is almost as good as any other top k list. However, we can show that the top k list produced is nearly optimal in an even stronger sense.

We say a partial ranking $\boldsymbol{\sigma}$ of type α is nearly optimal *in the strong sense* if there is some partial ranking $\boldsymbol{\sigma}'$ such that $\boldsymbol{\sigma} = \langle \boldsymbol{\sigma}' \rangle_\alpha$ and further $\boldsymbol{\sigma}'$ is nearly optimal. For instance, a top k list is nearly optimal in the strong sense if it represents the k most highly-ranked objects for some nearly optimal partial ranking.

We note first that this notion implies the weaker notion. Specifically, we have the following theorem.

Theorem 33 *Let $f_1, \ldots, f_m$ be functions mapping $D \to \mathcal{R}$. Suppose that partial ranking $\boldsymbol{\sigma}$ of type α is nearly optimal in the strong sense. More precisely, let $\boldsymbol{\sigma}'$ be a partial ranking such that $\boldsymbol{\sigma} = \langle \boldsymbol{\sigma}' \rangle_\alpha$ and for all partial rankings $\boldsymbol{\tau}'$,*

$$\sum_{i=1}^{m} L_1(\boldsymbol{\sigma}', f_i) \leq c \sum_{i=1}^{m} L_1(\boldsymbol{\tau}', f_i).$$

for some constant c. Then for all partial rankings $\boldsymbol{\tau}$ of type α, we have

$$\sum_{i=1}^{m} L_1(\boldsymbol{\sigma}, f_i) \leq (2c+1) \sum_{i=1}^{m} L_1(\boldsymbol{\tau}, f_i).$$

Proof.

$$\begin{aligned}
\sum_{i=1}^{m} L_1(\boldsymbol{\sigma}, f_i) &\leq \sum_{i=1}^{m} (L_1(\boldsymbol{\sigma}, \boldsymbol{\sigma}') + L_1(\boldsymbol{\sigma}', f_i)) \text{ by the triangle inequality} \\
&\leq \sum_{i=1}^{m} (L_1(\boldsymbol{\tau}, \boldsymbol{\sigma}') + L_1(\boldsymbol{\sigma}', f_i)) \text{ by Lemma 27} \\
&\leq \sum_{i=1}^{m} (L_1(\boldsymbol{\tau}, f_i) + L_1(f_i, \boldsymbol{\sigma}') + L_1(\boldsymbol{\sigma}', f_i)) \text{ by the triangle inequality} \\
&\leq (2c+1) \sum_{i=1}^{m} L_1(\boldsymbol{\tau}, f_i) \text{ by assumption}
\end{aligned}$$

□

Lemma 34 *Let α and β be types, let f be a function, and suppose that $\boldsymbol{\sigma} \in \langle f \rangle_\alpha$. Then there is a partial ranking $\boldsymbol{\sigma}' \in \langle f \rangle_\beta$ such that $\boldsymbol{\sigma} \in \langle \boldsymbol{\sigma}' \rangle_\alpha$.*

Proof. We first define a partial ranking $\boldsymbol{\rho}$ that is a refinement of both $\boldsymbol{\sigma}$ and $\hat{f}$, the induced partial ranking associated with f. For each pair i, j, if $\boldsymbol{\sigma}(i) < \boldsymbol{\sigma}(j)$ then set $\boldsymbol{\rho}(i) < \boldsymbol{\rho}(j)$. If $\boldsymbol{\sigma}(i) = \boldsymbol{\sigma}(j)$ and $f(i) < f(j)$, then set $\boldsymbol{\rho}(i) < \boldsymbol{\rho}(j)$. If $\boldsymbol{\sigma}(i) = \boldsymbol{\sigma}(j)$ and $f(i) = f(j)$, then set $\boldsymbol{\rho}(i) = \boldsymbol{\rho}(j)$. Notice that if $f(i) < f(j)$, then $\boldsymbol{\sigma}(i) \leq \boldsymbol{\sigma}(j)$ since f and $\boldsymbol{\sigma}$ are consistent with each other. So by definition, $\boldsymbol{\rho}(i) < \boldsymbol{\rho}(j)$ as well.

Assume $\boldsymbol{\sigma}' \in \langle \boldsymbol{\rho} \rangle_\beta$. We claim that $\boldsymbol{\sigma}'$ is consistent with f and with $\boldsymbol{\sigma}$. To see the first part, consider any i, j that $f(i) < f(j)$. Then $\boldsymbol{\rho}(i) < \boldsymbol{\rho}(j)$ as we noted above. Hence $\boldsymbol{\sigma}'(i) \leq \boldsymbol{\sigma}'(j)$ since $\boldsymbol{\sigma}'$ is consistent with $\boldsymbol{\rho}$. So $\boldsymbol{\sigma}'$ is consistent with f. Hence, $\boldsymbol{\sigma}' \in \langle f \rangle_\beta$. Similarly, if $\boldsymbol{\sigma}(i) < \boldsymbol{\sigma}(j)$, then $\boldsymbol{\rho}(i) < \boldsymbol{\rho}(j)$ by definition. Hence $\boldsymbol{\sigma}'(i) \leq \boldsymbol{\sigma}'(j)$. So $\boldsymbol{\sigma}$ is consistent with $\boldsymbol{\sigma}'$. Thus, $\boldsymbol{\sigma} \in \langle \boldsymbol{\sigma}' \rangle_\alpha$, as we wanted. □

We now prove that the median aggregation algorithm also produces a top k list that is nearly optimal in the strong sense. We actually prove a slightly more general theorem.

Theorem 35 *Let α be a type and let $f_1, \ldots, f_m$ be functions. Assume $f \in \text{median}(f_1, \ldots, f_m)$, and suppose that $\boldsymbol{\sigma} \in \langle f \rangle_\alpha$. Then there is a partial ranking $\boldsymbol{\sigma}'$ such that $\boldsymbol{\sigma} \in \langle \boldsymbol{\sigma}' \rangle_\alpha$ and for all partial rankings, $\boldsymbol{\tau}$, we have*

$$\sum_{i=1}^{m} L_1(\boldsymbol{\sigma}', f_i) \leq 3 \sum_{i=1}^{m} L_1(\boldsymbol{\tau}, f_i)$$

Furthermore, if the functions $f_1, \ldots, f_m$ are in fact partial rankings then we have for all functions, g,

$$\sum_{i=1}^{m} L_1(\boldsymbol{\sigma}', f_i) \leq 2 \sum_{i=1}^{m} L_1(g, f_i)$$

Proof. Let $f^\dagger$ be a partial ranking such that $L_1(f^\dagger, f)$ is minimized. Let β be the type of $f^\dagger$. By Lemma 34, there is a partial ranking $\boldsymbol{\sigma}'$ such that $\boldsymbol{\sigma}' \in \langle f \rangle_\beta$ and $\boldsymbol{\sigma} \in \langle \boldsymbol{\sigma}' \rangle_\alpha$. Since $\boldsymbol{\sigma}' \in \langle f \rangle_\beta$, Lemma 27 guarantees that $L_1(\boldsymbol{\sigma}', f) = L_1(f^\dagger, f)$. Hence, by Theorem 10, the corollary follows. □

A.6.4 The dynamic programming algorithm

Let $|D| = n$. We now describe an algorithm that given a function $f \in \text{median}(f_1, \ldots, f_m)$ for partial orders $f_1, \ldots, f_m$, finds a partial ranking $f^\dagger$ so that $L_1(f, f^\dagger)$ is minimized. (Note that the algorithm does not actually need f to be a median function.) It is easy to produce an algorithm running in time $O(n^2)$ if we are allowed to use $O(n^2)$ space. If we make the additional assumption that $2f(i)$ is integral for all i, then we have an algorithm that runs in linear space and time $O(n^2)$. Note that this assumption is not very restrictive, since the median function for a set of partial orders will always satisfy this when the median does not average two values (For instance, if we have a set of m values $a_1 \leq a_2 \leq \ldots \leq a_m$, we take the median value to be $a_{\lfloor \frac{m+1}{2} \rfloor}$.)

Suppose without loss of generality that $f(1) \leq f(2) \leq \ldots \leq f(n)$. For convenience, define $f(0) = -\infty$. Let π be the total order $1, 2, \ldots, n$. By Lemma 27, there is some type α such that if $f^\dagger \in \langle \pi \rangle_\alpha$ then $L_1(f, f^\dagger) \leq L_1(f, \boldsymbol{\tau})$ for all partial ranking $\boldsymbol{\tau}$. We can determine such a minimal type using dynamic programming.

To do so, we first need several definitions. For any i, j with $0 \leq i < j \leq n$, we define

$$c(i,j) = \sum_{\ell=i+1}^{j} \left| f(\ell) - \frac{i+j+1}{2} \right|$$

To motivate our definition of $c(i,j)$, imagine that we alter the type of π so that there is a bucket starting at $i+1$ and going until j. Then the position of that bucket is $\frac{i+j+1}{2}$, and the distance between that bucket and f (on the values $\{i+1, i+2, \ldots, j\}$) is precisely $c(i,j)$.

In general, let $\mathcal{S}$ be a sequence $s_0 < s_1 < \cdots < s_t$. Then we define

$$c(\mathcal{S}) = \sum_{\ell=0}^{t-1} c(s_\ell, s_{\ell+1})$$

Intuitively, we think of each s_ℓ as marking a point where one bucket starts and the next begins. The important thing to notice about this is that there is a one-to-one correspondence between types on the domain $\{1, 2, \ldots, n\}$ and strictly increasing sequences that begin with 0 and end with n. More precisely, let β be a type with t buckets represented by the sequence $b_1, b_2, \ldots, b_t$. Define $seq(\beta)$ to be the sequence $s_0, s_1, \ldots, s_t$, where $s_0 = 0$ and $s_{\ell+1} = s_\ell + b_\ell$ for all $\ell \geq 0$. It is easy to check that the function $seq(\cdot)$ is one-to-one. Further, we see immediately that if $\boldsymbol{\tau} \in \langle \pi \rangle_\beta$ then

$$L_1(f, \boldsymbol{\tau}) \;=\; \sum_{\ell=0}^{t-1} c(s_\ell, s_{\ell+1}) = c(seq(\beta)) \tag{13}$$

Our dynamic programming algorithm will calculate a sequence $\mathcal{S}_n$ starting with 0 and ending with n such that $c(\mathcal{S}_n)$ is minimized. To this end, we find $n+1$ different sequences, $\mathcal{S}_0, \mathcal{S}_1, \mathcal{S}_2, \ldots, \mathcal{S}_n$. For all $j > 0$, the sequence $\mathcal{S}_j$ will have the property that its first element is 0, and its last element is j. Our goal is to have $c(\mathcal{S}_j)$ minimal over all such sequences.

To this end, define $\mathcal{S}_0 = 0$, and recursively define $\mathcal{S}_j = \mathcal{S}_{i_0}, j$, where $i_0 = \text{argmin}_i[c(S_i) + c(i,j)]$. Then we have the following.

Lemma 36 *Let $\mathcal{S}_0, \mathcal{S}_1, \ldots, \mathcal{S}_n$ be defined as above. Then for all j and for all strictly increasing sequences $\mathcal{S}'_j$ that start with 0 and end with j, we have $c(\mathcal{S}'_j) \geq c(\mathcal{S}_j)$.*

Proof. We proceed by induction. The case $j = 0$ is trivially true. So assume that $j > 0$ and that our claim is true for all indices smaller than j.

Given f such that $f(1) \leq f(2) \leq \ldots \leq f(n)$
1 **FOR** $j := 1$ **TO** n
2 Set $c(0,j) := \sum_{\ell=1}^{j-1} |f(\ell) - \frac{j}{2}|$.
3 Set $k := 0$.
4 **FOR** $i := 1$ **TO** $j-1$
5 **WHILE** ($k \leq j$ and $f(k) < \frac{i+j+1}{2}$) increment k
6 Set $c(i,j) := c(i-1,j) - \left|f(i) - \frac{i+j}{2}\right| + \frac{2k-i-j-2}{2}$.
7 Find $i_0 \geq 0$ such that $[c(\mathcal{S}_{i_0}) + c(i_0,j)]$ minimized.
8 Set $\mathcal{S}_j = \mathcal{S}_{i_0}, j$.
9 Output $\mathcal{S}_n$.

Figure 1: Pseudocode to compute optimal sequence $\mathcal{S}_n$.

Now, let $\mathcal{S}'_j$ be a strictly increasing sequence starting with 0 and ending with j. Suppose the penultimate element of $\mathcal{S}'_j$ is i. Then there is a strictly increasing sequence $\mathcal{S}'_i$ ending with i such that $\mathcal{S}'_j = \mathcal{S}'_i, j$. By definition, $c(\mathcal{S}'_j) = c(\mathcal{S}'_i) + c(i,j)$. But by induction, $c(\mathcal{S}'_i) \geq c(\mathcal{S}_i)$. Hence, $c(\mathcal{S}'_j) \geq c(\mathcal{S}_i) + c(i,j) \geq c(\mathcal{S}_j)$. □

Using Lemma 36, it is easy to see that the type associated with $\mathcal{S}_n$ is optimal. That is, suppose $\mathcal{S}_n$ is the sequence $s_0 < s_1 < \ldots < s_t$, and let α be the shape given by the sequence $s_1 - s_0, s_2 - s_1, \ldots, s_t - s_{t-1}$. Let $f^\dagger \in \langle \pi \rangle_\alpha$. Then for any partial ranking $\boldsymbol{\tau}$, we have

$$L_1(f, f^\dagger) = c(\mathcal{S}_n) \leq c(seq(type(\boldsymbol{\tau}))) = L_1(f, \boldsymbol{\tau})$$

by equation 13.

Given the recurrence relation, it is a simple matter to calculate $\mathcal{S}_n$. Since $c(i,j)$ can be calculated in $O(n)$ time for all i, j, we see there is a simple algorithm to calculate $\mathcal{S}_n$ is time $O(n^3)$. However, we can in fact calculate $c(i,j)$ is amortized $O(1)$ time. In the case where we do not have memory restrictions, we simply utilize the following recurrence:

$$c(i-1, j+1) = c(i,j) + \left|f(i-1) - \frac{i+j}{2}\right| + \left|f(j+1) - \frac{i+j}{2}\right|$$

Using this, we can calculate $c(i,j)$ for all i, j in $O(n^2)$ time, but $O(n^2)$ space.

If $2f(i)$ is integral for all i, then we can calculate $\mathcal{S}_n$ in linear space and time $O(n^2)$ using a slightly more complicated algorithm. The pseudocode is shown in Figure 1. The following lemma is the key idea.

Lemma 37 *Let i, j, k be integers, and suppose that either (1) $f(k-1) \leq \frac{i+j}{2} < \frac{i+j+1}{2} \leq f(k)$ with $k \leq j$, or (2) $f(j) \leq \frac{i+j}{2}$ with $k = j+1$. Then in both cases, $c(i,j) = c(i-1,j) - \left|f(i) - \frac{i+j}{2}\right| + \frac{2k-i-j-2}{2}$.*

Proof. We first consider the case $f(k-1) \leq \frac{i+j}{2} < \frac{i+j+1}{2} \leq f(k)$.

$$\begin{aligned} c(i,j) &= \sum_{\ell=i+1}^{j} \left|f(\ell) - \frac{i+j+1}{2}\right| \\ &= \sum_{\ell=i+1}^{k-1} \left|f(\ell) - \frac{i+j+1}{2}\right| + \sum_{\ell=k}^{j} \left|f(\ell) - \frac{i+j+1}{2}\right| \end{aligned}$$

$$
\begin{aligned}
&= \sum_{\ell=i+1}^{k-1} \left(\frac{i+j+1}{2} - f(\ell) \right) + \sum_{\ell=k}^{j} \left(f(\ell) - \frac{i+j+1}{2} \right) \\
&= \sum_{\ell=i+1}^{k-1} \left(\frac{i+j}{2} - f(\ell) \right) + \frac{k-i-1}{2} + \sum_{\ell=k}^{j} \left(f(\ell) - \frac{i+j}{2} \right) - \frac{j-k+1}{2} \\
&= \sum_{\ell=i+1}^{k-1} \left| f(\ell) - \frac{i+j}{2} \right| + \sum_{\ell=k}^{j} \left| f(\ell) - \frac{i+j}{2} \right| + \frac{2k-i-j-2}{2} \\
&= c(i-1,j) - \left| f(i) - \frac{i+j}{2} \right| + \frac{2k-i-j-2}{2}
\end{aligned}
$$

As for the case $f(j) \leq \frac{i+j}{2}$ with $k = j+1$, we have

$$
\begin{aligned}
c(i,j) &= \sum_{\ell=i+1}^{j} \left| f(\ell) - \frac{i+j+1}{2} \right| \\
&= \sum_{\ell=i+1}^{j} \left(\frac{i+j+1}{2} - f(\ell) \right) \\
&= \sum_{\ell=i+1}^{j} \left| \frac{i+j}{2} - f(\ell) \right| + \frac{j-i}{2} \\
&= c(i-1,j) - \left| f(i) - \frac{i+j}{2} \right| + \frac{2k-i-j-2}{2}
\end{aligned}
$$

□

Using Lemma 37, we can finish analyzing the algorithm. Referring to the pseudocode in Figure 1, notice that if $2f(i)$ is integral for all i, then whenever we exit the **WHILE** loop, one of the two conditions from the previous lemma holds. So the algorithm correctly computes $c(i,j)$ for each i, j. Turning to the running time, notice that for each iteration of the outer loop, the value of k increases from 0 to at most $j+1$. Likewise, i increases from 1 to j. Finally, calculating both $c(0,j)$ in step 2 and the minimal i_0 in step 7 can be done in $O(n)$ time. So in total, the algorithm runs in time $O(n^2)$.