A Weighted Generalization of the Graham-Diaconis Inequality for Ranked List Similarity

Ali Dasdan
KD Consulting
Saratoga, CA, USA
alidasdan@gmail.com
April 17, 2018

Abstract

The Graham-Diaconis inequality shows the equivalence between two well-known methods of measuring the similarity of two given ranked lists of items: Spearman’s footrule and Kendall’s tau. The original inequality assumes unweighted items in input lists. In this paper, we first define versions of these methods for weighted items. We then prove a generalization of the inequality for the weighted versions.

1 Introduction

The field of similarity computation is more than a century old, e.g., see the review at [2]; it offers many measures to compute similarity depending on how the inputs are modelled. In this paper, we focus on computing the similarity of ranked lists.

Among the many list similarity measures in the literature, Spearman’s footrule and Kendall’s tau are commonly used. In this paper, we generalize these measures to include weights and also to work for partial lists as well as permutations. The main contribution of this paper is to prove the equivalence
of these weighted versions, by generalizing a proof by Diaconis and Graham for the unweighted versions. Here equivalence means these measures are within small constant multiples of one another.

2 Related Work

A detailed related work on similarity in general is given in [2]. For more recent work, refer to [1, 5, 6, 8]. For other ways of incorporating weights with the similarity measures in question, refer to [1, 5, 6, 8, 9, 10].

The work in this paper and that in [2] were actually part of a comprehensive multi-year project on all forms of similarity for web search metrics at Yahoo! Web Search; we started the project around 2007. An internal version of [2] from the year 2009 with our proof was cited in [8].

3 Rank Assignment

Consider the ordered or ranked lists \( \sigma = (a, b, c, d, e, f) \) and \( \pi = (b, f, a, e, d, c) \). These lists are full in that they are permutations of each other, which in turn means these lists contain the same items or elements, possibly in a different order. The rank of an element \( i \) in \( \sigma \) gives its order and is denoted by \( \sigma(i) \). For example, the rank of \( e \) in \( \sigma \) is \( \sigma(e) = 5 \) whereas its rank in \( \pi \) is \( \pi(e) = 4 \).

Now consider the ordered lists \( \sigma = (a, b, c) \) and \( \pi = (b, e, c, f) \). These lists are partial in that they are not permutations of each other. This means one list may contain elements that the other list does not. Moreover, their sizes may be different too. For example, \( a \) is present in \( \sigma \) but absent from \( \pi \); also, \( \sigma \) has a length of 3 whereas \( \pi \) has a length of 4. For each list, the rank of an element in that list is well defined.

For comparing ranked partial lists, it is convenient to consider that these partial lists are actually permutations of each other, with missing elements somehow added at the end of each partial list. The motivation for this view comes from web search engines in that a search result that is missing from the first page of shown (usually 10) results is most probably still in the search index, which is huge, but not shown.

To figure out how to add the missing elements, consider again our example partial lists \( \sigma = (a, b, c) \) and \( \pi = (b, d, c, e) \). Completing these lists means having each list contain all the elements from their union, while pre-
serving the rank of their existing elements. The union of $\sigma$ and $\pi$ is equal to \{a, b, c, d, e\}, where we use the set notation to imply unordering. Comparing these lists to their union shows that $\sigma$ is missing \{d, e\} whereas $\pi$ is missing \{a\}. Let $\sigma'$ denote the completion of $\sigma$. Here we have two options for $\sigma'$: 1) $\sigma' = (a, b, c, d, e)$ or 2) $\sigma' = (a, b, c, e, d)$. For $\pi$, we have a single option: $\pi' = (b, d, c, e, a)$. In the literature [4], another option is to put $d$ and $e$ at the same rank; we will not consider it here due to the same search engine analogy above, i.e., if the search engine returns more results, they will again be ranked.

Now consider the similarity between $\sigma'$ and $\pi'$. Since $d$ is before $e$ in $\pi$ as well as $\pi'$, the first option for $\sigma'$ increases the similarity whereas the second option decreases the similarity. Either option can be used for completing partial lists to permutations; our choice in this paper is the first option yet again due to the same search engine analogy: Assuming a competitor search engine has the same elements at the same rank will provide motivation to speed up innovation to beat the competition.

Now that we know how to transform partial lists to become full lists, we will focus only on full lists in the sequel. We will do one more simplification in that without loss of generality, we represent the lists using the ranks of their elements rather than the elements themselves. For example, $\sigma = (a, b, c, d, e, f)$ and $\pi = (b, f, a, e, d, c)$ become $\sigma = (1, 2, 3, 4, 5, 6)$ and $\pi = (2, 6, 1, 5, 4, 3)$. Without loss of generality, we could also have used $\pi$ as the reference list to determine the integer names of the elements in $\sigma$.

4 Weighted Measures

We now define the weighted versions of Spearman’s footrule and Kendall’s tau. We also provide examples.

Note that the normalized forms of these measures map to the interval $[0, 1]$ where 0 means the two input lists are ranked in the same order and 1 means the two input lists are ranked in the opposite order. If a mapping to $[-1, 1]$ is desired, where 1 and $-1$ indicate the same and opposite orders respectively, the normalized value $v$ needs to be transformed to $1 - 2v$. 

3
4.1 Weighted Spearman’s Footrule

We define the weighted version of Spearman’s footrule \([3, 12]\) for lists of length \(n\) as

\[
S_w(\sigma, \pi) = \sum_{i \in \sigma \cup \pi} w(i) |\sigma(i) - \pi(i)|. \tag{1}
\]

where \(w(i)\) returns a positive number as the weight of the element \(i\).

The measure \(S_w\) can be normalized to the interval \([0, 1]\) as

\[
s_w(\sigma, \pi) = \frac{S_w(\sigma, \pi)}{\sum_{i=1}^{n} w(i)|i - (n - i + 1)|} \tag{2}
\]

where the denominator reaches its maximum when both lists are sorted but in opposite orders.

Fig. 1 shows the algorithms, written in the Python programming language, to compute both the numerator and the denominator of \(s_w(\sigma, \pi)\) in Eq. 2 as well as \(s_w(\sigma, \pi)\) itself as their ratio. Using unity weights lead to the unweighted versions of these algorithms.

4.2 Weighted Kendall’s Tau

The unweighted Kendall’s tau is the number of swaps we would perform during the bubble sort to reduce one permutation to another. As we described
Figure 2: Weighted Kendall’s tau in the Python programming language. Use unity weights to derive the unweighted versions.

The way we determine the ranks of the extended lists (§3), we can always assume that the first list \( \sigma \) is the identity (increasing from 1 to \( n \)), and what we need to compute is the number of swaps to sort the permutation \( \pi \) back to the identity permutation (increasing). Here, a weight will be associate to each swap.

We define a weighted version of Kendall’s tau [7, 11] for lists of length \( n \) as

\[
K_w(\sigma = \iota, \pi) = \sum_{1 \leq i < j \leq n} \frac{w(i) + w(j)}{2} \left[ \pi(i) > \pi(j) \right] 
\]  

where \([x]\) is equal to 1 if the condition \( x \) is true and 0 otherwise.

The measure \( K_w \) can be normalized to the interval [0, 1] as

\[
k_w = \frac{K_w(\sigma, \pi)}{\sum_{\{i,j\in\sigma \cup \pi; i<j\}} \frac{w(i) + w(j)}{2}}
\]

where the value of the denominator is exactly the maximum value that the numerator can reach when both lists are sorted but in opposite orders.

Fig. 2 shows the algorithms, written in the Python programming language, to compute both the numerator and the denominator of \( k_w(\sigma, \pi) \) in Eq. 4 as well as \( k_w(\sigma, \pi) \) itself as their ratio. Using unity weights lead to the unweighted versions of these algorithms.
4.3 Examples

For the sake of simplicity, let us assume that \( w(i) = w \) for all \( i \) in this section. We will have three examples. Let us compute \( S_w, K_w, s_w, \) and \( k_w \) for each example.

**Example 1.** Given \( \sigma = (a, b, c, d, e) \) and \( \pi = (a, b, c, d, e) \), we have \( \sigma' = (a, b, c, d, e) \sim (1, 2, 3, 4, 5) \) and \( \pi' = (a, b, c, d, e) \sim (1, 2, 3, 4, 5) \). Then,

\[
S_w = 0w = 0 \quad \text{and} \quad K_w = 0w = 0.
\]

Also,

\[
s_w = \frac{0w}{12w} = 0 \quad \text{and} \quad k_w = \frac{0w}{10w} = 0.
\]

**Example 2.** Given \( \sigma = (a, b, c, d, e) \) and \( \pi = (e, d, c, b, a) \), we have \( \sigma' = (a, b, c, d, e) \sim (1, 2, 3, 4, 5) \) and \( \pi' = (e, d, c, b, a) \sim (5, 4, 3, 2, 1) \). Then,

\[
S_w = w(|1 - 5| + |2 - 4| + |3 - 3| + |4 - 2| + |5 - 1|) = w(4 + 2 + 0 + 2 + 4) = 12w
\]

and

\[
K_w = w([5 > 4] + [5 > 3] + [5 > 2] + [5 > 1] + [4 > 3] + [4 > 2] + [4 > 1] + [3 > 2] + [3 > 1] + [2 > 1]) = 10w.
\]

Also,

\[
s_w = \frac{12w}{12w} = 1 \quad \text{and} \quad k_w = \frac{10w}{10w} = 1.
\]

**Example 3.** Given \( \sigma = (a, b, c) \) and \( \pi = (b, d, c, e) \), we first extend them to full lists and replace elements by their ranks with \( \sigma \) as the reference list: \( \sigma' = (a, b, c, d, e) \sim \sigma' = (1, 2, 3, 4, 5) \) and \( \pi' = (b, d, c, e, a) \sim \pi' = (2, 4, 3, 5, 1) \). Then,

\[
S_w = w(|1 - 2| + |2 - 4| + |3 - 3| + |4 - 5| + |5 - 1|) = w(1 + 2 + 0 + 1 + 4) = 8w
\]
and

\[ K_w = w([2 > 1] + [4 > 3] + [4 > 1] + [3 > 1] + [5 > 1]) \]
\[ = w(1 + 1 + 1 + 1 + 1) \]
\[ = 5w, \]

where all the other comparisons return zero.

Then the normalized measures become

\[ s_w = \frac{8w}{12w} = \frac{2}{3} \approx 0.67 \text{ and } k_w = \frac{5w}{10w} = \frac{1}{2} = 0.5. \]

Notice that \( K_w \leq S_w \leq 2K_w \) because \( 5w \leq 8w \leq 10w \).

5 Equivalence Between Measures

We now prove that the Graham-Diaconis inequality between Spearman’s footrule and Kendall’s tau is valid for weighted ranked lists too. This inequality shows that these measures with or without weights are within small constant multiples of each other, which is another way of saying that these measures are equivalent [4].

Denote by \( S \) and \( K \) the unweighted versions of \( S_w \) and \( K_w \), respectively. The unweighted versions are obtained by setting each weight in \( S_w \) and \( K_w \) to unity. This equivalence proof allows us to use the simpler of these two measures, Spearman’s footrule, as our list similarity measure even for the weighted case.

For permutations, the equivalence between the unweighted versions \( S \) and \( K \) is well known from the following classical result [3]:

**Theorem 5.1** (Diaconis-Graham) For every two permutations \( \sigma \) and \( \pi \)

\[ K(\sigma, \pi) \leq S(\sigma, \pi) \leq 2K(\sigma, \pi). \]  \( (5) \)

For a discussion on the equivalence of other unweighted list similarity measures, see [4].

For weighted permutations, we generalize this result to the equivalence between \( S_w \) and \( K_w \).

**Theorem 5.2** For every two permutations \( \sigma \) and \( \pi \),

\[ K_w(\sigma, \pi) \leq S_w(\sigma, \pi) \leq 2K_w(\sigma, \pi) \]  \( (6) \)

where every weight is a positive number.
Our proof below closely follows the notation and reasoning of the original proof in [3] and extends it to the weighted case.

**Proof.** Before proving this theorem, we need the following preliminary facts, which are the same as in [3]. Note that the element weights do not invalidate these facts.

Assume that all permutations below are defined on the same set, where a permutation is a bijection (i.e., one-to-one and onto function) from some set \(X\) of size \(n\) to \(\{1, \ldots, n\}\).

**Metric space.** Both \(S\) and \(K\) are metrics, meaning that if \(d\) denotes one of these measures, then \(d\) satisfies the metric properties: \(d(\sigma, \pi) \geq 0\) (i.e., non-negativity); \(d(\sigma, \pi) = 0\) if and only if (iff) \(\sigma = \pi\) (i.e., identity of indiscernible); \(d(\sigma, \pi) = d(\pi, \sigma)\) (i.e., symmetry); and, \(d(\sigma, \pi) \leq d(\sigma, \eta) + d(\eta, \pi)\) for some permutation \(\eta\) (i.e., triangle inequality).

**Right invariance.** Both \(S\) and \(K\) are right invariant, meaning if \(d\) denotes one of these measures, then \(d\) is right invariant if a permutation \(\eta\) exists such that \(d(\sigma, \pi) = d(\sigma \eta, \pi \eta)\). In particular, \(d(\iota, \sigma) = d(\sigma^{-1}, \iota) = d(\iota, \sigma^{-1})\) where \(\iota\) stands for the identity permutation on \(X\) and \(\sigma^{-1}\) is the permutation inverse to \(\sigma\) (i.e., \(\sigma \sigma^{-1} = \iota\)). To simplify the notation, we abbreviate \(d(\iota, \sigma)\) by \(d(\sigma)\), hence, \(d(\sigma) = d(\sigma^{-1})\).

Now we come to the proof of the theorem. We divide the proof into two parts. We prove first \(S_w(\sigma) \leq 2K_w(\sigma)\) and then \(K_w(\sigma) \leq S_w(\sigma)\).

The proof of \(S_w(\sigma) \leq 2K_w(\sigma)\). Recall that \(K(\sigma) = K(\sigma^{-1})\) is the smallest number of pairwise adjacent transpositions or swaps required to bring \(\sigma\) to the identity \(\iota\). Note that bubble sort will make the same number of swaps to sort its input list. Let \(x_i, 1 \leq i \leq K(\sigma)\), be a sequence of integers that indexes a sequence of transpositions transforming \(\iota\) to \(\sigma_1\) to \(\sigma_2\) to \ldots to \(\sigma\). The \(i\)-th transposition transforms \(\sigma_i\) to \(\sigma_{i+1}\) by interchanging \(\sigma_i(x_i)\) to \(\sigma_i(x_i + 1)\), i.e., the element at index \(x_i\) and the next element at index \(x_i + 1\). Without loss of generality, we may assume that \(\sigma_i(x_i) < \sigma_i(x_i + 1)\). Consider the difference

\[
\Delta_{i+1} = S_w(\sigma_{i+1}) - S_w(\sigma_i) = (w(x_i + 1)|x_i - \sigma_i(x_i + 1)| + w(x_i)|x_i + 1 - \sigma_i(x_i)|) - (w(x_i)|x_i - \sigma_i(x_i)| + w(x_i + 1)|x_i + 1 - \sigma_i(x_i + 1)|)
\]

where the contributions from the elements whose positions have not changed, i.e., the elements at indices other than \(x_i\) and \(x_i + 1\), cancel out.
There are three possibilities, each of which removes the absolute values in Eq. 7 to compute the difference.

- **Case 1.** If \( \sigma_i(x_i) < \sigma_i(x_i + 1) \leq x_i < x_i + 1 \), then \( \Delta_{i+1} = -w(x_i + 1) + w(x_i) \).

- **Case 2.** If \( x_i < x_i + 1 \leq \sigma_i(x_i) < \sigma_i(x_i + 1) \), then \( \Delta_{i+1} = w(x_i + 1) - w(x_i) \).

- **Case 3.** If \( \sigma_i(x_i) \leq x_i < x_i + 1 \leq \sigma_i(x_i + 1) \), then \( \Delta_{i+1} = w(x_i + 1) + w(x_i) \).

Thus \( S_w(\sigma) = \sum_{i=1}^{K(\sigma)} \Delta_i \leq 2K_w(\sigma) \) because \( \Delta_{i+1} \leq 2((w(x_i + 1) + w(x_i))/2) \) for each of the three cases above. Note that the expression of \( \Delta_{i+1} \) also explains our choice of the additive weight aggregation in \( K_w \).

**The proof of** \( K_w(\sigma) \leq S_w(\sigma) \). To prove the left-hand side of the inequality, first denote the inversion \( \sigma(i) > \sigma(j) \) with \( i < j \) by \([i; j]\). We then simplify Eq. 3 for \( K_w(\sigma) \) as

\[
K_w(\sigma) = \sum_{1 \leq i < j \leq n} \frac{w(i) + w(j)}{2}[\sigma(i) > \sigma(j)]
\]

or

\[
2K_w(\sigma) = \sum_{1 \leq i < j \leq n} w(i)[\sigma(i) > \sigma(j)] + \sum_{1 \leq i < j \leq n} w(j)[\sigma(i) > \sigma(j)],
\]  

(8)

where \([x]\) is equal to 1 if the condition \( x \) is true and 0 otherwise. Note that \( 2K_w \) is the sum of all inversions \([i; j]\) in \( \sigma \) and the weight for each inversion is added twice, once for \( \sigma(i) \) and another for \( \sigma(j) \). For the proof, we will show that \( S_w \) upper-bounds the total number of inversions, hence, each term in \( 2K_w \) separately.

**Define two types of inversions:** Type I and Type II. Call an inversion \([i; j]\) a Type I inversion if \( \sigma(i) \geq j \) and a Type II inversion if \( \sigma(i) \leq j \). Note that every inversion of \( \sigma \) is either a Type I or a Type II inversion or both. Similarly, the sum of Type I and Type II inversions in \( \sigma \) upper-bounds \( K_w(\sigma) \).
For a fixed $i$, if $[i; k]$ is a Type I inversion, then we must have $i < k \leq \sigma(i)$. Thus, the number of Type I inversions, or the number of possible $k$, is at most $\sigma(i) - i$ and the total weight is at most $w(i)(\sigma(i) - i)$. Similarly, if $[k; j]$ is a Type II inversion, then we must have $\sigma(j) < \sigma(k) \leq j$. Thus, the number of Type II inversions, or the number of possible $k$, is at most $j - \sigma(j)$ and the total weight is at most $w(i)(j - \sigma(j))$. Then, it follows that the first term in Eq. 8 is at most the sum of the number of Type I inversions and the number of Type II inversions, which is further upper-bounded as

$$\sum_{\sigma(i) \geq i} w(i)(\sigma(i) - i) + \sum_{\sigma(j) \leq j} w(i)(j - \sigma(j)) = \sum_{i} w(i)|\sigma(i) - i| = S_w(\sigma).$$

Similarly, for a fixed $j$, an argument similar to the case for $i$ and $w(i)$ can be carried out for the case for $j$ and $w(j)$ to prove that the second term in Eq. 8 is at most $S_w(\sigma)$. Combined, these two arguments show that $2K_w(\sigma) \leq 2S_w(\sigma)$ or $K_w(\sigma) \leq S_w(\sigma)$. QED.

We generalize this result to the equivalence between $S_w$ and $K_w$ for ranked partial lists.

**Theorem 5.3** For partial lists $\sigma$ and $\pi$ with the rank assignment for missing elements as described in §3.

$$K_w(\sigma, \pi) \leq S_w(\sigma, \pi) \leq 2K_w(\sigma, \pi) \quad (9)$$

where every weight is a positive number.

**Proof.** The proof directly follows from the way we assign ranks of the missing elements in §3. This is because the resulting lists become permutations of each other. Thus, Theorem 5.2 applies. QED.

### 6 Experimental Results

To provide more insight into these algorithms, we performed two sets of experiments.

The first set of experiments is about understanding the distribution of the ratio of Spearman’s footrule to Kendall’s tau. We computed this ratio over all permutations of a 10-element unweighted list. By Theorem 5.1, this ratio
Figure 3: Distribution of the ratio of Spearman’s footrule to Kendall’s tau over all permutations of a 10-element unweighted list.

fits in the range from 1 to 2, inclusive. Fig. 3 shows the outcome distribution with orange lines marking the multiples of the standard deviation (0.14). The distribution parameters are also shown in this figure. Note that the median, mean, and mode are at or very close to 1.50, the middle value of the range; although this may almost imply a balanced distribution, the distribution is actually slightly right-skewed with a skewness of 0.42, which is also noticeable visually.

The second set of experiments is about understanding the distributions of the normalized measures. We again computed the normalized values over all permutations of a 10-element unweighted list. The normalized values fit in the range from 0 to 1, inclusive. Fig. 4 shows the outcome distributions for both the normalized measures. As is also obvious visually, the normalized Kendall’s tau is distributed in a balanced way with the median, mean, and mode are at the middle value 0.50 of the range (with a standard deviation of 0.13) whereas the normalized Spearman’s footrule is distributed with a negative skewness (i.e., left-skewed) of -0.18. The other parameters for the normalized Spearman’s footrule are 0.66, 0.66, and 0.68 for the median, mean, and mode, respectively. The standard deviation is 0.14.
Figure 4: Distribution of the normalized Spearman’s footrule and Kendall’s tau over all permutations of a 10-element unweighted list.

7 Conclusions

The field of similarity computation is more than a century old. In this paper, we focus only on the similarity between two ranked lists. Such lists may be full (permutations) or partial. For permutations, the classical Graham-Diaconis inequality show their equivalence using two well-known similarity measures: Spearman’s footrule and Kendall’s tau.

In this paper, we consider ranked lists that may be partial and ranked lists that may be weighted (with element weights) or both. We first propose a rank assignment method to convert partial lists to permutations. Next, we define weighted versions of Spearman’s footrule and Kendall’s tau. Finally, we generalize the Graham-Diaconis inequality to permutations with element weights. Due to the form of our rank assignment, we also show that the same weighted generalization applies to partial lists.

Acknowledgments

I thank Paolo D’Alberto for a joint work that subsumed this work and Ravi Kumar for discussions on the weighted form of Kendall’s tau.
References

[1] J. Chen, Y. Li, and L. Deng. On the equivalence of weighted metrics for distance measures between two rankings. *J. of Information and Computational Science*, 11:13:4477–85, 2014.

[2] P. D’Alberto and A. Dasdan. On the weaknesses of correlation. *CoRR*, abs/1107.2691, 2011.

[3] P. Diaconis and R. Graham. Spearman’s footrule as a measure of disarray. *J. Roy. Statistics Soc.*, 39(Ser. B):262–268, 1977.

[4] R. Fagin, R. Kumar, and D. Sivakumar. Comparing top k lists. *SIAM J. Discrete Math.*, 17(1):134–160, 2003.

[5] F. Farnoud, B. Touri, and O. Milenkovic. Nonuniform vote aggregation algorithms. *CoRR*, abs/1206.5343, 2012.

[6] F. F. Hassanzadeh, O. Milenkovic, G. J. Puleo, and L. Su. Computing similarity distances between rankings. *Discrete Applied Mathematics*, 232:157–175, 2017.

[7] M. G. Kendall. A new measure of rank correlation. *Biometrika*, 30(1–2):81–93, Jun. 1938.

[8] R. Kumar and S. Vassilvitskii. Generalized distances between rankings. In *Proc. Int. Conf. World Wide Web (WWW)*, pages 571–580, 2010.

[9] G. B. Shieh. A weighted Kendall’s tau statistic. *Statist. Probab. Lett.*, 39:17–24, 1998.

[10] G. B. Shieh, Z. Bai, and W. Y. Tsai. Rank tests for independence - with a weighted contamination alternative. *Statistica Sinica*, 10:577–593, 2000.

[11] G. L. Sievers. Weighted rank statistics for simple linear regression. *J. of the American Stat. Assoc.*, 73(363):628–631, Sep 1978.

[12] C. Spearman. A footrule for measuring correlation. *British J of Psychology*, 2:89–108, 1906.