Statistical Applications in Genetics and Molecular Biology

Volume 9, *Issue* 1 2010 *Article* 20

Space Oriented Rank-Based Data Integration

Shili Lin[*]

[*]The Ohio State University, shili@stat.ohio-state.edu

Space Oriented Rank-Based Data Integration*

Shili Lin

Abstract

Integration of data from multiple omics platforms has become a major challenge in studying complex systems and traits. For integrating data from multiple platforms, the underlying spaces from which the top ranked elements come from are likely to be different. Thus, taking the underlying spaces into consideration explicitly is important, as failure to do so would lead to inefficient use of data and might render biases and/or sub-optimal results. We propose two space oriented classes of heuristic algorithms for integrating ranked lists from omic scale data. These algorithms are either Borda inspired or Markov chain based that take the underlying spaces of the individual ranked lists into account explicitly. We applied this set of algorithms to a number of problems, including one that aims at aggregating results from three cDNA and two Affymetrix gene expression studies in which the underlying spaces between Affymetrix and cDNA platforms are clearly different.

KEYWORDS: Borda's method, Markov chain, omic-scale data, rank aggregation, top-k lists, underlying space

*This work was supported partially by an NSF grant to the Mathematical Biosciences Institute DMS-0112050. The author would like to thank two referees for their suggestions and comments.

1 Introduction

Integration of data from multiple omics platforms has become a major challenge in studying complex systems and traits. Each of the omic-scale data contains valuable information on various aspects of the whole biological system, and analyzing them jointly in an integrative fashion would most likely lead to greater insight. Integration of omic-scale data may be approached through "low-level" meta-analytic schemes by modeling the raw data directly (Schadt et al., 2005). In many situations, however, it would be hard (if not impossible given the limited understanding of the whole biological system to date) to model the interplay of the raw data to make use of all the information available. In such cases, a "high-level" data analytic method such as those based on rankings offers an alternative to integrate individual results to arrive at some consensus that is more "reliable" than any of the individual studies (Lin and Ding, 2009). In fact, ranked-based methods have been increasingly used in analyzing omics data (e.g., Conlon et al., 2007; Fishel et al., 2009; Liu et al., 2008; Tan et al., 2005; Yuen et al., 2002). The properties of being invariant to transformation and normalization are the important features of ranked-based methods for integrating data from different platforms.

There has been a number of methods adapted or specifically proposed for aggregating ranked lists from omics platforms (DeConde et al., 2006; Hall and Schimek, 2009; Lin and Ding, 2009). Such methods have been demonstrated to be able to handle multiple platform data. For example, suppose data are available for copy number variation from SNP arrays, gene expression from Affymetrix arrays, and DNA-protein binding data from ChIP-seq, and the goal is to find putative genes whose functionality may have been altered in a particular disease. In such a case, a rank-based methods may be used to synthesize information from the individual studies, each of which contains the ranking of a set of important genes. Another scenario may be dealing with multiple independent studies of the same kind. For example, multiple studies may be carried out to identify genes that are differentially expressed between disease cases and normal controls. Some of the studies may be based on the microarray technologies (cDNA or Affymetrix gene chips), whereas more recent studies may be next-generation-sequencing technology based. Even though all the studies aim at finding genes that are differentially expressed, and as such containing common information, the results are likely to vary from study to study. Synthesizing information from all the studies would increase the power to identify true biomarkers while keeping the false positive rate low (Xu et al., 2005; Fishel et al., 2007).

For integrating data from multiple platforms, the underlying spaces from which the top ranked elements come from are likely to be different. Thus, taking the underlying spaces into consideration explicitly is essential, as failure to do so would lead to inefficient use of data and might render biases and/or sub-optimal results; see Lin (2010) for further comments on the effects of the underlying spaces. However, this important aspect is usually overlooked in the literature on rank-based integration methods for omic-scale data. Nevertheless, although usually no assumptions about the underlying spaces are explicitly stated, careful dissections of such algorithms reveal implicit assumptions about the spaces regardless of whether such assumptions are valid for a particular integration problem.

For instance, in the "transitivity example" of DeConde et al. (2006), a set of three genes, $\{a, b, z\}$, were considered, where the ordering of the elements within the curly brackets $\{\ \}$ is arbitrary. Suppose we have a series of comparisons resulting in 20 pairwise orderings: nine (a, z), one (z, a), seven (b, z), and three (z, b), where the elements within the parentheses ( ) are ordered according to their rankings, from the higher to the lower. The implicit assumption in DeConde et al. (2006) is that the two elements in each comparison constitute the underlying space. In this case, the ranking of the three elements when information from all the 20 lists are aggregated is (a, b, z). However, if we treat each list as the top two elements with all three elements being compared, then the ranking of the elements would be (z, a, b) (Tables 1 and 2). These results are intuitively sensible when interpreting based on the underlying spaces from which the lists are derived. In the pairwise comparison case, element a wins 90% of the time when it is being compared, element b 70% of the time, while z only 20% of the times. On the other hand, if all three elements are always being compared with one another, then z emerges as a winner since it is ranked either first or second in all the comparisons while a and b appear in the top two in only half of the 3-way comparisons. This simple example demonstrates the important role played by the underlying spaces.

In this paper, we describe two broad classes of heuristic algorithms for integrating omics data that explicitly take the underlying spaces into consideration. One class consists of algorithms that are extensions of the Border's method to accommodate the situation of potentially different underlying spaces. The other class of algorithms are Markov chain based, which may be thought of as generalizations of the same type of algorithms devised originally for integrating internet search engine results.

2 Methods

In this section, before we describe the proposed algorithms, we first define underlying spaces and top-k lists. We also introduce an extension to Kendall's tau distance that accommodates top-k lists from different spaces, and we use it as one measure of relative performances of the methods.

2.1 Spaces and top-k lists

Given a discrete space T (e.g., genes in the human genome) that contains $|T|$ elements, we associate each one with a unique label such that T can be viewed as a list: $T = \{1, 2, \cdots, |T|\}$. A permutation of $T, \tau(T) = (t_1, t_2, \cdots, t_{|T|})$, such that $R_\tau(t_i) < R_\tau(t_j)$ for any $i < j$, is a complete ranking of the elements in T. We refer to $\tau(T)$ as a full ranked list, and we further refer to $R_\tau(t)$ as the rank of element t under the ranking mechanism (or function; in this case it is a permutation) τ to distinguish it from the rank of t under a different ranking mechanism. As denoted earlier, we use curly brackets for a set of elements whose ordering is arbitrary, whereas parentheses are used for an ordered list ranked from the highest to the lowest.

In many situations, including the biological problems being addressed in this paper, a full ranked list is typically not desirable (nor available). Instead, one is interested in only a partial ranked list $S \subset T$ under ranking mechanism τ. Without loss of generality, we assume that the partial list $S = (s_1, s_2, \cdots, s_{|S|})$ is ordered according to their ranks such that $R_\tau(s_i) < R_\tau(s_j)$ for $i < j$. In particular, the situation that is of particular interest is when S is composed of the top $k = |S|$ elements in T. We refer to such a ranked list as the top-k list. It is implicitly assumed that all the elements that are in T but not in S are ranked lower than k.

In the context of the problem being addressed here, we have L top-k lists, S_l, with $|S_l| = k_l, l = 1, \cdots, L$. The lengths of the lists may not necessarily be the same, and further they may not come from the same underlying space. Our objective is then to aggregate these L lists to arrive at a new ranking of the elements in $S = \cup_{l=1}^{L} S_l$, or a top-k list of S, that synthesizes information contained in all the individual lists, regardless of whether the underlying spaces are the same or not.

2.2 Kendall's tau distance and criterion

For two full ranked lists defined on a common space, Kendall's tau distance is essentially counting the number of pairwise discordance between the two lists (Kendall, 1970). To deal with the situation of top-k lists and different underlying spaces, the Kendall's tau distance is modified as follows. Let S_1 be a top-k list of length $|S_1| = k_1$ with respect to space T_1 and S_2 a top-k list of length $|S_2| = k_2$ with respect to space T_2. Let τ_1 and τ_2 be the underlying associated ranking mechanisms over the spaces T_1 and T_2, respectively. Let $S = S_1 \cup S_2$. For each $u \in S$, if $u \in S_l$, then the rank $R_{\tau_l}(u)$ is as defined in the original top-k lists; if $u \in T_l \cap S_l^c$, then define $R_{\tau_l}(u) = k_l + 1$, otherwise the rank is left undefined, that is, $R_{\tau_l}(u) = \text{NA}$, $l = 1, 2$. For each pair of elements $u, v \in S$, let D denote the collection of pairs that are both in the spaces T_1 and T_2. We further let B be the collection of pairs of elements with the following property: both elements of each pair belong to one of the two lists but neither belongs to the other list. Let

$$d_k(u,v) = \begin{cases} I\{[R_{\tau_1}(u) - R_{\tau_1}(v)][R_{\tau_2}(u) - R_{\tau_2}(v)] < 0\} & \text{if } (u,v) \in D \cap B^c \\ p & \text{otherwise ,} \end{cases} \tag{1}$$

where $I\{\cdot\}$ is the usual indicator function that takes the value of 1 or 0 depending on whether the condition within the curly brackets is satisfied or not. Further, p is a parameter between 0 and 1. Since the distance as defined in (1) is equivalent to that given in Fagin et al. (2004) when $T_1 = T_2$, and thus can be regarded as a generalization of Fagin's modified Kendall's tau distance, we set $p = 1/2$ for a "neutral approach" following Fagin. Then the modified Kendall's tau distance is defined as

$$K(S_1, S_2) = \frac{1}{2} \sum_{u,v \in S} d_K(u,v).$$

For aggregating top-k lists, the distances to be computed are those between the candidate aggregate list A and each of the input list S_l. Suppose the lengths of the input lists, $k_l, l = 1, \cdots, L$, are not all the same, then the scaled Kendall's distance would be more appropriate. For example, $k_l(k_l - 1)/2$, the largest Kendall's distance between two full lists of length k_l, would be a reasonable choice as a scaling factor. This leads to the following evaluation criterion for measuring the relative performances of aggregation algorithms

$$\text{Kendall's criterion } = \sum_{l=1}^{L} w_l K(A, S_l),$$

where w_l is the weight for list l that can accommodate the scaling factor as discussed above or any prior information on the relative importance of the list. An aggregate list with a smaller value of Kendall's criterion is deemed to be superior to one with a larger value when comparing results from different aggregation algorithms with the same assumption about the underlying spaces.

2.3 Borda's methods

The collection of Borda-inspired methods are intuitive and easy to understand. In the original method proposed by Jean-Charles de Borda (Borda, 1781), aggregate ranks were computed based on arithmetic average for full ranked lists. Many other aggregation functions and modifications have been proposed and used, and are applicable to top-k lists (Dwork et al., 2001). We first describe Borda's method for full ranked lists. Then necessary modifications are made to accommodate top-k lists with potentially different underlying spaces.

The algorithms.

Given full ordered lists $\tau_1, \cdots, \tau_L$, each a permutation of the underlying space T, we let $R_{\tau_l}(u)$ be the rank of element $u \in T$ in list τ_l. We let $B_l(u)$ denote the Borda's score in general, with $B_l(u) = R_{\tau_l}(u)$ being a special case. Let $Bu = f(B_1(u), B_2(u), \cdots, B_L(u))$ be an aggregate function of the Borda scores. Then one sorts the Bu's to obtain an aggregate ranked list $\tau(T)$. Frequently suggested aggregation functions include

$$\begin{array}{lll} f(x_1, \cdots, x_L) = \text{median}\{|x_1|, \cdots, |x_L|\} & & \text{(median)} \\ f(x_1, \cdots, x_L) = \left(\prod_{l=1}^{L} |x_l|\right)^{1/L} & & \text{(geometric mean)} \\ f(x_1, \cdots, x_L) = \sum_{l=1}^{L} |x_l|^p / L & & (p-\text{norm}). \end{array}$$

Note that the method proposed by Borda (Borda, 1781) is a special case of $p-$ norm when $p = 1$ (arithmetic mean) and $B_l(u) = R_{\tau_l}(u)$, apart from a scaling factor. Although the most frequently used Borda score is rank, in situations where other information, beyond the rankings, is available, Borda's score may be defined accordingly to take into account of such additional information.

Now let $S_1, \cdots, S_n$ be n top-k lists from potentially different underlying spaces $T_1, \cdots, T_n$ with $\tau_1, \cdots, \tau_n$ as the associated ranking mechanisms. Let $S = \cup_{i=1}^{n} S_l$. We define ranks for all elements in S under each ranking mechanism as in 2.2, and we refer to them as expanded top-k lists. Since Borda's method is based on a function of the 'positions' across all lists, it appears to be a reasonable choice to ignore any NA's in the computation of the aggregation function.

Transitivity example.

The transitivity example is simple but useful to demonstrate the algorithms and the effects of the assumptions about the underlying spaces on the results.

Table 1: Expanded ranked lists and results from three Borda algorithms[a]

	Type (# lists)				Border					
	T1	T2	T3	T4	ARM		MED		GEO	
	(9)	(1)	(7)	(3)	SC	RK	SC	RK	SC	RK
A. Com Space										
a	1	2	3	3	2.05	2	2.5	2	11.68	2
b	3	3	1	2	2.15	3	2.5	2	13.07	3
z	2	1	2	1	1.80	1	2.0	1	11.09	1
B. Diff Space										
a	1	2	-	-	1.1	1	1.0	1	0.07	1
b	-	-	1	2	1.3	2	1.0	1	0.21	2
z	2	1	2	1	1.8	3	2.0	3	0.55	3

[a] ARM: arithmetic mean; MED: median; GEO: geometric mean;
SC: score; RK: rank

When the individual outcomes are considered as top-2 lists from the common underlying space $T = \{a, b, z\}$, the element not being included in each list was assigned a rank of 3 in the expanded top-k lists (Table 1A). All three Borda's algorithms selected z as the top ranked element. Further, both the arithmetic mean (ARM) and the geometric mean (GEO) aggregation functions selected a as the second ranked, although, not surprisingly, the median (MED) aggregation function failed to rank a and b due to the information loss with a robust aggregate function. On the other hand, when all lists are treated as simply the results of pairwise comparisons as in DeConde et al. (2006), the underlying spaces become different, and the expanded ranked lists contain NA's (the -'s in Table 1B). Both ARM and GEO give results as expected, whereas MED once again failed to separate a and b .

2.4 Markov chain methods

Markov chain methods provide a more elegant but less intuitive alternative to Borda's methods. Using the same notation as in 2.3, we treat $S = \cup_{l=1}^{L} S_l$ as the state space of the Markov chain. The expanded rankings for elements in S are as defined in 2.2 under each ranking mechanism. The idea is then to construct the transition matrix of an ergodic Markov chain such that its stationary distribution will assign a larger probability to a state that is ranked higher. Thus, the stationary distribution will determine the aggregate rankings of the items. Information on pairwise rankings will be used to construct the transition probabilities. A number of ways of assigning such probabilities are possible depending on one's objectives (Dwork et al., 2001; DeConde et al., 2006). We discuss three of them below, which may be viewed as generalization of algorithms in Dwork et al. (2001) and DeConde et al. (2006) in that the underlying spaces are being taken into account explicitly. That is, our algorithms as presented here can accommodate full ranked lists as well as top-k lists with either a common space or different underlying spaces.

MC1.

For each $u \in S$, let

$$P(u \rightarrow v) = \begin{cases} 1/|S| & \text{if } R_{\tau_l}(u) > R_{\tau_l}(v) \text{ for at least one of the input lists,} \\ 0.5/|S| & \text{if } R_{\tau_i}(u) = \text{NA or } R_{\tau_j}(v) = \text{NA for all the lists,} \\ 0 & \text{otherwise,} \end{cases}$$

for each v $(\neq u) \in S$. Then define $P(u \rightarrow u) = 1 - \sum_{v \neq u} P(u \rightarrow v)$. The general idea in this construction of the transition matrix is that the chain will either move to a state with better ranking in at least one of the lists or stay at the same state.

MC2.

Whereas MC1 may move to any state with a higher ranking in at least one of the input lists with equal probability, MC2 is a majority rule algorithm. Specifically, for each $u \in S$, let

$$P(u \rightarrow v) = \begin{cases} 1/|S| & \text{if } R_{\tau_l}(u) > R_{\tau_l}(v) \text{ for a majority of the input lists} \\ & \text{for which both}R_{\tau_i}(u) \neq \text{NA and } R_{\tau_i}(v) \neq \text{NA,} \\ 0.5/|S| & \text{if } R_{\tau_i}(u) = \text{NA or } R_{\tau_j}(v) = \text{NA for all the lists,} \\ 0 & \text{otherwise,} \end{cases}$$

for each v $(\neq u) \in S$, and define the diagonal probabilities as in MC1. The general idea in this construction of the transition matrix is that the chain will move to a state with better rankings in at least half of the input lists or when u and v being compared are never in the same space for any of the input lists.

MC3.

While the original MC2 was proposed in Dwork et al. (2001) as a spam fighting algorithm due to its majority-rule nature, MC3 may be more appropriate for multi-platform omics problems given the unique feature of each data

Table 2: Expanded ranked lists and results from Markov chain algorithms[a]

	Type (# lists)				Border					
	T1	T2	T3	T4	MC1		MC2		MC3	
	(9)	(1)	(7)	(3)	PR	RK	PR	RK	PR	RK
A. Com Space										
a	1	2	3	3	0.33	1	0.05	2	0.31	2
b	3	3	1	2	0.33	1	0.05	2	0.26	3
z	2	1	2	1	0.33	1	0.91	1	0.43	1
B. Diff Space										
a	1	2	-	-	0.33	1	0.49	1	0.49	1
b	-	-	1	2	0.33	1	0.49	1	0.40	2
z	2	1	2	1	0.33	1	0.02	3	0.11	3

[a] PR: stationary probability; RK: rank

type. Consider each $u \in S$. For each v $(\neq u) \in S$, let $A(v) = \{l | R_{\tau_l}(u) \neq \text{NA and } R_{\tau_l}(v) \neq \text{NA}\}$. Then

$$P(u \to v) = \begin{cases} \sum_{l \in A(v)} I(R_{\tau_l}(u) > R_{\tau_l}(v))/|A(v)||S| & \text{if } |A(v)| \neq 0, \\ 0.5/|S| & \text{if } |A(v)| = 0, \end{cases}$$

where I is the usual indicator function as defined before and $|\cdot|$ is the cardinality of the set. The general idea in this construction of the transition matrix is that the chain will move to a state with probability proportional to the number of lists that rank the new state higher than the current one.

Technical note.

In the MC procedures, if needed as suggested by DeConde et al. (2006), the transition probability may be modified as follows to ensure the existence of a unique stationary distribution:

$$P'(u \to v) = (1 - \delta)P(u \to v) + \delta/|S|,$$

where δ is a tuning parameter and is usually set to be small, say 0.05, though our exploration has indicated that the results are rather insensitive to the choice of δ between 0 and 1/2.

Transitivity example.

The transitivity example is simple enough to illustrate and compare the transition matrices constructed from the different MC algorithms and different assumptions about the underlying spaces. Under the assumption of a common underlying space, the transition matrices from the three MC algorithms are given in Figure 1(A). We can see that both MC1 and MC3 have all non-zero off-diagonal probabilities, albeit with potentially smaller probabilities for MC3 due to its use of proportionality. Thus both corresponding Markov chains are ergodic, that is, any state in $S = \{a, b, z\}$ can be reached from any other state aperiodically. On the other hand, because of the more stringent requirement for MC2, there are zero entries in its transition matrix. For example, a is preferred over z in only 9 out of 20 triple comparisons (not a majority) and thus $P(z \to a) = 0$. As a consequence, the corresponding Markov chain is no longer ergodic. As can be seen from Figure 1(B), state z is an absorbent state (the Markov chain will stay at state z as soon as it reaches there). Nevertheless, as discussed in the technical note, the transition matrix can be easily modified to make it ergodic (right matrix of second row in Figure 1(A)). Table 2A shows the stationary distributions and the corresponding rankings. As can be seen from the table, MC3 yielded the same results as ARM and GEO, whereas MC2 ranked z ahead of the other two elements, but MC1 (being the most simple algorithm) failed to distinguish the rankings because none of the three elements completely dominates the other two.

On the other hand, under the assumption of pairwise comparisons (that is, the underlying space for each list consists of only the two elements being compared), the results are different, as expected (Table 2B). MC3 is able to rank all three items in S, and the result coincides with those from ARM and GEO. However, MC2 once again could not rank between a and b, while MC1 failed to distinguish all three. Also, as before, MC1 and MC3 are ergodic while MC2 is not. The transition matrices and the graphical representation of MC2 showing its non-ergodic nature are given in Figure 2.

3 Applications

Two examples are presented here to illustrate the algorithms with varying assumptions about the underlying spaces. The first is on a dataset consisting of three top-k lists of different lengths, which is contrive but illuminating for interpreting the results. The second is an application to aggregating results from five gene expression studies. These studies used different microarray platforms and thus the assumption about the underlying spaces is a key issue.

(A)

$$\mathrm{MC1} = \begin{bmatrix} 0.33 & 0.33 & 0.33 \\ 0.33 & 0.33 & 0.33 \\ 0.33 & 0.33 & 0.33 \end{bmatrix}$$

$$\mathrm{MC2} = \begin{bmatrix} 0.33 & 0.33 & 0.33 \\ 0.33 & 0.33 & 0.33 \\ 0.00 & 0.00 & 1.00 \end{bmatrix} \overset{\delta=0.05}{\Longrightarrow} \begin{bmatrix} 0.33 & 0.33 & 0.33 \\ 0.33 & 0.33 & 0.33 \\ 0.02 & 0.02 & 0.98 \end{bmatrix}$$

$$\mathrm{MC3} = \begin{bmatrix} 0.65 & 0.16 & 0.18 \\ 0.16 & 0.67 & 0.21 \\ 0.15 & 0.12 & 0.73 \end{bmatrix}$$

(B)

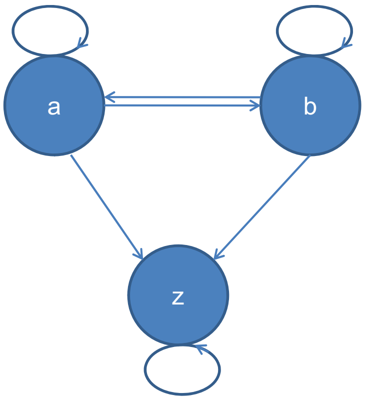


Figure 1: Dissection of the Markov chain algorithms under the common space assumption. (A). Transition matrices built from the MC algorithms and the ergodic counter part of MC2 with tuning parameter $\delta = 0.05$. (B). Graphical representation of the transition matrix built from MC2.

3.1 Top-k lists of variable lengths

Consider the problem of obtaining an integrated top-k list from three individual ranked lists, L1, L2, and L3, of lengths 30, 25, and 20 (Table 3) that come from three spaces (platforms) with 6000, 5000, and 4000 elements, respectively, but all contain elements 1-40. As can be seen from the table, there are 40 items (elements 1-40) in the aggregate candidate set S. Elements 1-

(A)

$$\text{MC1} = \begin{bmatrix} 0.50 & 0.17 & 0.33 \\ 0.17 & 0.50 & 0.33 \\ 0.33 & 0.33 & 0.33 \end{bmatrix}$$

$$\text{MC2} = \begin{bmatrix} 0.83 & 0.17 & 0.00 \\ 0.17 & 0.83 & 0.00 \\ 0.33 & 0.33 & 0.33 \end{bmatrix} \overset{\delta=0.05}{\Longrightarrow} \begin{bmatrix} 0.81 & 0.18 & 0.02 \\ 0.18 & 0.81 & 0.02 \\ 0.33 & 0.33 & 0.33 \end{bmatrix}$$

$$\text{MC3} = \begin{bmatrix} 0.80 & 0.17 & 0.03 \\ 0.17 & 0.73 & 0.10 \\ 0.30 & 0.23 & 0.47 \end{bmatrix}$$

(B)

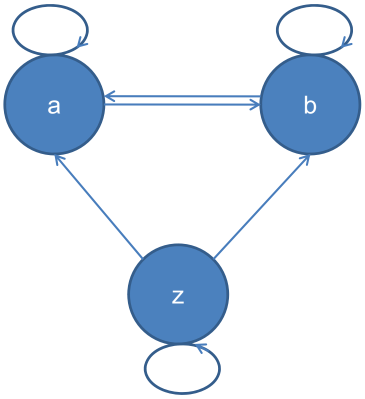


Figure 2: Dissection of the Markov chain algorithms under the different spaces assumption. (A). Transition matrices built from the MC algorithms and the ergodic counter part of MC2 with tuning parameter $\delta = 0.05$. (B). Graphical representation of the transition matrix built from MC2.

10 appear as top 1-10, in that order, in all three lists. There are five other elements (11-15) that also appear in all three lists, but with different rankings. In addition, five of the elements (16-20) are contained in two of the top-k lists, whereas the remaining 20 elements (21-40) only appear in one of the three top-k lists. We applied both the Borda's and the Markov chain methods to analyze the data under three different assumptions about the underlying spaces: *common-space, platform-dependent,* and *top-k-space.* Under *common-space*,

we assumed that the three top-k lists come from a single common space (i.e., the studies were done using the same platform). In the *platform-dependent* assumption, we used the known spaces from which the top-k lists were generated in the aggregation. For aggregation under the *top-k-space* assumption, we treated each top-k list as its own space. Note that in real data application, information on the underlying spaces can usually be obtained for known platforms. In this example, results obtained under the *common-space* and the *platform-dependent* assumptions turn out to be the same since all elements in the candidate set S are contained in all three platforms. We also note that the Markov chain algorithms described in DeConde et al (2006) make use of the *top-k-space* assumption implicitly.

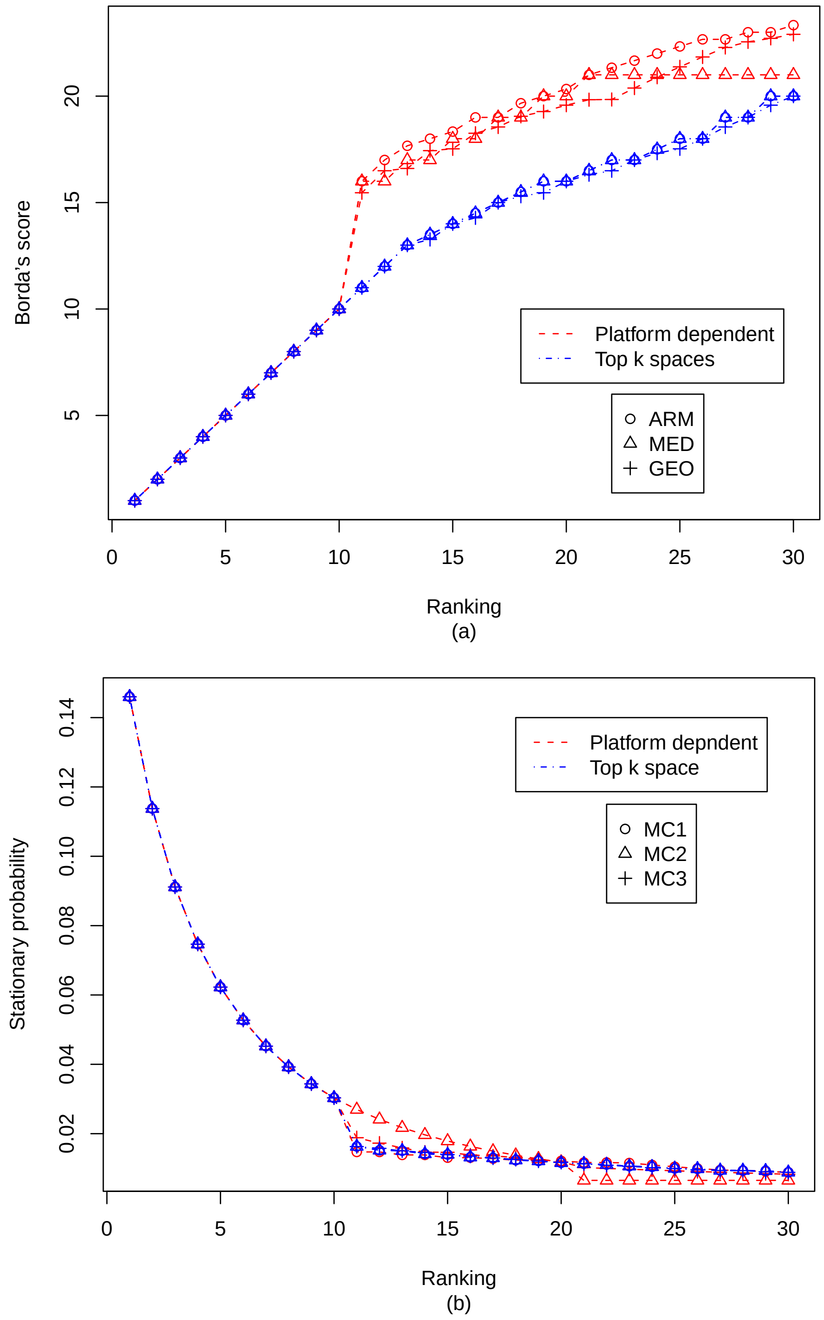
As we can see from Table 3, regardless of the assumption about the underlying spaces, elements 1-10 were selected as the top 10, in that order, in the aggregate lists by all of the algorithms, as one would expect. On the other hand, the results for ordering the remaining elements differ depending on the space assumption and algorithm, but the discrepancies are much more profound (and fundamental) under different space assumptions than with different algorithms. In fact, the Kendall's criterion gave similar values for the results under the same space assumptions but vastly different values across the various space assumptions, where the variable lengths in the lists are accounted for in w_l in computing Kendall's criterion. Specifically, the maximum relative difference is 14% across all algorithms within the same space assumption versus 115% across various space assumptions.

Under the *platform-dependent* assumption, elements 11-20 (which appear in at least two of the top-k lists) were ranked in the top 11-20 (not necessarily in that order) by most of the algorithms. In contrast, under the *top-k-space* assumption, elements 31-35 were selected to be among the top 11-20, regardless of the algorithm used. Although seemingly counter intuitive (since element 31-35 only appear in one of the top-k lists), these results are easily explained casting in the context of the underlying spaces: in the only study that included 31-35 (L2), as a group, they were ranked at least as high as any group of five other elements in all of the studies, apart from the top 10 elements. On the other hand, the implicit information that 31-35 were ranked lower than 30 (for L1) and 20 (for L3) was fully utilized under the *platform-dependent* assumption, diminishing their chance of making it to the top 11-20. The relative rankings of elements 11-15 and 16-20 is another example that exemplifies the effect of assumption about the underlying spaces. Under the *top-k-space* assumption, there were only two studies that included elements 16-20, and they were ranked at 11-15 (L3) and 16-20 (L1). These rankings match the rankings of elements 11-15 in two of the three studies (L1 and L3), but elements

Table 3: Three variable length top-k lists and aggregate results[a]

				Borda						Markov Chain					
				Platform Depedent			Top-k Space			Platform Dependent			Top-k Space		
Rank	L1	L2	L3	ARM	MED	GEO	ARM	MED	GEO	MC1	MC2	MC3	MC1	MC2	MC3
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5
6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6
7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7
8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8
9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9
10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10
11	11	31	16	11	11	11	31	31	31	16	11	11	31	31	31
12	12	32	17	12	16	12	32	32	32	11	12	12	16	16	16
13	13	33	18	16	12	16	33	33	33	17	13	13	32	32	32
14	14	34	19	13	17	17	16	16	16	12	14	14	17	17	17
15	15	35	20	17	13	13	34	34	34	18	15	16	33	33	33
16	16	36	11	14	18	18	17	17	17	13	16	17	18	18	18
17	17	37	12	18	14	14	35	35	35	31	17	15	34	34	34
18	18	38	13	19	19	19	18	18	18	19	18	18	19	19	19
19	19	39	14	15	15	31	11	11	11	14	19	19	35	35	35
20	20	40	15	20	20	15	36	36	36	32	20	20	20	20	20
21	21	11		31	21	20	19	19	19	20	40	31	36	36	36
22	22	12		32	31	32	12	12	12	15	39	32	11	11	11
23	23	13		33	32	33	37	37	37	33	38	33	37	37	37
24	24	14		34	33	34	20	20	20	34	37	34	12	12	12
25	25	15		35	34	35	13	13	13	35	36	35	38	38	38
26	26			21	35	36	38	38	38	36	35	36	13	13	13
27	27			36	36	37	14	14	14	37	34	37	21	21	21
28	28			22	37	21	39	39	39	21	33	38	39	39	39
29	29			37	38	38	15	15	15	38	32	39	14	14	14
30	30			23	39	22	40	40	40	22	31	40	22	22	22

[a] Results under *common-space* assumption are the same as those under *platform-dependent* assumption and are not shown separately.



(a)

(b)

Figure 3: Measures of informativeness for the ranked elements: (a) based on Borda's score; (b) based on Markov chain stationary probability.

11-15 were also included in the second study and ranked lower, at 21-25 (L2). As such, elements 16-20, as a group, were ranked higher than elements 11-15 under the *top-k-space* assumption, especially in the Markov chain algorithms. In contrast, under the *platform-dependent* assumption, the information that elements 16-20 were ranked lower than elements 11-15 (L2) was utilized, which led to higher rankings for elements 11-15, as a group, especially for MC2 and MC3.

The relative rankings of the elements ranked below 20 were much more mixed (rankings below 30 were not shown in Table 3). However, such discrepancies are unimportant as it is generally known that the relative rankings of the lowly ranked elements are unreliable due to information degradation (Hall and Schimek, 2009; Lin and Ding, 2009). As such, the rankings of the bottom half of the elements should be taken as a grain of salt. By examining the Borda scores or the stationary probabilities of the Markov chain of the top 30 elements (Figure 3), one can see that the changes (either relative or absolute) in the scores or the probabilities are smaller for the lowly ranked items, and there may be a large gap in the scores/probabilities when the information for ranking starts to diminish. Indeed, as can be seen in Figure 3(a), under the *platform-dependent* assumption, there is a jump in the Borda score in the ranking from 10-11, and the changes in the scores also slow down, signaling the degradation of information for the relative ranking after this point. The stationary probabilities plotted against the ranks for the Markov chain algorithms, shown in Figure 3(b), also contains useful information regarding the relative rankings of elements. Other than MC2 under the *platform-dependent* assumption, there appears to be limited information for ranking the elements beyond the top 10.

3.2 Prostate cancer gene expression studies

We applied the Borda and Markov chain algorithms to aggregate results from five microarray studies that aimed at finding genes that are differentially expressed between prostate cancers and normal prostate tissues (Dhanasekaran et al., 2001; Luo et al., 2001; Singh et al., 2002; True et al., 2006; Welsh et al., 2001). In particular, the data being employed in this example are the five lists of top-25 up-regulated genes used by DeConde et al. (2006) and Lin and Ding (2009), which are reproduced in columns 2-6 of Table 4. Given it is fairly reasonable to assume that the three cDNA studies (Dhanasekaran (Dhana), Luo, and True) have a common underlying space while the two Affymetrix studies (Singh and Welsh) have their own common space yet the two common spaces are different (although with a fair amount of overlaps), we aggregated the results from all five studies under such assumption (*platform dependent*). The results show that the aggregate list from MC3 has the smallest value of the Kendall's criterion (Table 4), although the differences are small as in the previous example. Despite the large differences in the ranked lists among the different algorithms, the top group of ranked genes are quite similar. Moreover, the top four elements from each of the two separate integration studies (integrating the three cDNA studies and integrating the two Affymetrix studies; results not shown) are all contained in the top seven genes of the overall aggregation from MC3, another indication that it is likely that reasonable results have been obtained. An examination of the Borda scores and the stationary probabilities show that there is little information for ranking genes beyond the top 10 (results not shown). As such, the discrepancies among the different algorithms are not surprising, but caution needs to be exercised if there is any desire to interpret the rankings of the genes ranked lower than 10.

To further examine the effect of space assumption on the results for real omic studies, we used the MC3 algorithm to reanalyze the data under two other assumptions, in addition to *platform-dependent*, about the underlying spaces: all five studies coming from the same space (*common-space* assumption) or all five studies coming from their own different spaces (*top-k-space* assumption). The stationary probabilities of MC3 versus the ranks were plotted in Figure 4 for each of the three space assumptions. As can be seen in Figure 4(c), because a great deal of information on the share common platforms were not utilized, the information degraded quickly beyond ranking the top two genes (HPN, AMACR, which are the same as the top two genes under the other two space assumptions). On the other hand, as shown in Figure 4(a) and 4(b), assuming all the underlying spaces are the same has much smaller effect on the ranking information, in this example, presumably because of the fair amount of overlaps between the cDNA and the Affymetrix platforms. Table 5 shows the results for only the top eight genes given that there is little information on the relative rankings beyond the top eight for any of the three analyses. It is reassuring to see that the top seven genes are identical under the *common-space* and *platform-dependent* scenarios. On the other hand, other than the top two genes, there are only two more genes (FASN and GDF15) in the top eight that are in common with the other two lists. This, together with the information in Figure 4, indicates that the *common-space* and *platform-dependent* assumptions are much similar than with the *top-k-space* assumption, and once again demonstrates the potential large effect of the underlying assumption about the spaces. Nevertheless, it is worth noting that HPN, AMACR, and FASN have been suggested to be involved in prostate cancer development and progression (Klezovitch et al., 2004), and they are contained in all there top-8 lists.

Table 4: Top-25 lists from five prostate cancer studies and aggregate results

	cDNA			Affymetrix		Borda			Markov Chain		
Rank	Dhana	Luo	True	Singh	Welsh	ARM	MED	GEO	MC1	MC2	MC3
1	OGT	HPN	AMACR	HPN	HPN	HPN	HPN	HPN	HPN	HPN	HPN
2	AMACR	AMACR	HPN	SLC25A6	AMACR	AMACR	AMACR	AMACR	AMACR	AMACR	AMACR
3	FASN	CYP1B1	NME2	EEF2	OACT2	CANX	FASN	OGT	GDF15	GDF15	GDF15
4	HPN	ATF5	CBX3	SAT	GDF15	GRP58	KRT18	FASN	FASN	EEF2	NME1
5	UAP1	BRCA1	GDF15	NME2	FASN	GDF15	CANX	SAT	NME1	NME1	FASN
6	GUCY1A3	LGALS3	MTHFD2	LDHA	ANK3	FASN	GRP58	GDF15	EEF2	FASN	EEF2
7	OACT2	MYC	MRPL3	CANX	KRT18	NME1	GDF15	CANX	CANX	KRT18	KRT18
8	SLC19A1	PCDHGC3	SLC25A6	NACA	UAP1	SAT	NME1	GRP58	GRP58	FMO5	CANX
9	KRT18	WT1	NME1	FASN	GRP58	KRT18	EEF2	NME2	KRT18	CANX	GRP58
10	EEF2	TFF3	COX6C	SND1	PPIB	EEF2	SAT	SLC25A6	OGT	GRP58	UAP1
11	STRA13	MARCKS	JTV1	KRT18	KRT7	ANK3	ANK3	EEF2	NME2	UAP1	NME2
12	ALCAM	OS-9	CCNG2	RPL15	NME1	LDHA	LDHA	ANK3	SLC25A6	OACT2	SLC25A6
13	GDF15	CCND2	AP3S1	TNFSF10	STRA13	TMEM4	TMEM4	LDHA	CYP1B1	SAT	OGT
14	NME1	NME1	EEF2	SERP1	DAPK1	NACA	NACA	CYP1B1	UAP1	TMEM4	OACT2
15	CALR	DYRK1A	RAN	GRP58	TMEM4	NME2	KRT7	OACT2	ATF5	LDHA	FMO5
16	SND1	TRAP1	PRKACA	ALCAM	CANX	OACT2	RPL15	KRT18	CBX3	SLC25A6	TMEM4
17	STAT6	FMO5	RAD23B	GDF15	TRA1	SLC25A6	TNFSF10	ATF5	SAT	ANK3	SAT
18	TCEB3	ZHX2	PSAP	TMEM4	PRSS8	OGT	DAPK1	CBX3	OACT2	NME2	CYP1B1
19	EIF4A1	RPL36AL	CCT2	CCT2	ENTPD6	UAP1	SERP1	NME1	BRCA1	NACA	ATF5
20	LMAN1	ITPR3	G3BP	SLC39A6	PPP1CA	CYP1B1	TRA1	NACA	TMEM4	KRT7	LDHA
21	MAOA	GCSH	EPRS	RPL5	ACADSB	KRT7	PRSS8	UAP1	LGALS3	RPL15	CBX3
22	ATP6V0B	DDB2	CKAP1	RPS13	PTPLB	ATF5	ENTPD6	BRCA1	ANK3	PPIB	BRCA1
23	PPIB	TFCP2	LIG3	MTHFD2	TMEM23	CBX3	PPP1CA	LGALS3	LDHA	TNFSF10	LGALS3
24	FMO5	TRAM1	SNX4	G3BP2	MRPL3	BRCA1	SLC39A6	GUCY1A3	MYC	SERP1	ANK3
25	SLC7A5	YTHDF3	NSMAF	UAP1	SLC19A1	RPL15	ACADSB	TMEM4	GUCY1A3	GUCY1A3	GUCY1A3
Kendall's Criterion						30.43	30.72	30.40	30.38	30.46	30.34

Table 5: Top-8 aggregate ranks from five prostate cancer studies using the MC3 algorithm with three different assumptions of the underlying spaces[a]

Rank	Common Space	Platform Dependent	Different Spaces
1	HPN	HPN	HPN
2	AMACR	AMACR	AMACR
3	GDF15	GDF15	NME2
4	NME1	NME1	FASN
5	FASN	FASN	SLC25A6
6	EEF2	EEF2	OGT
7	KRT18	KRT18	OACT2
8	UAP1	CANX	GDF15

[a] Common Space: all five studies share a common space; Platform Dependent: studies from same platform share same space; Different Spaces: all studies have different spaces.

4 Discussion

In this paper, we propose two classes of space oriented heuristic algorithms for integrating ranked lists from various sources. Although this work is motivated by aggregating results from omic-scale data, the algorithms are applicable to aggregating results from other settings as long as the underlying spaces are known or can be found fairly accurately. These algorithms are either Borda inspired or Markov chain based that take the underlying spaces from which the individual ranked lists come from into account explicitly. We believe that this is an important development and a significant departure from existing methods as omic-scale data from different platforms clearly have different underlying sets of elements being considered but there is usually substantial overlapping to warrant the application of aggregation methods. For example, 30-50% of the genes on various Affymetrix platforms are also present in cDNA gene chips. Thus any algorithm that strives to make full use of information available and to avoid biases in the results should be space dependent. As we show through the examples, results from aggregation algorithms can be rather sensitive to the assumption about the underlying spaces, and as such, any outcome without considering the underlying spaces from which the ranked lists are derived should not be accepted without careful scrutiny. Unfortunately, existing methods rarely take spaces into account explicitly, and so we hope this work would inspire consideration in this direction in future research in this area.

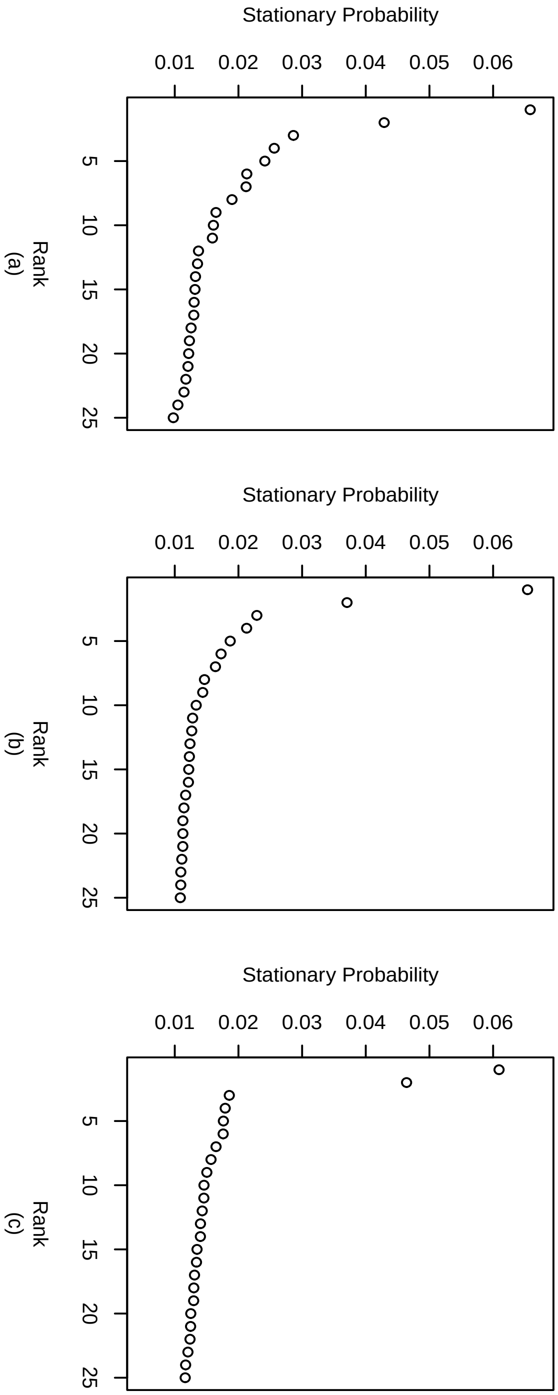


Figure 4: Measures of informativeness for the ranked elements: (a) based on Borda's score; (b) based on Markov chain stationary probability.

Unless information indicates otherwise, the *platform-dependent* assumption is usually the one that should be used to fully and correctly utilize the information provided by the underlying spaces. The three different assumptions used in the two examples in the Application section are merely for the purpose of demonstrating the procedures and showing the effects of assumptions about the underlying spaces. In an omic study, the elements (genes) studied under each platform are usually available and can be retrieved for use as demonstrated.

The algorithm as presented in this paper will return a full ranked list or a top-k list of all the elements in the set composed of the input lists. However, as discussed in Hall and Schimek (2009) and Lin and Ding (2009), information about relative rankings of element degrades quickly such that the rankings of elements not on the tops are not reliable. While Hall and Schimek (2009) provides an elegant and formal treatment of this subject, we show in this paper that an informal procedure through monitoring the scores of the Borda algorithms and the stationary probabilities of the Markov chain algorithms may offer a quick and effective alternative. More specifically, we may ascertain where information starts to degrade by examining the scores/probabilities to detect the “change point” at which the (relative) change of the scores/probabilities become much smaller. As we can see from the examples, elements may form “equivalent classes” (similar scores/probabilities) after a certain point, and it is virtually impossible to decipher the relative rankings of the elements within the same equivalent class.

As we have pointed out earlier, the two classes of algorithms presented in this paper are heuristic in nature, as they are neither distribution based nor intending to optimize any criterion. However, they may still be evaluated for their relative performances using an objective criterion such as the Kendall’s criterion that we used in this paper. From this perspective, the algorithm MC3 appears to slightly outperform the others, consistent with the findings in DeConde et al. (2006) where underlying spaces were not taken into consideration explicitly, although the difference among the algorithms appear to be small and unsubstantial in our examples. Nevertheless, from a methodological standpoint, it may be of interest to explore optimization based algorithms such as the cross entropy Monte Carlo method proposed by Lin and Ding (2009) to gauge their performances when they are extended to be space oriented. However, although we expect such extensions to be feasible since we have already extended the Kendall’s tau distance for two top-k lists from potentially different spaces, the computational cost is much greater, and may not be cost effective from a practical standpoint for the following reason. From the results in all three examples, it appears that the assumption about the underlying

DOI: 10.2202/1544-6115.1534

spaces has much greater effect than the specific algorithm. As such, the choice of a particular algorithm is much less critical than taking into account the information contained in the underlying spaces. If a single algorithm for analyzing the data is desired, then MC3 would be recommendable for analyzing omic data. However, since all algorithms discussed here are computationally very efficient, it would not be unreasonable to run all the algorithms. The elements that are labeled as the top ones in multiple algorithms would increase one's confidence in their relevance in the particular study.

5 Software information

R codes that implement the algorithms are available at `http://www.stat.osu.edu/~statgen/SOFTWARE/TopKCEMC/`

REFERENCES

Borda, JC. Memoire sur les elections au scrutin. Histoire de l'Academie des Sciences, 1781.

Conlon, EM., Song, JJ., Liu, A. Bayesian meta-analysis models for microarray data: a comparative study. BMC Bioinformatics 2007; 8:80

DeConde, RP., Hawley, S., Falcon, S., Clegg, N., Knudsen, B., and Etzioni, R. Combining results of microarray experiments: a rank aggregation approach. Statistical Applications in Genetics and Molecular Biology. 2006; 5:15.

Dhanasekaran, SM., Barrette, TR., Ghosh, D., Shah, R., Varambally, S., Kurachi, K., Pienta, KJ., Rubin, MS. and Chinnaiyan, AM. Delineation of prog-nostic biomarkers in prostate cancer. Nature, 2001; 412:822-826.

Dwork, C., Kumar, R., Naor, M., and Sivakumar, D. Rank aggregation methods for the web. in Proceedings of the 10th International World Wide Web Conference. 2001. Pages 613-622. New York.

Fagin, R., Kumar, R. and Sivakumar, D. Comparing top k lists. SIAM Journal of Discrete Mathematics. 2003; 17:134-160.

Fishel, I., Kaufman, A. and Ruppin, E. Meta-analysis of gene expression data: a predictor-based approach. Bioinformatics. 2007; 23:1599-1606.

Hall, P. and Schimek, MG. Moderate deviation-based inference for random degeneration in paired rank lists. 2009. Submitted manuscript.

Kendall, M. Rank Correlation Methods 4th ed. 1970. Griffin, London.

Klezovitch, O., Chevillet, J., Mirosevich, J., Roberts, R., Matusik, R., and Vasioukhin, V. (2004) Hepsin promotes prostate cancer and metastasis. Cancer Cell. 2004; 6:185-195.

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

Lin, S. Rank aggregation methods. Wiley Interdisciplinary Reviews: Computational Statistics. 2010; in press.

Liu, HC., Chen, CY., Liu, YT., Chu, CB., Liang, DC., Shih, LY. Lin, CJ. Cross-generation and cross-laboratory predictions of Affymetrix microarrays by rank-based methods. Journal of Biomedical Informatics 2008; 41:570-579

Luo, J., Duggan, DJ., Chen, Y., Sauvageot, J., Ewing, M., Bittner, ML., Trent, JM. and Isaacs, WB. Human prostate cancer and benign prostatic hyperplasia: molecular dissection by gene expression profiling. Cancer Research. 2001; 61:4683-4688.

Schadt, E. E., Lamb, J., Yang, X., Zhu, J., Edwards, S., Guhathakurta, D., Sieberts, S. K., Monks, S., Reitman, M., Zhang, C., Lum, P. Y., Leonardson, A., Thieringer, R., Metzger, J. M., Yang, L., Castle, J., Zhu, H., Kash, S. F., Drake, T. A., Sachs, A., and Lusis, A. J. (2005) An integrative genomics approach to infer causal associations between gene expression and disease. *Nature Genetics*, **37**, 710-7.

Singh, D., Febbo, PG., Ross, K., Jackson, DG., Manola, J., Ladd, C., Tamayo, P., Renshaw, AA., Amico, AV., Richie, JP., Lander, ES., Loda, M., Kantoff, PW., Golub, TR. and Sellers, WR. Gene expression correlates of clinical prostate cancer behavior. Cancer Cell. 2002; 1:203-209.

Tan, AC., Naiman1, DQ., Xu, L, Winslow, RL. and Geman, D. Simple decision rules for classifying human cancers from gene expression proles. Bioinformatics. 2005; 21(20): 3896-3904.

True, L., Coleman, I., Hawley, S., Huang, A., Gifford, D., Coleman, R., Beer, T., Gelman, E., Datta, M., Mostaghel, E., Knudsen, B., Lange, P., Vessella, R., Lin, D., Hood, L. and Nelson, P. A Molecular Correlate to the Gleason Grading System for Prostate Adenocarcinoma. Proceedings of the National Academy of Sciences of the USA. 2006; 103:10991-10996.

Welsh, JB., Sapinoso, LM., Su, AI., Kern, SG., Wang-Rodriguez, J., Moskaluk, CA., Frierson, HF.Jr. and Hampton, GM. Analysis of gene expression identifies candidate markers and pharmacological targets in prostate cancer. Cancer Research. 2001; 61:5974-5978.

Xu L., Tan AC., Naiman DQ., Geman D. and Winslow RL. Robust can-cer marker genes emerge from direct integration of inter-study microarray data. Bioinformatics. 2005; 21:3905-3911.

Yuen, T., Wurmbach, E., Pfeffer, RL., Ebersole, BJ. and Sealfon, SC. Accuracy and calibration of commercial oligonucleotide and custom cDNA mi-croarrays. Nucleic Acids Research. 2002; 30:e48.