Identifying Differentially-Expressed Genes via Weighted Rank Aggregation

Qiong Fang
Dept of Computer Science and Engineering
HKUST
Hong Kong, China
fang@cse.ust.hk

Jianlin Feng
School of Software
Sun Yat-Sen University
Guangzhou, China
fengjlin@mail.sysu.edu.cn

Wilfred Ng
Dept of Computer Science and Engineering
HKUST
Hong Kong, China
wilfred@cse.ust.hk

***Abstract*—Identifying differentially-expressed genes is an important problem in gene expression analysis, since these genes, exhibiting sufficiently different expression levels under distinct experiment conditions, could be critical for tracing the progression of a disease. In a microarray study, genes are usually sorted in terms of their differentiation ability with the more differentially expressed genes being ranked higher in the list. As more microarray studies are conducted, rank aggregation becomes an important means to combine such ranked gene lists in order to discover more reliable differentially-expressed genes.**

In this paper, we study a novel weighted gene rank aggregation problem whose complexity is at least NP-hard. To tackle the problem, we develop a new Markov-chain based rank aggregation method called *Weighted MC* (WMC). The WMC algorithm makes use of the rank-based weight information to generate the transition matrix, and then outputs an aggregated order that better facilitates the effect of the weighted distance functions. Extensive experiments on the real biological datasets show that our approach is more efficient in aggregating long gene lists. Importantly, the WMC method is much more robust for identifying biologically significant genes compared with the state-of-the-art methods.

***Keywords*-differential expression; rank aggregation; Markov chain; ordering disagreement; existence disagreement;**

I. Introduction

Identifying genes that exhibit sufficiently different expression levels under distinct experiment conditions (e.g., diseased vs. normal tissues) is an important problem in gene expression analysis. Such *differentially expressed genes* are critical for studying the progression of a disease, since the difference indicates that the functionalities of the genes may have been altered at a particular disease stage.

As microarray technologies advance, extensive studies have been done to identify differentially expressed genes for various diseases [1]–[5]. However, due to different experimental platforms, biological techniques, and many other factors, even when the experiments are designed for achieving the same research goal (e.g. for studying the same disease), the identified differentially expressed genes may vary from study to study. The high variability of the experiment results obtained from different studies motivates researchers to combine the studies in order to obtain more "reliable" findings [6], [7]. While it is well recognized that combining the raw gene expression data is too difficult due to the incomparability of the measurements of gene expression across studies, the ranked gene lists produced by different studies in terms of the differentiation abilities of genes are more comparable [8]. Motivated by this observation, DeConde et al. [6] first formulated the problem of identifying the differentially expressed genes as a rank aggregation problem, which takes the ranked gene lists obtained from individual microarray studies as input and generate an aggregated gene list. The extended Kendall's tau distance for top-k lists [9] is adopted to measure the distance between the gene lists. Later, Lin [7], [10] also studied the problem of using rank aggregation to identify differentially expressed genes, and adopted the same formulation as that of DeConde et al.

The gene list obtained from a single microarray study usually contains up to thousands of genes. However, due to the complexity of biological validation, only the top-rated genes will be further checked to see whether or not they are *biologically significant* [6], [11], i.e., validated to be important for a certain disease. Therefore, through combining the gene lists from different studies, we expect that the aggregated gene list be more "*reliable*" than every input one, in the sense that more top-rated genes in the aggregated gene list are biologically significant. From this viewpoint, we can say that a gene list that minimizes its disagreement with the gene lists of other studies, especially among the top-rated genes, is a more "reliable" result.

However, the extended Kendall's tau distance adopted by DeConde et al. only counts the number of discordant gene pairs between two gene lists but ignores the ranks of the genes in the lists. Therefore, the disagreements among the top-rated genes and among the low-rated genes are treated with equal importance. In order to overcome the limitations of the above distance measure, we define two new distance functions, respectively called *weighted Kendall's tau* and *weighted Spearman's footrule*, both of which regard the disagreements among the top-rated genes to be more critical than those among the low-rated genes. Accordingly, we formulate the *weighted rank aggregation problem* (WRA) by adopting the proposed weighted distance functions.

Various rank aggregation methods have been applied to

aggregate the gene lists in order to identify differentially expressed genes. Among them, the *Markov Chain* (MC) based methods have shown their advantages in aggregating very few but long lists [12], which well conforms to the nature of the gene rank aggregation problem. DeConde et al. [6] considered two MC-based methods to aggregate the ranked gene lists, which leads to the identification of several biologically significant genes. We take advantage of the MC technique, and propose a new MC-based method called *Weighted MC* (WMC). The WMC method essentially determines the transition probability from one gene to another by incorporating the rank-based information, so that it better facilitates the effect of the weighted distance functions. Extensive experiments on the real biological datasets show that the WMC method is more efficient in aggregating long gene lists, and it is also more robust for identifying biologically significant genes, in the sense that these genes are constantly top-rated in the aggregated orders.

The rest of the paper is organized as follows. We first define some concepts and notations in Section II. Then, we propose the new weighted distance functions and formulate the WRA problem in Section III. An efficient MC-based method called WMC is presented in Section IV. Extensive experimental studies using the real biological datasets are conducted in Section V. Finally, we present the related work in Section VI, and conclude the paper in Section VII.

II. Preliminaries

We now introduce the basic notations that are used throughout the paper.

Definition 2.1: **(Linear Order)** Given a set of items $T = \{t_1, \ldots, t_n\}$, a *linear order* O of T is a binary relation on T, where the relation satisfies the properties of *antisymmetry*, *transitivity* and *totality*. We call T the *associated itemset* of O and denote by $O(t_i)$ the *rank* of item t_i in O.

We may simply write $O = [t_{i_1} \succ \cdots \succ t_{i_n}]$, where the ordered pair "$t_k \succ t_s$" means that t_k is preferred to t_s. In particular, the rank of the most preferred item in O, i.e., t_{i_1}, is 1. Suppose there is a set of linear orders $\{O_1, \ldots, O_m\}$, and their corresponding associated itemsets are $T_1, \ldots, T_m$, which are not necessarily to be identical. A *union itemset* is given by $T = \bigcup_i T_i$. As $T_i \subset T$, we say that O_i is an *incomplete linear order* with respect to T. In practice, different microarray experiments usually monitor different sets of genes [6]. Let T represent the union gene set which contains all the genes that appear in at least one ranked gene list produced by a particular study. A single gene list thus may not contain all the genes in T, and it can be regarded as an incomplete linear order with respect to T.

III. Weighted Distance Functions

In this section, we define the new *weighted Kendall's tau* and *weighted Spearman's footrule* distance functions to realize the following two goals.

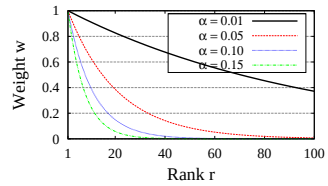


Figure 1. The rank-based weight function

Goal 1: The disagreement among top-rated items should be more critical than the disagreement among low-rated items.

Goal 2: The consistency among top-rated items should be more important than the consistency among low-rated items.

Before proposing the weighted distance functions, we first define a rank-based weight function as follows:

$$w(r) = \frac{e^{-\alpha r}}{e^{-\alpha}} = e^{-\alpha(r-1)}, \tag{1}$$

where r is the rank of item in the order and α is a small positive constant that controls how rapidly the weight w decreases as r increases. The weight function is monotonically decreasing, and is normalized by the largest value $e^{-\alpha}$, which is shown in Figure 1 with different α values.

A. Weighted Kendall's tau

We make use of the above rank-based weight function to define the penalties for all the disagreements between two incomplete linear orders, and accordingly propose our new *weighted Kendall's tau* distance function.

We consider two types of disagreements between two orders: the *ordering disagreement* and the *existence disagreement*. An item pair, existing in both orders, is said to cause an ordering disagreement if their ordering relationships in two orders disagree. An item existing in one order but not in the other is said to cause an existence disagreement. We define the penalties for these two types of disagreements as follows. Suppose there are two linear orders O_1 and O_2 with the associated itemsets respectively to be T_1 and T_2. Let T be the union of T_1 and T_2, i.e., $T = T_1 \cup T_2$.

- *Ordering disagreement* ($\mathcal{K}^o$)*:* For a pair of items $t_i, t_j \in (T_1 \cap T_2)$, if the ordering relationships between t_i and t_j are different in O_1 and O_2, the penalty caused by (t_i, t_j) is given by

$$\mathcal{K}^o_{i,j}(O_1, O_2) = w(r^*),$$

where $r^* = \frac{\min\{O_1(t_i), O_1(t_j)\} + \min\{O_2(t_i), O_2(t_j)\}}{2}$. The disagreement is emphasized by setting the weight using the smaller ranks of the pair of items in the orders. If the ordering relationship between t_i and t_j are the same in O_1 and O_2, the penalty is 0. This idea can be uniformly represented as:

$$\mathcal{K}^o_{i,j}(O_1, O_2) = \\ w(r^*) I[(O_1(t_i) - O_1(t_j))(O_2(t_i) - O_2(t_j)) < 0],$$

where $I[\cdot]$ is the indicator function. If the ordering disagreement happens among genes with smaller ranks, the r^* value will be smaller and accordingly a larger weight is counted, which satisfies Goal 1.

- *Existence disagreement* ($\mathcal{K}^e$)*:* For an item $t_i \in (T - (T_1 \cap T_2))$, since it exists in one order but is absent in the other, we define the penalty for its existence disagreement as follows:

$$\mathcal{K}_i^e(O_1, O_2) = w(O_1(t_i))I[t_i \in T - T_2] + w(O_2(t_i))I[t_i \in T - T_1].$$

If t_i gets a smaller rank in one order, i.e., top-rated, a larger penalty should be counted for its absence in the other order. Counting a large penalty for the existence disagreement of an item with small rank encourages other items that appear in both orders be ranked higher, which conforms to Goal 2.

Summing up the penalties of all possible disagreements, the weighted Kendall's tau distance between O_1 and O_2, denoted as $\mathcal{K}(O_1, O_2)$, is defined as:

$$\mathcal{K}(O_1, O_2) = \sum_{t_i, t_j \in (T_1 \cap T_2)} \mathcal{K}_{i,j}^o(O_1, O_2) + \sum_{t_i \in (T - (T_1 \cap T_2))} \mathcal{K}_i^e(O_1, O_2).$$

B. Weighted Spearman's footrule

We similarly consider the ordering disagreement and the existence disagreement, and define the weighted Spearman's footrule distance function between two linear orders O_1 and O_2, which is denoted as $\mathcal{S}(O_1, O_2)$.

- *Ordering disagreement:* For an item $t_i \in (T_1 \cap T_2)$, the penalty is defined as

$$\mathcal{S}_i^o(O_1, O_2) = w(r^*)|O_1(t_i) - O_2(t_i)|,$$

where $r^* = \min\{O_1(t_i), O_2(t_i)\}$. $|O_1(t_i) - O_2(t_i)|$ is the difference between the ranks of t_i in O_1 and O_2, which is additionally weighted by the location of t_i in the orders. The smaller rank t_i has in either of the orders, a larger weight is counted.

- *Existence disagreement:* For an item $t_i \in T - (T_1 \cap T_2)$, we define the penalty for its existence disgreement as:

$$\mathcal{S}_i^e(O_1, O_2) = w(O_1(t_i))I[t_i \in T - T_2] + w(O_2(t_i))I[t_i \in T - T_1].$$

Summing up the penalties of all possible disagreements, the weighted Spearman's footrule distance $\mathcal{S}(O_1, O_2)$ is defined as:

$$\mathcal{S}(O_1, O_2) = \sum_{t_i \in (T_1 \cap T_2)} \mathcal{S}_i^o(O_1, O_2) + \sum_{t_i \in (T - (T_1 \cap T_2))} \mathcal{S}_i^e(O_1, O_2).$$

C. Weighted Rank Aggregation

Having defined the weighted distance functions, we now formulate the *weighted rank aggregation* problem as follows.

Definition 3.1: **(Weighted Rank Aggregation (WRA))**: Given a set of linear orders $\{O_1, \ldots, O_m\}$ with $T = \bigcup T_i$, we aim to find a linear order O^* of T such that $\sum_{i=1}^m \mathcal{D}(O^*, O_i)$ is minimized, where $\mathcal{D}$ is the distance function, and it can either be the weighted Kendall's tau $\mathcal{K}$ or the weighted Spearman's footrule $\mathcal{S}$.

Complexity Note. The WRA problem adopts the weighted Kendall's tau distance function. Thus, if we set the weight function to be a constant, say $w(r) = 1$, and require that all the input orders contain the same set of items, the WRA problem is reduced to the Kendall-optimal rank aggregation problem, which is known to be NP-hard [12]. Thus, the WRA problem is at least NP-hard.

IV. Algorithm

In this section, we propose a new Markov chain method called *Weighted Markov Chain* (WMC) and present an algorithm that takes both the ordering relationship and the rank-based weight of items to construct the transition matrix.

A. Constructing Transition Matrices

Figure 2 shows the algorithm called TransMatrix, which constructs the transition matrix from an input set of linear orders. Given a set of linear orders, and the weight parameter α_w used for computing the weight by Equation (1), we construct a $|T|$-by-$|T|$ transition matrix $\mathcal{M}$, where T consists of all the items appearing in the input orders, and each entry $\mathcal{M}(t_i, t_j)$ corresponds to the probability that the Markov-chain process transfers from state (i.e., item) t_i to state t_j (Lines 1-2). The construction of the transition matrix can be accomplished by the following three steps.

Step One: We first impose on $\mathcal{M}$ the probability induced from the ordering relationship between every pair of items (Lines 3-6). For a pair of items t_i and t_j which co-occur in at least one input order, the probability that t_i transfers to t_j should intuitively be proportional to the weight with which the ordering relationship "$t_j \succ t_i$" is supported. For a specific order O_k, the weight with which O_k supports "$t_j \succ t_i$" is $w(O_k(t_j))$. Thus, the weight with which the relationship "$t_j \succ t_i$" is supported is (Line 4):

$$\mathcal{M}^*(t_i, t_j) = \sum_{k: O_k(t_j) < O_k(t_i)} w(O_k(t_j)).$$

Then, the entry $\mathcal{M}(t_i, t_j)$ is determined by (Line 5)

$$\mathcal{M}(t_i, t_j) = \frac{\mathcal{M}^*(t_i, t_j)}{\mathcal{M}^*(t_i, t_j) + \mathcal{M}^*(t_j, t_i)}.$$

Step Two: We consider the probability that every item should stay in its own state during the transition. We compute a rank-based weight for each item, and the larger the weight is, the more probable the item is to transfer to itself (Lines 7-14). We first obtain the set of ranks that t_i has in the input orders, and call the set the *rank profile* of t_i, denoted as $P(t_i)$ (Line 8). To combat the influence of noise that t_i is placed with exceptional ranks in some orders, we take the median of the ranks in $P(t_i)$, denoted as $r_m(t_i)$,

TRANSMATRIX Algorithm
Input: A set of orders $\{O_1, \ldots, O_m\}$; weight para α_w.
Output: The transition matrix $\mathcal{M}$.
Variables: $\mathcal{O}(t_i)$- the set of orders containing t_i.

1: $T = \bigcup_{i=1}^{m} T_i$
2: Initialize the $|T|$-by-$|T|$ matrix $\mathcal{M}$ to be $\mathbf{0}$.
3: **for** $t_i, t_j \in T$ with $t_i \neq t_j$ and $\mathcal{O}(t_i) \cap \mathcal{O}(t_j) \neq \phi$ **do**
4: $\quad \mathcal{M}^*(t_i, t_j) = \sum_{k:O_k(t_j)<O_k(t_i)} w(O_k(t_j))$
5: $\quad \mathcal{M}(t_i, t_j) = \frac{\mathcal{M}^*(t_i,t_j)}{\mathcal{M}^*(t_i,t_j)+\mathcal{M}^*(t_j,t_i)}$
6: **end for**
7: **for** $t_i \in T$ **do**
8: $\quad P(t_i) = \{O_k(t_i) | t_i \in T_k\}$
9: $\quad$ Set $r_m(t_i)$ to be the median of ranks in $P(t_i)$
10: $\quad \mathcal{M}(t_i, t_i) = \frac{|P(t_i)| * w(r_m(t_i))}{m}$
11: $\quad$ **for** $t_j \in T$ **do**
12: $\quad\quad \mathcal{M}(t_i, t_j) += \frac{1}{m}(1 - \frac{|P(t_i)| * w(r_m(t_i))}{m})$
13: $\quad$ **end for**
14: **end for**
15: **for** $t_i \in T$ **do**
16: $\quad \mathcal{M}(t_i, t_j) = \frac{\mathcal{M}(t_i,t_j)}{\sum_{t_k \in T} \mathcal{M}(t_i,t_k)}$ for all $t_j \in T$
17: **end for**
18: **for** $t_i, t_j \in T$ **do**
19: $\quad \mathcal{M}(t_i, t_j) = (1 - \epsilon)\mathcal{M}(t_i, t_j) + \epsilon/|T|$
20: **end for**

Figure 2. TRANSMATRIX for constructing the transition matrix of WMC

and set the weight of t_i to be $w(r_m(t_i))$. The rank-based weight of t_i is further weighted by the frequency that t_i appears in the input orders, and thus the entry $\mathcal{M}(t_i, t_i)$ is

$$\mathcal{M}(t_i, t_i) = \frac{|P(t_i)|}{m} w(r_m(t_i)).$$

We then uniformly distribute the remaining probability $(1 - \mathcal{M}(t_i, t_i))$ to all the items in T. This implies that, regardless of any other factors, the probability that t_i transfers to any state in T is equal (Lines 11-13).

Step Three: $\mathcal{M}$ is first normalized such that the sum of every row is 1 (Lines 15-17). Then, in order to guarantee that the transition matrix be *ergodic*, all the entries are multiplied by $(1-\epsilon)$ and then added with $\epsilon/|T|$ (Lines 18-20), where ϵ is a small positive number. In practice, we choose $\epsilon = 0.15$.

B. Markov Chain Process

Having the transition matrix constructed, the WMC algorithm sets the initial probability of every item to be equal and starts the Markov chain process. After each iteration, items are ordered in decreasing order of their updated probabilities. When the order becomes stable, the Markov chain process stops, and this order is output as the aggregated order.

C. Comparisons with MC4 and MCT

The MC4 and MCT algorithms were adopted by DeConde et al. [6] for gene rank aggregation. Both MC4 and MCT determine the transition probability merely by the ordering relationships of item pairs and disregard the location where a certain ordering relationship is. Thus, when the similarities of the top-rated items in the aggregated order are more important, they usually truncate the input orders and only keep the top-k lists. The reason is that, in order to pursue the overall similarity when long lists are aggregated, the local similarity like top-rated results may be sacrificed.

In contrast, our WMC method makes use of the rank-based information to determine the transition matrix, which emphasizes the accuracy of top-rated results and does not need to truncate the input orders. Suppose there is a pair of items a and b, and the relationship "$a \succ b$" prevails in a few, say l, orders with small ranks. The relationship "$b \succ a$" also appears in $l+1$ other orders, but with very large ranks. In this case, if the full lists are aggregated, both MC4 and MCT algorithms prefer the transition from b to a. However, since the weight of "$a \succ b$", which is associated with very small rank, may still be very much larger than the weight of "$b \succ a$", which is associated with large rank. The WMC algorithm still prefers the transition from a to b.

V. EXPERIMENTS

In this section, we study the effectiveness of our proposed distance functions and evaluate the performance of the WMC algorithm using the real data from five prostate cancer microarray studies [1]–[5].

We adopt the methods introduced in [6] to generate the ranked gene list from each study. Then, we vary m to be $\{300, 600, \ldots, 2100\}$, and respectively take the top-m genes of each gene list to form a set of five input orders with length m. The seven datasets of different lengths are used to evaluate our WMC method, and two other MC-based methods, MC4 and MCT. We also compare with a statistical method for discovering differentially expressed genes from multiple studies, which is proposed by Rhodes et al. [11] and denoted as RHODES. All the experiments are conducted on a Macbook Pro with 2.53GHZ CPU and 4G memory.

As the top-100 genes in the ranked gene lists are more important for further scientific investigation [6], we need to ensure that enough penalties are counted for the disagreement happened among the top-100 results. Thus, according to Figure 1, we set the α value for computing the weighted distance functions to be 0.05.

A. Efficiency of WMC

We first study the efficiency of the three MC-based methods with respect to the length of the input orders. The running time of each method is shown in Figure 3. For WMC, we run it by setting α_w to be 0.01, 0.05, and 0.09, which correspond to the series "WMC (0.01)", "WMC (0.05)", and "WMC (0.09)" in the figure.

While the running time of the three methods are very close on small datasets, the running time of WMC under

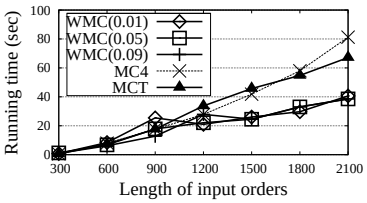


Figure 3. Running time

all α_w values becomes apparently shorter than that of MC4 and MCT when the dataset gets larger. The reason is that, when the input dataset gets larger, both MC4 and MCT need more iterations to reach the stable states, while less number of iterations are needed by WMC. As more time will be used for one iteration based on the transition matrix induced from the larger datasets, the WMC method becomes more efficient than MC4 and MCT for the large datasets. The reason why our WMC method consumes less number of iterations to reach the stable state on large datasets can be explained as follows. When a small dataset gets larger with more items appended at the end of each order, the ordering relationships between the appended items and other highly-ranked ones reinforce the probabilities of transferring to the highly-ranked items from them. Moreover, since the rank-based weight of the newly-appended items is small due to their large ranks, the weight contributes little to the transition probabilities that other items may transfer to the appended items. Thus, comparing to the transition matrix constructed from the smaller dataset, the transition matrix constructed from the larger one makes the Markov chain process more quickly to reach the state state.

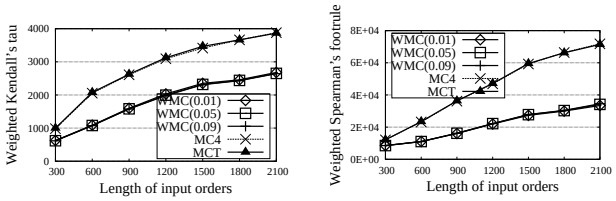
B. Comparison of Distance Functions

We then study the effectiveness of the weighted distance functions. Figures 4(a) and 4(b) show the sum of the distance between the aggregated order and the input orders when the weighted Kenall's tau distance and the weighted Spearman's footrule distance are respectively taken.

The weighted Kendall's tau distance of WMC is smaller than that of MC4 and MCT for all the datasets. The reason is that, the ordering disagreements (OD) between the aggregated order of WMC and the input orders happen among items with larger ranks, and thus the penalties caused by OD are much smaller. It implies that, comparing to MC4 and MCT, WMC tends to generate the aggregated orders that better preserve the ordering relationship among top-rated items. The weighted Spearman's footrule distance function counts for each OD the penalty of the location-based weight multiplied by the rank difference of the item causing the disagreement, which even amplifies the advantage of WMC in reducing OD. Thus, the weighted Spearman's footrule distance of WMC is even smaller, as shown in Figure 4(b).

C. Biological Significance

Identifying the differentially expressed genes are critical for tracing the progression of a disease. [6] lists the following eight differentially expressed genes: **HPN**, **AMACR**, **FASN**, **GUCY1A3**, **STRA13**, **CCT2**, **CANX**, and **TRAP1**, which have already been verified to be clinically important for tracing the development of the prostate cancer.



(a) Weighted Kendall's tau (b) Weighted Spearman's footrule

Figure 4. Effects of the distance function

In order to evaluate the quality of aggregation, we collect the ranks of these genes in the aggregated orders generated by the four methods, and check whether these genes are also "successfully" identified. We adopt the benchmark defined by [6]: a gene is said to be successfully identified if its rank in an aggregated order is smaller than 100. Since the setting of α_w of WMC does not influence much the ranks of the genes, we only show the result when $\alpha_w = 0.01$ for clarity.

Due to space limit, we only show the rank distributions of genes HPN, AMACR, FASN, and CANX in Figure 5. In Table I, the average ranks of the eight genes over the seven datasets are listed for three MC-based methods. RHODES performs bad on small datasets, and thus we show its best results acquired based on the input orders of length 2100. Comparing WMC, MC4 and MCT, all the genes except HPN get much smaller average ranks when the WMC method is adopted, and the average ranks of six out of eight genes, i.e., HPN, AMACR, FASN, GUCY1A3, STRA13, and CANX, are smaller than 100. In contrast, only HPN and AMACR get the average ranks smaller than 100 if either MCT or MC4 is adopted. Four genes get the ranks smaller than 100 when RHODES is adopted. However, the accuracy of RHODES is achieved by running a large number of permutation testing for every single gene, which is time consuming. According to the settings in [11], RHODES spends 506 seconds for permutation testing on the input orders of length 2100, which is 12 times longer than the time consumed by WMC.

In addition, when WMC is adopted, the ranks of these genes remain quite stable, no matter which dataset is taken for aggregation, while the rank of these genes changes a lot if either MC4 or MCT is adopted. The instability of the ranks of the genes over different datasets makes it a difficult problem for the MCT and MC4 methods to determine the proper length of input orders so that more significant genes can be ranked reasonably high. However,

Table I
THE AVERAGE RANKS OF GENES OVER SEVEN DATASETS

	WMC	MC4	MCT	RHODES
HPN	**2.1**	**1.0**	**1.0**	**1**
AMACR	**1.1**	**9.7**	**7.86**	**7**
FASN	**51.1**	194	202	554
GUCY1A3	**17.9**	589.3	595.6	**27**
STRA13	**97.7**	139.4	140.6	662
CCT2	111.3	207.6	200.3	282
CANX	**10.3, 2.7**	276.9	253.7	3072
TRAP1	165.4	215.4	216.1	**3**

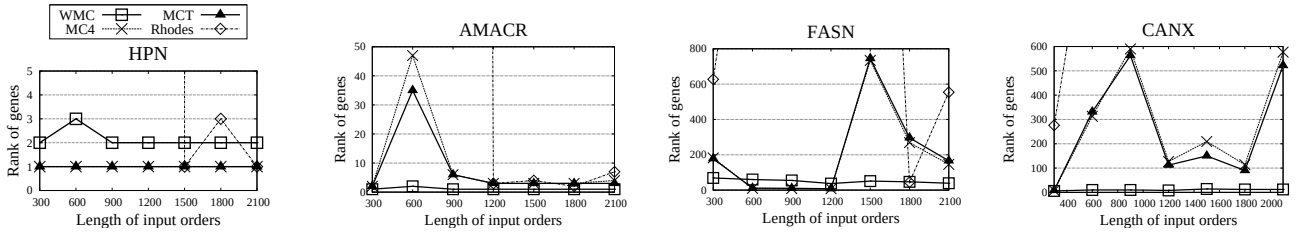


Figure 5. Stability of the ranks of four identified prostate cancer genes

as we have explained in the first set of experiments, the items with very low ranks in the input orders may only reinforce the probabilities that those highly-ranked items can be transferred to, and thus the datasets with longer orders are always perferred by WMC.

VI. Related Work

A few representative gene rank aggregation methods are: Borda's method [7], Fagin's median-rank based method [13], Thurstone's order-statistic method [6], and Markov chain based methods [6], [7]. Both Borda's method and Fagin's median-rank based method are positional rank aggregation methods, which are simple and efficient. However, their performance in optimizing some distance criteria, like Kendall's tau, may not be good enough when incomplete orders are aggregated. Thurstone's model assumes that each item follows a normal distribution, and items are ordered in terms of their estimated mean values, which leads to the aggregated order. However, for the gene rank aggregation problem considered in this paper, Thurstone's method may suffer from the problem that the samplings (i.e., input orders) are usually too few, which may affect the accuracy of parameter estimation. The MC-based methods, i.e., MC1 to MC4 [12] and MCT [6], have shown their advantage in aggregating few number of long top-k lists, which well conforms to the gene rank aggregation problem. However, none of the five MC-based methods make use of the rank-based information, and the generated aggregated orders do not favor the accuracy of top-rated items either.

VII. Conclusions

In this paper, we study a novel weighted rank aggregation problem, which aims to identify more reliable differentially expressed genes by combining the results obtained from different microarray studies. We propose two new distance functions, the weighted Kendall's tau and the weighted Spearman's footrule, which measure the difference between gene lists by incorporating rank-based weight. These distance functions emphasize the similarities as well as the disagreements among top-rated items, which better favors the domain specific need that the reliable differentially expressed genes are expected to be top-rated.

We develop a new Markov chain based rank aggregation method called WMC. The WMC method makes use of the rank-based information to determine the transition from one state to another. Our experiments based on the real biological datasets show that the WMC method better facilitates the effect of the weighted distance functions, comparing with two other MC-based methods. The WMC method is also more efficient in aggregating long ranked gene lists, since the stable state can be reached more quickly. Importantly, by checking the rank of eight well-identified differentially expressed genes in the datasets, we confirm that WMC is far more robust in identifying truly differentially expressed genes than its counterparts, in the sense that most of such genes are constantly top-rated in the aggregated orders.

References

[1] S. Dhanasekaran, T. Barrette, D. Ghosh, R. Shah, and et al., "Delineation of prognostic biomarkers in prostate cancer," *Nature*, vol. 412, pp. 822–826, 2001.

[2] J. Luo, D. Duggan, Y. Chen, J. Sauvageot, and et al., "Human prostate cancer and benign prostatic hyperplasia: Molecular dissection by gene expression profiling," *Cancer Research*, vol. 61, pp. 4683–4688, 2001.

[3] D. Singh, P. G. Febbo, K. Ross, D. G. Jackson, and et al., "Gene expression correlates of clinical prostate cancer behavior," in *Cancer Cell*, vol. 1, 2002, pp. 203–209.

[4] L. True, I. Coleman, S. Hawley, C.-Y. Huang, and et al., "A molecular correlate to the gleason grading system for prostate adenocarcinoma," *Proc Natl Acad Sci USA*, vol. 103, no. 29, pp. 10 991–10 996, 2006.

[5] J. Welsh, L. Sapinoso, A. Su, and et al., "Analysis of gene expression identifies candidate markers and pharmacological targets in prostate cancer," *Cancer Research*, vol. 61, 2001.

[6] R. P. DeConde, S. Hawley, S. Falcon, and et al., "Combining results of microarray experiments: A rank aggregation approach," *Stat Appl Genet Mol Biol*, vol. 5, no. 1, 2006.

[7] S. Lin, "Rank aggregation methods," *Wiley Interdisciplinary Reviews: Computational Statistics*, vol. 2, pp. 555–570, 2010.

[8] T. Yuen, E. Wurmbach, R. Pfeffer, B. Ebersole, and et al., "Accuracy and calibration of commerical oligonucleotide and custom cdna microarrays," *Nucleic Acids Res.*, 2002.

[9] R. Fagin, R. Kumar, and D. Sivakumar, "Comparing top-k lists," in *SODA '03*, 2003, pp. 28–36.

[10] S. Lin, "Space oriented rank-based data integration," *Stat Appl Genet Mol Biol*, vol. 9, no. 1, 2010.

[11] D. R. Rhodes, T. R. Barrette, and et al., "Meta-analysis of micorarrays: Interstudy validation of gene expression profiles reveals pathway dysregulation in prostate cancer," *Cancer Research*, vol. 62, no. 15, pp. 4427–4433, 2002.

[12] C. Dwork, R. Kumar, M. Naor, and D. Sivakumar, "Rank aggregation methods for the web," in *WWW' 01*, 2001.

[13] R. Fagin, R. Kumar, and D. Sivakumar, "Efficient similarity search and classification via rank aggregation," in *SIGMOD '03*, 2003, pp. 301–312.