This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TNB.2014.2359494, IEEE Transactions on NanoBioscience

RANWAR: Rank-Based Weighted Association Rule Mining from Gene Expression and Methylation Data

Saurav Mallik*, Anirban Mukhopadhyay†, *Senior Member, IEEE*, and Ujjwal Maulik‡, *Senior Member, IEEE*

***Abstract*—Ranking of association rules is currently an interesting topic in data mining and bioinformatics. The huge number of evolved rules of items (or, genes) by association rule mining (ARM) algorithms makes confusion to the decision maker. In this article, we propose a weighted rule-mining technique (say, $RANWAR$ or rank-based weighted association rule-mining) to rank the rules using two novel rule-interestingness measures, viz., rank-based weighted condensed support (wcs) and weighted condensed confidence (wcc) measures to bypass the problem. These measures are basically depended on the rank of items (genes). Using the rank, we assign weight to each item. $RANWAR$ generates much less number of frequent itemsets than the state-of-the-art association rule mining algorithms. Thus, it saves time of execution of the algorithm. We run $RANWAR$ on gene expression and methylation datasets. The genes of the top rules are biologically validated by Gene Ontologies (GOs) and KEGG pathway analyses. Many top ranked rules extracted from $RANWAR$ that hold poor ranks in traditional Apriori, are highly biologically significant to the related diseases. Finally, the top rules evolved from $RANWAR$, that are not in Apriori, are reported.**

***Index Terms*—Weighted association rule mining, wcs, wcc, Limma, gene-weight, gene-ranking, $RANWAR$.**

I. Introduction

KNOWLEDGE Discovery and Data Mining (KDD) is an interdisciplinary domain that mainly focus on the systematic ways of acquiring interesting rules and patterns from the data. The significant common patterns which are estimated by interestingness measures include association rules. Association rule mining (ARM) [3], an important data mining technique is utilized for detecting interesting relationships between items.

Huge number of rules always creates problem to select top among them. Therefore, the ranking of rules from the biological data is very important area for research. For this, different rule-interestingness measures (viz., support, confidence, lift, conviction etc.) were proposed. But, these still generate huge number of frequent itemsets, and thereby these generate huge number of association rules. Thus, lot of time is taken to run these algorithms. In this article, we propose a weighted rule-mining technique (viz., $RANWAR$ or Rank-based Weighted Association Rule-Mining) which has been developed using two novel measures rank-based weighted condensed support (say, wcs) and rank-based weighted condensed confidence (say, wcc) measures for extracting rules from the data. Sometime it happens that a lot of rules have same support and same confidence. At this moment, if we need some of them, it is difficult to differentiate among them. Therefore, if we apply the wcs and wcc, we can easily categorize them. The major benefit of $RANWAR$ is that it generates much less number of frequent itemsets than state-of-the-art association rule mining algorithms for same minimum support value. There is no such ARM method which generates lesser number of frequent itemsets than $RANWAR$. Thereby, it takes much less time than the other algorithms. Another benefit of $RANWAR$ is that some of the rules which hold low rank in traditional rule mining algorithms, hold good rank in $RANWAR$. Some evidences of biological significance of the genes of the evolved rules are also found.

As we know that if the number of genes in data is large, the number of itemsets will be also large, thus, using Limma [2] statistical test, we have just taken into account the top differentially expressed (i.e., DE) or differentially methylated (i.e., DM) genes. Limma is an useful statistical test which performs well for both normally and non-normally distributed data for all types of sample-size (i.e., small, medium, large). Our proposed measures are basically rank-based weighted measures. Therefore, ranking of genes has a significant role here. Limma test provides a rank-wise gene-list according to their p-values from best to worst cases. Thereafter, we assign weight to each item/gene w.r.t. their p-value ranking, and include these into the measures. Therefore, our measures give importance to each item (gene). Our proposed measures (viz., wcs and wcc) are condensed form of the traditional support and confidence measures. Furthermore, two gene expression datasets and two methylation datasets are used to test the performance of $RANWAR$. We have made a comparative analysis of it with the traditional Apriori algorithm [1] and other state-of-the-art rule mining algorithms. For validation of the rules, GO terms and KEGG pathways of the genes in the rules are identified. The genes of the evolved rules involving highest number of GOs/pathways are reported for biologically benevolent aspects. Finally, we report many top ranked rules produced by $RANWAR$ that hold poor ranks in traditional Apriori, but are highly biologically significant to related diseases.

The rest of the article is organized as follows. Section II presents literature review. Section III and section IV elaborate the proposed measures, and the proposed ARM method, respectively. The source and description about the dataset are given in section V. Section VI presents the experimental results and discussion, where section VII shows utility of ARM in biology and our novel findings. Finally, Section VIII concludes the article.

II. Literature Review

ARM [1] is a popular technique to estimate interesting relationships among different items (i.e., genes). Suppose, $Itmset = \{i_1, i_2, ..., i_n\}$ be an itemset (i.e., set of genes) and $S = \{s_1, s_2, ..., s_m\}$ be a set of transactions (samples). Thus, a rule might be described as $A \Rightarrow C$, where $A, C \subseteq Itmset$ and $A \bigcap C = \phi$. Here, A is called as antecedent and C is called as consequent. In a transaction database, a transaction may consist of a set of items purchased in it. In a similar sense, in gene expression / methylation [3] dataset, in any tissue sample (trans-

*S. Mallik is with Machine Intelligence Unit, Indian Statistical Institute, Kolkata, India. E-mail: sauravmtech2@gmail.com, chasaurav_r@isical.ac.in

†A. Mukhopadhyay is with the Department of Computer Science and Engineering, University of Kalyani, Nadia, India. E-mail: anirban@klyuniv.ac.in

‡U. Maulik is with the Department of Computer Science and Engineering, Jadavpur University, Kolkata, India. E-mail: umaulik@cse.jdvu.ac.in

action), a set of genes may occur together. Some of them are up-regulated/hyper-methylated, and some are down-regulated/hypo-methylated, and remaining are not differentially expressed / methylated (i.e., neither up-regulated/hyper-methylated nor down-regulated/hypo-methylated). In case of a biological transaction, suppose, $\{gene1+, gene2-, gene3_{nde} \Rightarrow gene4+\}$ is an association rule which states that if gene1 is up-regulated (denoted by '+'), gene2 is down-regulated (denoted by '-'), and $gene3_{nde}$ is non-differentially expressed (depicted as 'nde') simultaneously, then it is likely that gene4 will be up-regulated. Therefore, those four genes must occur in some of transactions simultaneously. The support of an itemset is defined as number of transactions in which all items of the itemset appear simultaneously. The itemset is frequent when its support is greater than any threshold value (i.e., minimum support). The confidence of the rule is defined as ratio of support of the itemset to the support of antecedent.

Apriori [1] is a basic algorithm for learning association rules to control on databases that have transactions. Apriori utilizes a "bottom-up" approach, where frequent subsets are extended one item at a time to generate each candidate and groups of the candidates are tested against the data. The algorithm terminates if there is no further successful extensions to be identified. The output of Apriori is actually the sets of rules that generate the occurrence of items in the dataset. Apriori follows breadth-first search to count the candidate itemsets. Apriori produces candidate itemsets of length k from itemsets of length $k-1$. Thereafter, it eliminates the candidates having an infrequent sub-pattern.

By further investigations, different limitations have been found in the traditional Apriori algorithm, like generation of huge number of frequent itemsets, high elapsed time, multiple-scan problem, load-imbalance problem, importing same importance to each item etc. For reducing those shortcomings, different enhancements have been performed on the original Apriori algorithm (viz., Orlando et al. in 2001 [14], Pavon et al. in 2006 [11], Yu et al. in 2008 [16], Oguz et al. in 2012 [13] etc.). Besides that, many other ARM methodologies have been proposed (e.g., Tao et al. in 2003 [19]). Tao et al. used weighted ARM technique in efficient way. But, very less improvement on reducing elapsed time has been done through Tao et al. Therefore, further new methodologies are proposed (Yun et al. (WIP) in 2006 [15], Hong et al. in 2008 [12], Sun et al. in 2008 [17], Ahmed et al. in 2008 [18]). But, it has been noticed that it is very difficult to reduce all such limitations simultaneously. Therefore, we have mainly focused how to reduce elapsed time for rule mining in such way that only top ranked items and their related highly significant rules will present in result for large transaction database.

III. Proposed measures

In this article, we propose two novel rule-interestingness measures, viz., rank-based *weighted condensed support* (wcs) and *weighted condensed confidence* (wcc) on basis of independency of the genes of a microarray dataset [3], [4] in *statistical scenario*. Suppose, input boolean matrix BIT is of size $m \times n$, where m denotes #sample and n denotes #gene. The assigned weights to genes are assumed to be $W = \{w_1, w_2, ..., w_n\}$. A weight w_i is attached to each gene g_i (i.e., $i = 1, 2, ..., n$). This denotes a pair of (g_i, w_i) which is stated as a weighted gene. The weight of the g_i gene in the k-th sample/transaction is denoted by w_{ki}, where $1 \leq k \leq m$. If the gene g_i presents in the k-th transaction (s_k), then value of w_{ki} will be the weight of the gene g_i, otherwise, value of w_{ki} becomes zero. In other words,

$$w_{ki} = \begin{cases} w_i, & \text{if } g_i \in s_k. \\ 0, & \text{otherwise.} \end{cases}$$

For microarray dataset, differential expression values (t-statistical values, or corresponding p-value in any statistical test) of genes are calculated on the assumption of independency of genes [7]. If all the genes are dependent, then p-values (specified probabilities) of the genes in the test will be invalid [7]. Therefore, on the basis of the independency of the genes of a microarray dataset, we have determined itemset-transaction weight. The weight of a gene is calculated through p-value ranking of the gene. Thus, itemset-transaction weight can be defined as multiplication of weights of all the genes (items) of the itemset in a transaction (sample) for a microarray dataset. It is estimated as:

$$W_k(Z) = \prod_{i=1}^{Q}{}_{(\forall g_i \epsilon Z, Q=|Z|)} w_{ki}, \tag{1}$$

where $W_k(Z)$ denotes itemset-transaction weight of itemset Z for k-th transaction, w_{ki} refers to the weight of gene g_i for k-th transaction, $\prod$ denotes multiplicative operator, Q refers to the total number of genes in the itemset Z.

The proposed support of the itemset (i.e., $wcs(Z)$) is defined in two folds. If the size of itemset is one, then the $wcs(Z)$ is the ratio of summation of all the itemset-transaction weights of the itemset (i.e., summation of all $W_k(Z)$, $\forall k$) to the total number of transactions/samples (i.e., m) in the database. But if the size of itemset is greater than one, then the $wcs(Z)$ is the ratio of summation of all the itemset-transaction weights of the itemset (i.e., summation of all $W_k(Z)$, $\forall k$) to the frequency of the highest frequent gene/item of the itemset (i.e., $m'(Z)$) instead of considering m. Thus, wcs is stated as the *condensed form* of the traditional support. The proposed support can be stated as:

$$wcs(Z) = \begin{cases} \frac{\sum_{k=1}^{m} W_k(Z)}{m'(Z)}, & \text{if } |Z| > 1 \\ \\ \frac{\sum_{k=1}^{m} W_k(Z)}{m}, & \text{if } |Z| = 1 \end{cases} \tag{2}$$

TABLE I: An example of calculating the proposed wcs and wcc.

Genes (Items) ⟶

Item weight (w_i)	g1	g2	g3	g4	g5
	1.0	0.2	0.8	0.6	0.4

Genes (Items) ⟶ / Transactions ↑

	g1	g2	g3	g4	g5
s1	1	1	0	1	1
s2	0	0	1	1	0
s3	0	1	1	1	1
s4	1	1	0	0	1

Suppose, Z is whole itemset of a rule, $\{g1, g2 \Rightarrow g5\}$; Here, antecedent, $A = \{g1, g2\}$; consequent, $C = \{g5\}$; and $Z = A \cup C = \{g1, g2, g5\}$; and
So, $W_1(Z) = w_{11} * w_{12} * w_{15} = w_1 * w_2 * w_5 = 1.00 * 0.2 * 0.4 = 0.08$;
$W_2(Z) = w_{21} * w_{22} * w_{25} = 0 * 0 * 0 = 0$;
$W_3(Z) = w_{31} * w_{32} * w_{35} = 0 * w_2 * w_5 = 0 * 0.2 * 0.4 = 0$;
$W_4(Z) = w_{41} * w_{42} * w_{45} = w_1 * w_2 * w_5 = 1.00 * 0.2 * 0.4 = 0.08$;

$$\begin{aligned} m'(Z) &= max\{\sum_{k=1}^{4} BIT_{k1}, \sum_{k=1}^{4} BIT_{k2}, \sum_{k=1}^{4} BIT_{k5}\} \\ &= max\{(1+0+0+1), (1+0+1+1), (1+0+1+1)\} \\ &= max\{2, 3, 3\} = 3 \end{aligned}$$

$wcs(Z) = \frac{W_1(Z)+W_2(Z)+W_3(Z)+W_4(Z)}{m'(Z)} = \frac{0.16}{3} = 0.053$;
Similarly, $W_1(A) = w_{11} * w_{12} = w_1 * w_2 = 1.00 * 0.2 = 0.2$;
$W_2(A) = w_{21} * w_{22} = 0 * 0 = 0$; $W_3(A) = w_{31} * w_{32} = 0 * w_2 = 0 * 0.2 = 0$;
$W_4(A) = w_{41} * w_{42} = w_1 * w_2 = 1.00 * 0.2 = 0.2$;

$$\begin{aligned} m'(A) &= max\{\sum_{k=1}^{4} BIT_{k1}, \sum_{k=1}^{4} BIT_{k2}\} \\ &= max\{(1+0+0+1), (1+0+1+1)\} \\ &= max\{2, 3\} = 3 \end{aligned}$$

$wcs(A) = \frac{W_1(A)+W_2(A)+W_3(A)+W_4(A)}{m'(A)} = \frac{0.4}{3} = 0.13$;
Therefore, $wcc(A \Rightarrow C) = \frac{wcs(Z)}{wcs(A)} = \frac{0.053}{0.13} = 0.41$;

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TNB.2014.2359494, IEEE Transactions on NanoBioscience

where $m'(Z)$ is described as follows:

$$m'(Z) = \max_{(\forall g_i \epsilon Z, Q=|Z|)} \left\{ \sum_{k=1}^{m} BIT_{k1}, \sum_{k=1}^{m} BIT_{k2}, ..., \sum_{k=1}^{m} BIT_{kQ} \right\}, \quad (3)$$

where Q denotes the total number of genes in the itemset Z ($|Z| > 1$), and BIT_{ki} denotes boolean value of the gene g_i for k-th sample/transaction (here, $i = 1, 2, ..., Q$). Here, the boolean sub-matrix of the itemset Z having size $|m \times Q|$ is considered.

The proposed confidence of a rule (viz., $wcc(A \longrightarrow C)$) is defined as the ratio of the support of the itemset (i.e., $wcs(Z)$, $Z = A \cup C$) to the support of the antecedent (i.e., $wcs(A)$).

$$wcc(A \longrightarrow C) = \frac{wcs(A \cup C)}{wcs(A)} = \frac{wcs(Z)}{wcs(A)}. \quad (4)$$

An example of calculating wcs and wcc is presented by Table I.

It should be mentioned that for rule mining using the wcs and wcc measures, we have to set two thresholds, one for wcs, and other wcc. It is well-known that support is the property of an itemset, and confidence is the property of a rule. Therefore, here, the above statement signifies that if wcs of an itemset is greater than equal to a minimum support threshold (say, min_wsupp), then the itemset is frequent. If the itemset is frequent, then wcc values of rules made from the itemset need to validate. If wcc of any rule is greater than equal to a minimum confidence threshold (say, min_wconf), then the rule can be selected.

IV. Proposed Rule Mining Approach

In this article, we propose a rank-based weighted association rule mining ($RANWAR$) using the two proposed rule-interestingness measures (wcs and wcc) specially for microarray/beadchip data [3], [7]. The steps of it are described below:

A. Finding differentially expressed/methylated genes and ranking

Microarray technique [7] is a useful tool for measuring gene expression data across different experimental and control samples. At first, some pre-filtering process is applied on the data (viz., removal of genes having low variance). In fact, due to the low variance of the gene, sometime lower p-value is produced which seems to be significant, but actually it is insignificant. Thus, it is needed to check the overall variance of the data according for each gene and filter out the genes having very low variance. Here, we have used a matlab function "genevarfilter" by which some user-defined percentile of the genes having low variance can be eliminated from the gene list. The filtered data should be normalized gene-wise as normalization converts the data from different scales into a common scale. In our experiment, we have used zero-mean normalization [7] which converts the data into such a form where mean of each gene becomes zero and standard deviation becomes one. The zero-mean normalization can be formulated as: $x_{ij}^{norm} = \frac{x_{ij}-\mu}{\sigma}$, where μ and σ refer to mean and standard deviation of the expression/methylation data of a gene i before normalization respectively; and x_{ij} and x_{ij}^{norm} denote the value of i-th gene at j-th sample before and after normalization, respectively.

For identifying DE/DM genes, a suitable non-parametric test should be applied correctly. Thus, we choose Limma [2] as it performs well for both normal and non-normal distributions for all sizes of data. The moderated t-statistic of Limma is stated as: $\tilde{t}_g = \frac{1}{\sqrt{\frac{1}{n_1}+\frac{1}{n_2}}} \frac{\hat{\beta}_g}{\tilde{s}_g}$, where sample size $n = n_1 + n_2$; $\hat{\beta}_g$ and $\tilde{s}_g^2$ denote the contrast estimator and posterior sample variance for the gene g, respectively. However, from the resulting value of the t-statistic, corresponding p-value is calculated from t-table or cumulative distribution function (cdf) [7]. If p-value of a gene is less than 0.05, then the gene is called DE/DM, otherwise not. The DE/DM genes are then ranked w.r.t. their p-values.

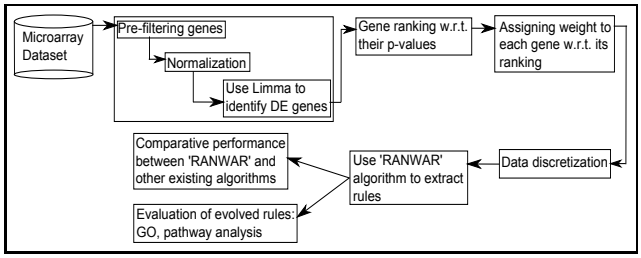


Fig. 1: The proposed rule mining approach from biological data.

B. Assigning weight to each gene

In our approach, all the genes have not same importance. That is why, some weight is assigned to each gene w.r.t. their p-value ranking mentioned earlier. Here, the weights of the genes are calculated in such a way that difference between the weights of any two consecutive ranked genes are same, and the weight of the first ranked gene is always 1 (see Table III for example). The ranges of weight lie in between 0 and 1. Suppose, n is number of genes. Thus, the weight of each gene (denoted by w_i, $1 \leq i \leq n$) is estimated from a function of the above rank (denoted by r_i, $1 \leq i \leq n$) and number of genes as described below:

$$w_i = \frac{1}{n} * (n - (r_i - 1)). \quad (5)$$

C. Data Discretization

Suppose, $I[r, c]$ is input data matrix. Here, r denotes genes, and c denotes samples. First of all, the matrix I is transposed. Suppose, IT be the resulting matrix. Now, discretization of the input data matrix is mandatory for applying association rule mining. For discretization purpose, we have utilized standard k-means clustering algorithm. But, before using k-means, we have chosen initial seed values for k-means using [6]. For doing this, at first, we choose first cluster center cc_1 uniformly at random from all the data points (χ). Thereafter, for each data-point y, we calculate the distance (D_y) between the data-point and nearest center which is already chosen. After that, choose the next cluster cc_j by taking $cc_j = y' \in \chi$ where the corresponding probability is $\frac{D(y')^2}{\sum_{y \in \chi} D(y)^2}$. Repeat the last step until we select k number of cluster centers. In this way, the initial centers are chosen that are used for the standard k-means clustering. Thereafter, we have run k-means clustering algorithm sample-wise (i.e., row-wise) on each row of IT where the number of clusters is 2 and the distance metric is the Euclidean distance. The cluster having higher centroid value is the cluster of up-regulated/hyper-methylated genes and the other cluster is the cluster of down-regulated/hypo-methylated genes.

According to IT matrix, '1'(red color) denotes up-regulated gene (DE_{up}) and '0'(green color) denotes down-regulated gene (DE_{down}) (see Fig. 2(b)). As in ARM,'1' and '0' signify presence and absence of some item (gene) in some transaction (sample), respectively for binary data, so here we can show only the

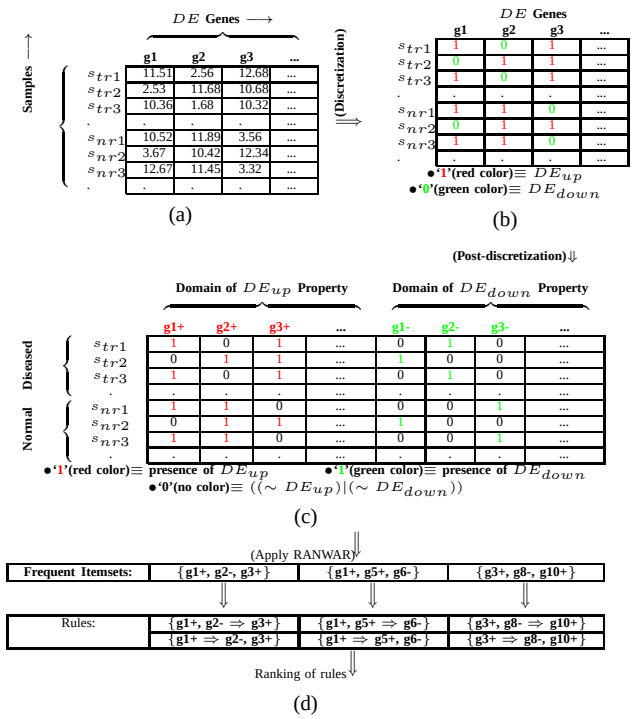


Fig. 2: An example of post-discretization using $RANWAR$.

Algorithm 1 RANWAR

Input: Data matrix D (rows=genes, columns=samples), original gene-list $A1$ according to D, rank-wise gene-list $A2$ (according to p-values of genes by Limma), flag of sorting the evolved rules $sortFlag$ (w.r.t. either wcs or wcc), minimum support threshold min_wsupp, minimum confidence threshold min_wconf.
Output: Set of rules $Rules$, support $RuleSupp$, confidence $RuleConf$.

procedure RANWAR
  Normalize the data-matrix D using zero-mean normalization.
  Calculate rank of genes (i.e., $rankk(:)$) according to original gene list $A1$.
  Assign weights $wt(:)$ to all genes according to their ranks $rankk(:)$.
  Transpose the normalized data-matrix.
  Choose initial seed values for using k-means clustering.
  Discretize the transposed matrix applying standard k-means clustering sample-wise.
  Apply post-discretization technique.
  Initialize $k = 1$.
  Find frequent 1-itemsets, $FI_k = \{i | i \in A1 \wedge wcs(i) \geq min\_wsupp\}$.
  repeat
    k=k+1.
    Generate candidate itemsets, CI_k from FI_{k-1} itemsets.
    for each candidate itemset, $c \in CI_k$ **do**
      Calculate $wcs(c)$ for each candidate itemset, c.
      if $wcs(c) >= min\_wsupp$ **then**
        $FI_k \leftarrow [FI_k; c]$.
        Generate rules, $rule(:)$ from the frequent itemset, c.
        Determine $wcc(:)$ for each $rule(:)$.
        for each evolved rule, $r \in rule(:)$ **do**
          if $wcc(r) >= min\_wconf$ **then**
            Store the r in the resulting rule-list $Rules$ with its wcs and wcc; $Rules \leftarrow r$, $RuleSupp \leftarrow wcs(r)$ and $RuleConf \leftarrow wcc(r)$.
          end if
        end for
      end if
    end for
  until $(FI_k = \emptyset)$
end procedure

presence of DE_{up} by using '1', and presence of DE_{down} by using '0'. But, we need to represent three categories of items, i.e., (i) DE_{up}, (ii) DE_{down} and (iii) $\sim DE_{up}| \sim DE_{down}$ using binary digits ('1' and '0'). Therefore, the number of columns of the IT is then set twice of the original (see Fig. 2(c)). Let, BIT be the resulting boolean matrix. In this case, for the first-half, '1'(red color) denotes presence of up-regulated gene (DE_{up}), and '0'(no color) represents absence of up-regulated gene ($\sim DE_{up}$). Similarly, for the second-half, '1'(green color) denotes presence of down-regulated gene (DE_{down}), and '0'(no color) represents absence of down-regulated gene ($\sim DE_{down}$). According to Fig. 2(c), s_{tr} and s_{nr} denote experimental/treated and normal/control samples respectively, where '+' and '-' denote up-regulation and down-regulation, respectively. Here, for the s_{tr1} sample, gene g_1 and gene g_3 are up-regulated, and gene g_2 is down-regulated. Thus, the cells $BIT(s_{tr1}, g_1+)$=1, $BIT(s_{tr1}, g_3+)$=1 and $BIT(s_{tr1}, g_2-)$=1. Thus, automatically, the cells $BIT(s_{tr1}, g_1-)$=0, $BIT(s_{tr1}, g_3-)$=0 and $BIT(s_{tr1}, g_2+)$=0. The same approach is applicable for the other samples/rows of the matrix BIT. Fig. 2 shows the procedure. If the genes become hyper-methylated (DM_{hyper}) and hypo-methylated (DM_{hypo}) instead of up-regulated (DE_{up}) and down-regulated (DE_{down}), respectively, same steps can be used.

D. Identification of frequent itemset and rule mining

After the post-discretization, we need to identify frequent itemsets. For this, at first, we evaluate wcs of the 1-itemsets, and then identify the frequent singleton itemsets (i.e., wcs of them are greater than equal to min_wsupp). Similarly, we calculate their supersets 2-itemsets and then determine frequent 2-itemsets. Then rules are extracted from the frequent 2-itemsets. Then, wcc of each rule is computed. The rules having wcc greater than equal to min_wconf value, are selected for resulting list of rules. Then, we determine their supersets 3-itemsets and then determine frequent 3-itemsets, and then extract significant rules from these, and so on. The algorithm terminates if there is no further successful extensions of frequent itemsets to be identified. Finally, the evolved rules are ranked w.r.t. wcs or wcc. For details, see Algorithm 1. $RANWAR$ is basically updated version of Apriori with the measures (i.e., wcs and wcc). However, the performance of $RANWAR$ is compared with other existing rule mining methods. The number of frequent itemsets are compared between $RANWAR$, and the other methods at different minimum support values. For validation of the rules, GO terms and KEGG pathways of the genes in the rules are identified. The genes of the evolved rules involving highest number of GOs/pathways are reported for biologically benevolent aspects. We report many top ranked rules produced by RANWAR that hold poor ranks in traditional Apriori, but are highly biologically significant to related diseases. A flowchart is provided to describe how to apply the measure to extract rules from biological dataset (see Fig. 1 and Fig. 2).

V. Real Datasets

Two real datasets are used which are described in Table II.

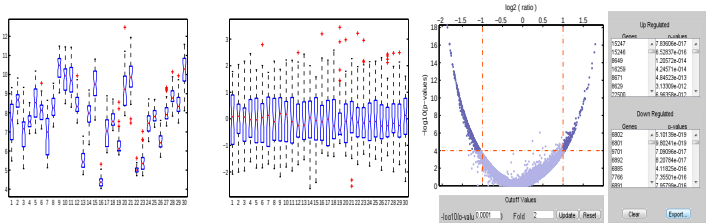
TABLE II: Information of used Datasets (DS).

id	Dataset information
DS1	Genome-wide DNA methylation dataset of Uterine cervical carcinogenesis (**NCBI Ref. id: GSE30760)** belonging 63 cancerous uterine cervix (CUC) treated samples and 152 normal uterine cervix (NUC) samples.
DS2	Gene expression dataset of cigarette smokers of lung adenocarcinoma (**NCBI Ref. id: GSE10072)** belonging 24 current-smoker (CS) treated samples and 16 never-smoker (NS) control samples, in Tumor category.
DS3	Expression dataset of Uterine Leiomyoma with 16 Uterine Leiomyoma (UL) samples and 16 normal myometrial (MM) samples (**NCBI Ref. id: GSE31699).**
DS4	Methylation dataset of Uterine Leiomyoma having the 18 UL samples and 18 MM samples (**NCBI Ref. id: GSE31699).**

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TNB.2014.2359494, IEEE Transactions on NanoBioscience

VI. Experimental Results and Discussion

We run $RANWAR$ on the two methylation datasets, and two expression datasets. As stated in section IV, at first, we eliminate top 10% genes which have lowest variance, and then normalize the data gene-wise using zero-mean normalization (see Fig. 3(a) and Fig. 3(b)). There are large number of genes in each dataset. So we run Limma test on the normalized dataset, and thereafter we make a rank-wise list of genes from best to worst according to their p-values in the test. We know that only highly (statistically) significant genes can make sense to the corresponding disease for biological data. So, considering all genes of a large dataset may increase elapsed time highly, and we will get a large number of rules in which most of them are redundant/insignificant. Thus, we consider top DE/DM genes, and generate only most significant rules instead of a large set of redundant rules. Here, we set different p-value cutoffs for different datasets for identifying different numbers of top ranked genes. E.g., p-value cutoff for DS1 is set 4.37E-40 to identify top 20 genes; that means only top 20 genes have p-values which are less than the cutoff, and p-values of remaining genes for the dataset are greater than equal to the mentioned cutoff. Similarly, for DS2, p-value cutoff is set as 0.00028 to identify top 30 genes. Thus, if the p-value cutoff is reduced, then more top ranked genes are identified than the current number of genes; and if it the cutoff is further increased, then less number of top genes are found. Hence, we pick up top 20 DM genes (including top 10 DM_{hyper} and top 10 DM_{hypo} genes) for DS1 (see Table III), and top 30 DE genes (including top 15 DE_{up} and top 15 DE_{down} genes) for DS2 (see Fig. 3(c)). Similarly, we select top 20 DE genes for DS3, and top 20 DM genes for DS4. Thereafter, for comparison purpose, we run $RANWAR$, and the other ARM methods (Apriori et al. (1993), Tao et al. (2003), Yun et al. (WIP) (2006), Sun et al. (2008), Ahmed et al. (2008) and Oguz et al. (2012)) at different minimum support values (viz., 0.1, 0.15, 0.2, ..., 0.5) and a fixed minimum confidence value (i.e., 0.5) on the datasets. Then we count the number of frequent itemsets and corresponding rules for each of the methods, and compare these for the datasets. We observe that $RANWAR$ produces very less number of frequent itemsets than the other existing methods (see Fig. 4). Therefore, $RANWAR$ takes less time to execute than the others (see Fig. 5).



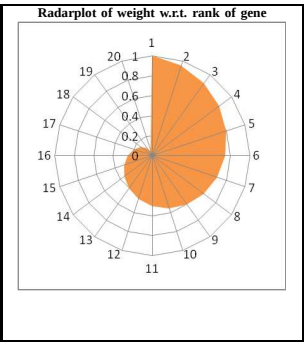
(a) (b) (c)

Fig. 3: Boxplots of data for top 30 DE genes before normalization (in (a)) and after normalization (in (b)), and volcanoplot for identifying DE_{up} and DE_{down} genes, in DS2 (in (c)).

At 0.10 minimum support (min_wsupp) and 0.50 minimum confidence (min_wconf) thresholds, we have obtained 5870 rules in $RANWAR$ whereas we get more than twenty lakh rules in Apriori for DS1. It is noticed that a large number of rules have same confidence and same support for Apriori. But, $RANWAR$ can differentiate them as they have different confidence or different support or both. For example, in Table IV(a), for DS1, rule ids 15 and 16 (highlighted by yellow color) have same confidence (i.e., 99.53% for both) and also same support (i.e., 97.67% for both) using Apriori. For $RANWAR$, these have same wcc (viz., 81.47% for both), but different wcs (viz., 71.59% for rule id 15, and 23.86% for rule id 16). Similarly, rule ids 18 and 19 (highlighted by green color) can be differentiated by their wcs (viz., 34.43% and 11.48%, respectively); and rule ids 21 and 22 can be differentiated by their wcs (viz., 33.92% and 13.57%, respectively). Some top common rules are listed in Table IV(a) and Table IV(b) for DS1 and DS2, respectively. There is another advantage of $RANWAR$ that a large number of rules which have low ranks in Apriori due to their same confidence and support within a wide range, have good ranks in $RANWAR$. For example, (for DS1,) in Table IV(a), the ranks of rule id 1 (viz., {C20orf85+, SLC22A18+, INMT+, ADCY2- $\Rightarrow$ PCDHAC2-}) are 284 in $RANWAR$, and 17166 in Apriori. Similarly, rule ids 2, 3, 4 and 5 stand 378th, 489th, 518th and 520th in $RANWAR$, and 20162, 9639, 38754 and 48072, respectively. (See Table IV.) For details, see "$RANWAR\_sup.pdf$" file.

TABLE III: The p-value, rank, weight (according to eq. 5) and radarplot of weight of top 20 DM genes of DS1.

Gene	P-value in Limma	Rank	Weight
ADCY2	7.47E-91	1	1
C20orf85	1.39E-86	2	0.95
PCDHAC2	4.13E-86	3	0.9
ARID3A	1.38E-80	4	0.85
SLC22A18	5.08E-78	5	0.8
ALX4	1.44E-76	6	0.75
SH3TC1	1.88E-74	7	0.7
SH3BP5	5.93E-74	8	0.65
NR1I2	1.15E-73	9	0.6
P2RY2	1.43E-73	10	0.55
INMT	2.76E-56	11	0.5
CCL14	1.35E-51	12	0.45
MLN	3.34E-50	13	0.4
CCL11	5.82E-50	14	0.35
C14orf115	1.07E-49	15	0.3
EPB42	1.17E-48	16	0.25
CSF1R	1.16E-46	17	0.2
GPR81	9.60E-44	18	0.15
SCNN1D	5.02E-43	19	0.1
PNMA6A	4.36E-40	20	0.05



VII. Utility of ARM in Biology and Novel Findings

ARM is utilized for discovering interesting relations between items in large databases using different measures of interestingness. E.g., a rule $\{\text{butter}, \text{bread}\} \Rightarrow \{\text{milk}\}$ states that if a customer purchases butter and bread together, then he is likely to also purchase milk. In addition to the example of marketing, in current decade, association rules are applied in many domains including continuous production, web usage mining, intrusion detection, and bioinformatics [5]. In bioinformatics, ARM can identify interesting relationships among genes across different experimental and control samples (conditions) from microarray gene expression [3], [4], [7] as well as methylation data [3]. The interesting associations among huge amount of gene mutation helps to identify the reason of mutation in diseases and tumors [3]. Using the rule mining, it is possible to identify sets of co-regulated genes from single-time-point gene expression data. Association rules are applied to obtain transcription factors associated with gene expressions. They are useful for time series datasets, for determining temporal activation or inhibition relations among genes. Besides that, they are utilized for involving multiple motifs (items) and for predicting expression in multiple cell types. They are also helpful for finding the biological data duplicates. In addition to the earlier domains, ant-based ARM is applied to find classification rules for 1 specified class only.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TNB.2014.2359494, IEEE Transactions on NanoBioscience

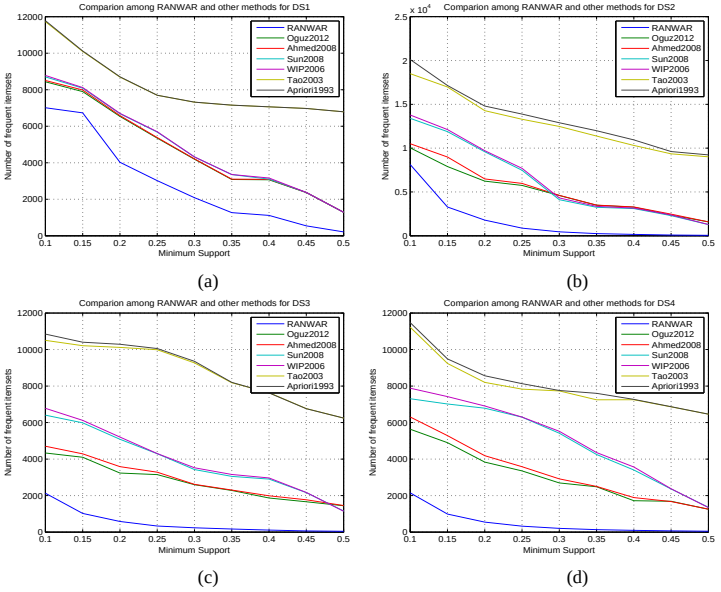


Fig. 4: Comparison of #frequent itemsets between $RANWAR$ and other existing ARM methods at different minimum support.

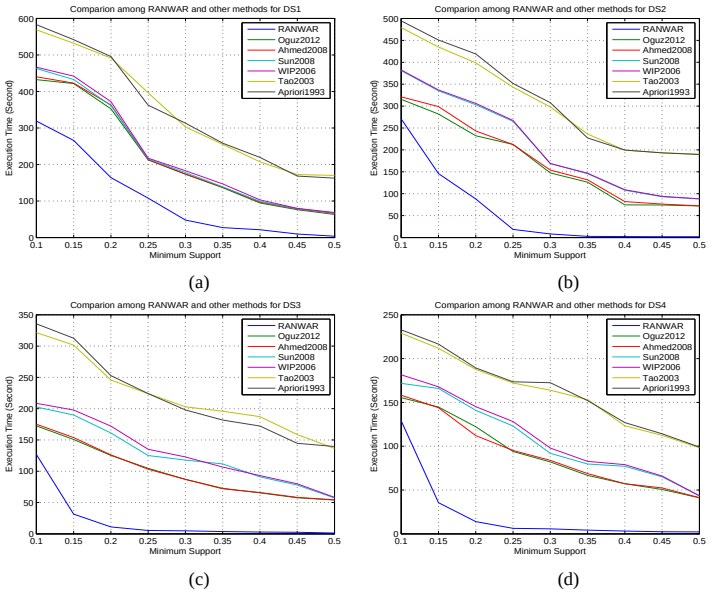


Fig. 5: Comparison of execution time between $RANWAR$ and the other existing ARM methods at different minimum support and a specified minimum confidence cutoff (here, 0.5).

In our experiment, we have performed Gene Ontology (GO) and KEGG pathway analysis on the top rules. If all the genes of any evolved rule involve together in any significant KEGG pathway or Gene-Ontology (GO-term) (whose p-value < 0.05), then the rule is biologically significant. For DS1, rule $\{$CCL11+$\Rightarrow$ CCL14+$\}$ (wcc= 80.63% and wcs= 17.54%) is existing in cytokine-cytokine receptor interaction pathway (viz., hsa04060) which is significantly important for cervical cancer (see [8]). Each of the rules $\{$CCL11+$\Rightarrow$ CCL14+$\}$ (wcc= 80.63% and wcs= 17.54%), $\{$EPB42- $\Rightarrow$ CCL11+$\}$ (wcc= 80.54% and wcs= 24.26%) and $\{$EPB42- $\Rightarrow$ CCL14+$\}$ (wcc= 79.88% and wcs= 30.94%) contain five significant GO-BPs (viz., GO:0050801 ion homeostasis, GO:0048878 chemical homeostasis, GO:0055066 di- and tri-valent inorganic cation homeostasis, GO:0055080 cation homeostasis and GO:0042592 homeostatic process) (see [10]). For DS2, rules $\{$AURKA+$\Rightarrow$ FBXO5-$\}$ (wcc= 87.50% and wcs= 58%) and $\{$CCL11+$\Rightarrow$ CCL14+$\}$ (wcc= 87.50% and wcs= 27.20%) both are existing in Oocyte meiosis pathway (hsa04114) that is important for lung adenocarcinoma (see [9]). In Fig. 7, we plot different combinations of GO-terms against different combinations of rules. From that, we have found that all genes of 3 rules exist in 23 GO:BPs, all genes of 1 rule in 16 GO:BPs etc. The 3 rules are: $\{$TTK-$\Rightarrow$ BUB1-$\}$, $\{$TTK+$\Rightarrow$ CENPF+$\}$ and $\{$CENPF-$\Rightarrow$ TTK-$\}$. All the three rules have same wcc. But, their wcs are different (viz., 36.09%, 25.42% and 16.18%, respectively). Similarly, we have identified that all genes of 1 rule involves in 7 GO:CCs, all genes of 2 rules in 6 Go:CCs etc.; and all genes of 4 rules involves in 8 GO:MFs, all genes of 2 rules in 7 Go:MFs etc. The 4 rules are $\{$AURKA+$\Rightarrow$ PLK4-$\}$, $\{$TTK-$\Rightarrow$ PLK4-$\}$, $\{$TTK-$\Rightarrow$ BUB1-$\}$ and $\{$BUB1-$\Rightarrow$ PLK4-$\}$. Here, some significant GO:BPs are M phase, regulation of cell cycle process etc.; similarly, some significant GO:CCs are spindle, intracellular organelle lumen etc.; and some significant GO:MFs are ATP binding, adenyl nucleotide binding etc. For details, see "RANWAR_sup2.pdf".

Furthermore, $RANWAR$ rules are not totally universal set of traditional Apriori rules. E.g., according to Fig. 6, suppose, there are two cases for rule extraction; viz., case 1: mining rules where min_Supp or min_wsupp is 0.2, and min_Conf or min_wconf is 0.9; and case 2: mining rules where min_Supp or min_wsupp is 0.15, and min_Conf or min_wconf is 0.9. Here, most of the rules generated by $RANWAR$ at case1 exist in the set of the rules generated by Apriori at case 1 as well as the set of the rules generated by Apriori at case 2. *But, there are some limited number of the rules generated by $RANWAR$ at case1 that do not exist in the set of rules generated by Apriori at case 1, but exist in the set of rules generated by Apriori at case 2; i.e.,* $(R_{c1} - A_{c1}) \subseteq R_{c1} \subseteq A_{c2}$, *but* $(R_{c1} - A_{c1}) \nsubseteq A_{c1}$ (all notations are denoted in Fig. 6). For details, see "RANWAR_sup2.pdf".

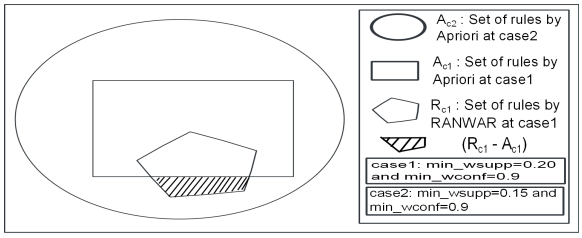


Fig. 6: Venn diagram of rules by $RANWAR$ and Apriori in different cases. $(R_{c1} - A_{c1}) \subseteq R_{c1} \subseteq A_{c2}$, but $(R_{c1} - A_{c1}) \nsubseteq A_{c1}$.

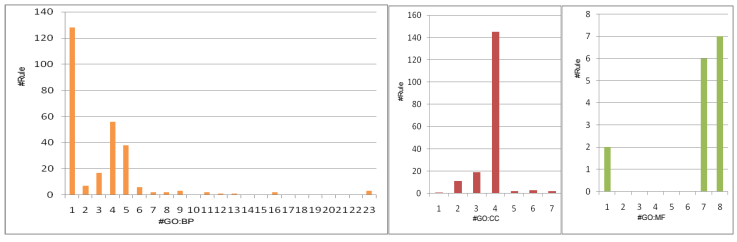


Fig. 7: #Rule existing in #GO:BP (in (a)), #GO:CC (in (b)) and #GO:MF (in (c)), respectively for DS2.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TNB.2014.2359494, IEEE Transactions on NanoBioscience

TABLE IV: Top common rules between $RANWAR$ and Apriori for (a) DS1 and (b) DS2. The highlighted rules in same color are differentiable in ranking by $RANWAR$, not by Apriori. Rk_R and Rk_A denote rank of rule in $RANWAR$ and Apriori, respectively; $Conf$ and $Supp$ denote confidence and support, respectively (where '+' denotes DM_{hyper}/DE_{up} and '-' denotes DM_{hypo}/DE_{down}).

(a)

Id.	Rule	wcc	wcs	Rk_R	Conf	Supp	Rk_A
1	C20orf85+, SLC22A18+, INMT+, ADCY2- ⇒ PCDHAC2-	93.02%	12.72%	284	100%	73.02%	17166
2	C20orf85+, SLC22A18+, CCL14+, ADCY2- ⇒ PCDHAC2-	93.02%	11.37%	378	100%	72.56%	20162
3	SLC22A18+, CCL14+, ADCY2-, SH3TC1- ⇒ PCDHAC2-	93.02%	10.43%	489	100%	74.88%	9639
4	SLC22A18+, INMT+, ADCY2-, SH3TC1- ⇒ PCDHAC2-	92.45%	11.66%	518	99.39%	75.35%	38754
5	C20orf85+, SLC22A18+, CCL14+ ⇒ ADCY2-, PCDHAC2-	92.43%	11.37%	520	99.36%	72.56%	48072
6	{C20orf85+ ⇒ PCDHAC2- }	81.86%	54%	1529	100%	81.86%	2661
7	{CCL14+ ⇒ PCDHAC2- }	81.86%	31.30%	1533	100%	94.88%	22
8	{CCL11+ ⇒ PCDHAC2- }	81.86%	23.74%	1535	100%	92.56%	136
9	{C14orf115+ ⇒ PCDHAC2- }	81.86%	20.45%	1536	100%	93.02%	72
10	{C20orf85+ ⇒ CSF1R+ }	81.86%	18%	1537	100%	81.86%	2659
11	{ARID3A- ⇒ ADCY2- }	81.86%	17.39%	1540	100%	71.16%	25735
12	{C14orf115+ ⇒ CCL14+ }	81.86%	15.34%	1541	100%	93.02%	69
13	{CSF1R+ ⇒ PCDHAC2- }	81.86%	14.59%	1543	100%	99.53%	2
14	{CCL14+ ⇒ CSF1R+ }	81.86%	10.43%	1549	100%	94.88%	20
15	{ADCY2- ⇒ PCDHAC2- }	81.47%	71.59%	1594	99.53%	97.67%	33019
16	{ADCY2- ⇒ CSF1R+ }	81.47%	23.86%	1595	99.53%	97.67%	33018
17	{CCL14+ ⇒ ADCY2- }	81.46%	51.90%	1599	99.51%	94.42%	33027
18	{INMT+ ⇒ PCDHAC2- }	81.46%	34.43%	1600	99.51%	93.95%	33055
19	{INMT+ ⇒ CSF1R+ }	81.46%	11.48%	1601	99.51%	93.95%	33054
20	{SLC22A18+ ⇒ PCDHAC2- }	81.45%	40.70%	1602	99.50%	92.56%	33064
21	{C14orf115+ ⇒ ADCY2- }	81.45%	33.92%	1603	99.50%	92.56%	33065
22	{SLC22A18+ ⇒ CSF1R+ }	81.45%	13.57%	1604	99.50%	92.56%	33063
23	{P2RY2- ⇒ PCDHAC2- }	81.43%	25.77%	1605	99.47%	87.91%	33595
24	{MLN+ ⇒ PCDHAC2- }	81.42%	24.95%	1606	99.46%	85.12%	34045
25	{SH3TC1- ⇒ ADCY2- }	81.38%	77.27%	1636	99.42%	79.07%	35539
26	{SH3TC1- ⇒ PCDHAC2- }	81.38%	46.36%	1637	99.42%	79.07%	35540
27	{SH3TC1- ⇒ CSF1R+ }	81.38%	15.45%	1638	99.42%	79.07%	35538

(b)

Id.	Rule	wcc(%)	wcs(%)	Rk_R	Conf(%)	Supp(%)	Rk_A
1	{KPNA2+ ⇒ MTHFD2+}	87.5	58.67	7	100	100	22781
2	{CKS1B+ ⇒ KPNA2+}	87.5	55.87	14	100	100	22787
3	{CKS1B+ ⇒ MTHFD2+}	87.5	53.33	19	100	100	22786
4	{FBXO5- ⇒ KPNA2+}	87.5	50.29	28	100	100	22785
5	{FBXO5- ⇒ MTHFD2+}	87.5	48	34	100	100	29
6	{CKS1B+ ⇒ FBXO5-}	87.5	45.71	40	100	100	27
7	{HMGB2+ ⇒ KPNA2+}	87.5	43.58	47	100	97.5	25
8	{KPNA2+ ⇒ PLK4-}	87.5	41.90	51	100	100	24
9	{HMGB2+ ⇒ MTHFD2+}	87.5	41.60	56	100	97.5	6
10	{MTHFD2+ ⇒ PLK4-}	87.5	40	61	100	100	33
11	{HMGB2+ ⇒ CKS1B+}	87.5	39.62	65	100	97.5	31
12	{CKS1B+ ⇒ PLK4-}	87.5	38.10	69	100	100	1937
13	{HMGB2+ ⇒ FBXO5-}	87.5	35.66	81	100	97.5	1942
14	{FBXO5- ⇒ PLK4-}	87.5	34.29	85	100	100	1933
15	{HMGB2+ ⇒ PLK4-}	87.5	29.71	99	100	97.5	1935
16	{APOD+ ⇒ KPNA2+}	87.5	28.43	107	100	92.5	1940
17	{APOD+ ⇒ MTHFD2+}	87.5	27.13	115	100	92.5	30
18	{SDC1+ ⇒ KPNA2+}	87.5	26.54	117	100	95	36
19	{APOD+ ⇒ CKS1B+}	87.5	25.84	120	100	92.5	12
20	{SDC1+ ⇒ MTHFD2+}	87.50	25.33	124	100	95	22
21	{KPNA2+ ⇒ MICAL2-}	87.5	25.14	126	100	100	40
22	{SDC1+ ⇒ CKS1B+}	87.5	24.13	129	100	95	5
23	{MICAL2- ⇒ MTHFD2+}	87.5	24	132	100	100	11
24	{APOD+ ⇒ FBXO5-}	87.5	23.26	135	100	92.5	1
25	{CKS1B+ ⇒ MICAL2-}	87.5	22.86	139	100	100	7
26	{IDS- ⇒ KPNA2+}	87.5	22.35	144	100	100	34
27	{SDC1+ ⇒ FBXO5-}	87.5	21.71	150	100	95	20
28	{IDS- ⇒ MTHFD2+}	87.5	21.33	154	100	100	38
29	{FBXO5- ⇒ MICAL2-}	87.5	20.57	159	100	100	15
30	{CKS1B+ ⇒ IDS-}	87.5	20.32	163	100	100	2
31	{APOD+ ⇒ PLK4-}	87.5	19.38	170	100	92.5	17
32	{FBXO5- ⇒ IDS-}	87.5	18.29	175	100	100	26

VIII. Conclusion

The huge number of evolved rules of items (or, genes) by ARM algorithms makes confusion to the decision maker to choose the top genes. Therefore, in this article, we have proposed two novel rank-based weighted condensed rule-interestingness measures (viz., wcs and wcc). A weighted rule-mining algorithm (viz., $RANWAR$) has been developed using the measures specially for microarray/beadchip data. $RANWAR$ uses a statistical test, Limma to compute p-value of each gene (item), and some weight is given to each gene based on their p-value ranking. $RANWAR$ is basically weighted updated form of Apriori. We use two gene expression datasets and two methylation datasets to compare the performance of $RANWAR$ with the state-of-the-art ARM algorithms. $RANWAR$ generates very less number of frequent itemsets than the others. Thus, it saves time of execution of the algorithm. Another advantage of $RANWAR$ is that some most biological significant rules stand top here which hold very low rank in Apriori. The rules are validated by GO-terms and KEGG pathways of genes of the rules. Some top rules extracted from $RANWAR$ that are not present in Apriori, but have high biological significance, are also reported.

IX. Supplementary Materials

The code and supplementary files are available online at: https://www.dropbox.com/sh/jq07k1iujaqm4vv/TIzKK3hzW_

References

[1] R. Agrawal, T. Imielinski and A. Swami, *Mining Association Rules between Sets of Items in large Databases*, Proc. ACM SIGMOD, New York, NY, USA: ACM, pp. 207-216, 1993.

[2] G. Smyth, *Linear Models and Empirical Bayes Methods for Assessing Differential Expression in Microarray Experiments*, Statistical Applications in Genetics and Molecular Biology, vol. 3, no. 1, pp. 3, 2004.

[3] S. Mallik and et al., Integrated Analysis of Gene Expression and Genome-wide DNA Methylation for Tumor Prediction: An Association Rule Mining-based Approach, CIBCB, IEEE SSCI, Singapore, pp. 120-127, 2013.

[4] S. Bandyopadhyay, U. Maulik and J.T.L. Wang, *Analysis of Biological Data: A Soft Computing Approach*, World Scientific, Singapore, 2007.

[5] M. Anandhavalli, M.K.Ghose and K.Gauthaman, *Association Rule Mining in Genomics*, International Journal of Computer Theory and Engineering, vol. 2, no. 2, pp. 1793-8201, 2010.

[6] D. Arthur and S. Vassilvitskii, *k-means++: the advantages of careful seeding*, In Proc. of ACM-SIAM SODA 2007, Society for Industrial and Applied Mathematics Philadelphia, PA, USA, pp. 1027-1035, 2007.

[7] S. Bandyopadhyay and et al., *A Survey and Comparative Study of Statistical Tests for Identifying Differential Expression from Microarray Data*, IEEE/ACM TCBB, 2014, DOI: 10.1109/TCBB.2013.147.

[8] A. Thomas and et al., *Expression profiling of cervical cancers in Indian women at different stages to identify gene signatures during progression of the disease*, Cancer Medicine, 2013, DOI: 10.1002/cam4.152.

[9] J. Liu and et al., *Identifying differentially expressed genes and pathways in two types of non-small cell lung cancer: adenocarcinoma and squamous cell carcinoma*, Genet Mol Res, 2014, DOI: http://dx.doi.org/10.4238/2014.January.8.8.

[10] W. Wei and et al., *The potassium-chloride cotransporter 2 promotes cervical cancer cell migration and invasion by an ion transport-independent mechanism*, J Physiol., 2011, DOI: 10.1113/jphysiol.2011.214635.

[11] J. Pavon, S. Viana and S. Gomez, *Matrix Apriori: Speeding Up the Search for Frequent Patterns*, In Proc. of IASTED, 24th Multi-Conference on Applied Informatics, Innsbruck, Austria, 2006.

[12] Y. Hong and et al., *Incrementally fast updated frequent pattern trees*, Expert Systems with Applications, 2008, DOI: 10.1016/j.eswa.2007.04.009.

[13] D. Oguz and B. Ergenc, *Incremental itemset mining based on matrix apriori algorithm*, Springer Berlin Heidelberg, pp. 192-204, 2012.

[14] S. Orlando and et al., *Enhancing the apriori algorithm for frequent set counting*, In Data Warehousing and Knowledge Discovery, Springer Berlin Heidelberg, pp. 71-82, 2001.

[15] U. Yun and et al., *WIP: mining Weighted Interesting Patterns with a strong weight and/or support affinity*, In SDM, vol. 6, pp. 3477-3499, 2006.

[16] K.M. Yu and J.L. Zhou, *A Weighted Load-Balancing Parallel Apriori Algorithm for Association Rule Mining*, In Proc. of IEEE GrC, 2008, DOI: 10.1109/GRC.2008.4664768.

[17] K. Sun and F. Bai, *Mining Weighted Association Rules without Preassigned Weights*, IEEE TKDE, vol. 20, no. 4, pp. 489-495, 2008, DOI: 10.1109/TKDE.2007.190723.

[18] C.F. Ahmed and et al., *Mining Weighted Frequent Patterns in Incremental Databases*, In PRICAI, pp. 933-938, Springer Berlin Heidelberg, 2008.

[19] F.Tao, *Weighted Association Rule Mining using Weighted Support and Significance Framework*, Proc. ACM SIGKDD, Washington D.C., USA, pp. 661-666, 2003.