Distance-Enhanced Association Rules for Gene Expression

Aleksandar Icev — Dept. of Computer Science

Carolina Ruiz[1] — Dept. of Computer Science

Elizabeth F. Ryder — Dept. of Biology and Biotechnology

Worcester Polytechnic Institute
Worcester, MA 01602 USA
{icev|ruiz|ryder}@wpi.edu

Abstract

We introduce a novel data mining technique for the analysis of gene expression. Gene expression is the effective production of the protein that a gene encodes. We focus on the characterization of the expression patterns of genes based on their promoter regions. The promoter region of a gene contains short sequences called *motifs* to which gene regulatory proteins may bind, thereby controlling when and in which cell types the gene is expressed. Our approach addresses two important aspects of gene expression analysis: (1) Binding of proteins at more than one motif is usually required, and several different types of proteins may need to bind several different types of motifs in order to confer transcriptional specificity. (2) Since proteins controlling transcription may need to interact physically, we know that the order and spacing in which motifs occur can affect expression.

We use association rules to address the combinatorial aspect. The association rules we employ have the ability to involve multiple motifs and to predict expression in multiple cell types. To address the second aspect, we enhance association rules with information about the distances among the motifs, or items, that are present in the rule. Rules of interest are those whose set of motifs *deviates properly*, i.e. set of motifs whose pair-wise distances are highly conserved in the promoter regions where these motifs occur. We describe the design, implementation, and evaluation of our *Distance-based Association Rule Mining* algorithm (DARM) to mine those rules. We show that these distance-based rules achieve higher classification performance than standard association rules over two real datasets.

1. Introduction

Context of this research and problem definition

Control of gene expression remains one of the fundamental unsolved problems of biology. The basic problem is deceptively simple. The primary sequences that control most gene expression (defined here as transcription of DNA into RNA) are known to be located in the non-coding DNA upstream from the coding region. If several genes are expressed in the same temporal and spatial pattern in an organism, then it seems there must be DNA sequences in common among the non-coding regions of these genes that control the timing and location of expression. Although the complete genome sequence for many organisms is now available, most sequences known to be involved in control of transcription have been identified by painstaking molecular and genetic analyses rather than through computational analysis comparing DNA sequences.

There are many reasons for the difficulty in translating knowledge of DNA sequence into understanding of transcriptional control. Molecular analysis has shown that the DNA sequences or *motifs* that control transcription act by allowing the binding of protein transcription factors to non-coding DNA (Figure 1; for a review, see [Whi01]). Motifs tend to be fairly short, and are not always completely conserved among instances. For example, the so-called 'GATA' transcription factors bind the motif (A or T) GATA (A or G). Every occurrence of such short sequences cannot be functional on its own; instead, control of transcription is often combinatorial. Binding of proteins at more than one motif is usually required, and several different types of proteins may need to bind several different types of motifs in order to confer transcriptional specificity. In addition, since proteins controlling transcription may need to

[1] Corresponding author.

interact physically, we know that the order and spacing in which motifs occur can affect expression. So far, however, most software packages that elicit putative motifs involved in the control of transcription identify motifs individually and are not able to consider relationships among motifs (for a review, see [ON01]).

We have used association rules as a computational tool towards combinatorial analysis of motifs involved in transcriptional control [TMPP+01]. As a test database, we have used genes from the simplest multi-cellular animal with a sequenced genome, *C. elegans*, and its close relative, *C. briggsae*. In many cases, the expression patterns of *C. elegans* genes are known to be conferred by relatively short promoter regions (typically 2-4 kb) directly upstream of the protein coding region of the gene. We used an existing software package, MEME [BE95], to elicit putative motifs from these promoter regions. We then built association rules based on these motifs to try to identify combinations of motifs important in controlling transcriptional specificity.

Contributions of this paper

In this work, we take a first step toward including the distances between motifs in the formulation of association rules and introduce an algorithm to mine distance-based association rules efficiently. The values that we use to measure the quality of the rules are the *support*, the *confidence*, and the *coefficient of variation of distances*. This last value is introduced to capture the clustering significance of all pairwise distances of motif members of a rule. Although it is possible for DNA to form loops that allow distant motifs and their associated transcription factors to come into close contact, we have so far considered only linear base pair distances between motifs. Even so, these distance-enhanced models show an improvement in predictive capabilities over models that do not consider distance.

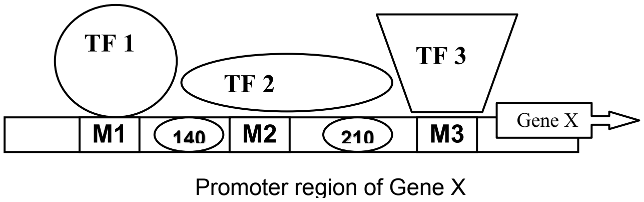

Figure 1. Gene expression. Transcription factors TF1, TF2, and TF3 bind to motifs M1, M2, and M3, respectively, and allow transcription of Gene X to occur. Numbers in ovals represent distances between motifs in base pairs.

2. Distance-based Association Rules

Association rules were introduced in [AIS93]. Association rules follow the form X => Y, where X and Y are disjoint sets of items (or *itemsets*). X is called the *antecedent*, and Y the *consequent* of the rule. The intended meaning of such a rule is that data instances that contain X are likely to contain Y as well. The extent to which the rule applies to a given dataset can be measured using various metrics, including *support* and *confidence*. The *support* of the rule is the prior probability of X and Y occurring together in an instance, Pr(X and Y). The *confidence* of the rule is the conditional probability of Y given X, Pr(Y|X). Here, probability is taken to be the observed frequency in the underlying dataset.

In our prior research [TMPP+01] we have used association rules to describe groups of motifs that when present in the promoter region of a gene make the gene likely to be expressed in the cell type of interest. The following example illustrates what these association rules look like and also points out the need to extend these association rules with distance information among the motifs.

2.1 Motivational Example

Consider the sample dataset shown in Figure 2. This sample consists of 9 data sequences related to 9 different gene promoter regions (PR1-PR9). Each data instance consists of the distinct motifs that are found present in the respective gene promoter region, and the cell type(s) where this gene is expressed (Neural or Muscle). Pairwise distances among the motifs (in DNA base pairs) are also shown in Figure 2.

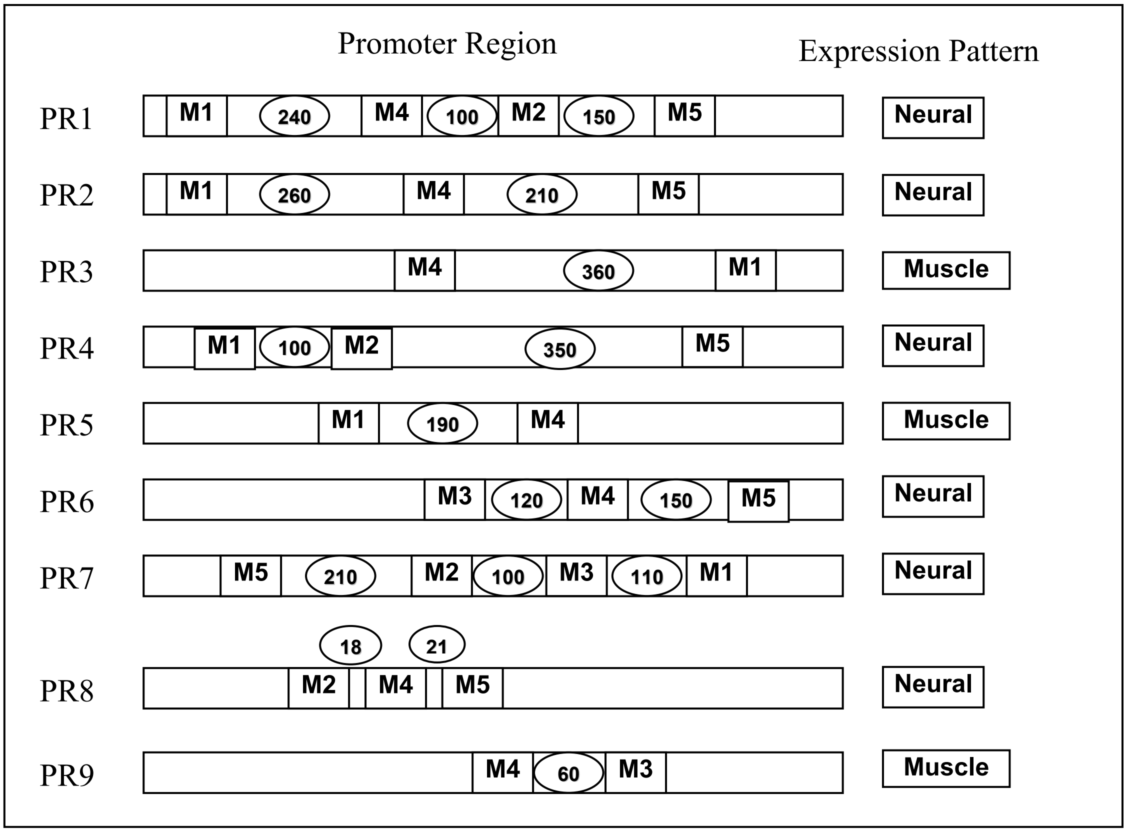
Promoter Region
Expression Pattern
PR1 M1 240 M4 100 M2 150 M5 Neural
PR2 M1 260 M4 210 M5 Neural
PR3 M4 360 M1 Muscle
PR4 M1 100 M2 350 M5 Neural
PR5 M1 190 M4 Muscle
PR6 M3 120 M4 150 M5 Neural
PR7 M5 210 M2 100 M3 110 M1 Neural
18 21
PR8 M2 M4 M5 Neural
PR9 M4 60 M3 Muscle

Figure 2. Motifs and expression paterns of a hypothetical data sample. PRx, promoter region of gene x. Mi, motif i. Numbers in ovals are distances from the start of one motif to the start of the next motif. Not drawn to scale.

Let us assume that we want association rules that have three motifs in the antecedent and one type of cell in the consequent. If the support threshold is (2/9)*100%=22.2% and the confidence threshold is 100%, applying the standard Apriori algorithm [AS94] to this dataset will generate the rules presented in Figure 3.

R1: M1, M2, M5=>Neural (sup=33%), (conf=100%) (M1, M2, M5 & Neural present in PR1, PR4, and PR7)

R2: M1, M4, M5=>Neural (sup =22%), (conf=100%) (M1, M4, M5 & Neural present in PR1, PR2)

R3: M2, M4, M5=>Neural (sup =22%), (conf=100%) (M2, M4, M5 & Neural present in PR1, PR8)

Figure 3. Rules obtained from the dataset shown in Figure 2 using the standard Apriori algorithm

Rules R2 and R3 in Figure 3 have the same values for support and confidence. Based on these measures, no distinction can be made between R2 and R3. However, visual examination of the promoters supporting these rules suggests that there is an important difference. Motifs M1, M4, and M5 are very similarly clustered in promoters PR1 and PR2, which provide the support for rule R2. In contrast, motifs M2, M4, and M5 are in a different order, and are further apart, in promoter PR1 than in PR8, which together provide the support for rule R3. Thus, rule R2 is more likely to be biologically significant than R3. In order to generate rules resembling rule R2, we introduce a new parameter that we call the *coefficient of variation of distances (cvd).* This coefficient will enable the generation of distance-based association rules.

2.2 Coefficient of Variation of Distances

We would expect variability of the distances among motifs to depend upon of the actual sizes of the distances. That is, larger distances would have bigger standard deviations than smaller distances. Thus, to determine whether distances represent similar clustering among promoters we use the coefficient of variation of distances (*cvd*) [Zar99]. The *cvd* of a pair of motifs with respect to a collection (or *itemset*) I of motifs is *the ratio between the standard deviation and the mean of the distances between the motifs in those promoter regions that contain all the motifs in* I. As an illustration, consider R1 from Figure 3. The collection of motifs present in this rule is IR1= {M1, M2, M5}. We augment the statistical information reported for this rule with the *cvd*'s of each pair of motifs present in the rule: cvd_{IR1} (M1,M2), cvd_{IR1} (M1,M5) and cvd_{IR1} (M2,M5). To calculate the *cvd* for the pair M1,M2 with respect to IR1, we note that the distances between M1 and M2 in the promoter regions that contain IR1, namely PR1, PR4, and PR7, are respectively 340, 100, and 210 basepairs. Thus,

$$cvd_{IR1}(M1,M2) = \frac{\sigma_{IR1}(M1,M2)}{\mu_{IR1}(M1,M2)} = \frac{120.14}{216.66} = 0.554$$

In the same manner, we calculate the other two *cvd*'s for R1.

Figure 4 depicts the enhanced versions of rules R1, R2, and R3 from Figure 3.

R1: M1, M2, M5=>Neural (sup=33%, conf=100%)			
		M2	M5
M1	cvd	0.554	0.076
	mean	216.6	462.0
	sdev	120.1	35.0
M2	cvd		0.433
	mean		237.0
	sdev		103.0

R2: M1, M4, M5=>Neural (sup=22%, conf=100%)			
		M4	M5
M1	cvd	0.056	0.036
	mean	250.0	488.0
	sdev	14.0	18.0
M4	cvd		0.136
	mean		233.0
	sdev		31.68

R3: M2, M4, M5=>Neural (sup =22%, conf=100%)			
		M4	M5
M2	cvd	0.982	0.772
	mean	59.0	97.0
	sdev	58.0	75.0
M4	cvd		1.199
	mean		138.0
	sdev		165.0

Figure 4. Distance-Based Association Rules obtained from the Association Rules in Figure 3

In addition to the support and the confidence values, an enhanced association rule contains distance information for each pair of motifs present in the rule. This distance information is given by the *cvd*, the mean, and the standard deviation of the distances between the two motifs of the pair in the set of promoter regions that provides support for the rule. We call such an enhanced rule a *Distance-based Association Rule (DAR).* Note that it would be enough to provide just two out of these three values (as the *cvd* is defined from the mean and the standard deviation) but for clarity we provide the three distance-related values.

Now we can illustrate what we want from the system: rules that satisfy the min support and min confidence thresholds, but also such that items in a rule preserve their distances in the dataset instances that support the rule; i.e. their *cvd*'s are below some maximal allowed threshold- *max-cvd*. The *max-cvd* is given by the user, and *cvd*'s for each pair of items in the rule should be less than the *max-cvd*. So, for the rules given in

Figure 4, if the user of the system sets *max-cvd* threshold to be maximum 0.15, rules 1 and 3 will be removed, while rule 2 will remain, since only for rule 2 are all pairwise *cvd*'s below the given *max-cvd*=0.15.

3. Mining Algorithm

3.1 Mining Task

The mining task can be specified as follows: *Given a dataset of instances D, a minimum support min-supp, a minimum confidence min-conf, and a maximum coefficient of variation of distances max-cdv; Find all distance-based association rules from D whose support and confidence are greater than or equal to the min-supp and min-conf thresholds and such that the cvd's of all the pairs of items in the rule are less than or equal to the maximum cvd threshold.*

3.2 Mining Distance-based Association Rules

Our algorithm to mine distance-based association rules from a dataset of instances extends the Apriori algorithm. The Apriori algorithm [AS94] accepts as inputs two thresholds, *min-supp* and *min-conf*, and mines (finds) all association rules having support and confidence greater than or equal to those thresholds. Apriori mines association rules using a two stage process. The first stage generates all the sets of items that satisfy the *min-supp* constraint, called *frequent itemsets*. The second stage constructs all the association rules that satisfy the *min-conf* constraint from those frequent itemsets.

In order to obtain distance-based association rules, one could use the Apriori algorithm to mine all association rules whose supports and confidences satisfy the thresholds, and then annotate those rules with the *cvd*'s of all the pair of items present in the rule, keeping as the end result of the algorithm, only those rules whose *cvd*'s satisfy the *max-cvd* threshold. We call this algorithm to mine distance-based association rules, *Naïve distance-Apriori.*

This naïve algorithm produces the desired association rules, but it is not particularly efficient in doing so, as it unnecessarily keep frequent itemsets during the first stage of the process that neither them nor their supersets satisfy the *max-cvd* constraint. The *Distance-based Association Rule Mining (DARM)* algorithm that we introduce in this paper prunes from consideration those unnecessary frequent itemsets, making the mining process more efficient. Section 5 presents experimental results that show the time savings of DARM over the naïve algorithm.

3.3 The DARM Algorithm

This algorithm follows Apriori's two stage process: It first generates all the frequent itemsets that satisfy the *max-cvd* constraint (we call them *cvd-frequent* itemsets), and then generates all association rules with the required confidence from those itemsets.

3.3.1 Generating cvd-frequent itemsets

The Apriori algorithm generates frequent itemsets level by level, first listing the collection L_1 of all the frequent itemsets of cardinality one, then the collection L_2 of all the frequent itemsets of cardinality two, and so on. It uses the fact that support monotonically decreases when the cardinality increases. This implies that, if an itemset is infrequent (i.e. its support is below the *min-supp* threshold), then all its supersets are also infrequent. This fact is known as the *Apriori Principle*. Hence, if an itemset is infrequent, it and all of its supersets can be pruned from consideration. In particular, candidate itemsets for L_{k+1} can be generated by joining together itemsets in L_k that differ in only one item.

In contrast with support satisfaction, the *max-cvd* constraint is a non-monotonic property. An itemset that does not satisfy this constraint may have supersets that do. As an illustration, assume that we want to mine association rules from the dataset in Figure 2 with *min-supp*=2/9*100% and *max-cvd=0.15*. Consider the itemset of motifs I={M1,M4}. This itemset is present in PR1, PR2, PR3, and PR5. The distance between the two motifs in those promoter regions are 240, 260, 360, and 190 respectively, and their *cvd* over those 4 promoter regions is cvd_I(M1,M4)=0.27. Hence this itemset I does not satisfy the *max-cvd* condition. However, for the superset J={M1,M4,M5} of I, cvd_J(M1,M4)=0.0564. Note that this superset J is supported by (i.e. contained in) promoter regions PR1 and PR2. This reduction in the set of promoters that supports the itemset makes it possible for the mean and the standard deviation of the distances of a pair of motifs to increase or to decrease, and consequently the *cvd* value either to increase or to decrease.

This example shows the non-monotonic behavior of the cvd values as the cardinality of the itemsets increases. Hence, one cannot remove from consideration an itemset that does not satisfy the *max-cvd* condition. Nevertheless, one can prune an itemset from consideration if it and all of its supersets violate the *max-cvd* condition. Generating *all* the supersets of an itemset to check this condition is very expensive in terms of computational time. We instead introduce a procedure that keeps under consideration only frequent itemsets that *deviate properly*.

Definition (Proper Deviation). Let *n* be the number of promoter regions (instances) in the dataset. Let I be a frequent itemset, and let S be the set of promoter regions that contain I. We say that I *deviates properly* if either:

1. I is *cvd-frequent.* That is, the *cvd* over S of each pair of motifs in I is less than or equal to the *max-cvd* threshold), or
2. For each pair of motifs P in I, there is a subset S' of S with cardinality greater than or equal to $\lceil$*min-supp***n*$\rceil$ such that the *cvd* over S' of P is less than or equal to the *max-cvd* threshold.

The *k*-level of itemsets kept by the DARM algorithm is the collection of frequent itemsets of cardinality *k* that deviate properly. Those itemsets are used to generate the (*k*+1)-level. If a frequent itemset does not deviate properly, it means that no matter what items are added to the itemset in higher levels of the Apriori itemset generation, the resulting superset either will fail to have the *min-supp* required or will contain a pair of items whose *cvd* is above the *max-cvd* allowed. Hence no rules can be generated from this itemset (or any of its supersets) and so the itemset can be removed from consideration.

Note that we do not require that there is one subset S' of S that works for all the pairs of motifs in I. This allows for easier parallel search for the appropriate subsets of S. We further speed up the search for an appropriate subset of S for a pair of motifs in I by sorting S according to the distance between of the pair of motifs in the promoter regions in S and considering only subsets of S formed by contiguous elements on that list. For example, for I={M2,M5} in Figure 2, the sorted version of S (annotated with the distance between the 2 motifs) is {PR8 (d=39), PR1 (d=150), PR7 (d=210), PR4 (d=350)}. One can prove that for each non-contiguous subset of S there is a contiguous subset of S over which the *cvd* value of the pair of motifs is smaller.

3.3.2 Generating rules from cvd-frequent itemsets

Once that all the *frequent* itemsets that deviate properly have been generated, distance-based association rules are constructed from those itemsets that satisfy the *max-cvd* constraint. As is the case with the Apriori algorithm, each possible split of such an itemset into two parts, one for the antecedent and one for the consequent of the rule, is considered. If the rule so formed satisfies the *min-conf* constraint, then the rule is added to the output.

3.4 Implementation

We have implemented our DARM algorithm in Java within the Weka environment [FW99]. This implementation is based on the work reported in [SR03].

4. Predicting Gene Expression using Distance-based Association Rules

In addition to using our distance based association rules to describe combinations of motifs that regulate gene expression, we also employ them to make predictions. For that purpose, we use our DARM algorithm constraining the antecedents of the rules to contain only motifs and the consequents to contain only cell types (see

Figure 4). We call those rules *class distance-based association rules* (or *class rules* for short). Once those rules have been mined, we select a subset of them to form part of a *predictive (or classification) model* as explain below.

4.1 Model Construction

One of the methods we have used to construct models follows the *CBA model* construction approach described in [LHM98]. First, rules are sorted in decreasing order of their confidence, with rules having equal confidence sorted by support. Rules are added to the model one at a time in the sorted order. Only the rules that classify correctly at least one instance from the training data not classified by the rules already in the model are kept. The resulting classifier is tested on the training instances for the error rate (the ratio of incorrect predictions over the training data).This process is repeated until exhausting the association rules or exhausting the training instances. The subset of the rules with lowest error rate is the final CBA model. This CBA model contains a default rule that is applied to test instances for which none of the other rules in the model apply. The default class is the majority class of the unclassified training instances.

4.2 Model Deployment

Given a novel gene (or instance), we apply to it the first rule (in the order in which rules are listed) in the model whose antecedent matches the instance. We say that *the antecedent of a rule matches an instance if all the motifs present in the antecedent of the rule are contained in the instance, and for each pair of those motifs the distance between them in the instance lies in the interval given by the mean plus/minus one standard deviation specified in the rule for that pair of motifs.* As an illustration, consider rule R2 from

Figure 4. If the novel gene contains motifs M1, M4, and M5, and the distances between M1 and M4, M1 and M5, and M4 and M5 lie in the intervals 250 ± 14, 488 ± 18, and 233 ± 32 respectively, we use the rule to predict that the gene is expressed in neural cells. Alternative criteria for rule application are certainly conceivable and worth experimenting with. The criterion described here yielded good experimental performance.

5. Experimental Evaluation

5.1 Data Description

As described in the Introduction, we used two datasets for our experiments. The *C. briggsae* dataset contains the promoter regions of 31 genes, and 5 cell types where the genes are assumed to be expressed in the same pattern as the homologous genes in *C. elegans*: 1. PanNeural (17 out of the 31 genes (54%) are expressed in all neurons); 2. ASENeuron (21 out of 31, 67%); 3. ASKNeuron (24 out of 31, 77%); 4. BodyWallMuscle (20 out of 31, 64%); and 5. OLLNeuron (19 out of 31, 61%), The *C. elegans* dataset contains the promoter regions of 57 genes, and 1 cell type, PanNeural, where 17 out of the 57 genes (29%) are expressed.

We obtained putative motifs for each cell type by running MEME [BE95] over the promoter regions associated with the genes expressed in the cell type. If there was more than one occurrence of a motif in a promoter region, we selected the occurrence of the motif with the lowest p-value. See [MPPT01, BFGMY03] for details on this process and the dataset collection.

5.2 Performance of Distance-based models

We constructed classification models for each cell type in the *C. briggsae* dataset and evaluated the models using several methods. See Figure 5, Figure 6, and Figure 7. The mining parameters for all these experiments are *max-cvd* = 0.5, *min-supp* = 20%, *min-conf* = 20%. The classification accuracies reported in these Figures can be compared against the accuracy of the classifier that always predicts the most frequent value of the classification target. For instance, such a classifier would achieve 54% accuracy for the PanNeural cell type in *C. briggsae* (see the percentages provided above in the data description section) which is considerably lower that those that we achieved for this cell type: 83% in Figure 5; 73% in Figure 6; and 80% in Figure 7.

5.3 Comparison of DARM models and Regular models

We compared the classification accuracy of models constructed from distance-based rules and standard rules. We report here the results obtained for the PanNeural cell type for *C. briggsae* in Figure 8, and for *C. elegans* in Figure 9. In each case, 66% of the data was used for training and 33% for testing. In both cases, our distance-based association rules outperformed the standard association rules.

5.4 Effectiveness of Deviating properly Pruning

Finally, we compared our DARM algorithm against the *Naïve distance-Apriori* described in Section 3. As expected, the naïve approach considers many more unnecessary itemsets than our DARM algorithm does. Figure 10 summarizes the number of itemsets considered by both methods over the five *C. briggsae* datasets with mining parameters *min-supp*=65%, *max-cvd*=0.6. The decrease of the number of the frequent itemsets yields time savings during the mining process.

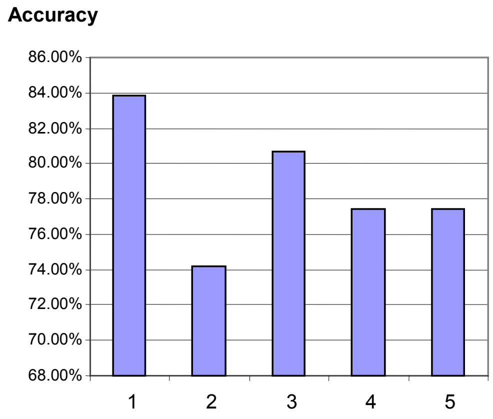	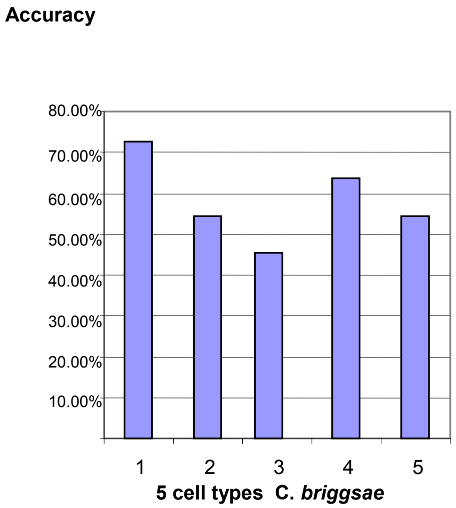
Figure 5. Testing over Training Data	Figure 6. 66% Training, 33% testing data
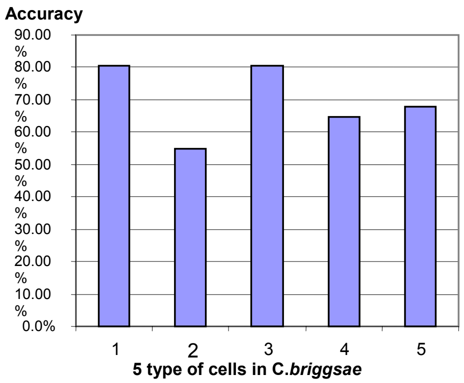	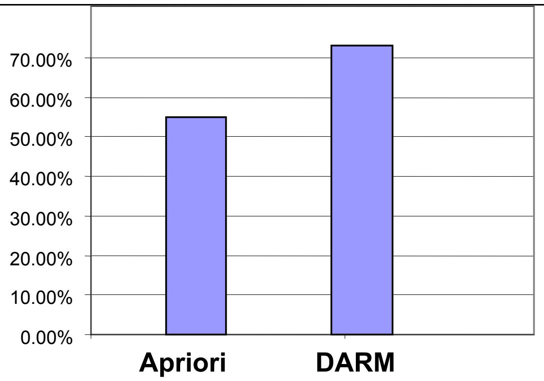
Figure 7. 10-fold cross-validation	Figure 8. Apriori vs. DARM (C. *briggsae*)
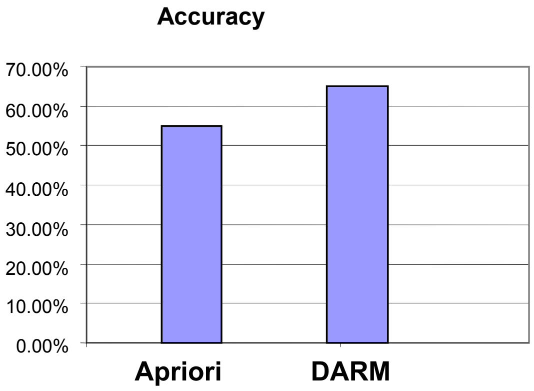	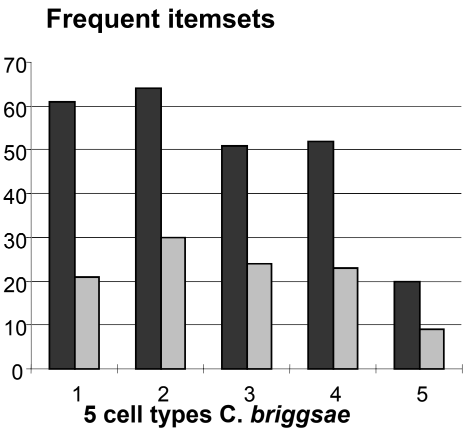
Figure 9. Apriori vs. DARM (C. *Elegans*)	Figure 10. DARM Savings.

6. Conclusions and Future Work

Our experimental results show that the distance-based association rules introduced in this paper achieved higher classification accuracy over gene expression data than standard association rules, and are better descriptors of this application domain. Further experimentation over larger datasets in this and other domains is planned. Also, an extension of the DARM algorithm that allows it to handle multiple occurrences of a motif in a promoter region is underway.

References

[AS94] R.Agrawal and R.Srikant. Fast Algorithms for Mining Association Rules. *In Proc. of the 20-th Very Large DataBases (VLDB) Conference*, pages 487-499, Santiago, Chile, 1994.

[BE95] T.Bailey and C.Elkan. Unsupervised Learning of Multiple Motifs in Biopolymers using Expectation and Maximization. *Machine Learning Journal,21,* pages 51-83, *1995.*

[BFGMY03] J.Baird, J.Farmer, R.Gougian,K.Monterio,P.Young. Motif Analysis of Gene Expression, Undergraduate Graduation Project (MQP), Worcester Polytechnic Institute, 2003.

[BG98] T.Bailey and M.Gribskov, "Combining Evidence using p-values: Application to Sequence Homology Searches", Bioinformatics, 14(48-54), 1998.

[FW99] E.Frank, I.Witten. Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations, Morgan Kaufmann, 1999.

[LHM98] B.Liu, W.Hsu, Y.Ma. Integrating Classification and Association Rule Mining. Proceedings of the Fourth International Conference on Knowledge Discovery and Data Mining , pages 80-86, New York, 1998.

[MPPT01] B.Murphy, D.Phu, I.Pushee, F.Tan. Motif-And Expression-Based Classification of DNA., Undergraduate Graduation Project (MQP), Worcester Polytechnic Institute, 2001.

[ON01] U. Ohler, H. Niemann. Identification and Analysis of Eukaryotic Promoters: Recent Computational Approaches. *TRENDS in Genetics* 17: 56-60. 2001.

[PBCB99] A.G.Pedersen, P.Baldi, Y.Chauvin. S.Brunak. The Biology of Eukaryotic Promoter Prediction - A Review. *Computers&Chemistry 23,* pages 191-207, 1999.

[SR03] C.Shoemaker, C.Ruiz. Association Rule Mining Algorithms for Set-valued Data. In *Proc. 4th Intl. Conf. on Intelligent Data Engineering and Automated Learning.* LNCS, Springer-Verlag. Hong Kong, March 2003.

[TMPP+01] F. Tan, B. Murphy, D. Phu, I. Pushee, C. Ruiz, J. Krushkal, E. Ryder. Identifying Promoter Motifs and Predicting Gene Expression Patterns in C.*elegans* using Data Mining Tools. In *Proc. 13th Intl. C.elegans Conf.* 2001.

[Whi01] R.White. Gene Transcription, Mechanisms and Control, *Blackwell science, 2001.*

[Zar99] J.Zar. Biostatistical Analysis. *Fourth Edition, Prentice hall.*, page 40, 1999.