Unsupervised Dense Regions Discovery in DNA Microarray Data

Andy M. Yip*, Edmond H. Wu**, Michael K. Ng**, and Tony F. Chan*

*Department of Mathematics, University of California,
405 Hilgard Avenue, Los Angeles, CA 90095-1555, USA.
{mhyip,chan}@math.ucla.edu
**Department of Mathematics, The University of Hong Kong
Pokfulam Road, Hong Kong.
hcwu@hkusua.hku.hk,mng@maths.hku.hk

Abstract. In this paper, we introduce the notion of dense regions in DNA microarray data and present algorithms for discovering them. We demonstrate that dense regions are of statistical and biological significance through experiments. A dataset containing gene expression levels of 23 primate brain samples is employed to test our algorithms. Subsets of potential genes distinguishing between species and a subset of samples with potential abnormalities are identified.

1 Introduction

In the analysis of massive microarray data, one typically employ techniques such as clustering, classification, association rule mining, correspondence analysis and/or multi-dimensional scaling plots to understand the structure of the datasets, identify abnormalities and generate a list of interesting genes for further analysis [4]. Besides the patterns discovered by aforementioned methods, we realize that *dense regions*, which are two-dimensional regions defined by subsets of genes and samples whose corresponding values are mostly constant, are another type of patterns that are of practical use and are significant. For example, one may want to find a subset of genes and samples that are co-regulated or are results of errors during preparation of the data.

To the best of the authors' knowledge, dense region patterns have not been previously studied. A similar but not identical notion is error-tolerant frequent itemset introduced by Yang et al in [7] which focuses on mining association rules. In fact, one may expect that the most natural way is to use clustering techniques with the results displayed as a heat map [3] and look for patches of similar color (dense regions). While such a method does allow identification of some of

* The research of this author is partially supported by grants from NSF under contracts DMS-9973341 and ACI-0072112, ONR under contract N00014-02-1-0015 and NIH under contract P20 MH65166.

** The research of this author is supported in part by Hong Kong Research Grants Council Grant Nos. HKU 7130/02P and HKU 7046/03P.

these regions, we find that there are many other significant "patches" that are hidden and are observable only after a suitable permutation of the genes and the samples, whereas cluster analysis only produces a single reordering of the genes and the samples. More importantly, cluster analysis generally uses information from all genes (when clustering samples) and all samples (when clustering genes) whereas dense regions allow a subset of identified genes to behave differently in some samples. The use of subspace clustering is also prohibited because each region depends on a different subspace and it will be computationally intensive to identify all these subspaces. Techniques such as multi-dimensional scaling and correspondence analysis do not emphasize the existence of dense regions as these techniques are not specifically designed for such purposes.

Given the usefulness of dense region patterns in the analysis of microarray data, it is essential to derive effective and efficient algorithms to discover such patterns, which is our goal in this paper. From the computational point of view, our algorithms are also very suitable for microarray data ($\sim$10000 genes and $\sim$100 samples) because the computational cost mostly depends on the smaller dimension of the data matrix which is around $\sim$100 in most studies. Due to the lack of space, we refer the readers to [8] for a thorough treatment on the basic theory of dense regions and justification of the correctness of the algorithms.

2 Definition of Dense Regions

Let X be a given n-by-p data matrix where n, p are the numbers of genes and samples respectively. Denote by $X(R, C)$, or simply $R \times C$, the submatrix of X defined by a subset of rows R and a subset of columns C.

Definition 1 (Dense regions (DRs)). *A submatrix $X(R, C)$ is called a maximal dense region with respect to v, or simply a dense region with respect to v, if $X(R, C)$ is a constant matrix whose entries are v (density), and, any proper superset of $X(R, C)$ is a non-constant matrix (maximality).*

Example 1. Let X be a data matrix given by the first matrix below. The DRs of X with value 1 are given by the four matrices in the brace.

$$\begin{pmatrix} 1&0&0&1\\ 1&1&0&1\\ 1&1&0&1\\ 1&1&2&0 \end{pmatrix}; \quad \left\{ \begin{pmatrix} 1&*&*&*\\ 1&*&*&*\\ 1&*&*&*\\ 1&*&*&* \end{pmatrix}, \begin{pmatrix} 1&*&*&1\\ 1&*&*&1\\ 1&*&*&1\\ *&*&*&* \end{pmatrix}, \begin{pmatrix} *&*&*&*\\ 1&1&*&1\\ 1&1&*&1\\ *&*&*&* \end{pmatrix}, \begin{pmatrix} *&*&*&*\\ 1&1&*&*\\ 1&1&*&*\\ 1&1&*&* \end{pmatrix} \right\}.$$

Definition 2 (μ-Dense regions (μ-DRs)). *A submatrix $X(R, C)$ is called a μ-dense region with respect to v if at least μ percent of the entries of $X(R, C)$ have value v, and, any proper superset of $X(R, C)$ has less than μ percent of entries with value v.*

3 The DRIFT Algorithm

The BasicDRIFT Algorithm. This starts from a given point (s, t) containing the target value v and returns two regions containing (s, t) where the first one

is obtained by a vertical-first-search; the other is by a horizontal-first-search. It is proven in [8, Theorem 1] that the two returned regions are in fact DRs.

Algorithm: BasicDRIFT(X, s, t)

$R_v \leftarrow \{1 \leq i \leq n | X_{it} = X_{st}\}$, $C_v \leftarrow \{1 \leq j \leq p | X_{ij} = X_{it} \forall i \in R_v\}$

$C_h \leftarrow \{1 \leq j \leq p | X_{sj} = X_{st}\}$, $R_h \leftarrow \{1 \leq i \leq n | X_{ij} = X_{sj} \forall j \in C_h\}$

Return $\{R_v \times C_v, R_h \times C_h\}$

To determine the time complexity, we suppose the two resulting DRs have dimensions n_v-by-p_v and n_h-by-p_h respectively. The number of computations required by the algorithm is $n + n_v p + p + p_h n$. Moreover, in practice, n_v, $n_h \ll n$ and p_v, $p_h \ll p$. In this case, the complexity is of $O(n + p)$ essentially.

The ExtendedBasicDRIFT Algorithm. The BasicDRIFT algorithm is very fast but it may miss some DRs. To remedy such a deficiency, we introduce the ExtendedBasicDRIFT algorithm which first obtains the set C_h as in the BasicDRIFT starting at (s, t), and then performs vertical searches over all possible subsets of $C_h \setminus \{t\}$. Non-redundant DRs are returned.

This algorithm returns all DRs containing (s, t). However, since it requires more computations than the BasicDRIFT does, we only invoke it to find DRs that the BasicDRIFT misses. The question now becomes how to combine the two algorithms in an effective way which is the purpose of our next algorithm.

The DRIFT Algorithm. We begin by introducing a key concept, called *isolated point*, which allows us to fully utilize the fast BasicDRIFT algorithm to find as many regions as possible while minimizes the use of the more expensive ExtendedBasicDRIFT algorithm.

Definition 3 (Isolated points). *A point (i, j) in a dense region D is isolated if it is not contained in any other dense region.*

By Theorem 3 in [8], (s, t) is an isolated point iff the two DRs obtained from the BasicDRIFT are identical. Moreover, each isolated point belongs only to one DR, hence, after we record this region, the removal of such a point does not delete any legitimate DR but enhances the search for other DRs by the BasicDRIFT. After we remove all the isolated points recursively, the ExtendedBasicDRIFT is run on the reduced data matrix to find all remaining DRs.

Algorithm: DRIFT(X, v)

Repeat

Start the BasicDRIFT at every point having value v

Record all the regions found that are legitimate DRs

Set the entries in X corresponding to the identified isolated points to be ∞

Until no further isolated point is found

Start the ExtendedBasicDRIFT at every point in the updated X having value v

Record all the regions found that are legitimate DRs

We remark that, a DR is "legitimate" if it is not a subset of any previously found DR. Moreover, one might want to discard DRs with small size. To do so, one may define a DR to be "illegitimate" if its size is below a user-specified threshold and thus it is not inserted into the output sequence.

The μDRIFT Algorithm. This algorithm takes the data matrix and the DRs found by DRIFT as inputs. For each of the input region, this algorithm greedily appends rows and columns to the region while maintains the percentage of the target value to be at least μ. Different starting 100%-DRs may result in the same region, such redundancies are removed from the final output list of μ-DRs.

4 Experimental Results

We employ the dataset consisting of gene expression measurements of 23 primate brain samples (7 human, 8 chimpanzees, 8 Rhesus macaques) studied by Cáceres et al in [1]. Oligonucleotide microarrays were used to measure expression levels of ∼10000 genes simultaneously. The purpose of the study was to explain phenotypic differences between human and chimpanzees at level of gene regulation using macaques are an outgroup, despite the fact that the two species have ∼99% of their DNA sequences in common.

For illustration purposes, a subset of 376 genes is selected based on the coefficient of variation and percentage of present calls generated by the dChip 1.3 software [5]. Next, model-based expression indices are calculated and all replicates are pooled resulting in a dataset with 13 samples and 376 genes. Each gene is then normalized to have mean 0 and standard deviation 1 across the samples. Finally, the values are discretized according to [6] to take integer values.

Example 1. We apply our algorithms to find DRs in the dataset. The results are shown in Fig. 1(a–d). The heat map of the dataset is shown in (a) where the genes and the samples are ordered according to the results of average linkage hierarchical clustering. Three sample DRs identified by μDRIFT are illustrated in (b–d). Moreover, we apply the annotation tool DAVID [2] to find the functional categories of genes in each DR. The region in (b) suggests that most of the genes under the study are down-regulated in the macaque brain samples (Mm1–Mm4). Among the 326 genes in this region, 130 (39.9%) of them are involved metabolism and 96 (29.4%) in cellular physiological process. The region in (c) mostly consists of genes that are down-regulated in the two human samples (Hs1–Hs2) but not other human samples (Hs3–Hs5). The samples Hs1 and Hs2 differ from the other three human samples by (i) they had longer postmortem intervals (∼13 hrs) so that the degradation of the RNA samples may have been more pronounced; (ii) they were collected from a different region (the frontal pole) which may show a different pattern of gene expression than samples collected from other regions. If case (i) is the major reason that causes the deviation of Hs1 and Hs2 from Hs3–Hs5, then one may want to discard Hs1 and Hs2 before any further analysis. Thus, it will be helpful to look at the functional categories of the genes in this region. Indeed, 15 genes (31.9%) are involved metabolism while 14 genes

(29.8%) in cellular physiological process. The region in (d) consists of genes that are consistently up-regulated in the human samples (Hs3–Hs5) but not in other samples. This gives a list of candidate genes to analyze the difference between human and chimpanzees while reducing the effects (i) and (ii) in Hs1 and Hs2 mentioned above. In this region, out of the 59 genes, 30 (50.8%) of them are involved in metabolism and 17 (28.8%) in cellular physiological process.

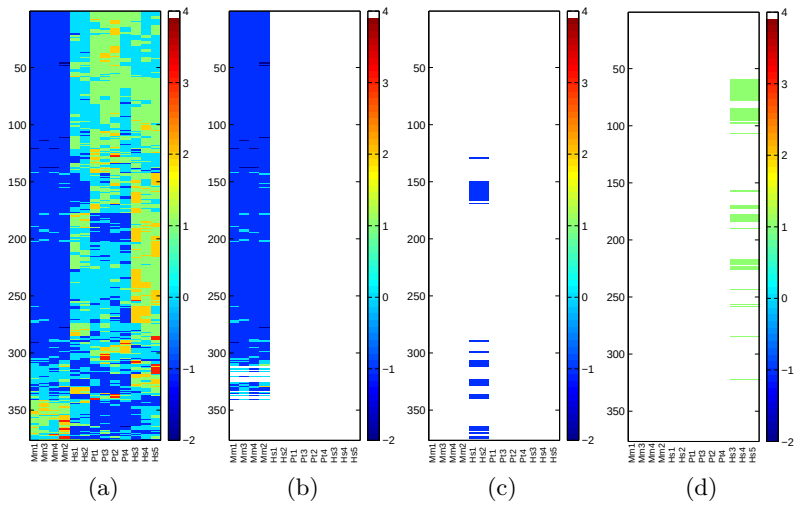
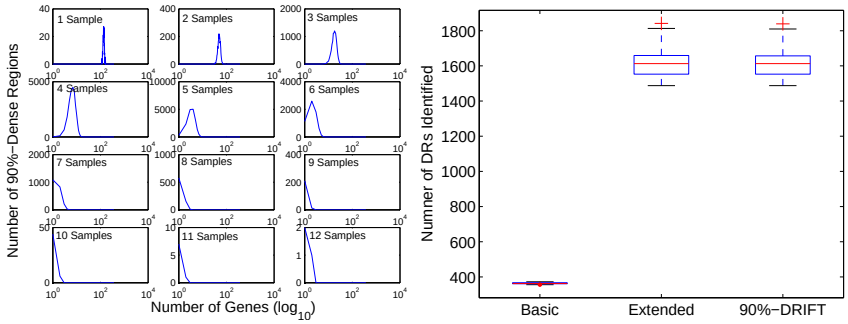


Fig. 1. (a) Heat map (ordered according the results of average linkage hierarchical clustering) of the expression level of 376 genes and 13 samples. (b) A 90%-DR with value −1 (326 genes, 4 samples). (c) A 90%-DR with value −1 (47 genes, 2 samples). (d) A 90%-DR with value 1 (59 genes, 3 samples).

Example 2. This example utilizes permutation tests to estimate statistical significance of DRs. We randomly permute the entries of the original data matrix, apply the μDRIFT with $v = -1, \mu = 90\%$ and record the size of each identified DR. The same procedure is repeated 40 times each with different permutations. The cumulated counts of the size of the DRs are visualized in Fig. 2(a). It can be seen that all of the DRs have very small sizes and that none of the regions have size equal to the ones in Fig. 1(b–c) found from the real dataset. Although we only use a small number of permutations, it is reasonable to expect that the probability of observing 90%-DRs having sizes equal to that in Fig. 1(b–c) from a randomly permuted matrix is extremely low.[1]

[1] One may calculate the exact p-value of each DR and obtain a ranking of the DRs.

Example 3. We evaluate the performance of our algorithms. In Fig. 2(b), the numbers of DRs found by the three algorithms using the dataset in Example 2 are shown as boxplots. We observe that the number of DRs found by BasicDRIFT is around 1/4 of that by the ExtendedBasicDRIFT. Moreover, the μDRIFT does not reduce the number of DRs found by the ExtendedBasicDRIFT at all indicating that most regions are of very small sizes. In contrast, the numbers of DRs found by the three algorithms in Example 1 are 119, 433 and 305 respectively. Thus, the μDRIFT is useful when the data matrix contains DRs of relatively large sizes.



(a) (b)

Fig. 2. (a) Frequency counts of the number of genes in the DRs. Each curve represents the regions with a fixed number of samples. (b) Boxplots of the number of DRs identified by the three algorithms. The data is generated by pooling 40 different permutations of the original data matrix.

Example 4. This example studies the effect of the choice of μ. The data matrix used is the original non-permuted one. In Table 1, we show the number of μ-dense regions found for various values of μ and $v = \pm 1$. The number of dense regions decreases as μ decreases. This is because many regions of small sizes are merged to form large regions. We empirically found that it is the most effective to use $\mu = 90\%$ followed by a filtering (at least 30 genes and 2 samples) to obtain a small list of most significant dense regions from which meaningful regions can be identified.

v	Basic	Extended	10%	20%	30%	40%	50%	60%	70%	80%	90%
−1	119	433	1	1	18	40	46	70	110	163	305
1	170	316	3	14	18	27	43	89	153	208	312

Table 1. Number of dense regions found by the BasicDRIFT, the ExtendedBasicDRIFT, and the μDRIFT with various values of μ.

5 Conclusion and Future Work

In this paper, we introduce the notion of dense regions in microarray data and present effective and efficient algorithms for discovering them. Such patterns are very natural that microarray users may want to look at in the beginning of high-level analysis. Moreover, traditional unsupervised or statistical methods fail to identify these patterns. We demonstrate the usefulness of our algorithms on a dataset with 23 primate brain tissue samples where several biologically interesting dense regions are discovered. We also employ permutation tests to assess statistical significance of dense regions where the particular three regions illustrated are very significant. Empirical studies of the behavior of the algorithms and the choice of the parameter μ are also given. We remark that although we use the matrix of expression values as input to our algorithms in our examples, it is equally-well to apply transformed data as input. For example, one may transform the matrix to a 0/1 matrix depending whether the corresponding gene is up or down-regulated in the sample. Thus, our method can be applied to answer a wider range of queries. We conclude that dense regions are very useful patterns in microarray data that many insights on the dataset can be gained by examining them. As a future work, we would like to incorporate genes' functional categories to enrich the information of the dense regions.

References

1. M. Cáceres et al, *Elevated gene expression levels distinguish human from non-human primate brains*, PNAS, 100, pp.13030–13035, 2003.
2. G. Dennis et al, *DAVID: database for annotation, visualization, and integrated discovery*, Genome Biology, 4(5):P3, 2003.
3. M. B. Eisen, P. T. Spellman, P. O Brown, and D. Botstein, *Cluster analysis and display of genome-wide expression patterns*, PNAS, 85, pp.14863–14868, 1998.
4. M. T. Lee, *Analysis of microarray gene expression data*, Kluwer Academic Publishers, 2004.
5. C. Li and W. H. Wong, *Model-based analysis of oligonecleotide arrays: expression index computation and outlier detection*, PNAS, 98, pp.31–pp.36, 2001.
6. E. H. Wu, M. K. Ng, A. M. Yip, and T. F. Chan, *Discretization of multidimensional web data for informative dense regions discovery*, submitted to 5-th Int. Conf. on Web-age Information Management, 2004.
7. C. Yang, U. Fayyad, and P. S. Bradley, *Efficient discovery of error-tolerant frequent itemsets in high dimensions*, Proc. of ACM SIGKDD Intl. Conf. on Knowledge Discovery and Data Mining: San Francisco, California, pp. 194–203, 2001.
8. A. M. Yip, E. H. Wu, M. K. Ng, and T. F. Chan, *An efficient algorithm for dense regions discovery from large-scale data streams* (extended version), UCLA CAM Reports 03-76, Math. Dept., University of California, Los Angeles, CA, 2003.