
©2009 IEEE. Personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution to servers or lists, or to reuse any copyrighted component of this work in other works must be obtained from the IEEE.
Data Pre-Processing for More Effective Gene Clustering

Jingyu Hou, Yi-Ping Phoebe Chen
School of Engineering and Information Technology
Deakin University
221 Burwood Highway, VIC 3125, Australia
{jingyu, phoebe@deakin.edu.au}

Abstract

The high-throughput experimental data from the new gene microarray technology has spurred numerous efforts to find effective ways of processing microarray data for revealing real biological relationships among genes. This work proposes an innovative data pre-processing approach to identify noise data in the data sets and eliminate or reduce the impact of the noise data on gene clustering. With the proposed algorithm, the pre-processed data sets make the clustering results stable across clustering algorithms with different similarity metrics, the important information of genes and features is kept, and the clustering quality is improved. The primary evaluation on real microarray data sets has shown the effectiveness of the proposed algorithm.

1. Introduction

Microarray technology has come into widespread use to allow the monitoring and comparing massive gene expression levels in a single hybridization experiment. It has emerged as a powerful molecular genetic tool for biomedical research [3]. A typical microarray data set includes expression levels for thousands of genes across hundreds of conditions. The massive quantity of the microarray data has spurred numerous efforts to find robust statistical and computational approaches for processing microarray data, drawing genomic-scale conclusions and generating new hypotheses [5] [13].

Among various approaches, clustering is a key step in the analysis of gene expression data. Gene clustering, which is also known as unsupervised pattern recognition, is to identify gene groups that show similar expression patterns over a wide range of experiments and to discover possible functional relationships among genes. However, microarray data is almost always described as containing large measurement noise or high variability. Although various algorithm have been proposed to filter and normalize the experimental microarray data [16][1], it is inevitable that the filtered and normalized microarray data still contain noises that have impact on the clustering results if the noises can not be dealt with properly.

To eliminate or reduce the impact of the microarray noise data on clustering results, many approaches have been proposed [4] [8] [10] [11] [12] [18]. These approaches could be roughly categorized into three types: gene selection, feature reduction and clustering on feature subsets. However, whatever the existing approaches we used, the genes or features that are to be eliminated from the data set were dealt with as a whole, i.e. the whole profile of the gene or the feature were eliminated. This treatment actually hid the individual performance evidences of the genes or features, i.e. the important but weak signals in the data set were thrown away. It is reasonable that some genes might express significantly under some conditions but insignificantly under other conditions. Simply removing these genes from the data set based on their statistical expression information across all conditions will lead to the elimination of the genes that might be important. On the other hand, statistical or other models that are used for removing genes and feature are based on some kinds of assumptions or predetermined performance patterns that might not match the real situations of the biological performance.

Instead of eliminating or removing genes or features from a microarray data set, in this work, we propose a new approach to pre-process the microarray data for clustering genes more effectively. This approach aims to adjust gene express values to more reasonably reflect the actual performance of individual genes under individual conditions, and to eliminate or reduce the impact of noise values on gene clustering. The adjustment to be made on each expression value is based on the expression value evaluation of the
corresponding gene across all conditions, as well as the expression value evaluation of the gene against all other peer genes. Therefore the adjustment pre-processes the microarray data without losing important information of genes and features in clustering.

The paper is organized as follows. In section 2, we present the proposed data pre-processing algorithm. Section 3 gives the primary evaluation results of the algorithm on real gene data sets. The conclusions and future work are presented in section 4.

2. Data pre-processing algorithm

The algorithm is based on a gene-condition correlation matrix which is written as $CM = [CM_{ij}]_{m \times n}$ where $m$ is the number of genes, and $n$ is the number of conditions. The value of $CM_{ij}$ is the expression value of gene $i$ under the condition $j$. To evaluate the expression values of the genes, we define the following values:

$$row\_avg[i] = \left( \sum_{j=1}^{n} CM_{ij} - \min_{i=1}^{n} - \max_{j=1}^{n} \right) / (n-2), \quad i = 1, \ldots, m,$$

$$column\_avg[j] = \left( \sum_{i=1}^{m} CM_{ij} - \min_{j=1}^{n} - \max_{j=1}^{n} \right) / (m-2), \quad j = 1, \ldots, n,$$

where $\min_{i=1}^{n} = \min_{j=1}^{n} (CM_{ij})$, $\max_{j=1}^{n} = \max_{j=1}^{n} (CM_{ij})$, and $\min_{j=1}^{n} = \min_{j=1}^{n} (CM_{ij})$, $\max_{j=1}^{n} = \max_{j=1}^{n} (CM_{ij})$.

The $row\_avg[i]$ excludes the minimum and maximum expression values of the gene $i$ to avoid the cases where some abnormal expression values dominate the calculation. The situation is the same for the definition of $column\_avg[j]$.

Based on the above defined values, the data pre-processing algorithm is described as the following pseudo code:

For $i = 1$ to $m$

For $j = 1$ to $n$

If ($CM_{ij} < row\_avg[i]$ and $CM_{ij} < column\_avg[j]$)

Then $CM_{ij} = 0$;

End

End

It is worth emphasizing that when identifying noise data, two conditions, $CM_{ij} < row\_avg[i]$ and $CM_{ij} < column\_avg[j]$, should be satisfied simultaneously in the above algorithm. Actually if the first condition, $CM_{ij} < row\_avg[i]$, is satisfied, it means the expression level of gene $i$ under the condition $j$ is below the average level across all conditions. This value provides less information in terms of conditions and might be a noise. Whether or not this value is a noise is to be further examined by the second condition. If the second condition $CM_{ij} < column\_avg[j]$ is satisfied as well, it means this value is still lower among all genes under the condition $j$. At this time, the value is likely to be a noise as it does provide less information in terms of conditions and genes.

The reason why the noise expression values are to be set to zeros is that most gene clustering algorithms are based on gene similarity or distance measures. Setting the noise values to zeros will eliminate or reduce the impact of the noise data on similarity or distance computation, and in turn to improve the clustering quality.

3. Evaluations

To evaluate the effectiveness of the algorithm in improving gene clustering results, we applied the algorithm to a publicly published yeast gene microarray data set [14]. Within the data set, each yeast gene contained 82 mRNA abundance values under 82 different experimental conditions. For evaluation purposes, we set up six sub data sets from this data set without any pre-defined selection criteria. The sizes of these data sets (i.e. the number of genes in the data sets) were 40, 250, 534, 1151, 1352 and 2001. The corresponding data sets were named G-40, G-250, G-534, G-1151, G-1352 and G-2001.

We chose two non-parametric clustering algorithms for evaluation. These two algorithms do not need any pre-determined thresholds for clustering. One of them is the Non-Parametric Global Gene Clustering (NPGGC) algorithm which is a hierarchical clustering algorithm [7], another one is the Correlation Search Technique (CST) algorithm [15] which is a non-hierarchical and $k$-means based algorithm. Meanwhile, we chose five commonly used similarity metrics for these two algorithms. For simplicity and convenience, here we use the same notations and definitions as those in section 2.

The first similarity metric, named $Metric\_1$, is the cosine measure, i.e. the similarity between genes $i$ and $j$, $SM_{ij}$, is defined as

$$SM_{ij} = \frac{\sum_{k=1}^{n} CM_{ik} \cdot CM_{jk}}{\sqrt{\sum_{k=1}^{n} CM_{ik}^2 \cdot \sum_{k=1}^{n} CM_{jk}^2}}$$

The second similarity metric, named $Metric\_2$, is the normalized cosine measure, i.e. the similarity between genes $i$ and $j$, $SM_{ij}$, is defined as

$$SM_{ij} = \frac{1}{2} \left( \sqrt{\sum_{k=1}^{n} CM_{ik}^2 \cdot \sum_{k=1}^{n} CM_{jk}^2} + 1 \right)$$

The third similarity metric, named Metric-3, is the Pearson’s correlation coefficient, i.e. the similarity between genes \(i\) and \(j\), \(SM_{ij}\), is defined as
\[
SM_{ij} = \frac{\sum_{k=1}^{n} (CM_{ik} - \bar{M}_i)(CM_{jk} - \bar{M}_j)}{\sqrt{\sum_{k=1}^{n} (CM_{ik} - \bar{M}_i)^2} \sqrt{\sum_{k=1}^{n} (CM_{jk} - \bar{M}_j)^2}}
\]
where \(\bar{M}_i\) and \(\bar{M}_j\) are the expression means of the genes \(i\) and \(j\) respectively. The fourth similarity metric, named Metric-4, is the absolute value of Pearson’s correlation coefficient, i.e. the similarity between genes \(i\) and \(j\), \(SM_{ij}\), is defined as
\[
SM_{ij} = \frac{|\sum_{k=1}^{n} (CM_{ik} - \bar{M}_i)(CM_{jk} - \bar{M}_j)|}{\sqrt{\sum_{k=1}^{n} (CM_{ik} - \bar{M}_i)^2} \sqrt{\sum_{k=1}^{n} (CM_{jk} - \bar{M}_j)^2}}
\]
The last similarity metric, named Metric-5, is the Euclidean distance based similarity, i.e. the similarity between genes \(i\) and \(j\), \(SM_{ij}\), is defined as
\[
SM_{ij} = e^{-d^2}, \quad \text{where} \quad d = \left[\frac{\sum_{k=1}^{n} (CM_{ik} - CM_{jk})^2}{n}\right]^{\frac{1}{2}}
\]
We used the Huberts’ \(\Gamma\) statistic index [6] to measure the clustering quality. The value of \(\Gamma\) is within the range \([-1, 1]\). A higher \(\Gamma\) value indicates the clustering quality is better.

Based on the above evaluation settings and definitions, we present the evaluation results of algorithms NPGGC and CST in tables 1 and 2 respectively. In each table, the Huberts’ \(\Gamma\) statistic index values of the clustering algorithm on the raw data sets and pre-processed (new) data sets are presented.

It was observed from Table 1 that for the hierarchical clustering algorithm NPGGC, most new results were better than the raw ones. It was worth noticing that the results of Euclidean distance (i.e. Metric-5) based clustering were improved significantly after the data sets were pre-processed by the algorithm.

It was obvious from Table 2 that for the \(k\)-means based non-hierarchical clustering algorithm CST, in most cases the new clustering results were as good as the raw ones. Similar to the performance of the NPGGC algorithm, the pre-processing algorithm also significantly improved the Euclidean distance based clustering results of the CST algorithm.

The above primary evaluation results showed that the data pre-processing algorithm was effective in eliminating or reducing the impact of the noise data on clustering with clustering quality improvement or without quality degradation, no matter what similarities were used for clustering. This implies that the pre-processing algorithm kept the important information of the genes and features while it eliminated the noise values.

### 4. Conclusions

This paper proposed an innovative data pre-processing algorithm for improving gene clustering results. The algorithm takes into account the gene profiles and feature profiles at the same time when identifying noise data. The algorithm pre-possesses the data sets without losing the importance information of genes and features. Furthermore, the pre-processed

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Metric-1</td>
<td>0.38</td>
<td>0.37</td>
<td>0.45</td>
<td>0.51</td>
<td>0.47</td>
<td>0.60</td>
<td>0.59</td>
<td>0.59</td>
</tr>
<tr>
<td>Metric-2</td>
<td>0.30</td>
<td>0.50</td>
<td>0.43</td>
<td>0.46</td>
<td>0.58</td>
<td>0.62</td>
<td>0.63</td>
<td>0.63</td>
</tr>
<tr>
<td>Metric-3</td>
<td>0.47</td>
<td>0.70</td>
<td>0.46</td>
<td>0.49</td>
<td>0.49</td>
<td>0.58</td>
<td>0.58</td>
<td>0.60</td>
</tr>
<tr>
<td>Metric-4</td>
<td>0.34</td>
<td>0.44</td>
<td>0.48</td>
<td>0.48</td>
<td>0.48</td>
<td>0.63</td>
<td>0.71</td>
<td>0.75</td>
</tr>
<tr>
<td>Metric-5</td>
<td>0.46</td>
<td>0.47</td>
<td>0.06</td>
<td>0.61</td>
<td>0.03</td>
<td>0.73</td>
<td>0.03</td>
<td>0.88</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Metric-1</td>
<td>0.48</td>
<td>0.49</td>
<td>0.53</td>
<td>0.53</td>
<td>0.54</td>
<td>0.56</td>
<td>0.54</td>
<td>0.55</td>
<td>0.55</td>
<td>0.56</td>
</tr>
<tr>
<td>Metric-2</td>
<td>0.50</td>
<td>0.51</td>
<td>0.53</td>
<td>0.53</td>
<td>0.54</td>
<td>0.56</td>
<td>0.54</td>
<td>0.55</td>
<td>0.55</td>
<td>0.56</td>
</tr>
<tr>
<td>Metric-3</td>
<td>0.48</td>
<td>0.52</td>
<td>0.51</td>
<td>0.51</td>
<td>0.54</td>
<td>0.55</td>
<td>0.54</td>
<td>0.54</td>
<td>0.53</td>
<td>0.54</td>
</tr>
<tr>
<td>Metric-4</td>
<td>0.52</td>
<td>0.53</td>
<td>0.54</td>
<td>0.54</td>
<td>0.55</td>
<td>0.56</td>
<td>0.55</td>
<td>0.56</td>
<td>0.55</td>
<td>0.56</td>
</tr>
<tr>
<td>Metric-5</td>
<td>0.61</td>
<td>0.59</td>
<td>0.76</td>
<td>0.83</td>
<td>0.54</td>
<td>0.90</td>
<td>0.51</td>
<td>0.90</td>
<td>0.50</td>
<td>0.90</td>
</tr>
</tbody>
</table>
data sets make the clustering results stable across clustering algorithms with different similarity metrics, which is important and practical in applications. The primary evaluation showed the effectiveness of the algorithm.

At the moment, our algorithm is to set the noise values of the raw data sets to zeros based on the idea of eliminating or reducing the impact of noise data on similarity computation, and in turn on clustering. Further research could be done to investigate if other ways of dealing with the identified noise data could be more effective. On the other hand, the conditions that are used to identify possible noise data are average expression values across features and genes (i.e. the average values of feature and gene profiles), other statistic measures could also be examined for more effectively identifying noise data. Another future work is to compare the clustering results on the data sets that are pre-processed by our algorithm with the results on the data sets that are pre-processed by the existing algorithms which eliminate genes or features from the original data sets.

References


