

奇异向量空间双聚类算法

徐晓华, 席艳秋, 潘舟金, 陆 林, 陈 峻

(扬州大学 信息工程学院, 江苏 扬州 225009)

摘 要: 本文针对 0/1 矩阵的双聚类问题提出一种奇异向量空间双聚类算法. 通过 SVD 分解将 0/1 矩阵映射到左右奇异向量空间上, 然后利用信息熵判断行聚类优先还是列聚类优先, 最后根据判断结果递归进行行聚类或列聚类, 直到满足停止条件. 实验显示奇异向量空间双聚类算法可以分辨出完全无重叠的子矩阵, 比较快速地得到硬的双簇.

关键词: SVD 分解; 0/1 矩阵; 行聚类; 列聚类; 布尔矩阵

中图分类号: TP181

文献标识码: A

文章编号: 1000-7180(2012)03-0078-06

Singular Vector Space Biclustering Algorithm

XU Xiao-hua, XI Yan-qiu, PAN Zhou-jin, LU Lin, CHEN Leng

(Department of Computer Science, Yangzhou University, Yangzhou 225009, China)

Abstract: In this paper, we propose a singular vector space biclustering algorithm for 0/1 binary clustering problems. At first, we compute the left and right singular vector matrices from the input 0/1 matrix, then the priority between the column-clustering and the row-clustering is determined based on information entropy. Finally, the column-clustering and the row-clustering are repeated iteratively until the stopping criterion is satisfied. Experimental results show that our algorithm can identify those completely non-overlapping sub-matrices and find the “hard” biclusters more efficiently.

Key words: singular value decomposition; 0/1 matrix; row-clustering; column-clustering; Boolean matrix

1 引言

生物学上在各种条件及各个样本下基因的表达水平显示出不同的数值效果, 但就其功能来说大概会发生两种情况: 基因在某个条件下发生了反应或在某条件下并未发生反应. 而 0/1 矩阵中只包含两种值(0 或 1), 正好符合基因在条件下的反应情况. 现今多种数据挖掘技术被提出用来支持挖掘大型 0/1 数据集, 而在探索性数据分析过程中经常使用聚类技术, 由此可以利用聚类算法来分析 0/1 数据集. 0/1 基因矩阵可用关系矩阵 R 表示, 如表 1 中基因 g_1 仅仅在条件 $t_3, t_6, t_7, t_8, t_9, t_{10}$ 下发生反应.

奇异值分解是一个有着很明显的物理意义的一

种方法, 它可以将一个比较复杂的矩阵用一些更小更简单的子矩阵的相乘来表示, 这些小矩阵描述的是该矩阵的重要特性. 本文利用 SVD 分解布尔矩阵, 选取能量主要集中在区域, 此方法可以去除一定程度的噪声数据. 文中利用信息熵来判断行聚类还是列聚类优先, 迭代使用行聚类或列聚类以得到 0/1 双聚类, 迭代终止的条件为子矩阵块中 1 元素的数量满足阈值. 本文可以分辨出哪些是完全无重叠的子矩阵, 哪些是有重叠的子矩阵, 并可以挖掘出无重叠的子矩阵. 对于那些在有交叠区域的 0/1 矩阵, 利用相似性差别的大小来划分行集与列集, 由此可以去除掉背景元素 0, 得到全 1 矩阵; 或根据相似性差别的阈值设定来得到满足阈值要求的双聚类.

收稿日期: 2011-07-12; **修回日期:** 2011-08-15

基金项目: 国家自然科学基金(61003180, 61070047); 江苏省科技厅自然科学基金(BK2010318); 江苏省教育厅自然科学基金(09KJB20013)

2 问题定义

基因集合 G 和条件集合 C , 比如表 1 中的 $G = \{g_1, g_2, \dots, g_5\}$ 和 $C = \{t_1, t_2, \dots, t_{10}\}$. 在布尔背景下需要挖掘 $r \subseteq G \times C$, 其中基因集合 $G = \{g_1, g_2, \dots, g_m\}$, $C = \{t_1, t_2, \dots, t_n\}$ 为一系列基因在各条件下的布尔属性. $(g_i, t_j) \in r$ 表明第 j 个条件支持第 i 条基因, 即第 i 条基因在第 j 个条件下发生反应, 此时值为 1.

表 1 布尔矩阵 R

	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9	t_{10}
g_1	1	1	1	1	0	1	1	0	0	0
g_2	1	1	1	1	0	0	0	0	1	1
g_3	1	1	1	1	0	0	0	0	1	1
g_4	0	0	0	0	1	1	1	1	1	1
g_5	1	0	1	0	1	1	1	1	0	0

对于一个典型的布尔型基因表达数据集, G 为基因集合, 条件集合 C 为对于 G 的反应. 矩阵记录了基因在各种条件下的表现情况, 比如过表达 (over-expressed) 的生物现象, 还可以考虑另外一些表现特性 (强烈变化, 低表达 under-expressed). 例如: 在表 1 中可以说基因 g_4, g_5 在条件 t_5, t_6 下过表达.

定义 1 1 频度 (Frequency of 1)^[1]

一个 bi-set 二值集合 (G, C) , $G = \{g_1, g_2, \dots, g_m\}$, $C = \{t_1, t_2, \dots, t_n\}$, $(g_i, t_j) = 1$, 表明第 i 条基因在第 j 个条件下发生反应. 1 频度 (Frequency of 1) 为 $f(G, C) = \text{size}(\{(g_i, t_j) = 1\}) / \text{size}((G, C))$, 其中 $\text{size}(\{(g_i, t_j) = 1\})$ 为集合 (G, C) 中 1 的个数, $\text{size}((G, C)) = m \times n$.

例如: 表 1 中 $f(R) = 30 / (5 \times 10) = 60\%$.

定义 2 1-矩阵和 0-矩阵

一个 bi-set 二值集合 (G, C) 为 1-矩阵当且仅当 $\forall g \in G$ 和 $\forall t \in C, (g, t) \in r$, 一个 bi-set 二值集合 (G, C) 为 0-矩阵当且仅当 $\forall g \in G$ 和 $\forall t \in C, (g, t) \notin r$.

定义 3 概念 (Concept)^[2]

若 0/1 基因矩阵 $R_{m \times n}$, $(G, C) \in 2^{\{1, \dots, m\}} \times 2^{\{1, \dots, n\}}$ 叫做一个概念 (Concept), 则当且仅当 (1) $\forall g \in G, \forall t \in C: (g, t) \in r$ 并且 (2) 不存在 $(G', C') \in 2^{\{1, \dots, m\}} \times 2^{\{1, \dots, n\}}$ 其中 (i) $\forall g' \in G, \forall t' \in C: (g', t') \in r$ 且 (ii) $G \subseteq G' \wedge C \subseteq C' \wedge (G', C') \neq (G, C)$. 一个概念也称为一个最大 1-矩阵.

例如: 表 1 中 $(\{g_1, g_2, g_3\}, \{t_1, t_2\})$ 是一个 1-矩

阵, 但却不是一个概念 (最大 1-矩阵). 在布尔矩阵 R 中存在两个概念 $(\{g_2, g_3\}, \{t_1, t_2, t_3, t_4, t_9, t_{10}\})$ 与 $(\{g_1, g_2, g_3, g_5\}, \{t_1, t_3\})$.

定义 4 频度概念 (Frequent Concept)^[3]

一个概念 (G, C) , 它的 1 频度 (Frequency of 1) 若满足 $f(G, C) \geq \sigma$ ($0 < \sigma \leq 1$), 其中 σ 为用户设定的阈值, 则称此概念为一个频度概念 (Frequent Concept).

通过以上的一些定义, 知道在 0/1 基因矩阵中进行聚类分析, 其主要的工作已转化为找出满足要求的频率概念, 也即是找出满足阈值要求的 1-矩阵.

3 奇异值向量空间双聚类算法

0/1 矩阵 R 的双聚类算法^[4-5]主要利用 SVD 分解矩阵, SVD 分解^[6-7]的数学公式为:

$$R = USV^T \quad (1)$$

$$\text{其中 } S = \begin{bmatrix} \sum & 0 \\ 0 & 0 \end{bmatrix}$$

$$\sum = \text{diag}(\sigma_1, \sigma_2, \dots, \sigma_r)$$

$$\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_r > 0$$

式中, σ_i ($i = 1, 2, \dots, r$) 为 R 的非零奇异值, 大部分情况下前 k ($0 < k < r$) 个奇异值之和就占全部奇异值之和的百分之九十几以上. 对于高维矩阵往往不需要所有的维度, 一般情况下只需要能量集中的前 k ($0 < k < r$) 个空间. 根据前 k ($0 < k < r$) 个奇异值, 相应地, 取得 $U = (u_1, u_2, \dots, u_k)$, $V = (v_1, v_2, \dots, v_k)$, 则 $USV^T = \sigma_1 u_1 v_1^T + \dots + \sigma_k u_k v_k^T$ (其中任意两个向量正交 $u_i^T u_j = 0$ 与 $v_i^T v_j = 0$).

SVD 分解将一个比较复杂的矩阵用几个更小更简单的子矩阵的相乘来表示, 这些小矩阵描述的是矩阵的重要的特性. 按照奇异值从大到小排序, 其中最大的奇异值对应的小矩阵反应了矩阵的最重要的特性. 所以首先分析最大奇异值对应的行向量 u_1 与列向量 v_1 , 并进行双向聚类.

行向量 $u_1 = (r_1, r_2, \dots, r_m)$ 反应了 0/1 矩阵行的能量分布情况, 1 元素越多的行 i 则 u_1 的 r_i 值越大. 设 u_1 中的值从大到小排列 $u_1 = (r_1, r_2, \dots, r_m)$ ($r_1 \geq r_2 \geq \dots \geq r_m$), 则对 0/1 矩阵 R 的行聚类即是对行向量 u_1 的聚类; 列向量 $v_1 = (c_1, c_2, \dots, c_n)$ 反应了 0/1 矩阵列的能量分布情况, 分析与行向量 u_1 的方法一样, 亦可得出对 0/1 矩阵 R 的列聚类即是对列向量 v_1 的聚类.

假设给定 0/1 矩阵中无重叠和噪声,那么数据集中的 1-矩阵间正交,当去掉为 0(或接近 0)的奇异值即去掉了背景数据 0 后,剩下的 k 个奇异值就对应着 k 个 1-矩阵. 算法 1 就是据此提出的,具体算法描述如下:

算法 1 奇异值向量空间双聚类算法

输入: 0/1 矩阵 R (无噪无重叠).

输出: 1-矩阵 A

1. SVD 分解矩阵 $R: R=USV^T$

2. for $u_i \in \{u_1, u_2, \dots, u_k\} (1 \leq i \leq k)$

记录不为 0 的值的行号 $\text{indexu}[i]$

endfor

3. for $v_i \in \{v_1, v_2, \dots, v_k\} (1 \leq i \leq k)$

记录不为 0 的值的行号 $\text{indexv}[i]$

endfor

4. for $\text{indexu}[i, j], \text{indexv}[i, j] (1 \leq i \leq k, 1 \leq j \leq k \text{ 且 } i \neq j)$

$\text{indexu}[i] \cap \text{indexu}[j] = \Phi$ and $\text{indexv}[i] \cap \text{indexv}[j] = \Phi$

endfor

5. 得到 k 个正交 1-矩阵 $R(\text{indexu}[i], \text{indexv}[i])$

算法 1 仅仅适应于简单的无噪声的 0/1 矩阵,且主要挖掘的是无重叠的 k 个 1-矩阵. 对于复杂的 0/1 矩阵就无能为力了,且挖掘的主要目的是频率概念(Frequent Concept),而不是 1-矩阵. 针对这些将作进一步的分析,以提出了更好的算法.

首先考虑 SVD 分解 0/1 矩阵 R 后得到的行向量 $u_1 = (r_1, r_2, \dots, r_m)$ 与列向量 $v_1 = (c_1, c_2, \dots, c_n)$, 聚类 u_1 与 v_1 . 此时面临一个问题: 行聚类优先还是列聚类优先. 由此引入信息熵^[8]的相关知识, 信息熵是信息论中用于度量信息量的一个概念. 一个系统越是有序, 信息熵就越低; 反之, 一个系统越是混乱, 信息熵就越高. 所以, 信息熵也可以说是系统有序化程度的一个度量. 假设存在变量 X , 可能的取值有 n 种, 分别为 $x_1, x_2, \dots, x_n$, 每一种值被取到的概率分别是 $P_1, P_2, \dots, P_n$, 那么 X 的熵就定义:

$$H(X) = H(P_1, P_2, \dots, P_n) \\ = - \sum_{i=1}^n P_i \times \log_2 P_i \sum_{i=1}^n P_i = 1 \quad (2)$$

0/1 基因矩阵 $R_{m \times n}, g \in G \equiv \{1, \dots, m\}$, G 表示矩阵 $R_{m \times n}$ 中的基因集合, 则矩阵 $R_{m \times n}$ 可以使用行向量 g 表示: $R_G = (g_1, g_2, \dots, g_m)^T$ 其中第 i 个行向量 $g_i = (R_{i1}, R_{i2}, \dots, R_{in})$; $t \in C \equiv \{1, \dots, n\}$, C 表示

矩阵 $R_{m \times n}$ 中的条件集合, 则矩阵 $R_{m \times n}$ 可以使用行向量 t 表示: $R_C = (t_1, t_2, \dots, t_n)$ 其中第 i 个列向量 $t_i = (R_{1i}, R_{2i}, \dots, R_{mi})$.

故 0/1 基因矩阵 $R_{m \times n}$ 的两个表现形式 R_G 和 R_C , 而且行矩阵 R_G 与列矩阵 R_C 所携带的信息量不同, 即它们的信息熵不同.

对于行矩阵 R_G , 信息熵为:

$$H(R_G) = - \sum_{i=1}^m P_i \times \log_2 P_i \\ P_i = \frac{r_i}{\text{sum}(u_k)} \quad (3)$$

其中 u_k 为第 k 个行特征向量 ($1 \leq k \leq r$ 的整数).

对于列矩阵 R_C , 信息熵为:

$$H(R_C) = - \sum_{i=1}^m P_i \times \log_2 P_i \\ P_i = \frac{c_i}{\text{sum}(v_k)} \quad (4)$$

其中 v_k 为第 k 个列特征向量 ($1 \leq k \leq r$ 的整数).

$H(R_G) \geq H(R_C)$ 则行聚类优先, 否则列聚类优先. 在假设行聚类优先的前提下, 给出改进算法, 步骤描述如下:

算法 2

第一步, SVD 分解 0/1 矩阵 $R (R=USV^T)$, $U = (u_1, u_2, \dots, u_r)$, $V = (v_1, v_2, \dots, v_r)$. 找出能量集中地前 $k (0 < k < r)$ 个奇异值, 本文主要利用比例来截取, 即 $(\sigma_1 + \dots + \sigma_k) / (\sigma_1 + \dots + \sigma_r) \geq \theta$, 其中 θ 为阈值 98%. 取出最大奇异值对应的行向量 u_1 与列向量 v_1 聚类.

第二步, 判断行聚类优先还是列聚类优先. 若行聚类优先则聚类行集, $u_1 = (r_1, r_2, \dots, r_m)$ 排序得到有序的向量 $u_1 = (r_1, r_2, \dots, r_m) (r_1 \geq r_2 \geq \dots \geq r_m \geq 0)$. 分析有序向量 u_1 , 先把背景元素 0 与噪声数据去除, 为此设定阈值 $\delta_0 (\delta_0 \geq 0)$, 当 $r_{NG} \geq \delta_0$ 且 $r_{NG+1} < \delta_0$ 则 $u_1 = (r_1, r_2, \dots, r_{NG}) (r_1 \geq r_2 \geq \dots \geq r_{NG} > 0)$ 为去掉背景元素与噪声后的行向量.

第三步, 因为向量 u_1 是排好序的, 每一行只与相邻的上下行最为相似, 因此计算相邻行间相似度向量 $s(r) = (s(r)_1, s(r)_2, \dots, s(r)_{NG-1})$, 其中 $s(r)_i = |r_{i+1} - r_i|$. 合并相邻行需满足的条价: 一个为 $s(r)_i < \Psi_r$ (Ψ_r 为行聚类相似度阈值); 另一个为 u_1 中元素与某个 r_i 相等的个数 $n < 3$, 为的是保证能量近似的行数不会过少, 即保证得到行聚类的基因数不得少于 3 条. 此步操作为, 把值为 r_i 的行合并到最为相似的行集.

第四步,计算行聚类得到的行子矩阵 $G=\{G_1, \dots, G_g\}$ 的 1-频度,若子矩阵 G_i 的 1-频率 $>$ 阈值 w ,则子矩阵 G_i 为一个频度概念,否则接着判断行聚类优先还是列聚类优先,转入第五步。

第五步,迭代执行第二步到第四步,直到满足条件退出。

算法 2 中只讲述了行向量 u_i 与列向量 v_i 上的聚类,但是这样并不能挖掘出所有的频率矩阵,最好的做法就是在前 k 个空间中都做一遍,但是这样就造成了效率高的问题。一般情况下,如若在某一空间内得到了较好的聚类效果则可以停止算法。

4 实验结果及分析

(1) 无噪声 0/1 人工数据集

为了测试算法 1 的正确性,生成一个黑色背景下有 2 个无重叠的矩阵 0/1 阵 A ,如图 1(a),打散其中的 1 元素生成矩阵 R ,如图 1(b)。

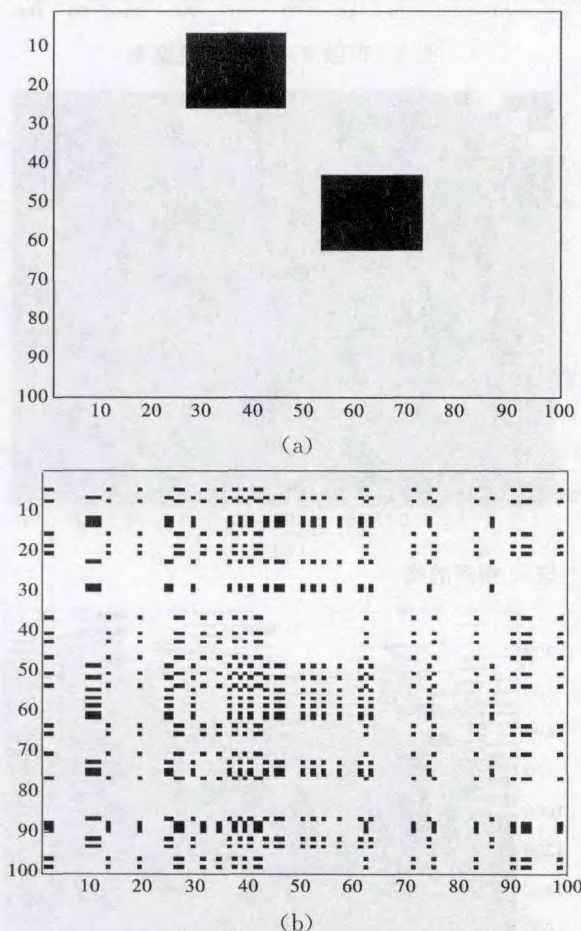


图1 0/1 阵 A 与矩阵 R

经过算法 1 的测试,结果取得 $\sigma_1=\sigma_2=20>0$,即 $k=2$. 在行向量 u_i 与列向量 v_i 上聚类得到 $\{g_5, g_8, g_{16}, g_{19}, g_{21}, g_{37}, g_{41}, g_{43}, g_{47}, g_{51}, g_{54}, g_{64},$

$g_{66}, g_{71}, g_{77}, g_{88}, g_{89}, g_{90}, g_{97}, g_{99}\}, \{t_1, t_2, t_{13}, t_{19}, t_{26}, t_{27}, t_{31}, t_{34}, t_{37}, t_{39}, t_{41}, t_{42}, t_{62}, t_{71}, t_{75}, t_{83}, t_{90}, t_{92}, t_{93}, t_{99}\} >$, $\{g_7, g_{12}, g_{13}, g_{14}, g_{23}, g_{29}, g_{30}, g_{49}, g_{52}, g_{55}, g_{57}, g_{59}, g_{61}, g_{62}, g_{73}, g_{75}, g_{76}, g_{87}, g_{92}, g_{94}\}, \{t_9, t_{10}, t_{11}, t_{24}, t_{25}, t_{29}, t_{36}, t_{38}, t_{40}, t_{43}, t_{45}, t_{46}, t_{50}, t_{52}, t_{54}, t_{57}, t_{61}, t_{63}, t_{74}, t_{86}\} >$. 两个矩阵的行标号与列标号并无重叠,所以挖掘的两个矩阵正交。

测试的第二个数据集 A ,如图 2(a),经过 SVD 分解可以把所有的 1 元素集中在矩阵块 B ,如图 2(b)。

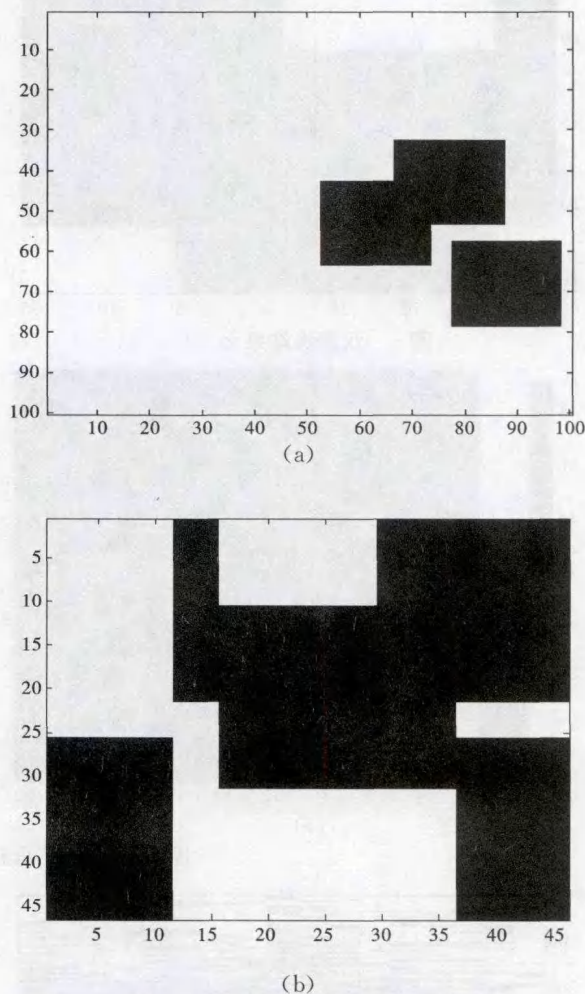


图2 0/1 阵 A 与最大 1 矩阵 B

根据算法 2 计算其中 1 元素的频率比重 $58.88\% < w = 75\%$,需要继续聚类矩阵 B ,经过迭代聚类得到三个双聚类,其结果之一,如图 3 所示。另外两个双聚类为 21×21 的全 1 方阵与 6×21 的全 1 矩阵。

(2) 有噪声 0/1 人工数据集

为输入有噪声 0/1 人工数据集,在规模为 100 行与 100 列的数据集中随机生成三个全 1 矩阵,然

后在其中混入少量的 0 元素,目的是找出频度概念.如图 4 中 A、B、C 为三个频度概念.

现在利用算法 2 聚类打散后的有噪声 0/1 数据集,分析图 5 知道, A 与 B 可以合并为一个满足 1-频率阈值要求的大类,矩阵 A 与矩阵 B 与矩阵 C 正交,所以在行向量 u_1 与列向量 v_1 上并不能挖掘出矩阵 C,在行向量 u_2 与列向量 v_2 上可以找到 C.

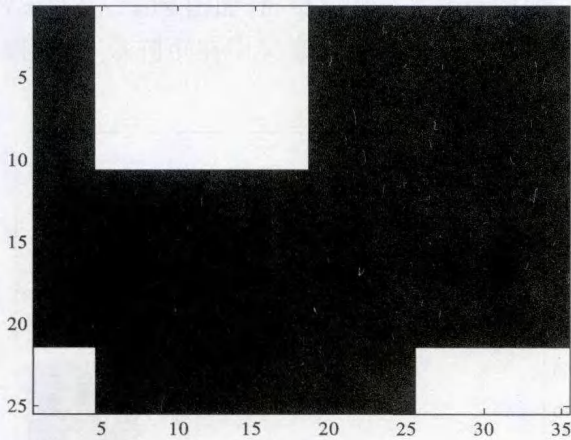
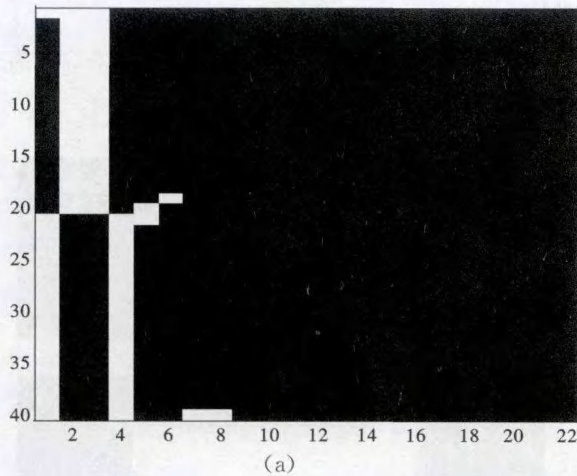
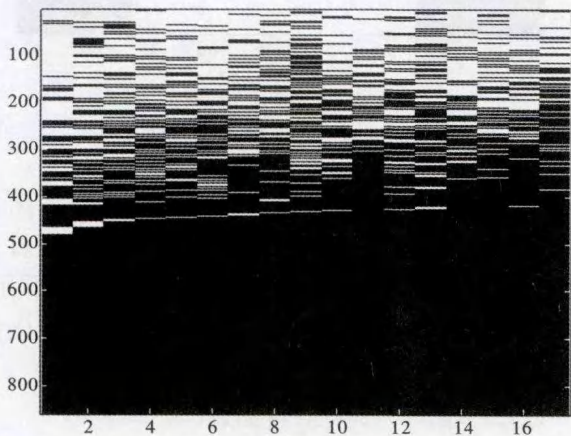


图 3 双聚类结果之一



(a)

图 5 u_1 与 v_1 得到的类和 u_2 与 v_2 得到的类



(a) 双聚类基因 858 条, 条件 17 个, 阈值 $t=0.0315$

(3) 实际数据集(yeast)

测试实数集时,首先对实数集数据进行二值化,二值化的过程为先标准化原矩阵再进行缩放,设置阈值 t ,数据大于 t 则表示该条基因在该条件下发生反应值为 1,否则为 0,即此基因在此条件下不发生反应.

图6(a)中阈值 $t=0.0315$,左图中基因858条,

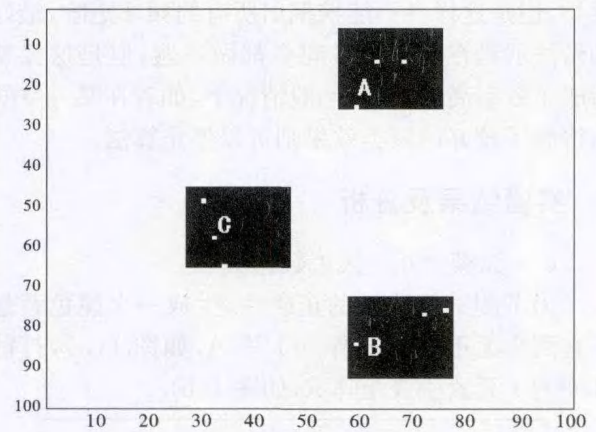
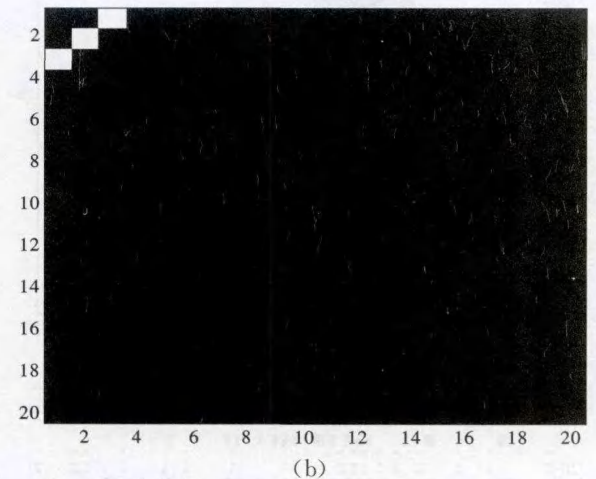
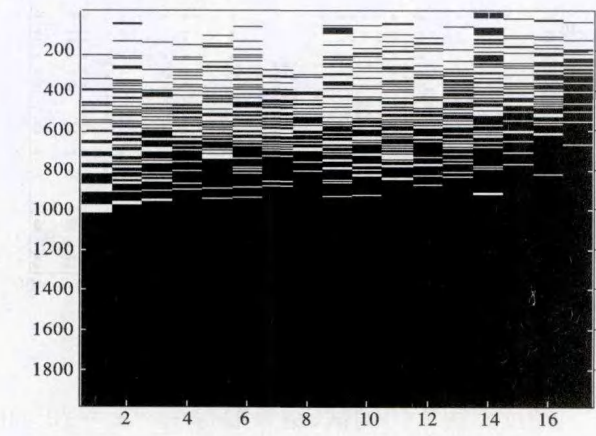


图 4 有噪声 0/1 人工数据集

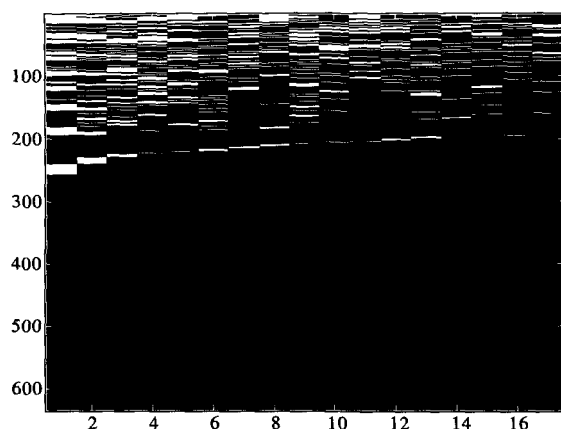
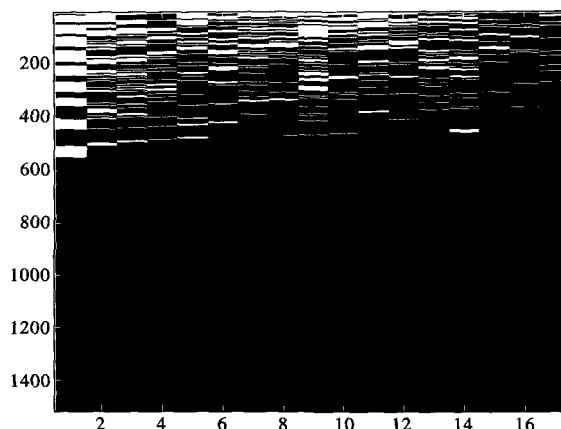


(b)



(b) 双聚类基因 1979 条, 条件 17 个, 阈值 $t=0.0210$

图 6 1 元素的频率比重阈值 $w=75\%$

(a) 双聚类基因 634 条,条件 17 个,阈值 $t=0.0315$ (b) 双聚类基因 1515 条,条件 17 个,阈值 $t=0.0210$ 图 7 1 元素的频率比重阈值 $w=91\%$

条件 17 个,1 元素的频率比重阈值 $w=75\%$;图 6 (b)中阈值 $t=0.0210$,基因 1979 条,条件 17 个,1 元素的频率比重阈值 $w=75\%$.图 7 中的 1 元素的频率比重阈值 $w=75\%$ 变为 $w=91\%$.

5 结束语

本文主要利用 SVD 进行双聚类 0/1 矩阵,通过 SVD 更加方便的了解原布尔矩阵,可以找出 1 元素的集中区域,分辨出原布尔矩阵中是否存在完全无重叠的子矩阵.通过奇异向量空间双聚类算法,可以挖掘出满足用户要求的“硬”矩阵,通过实验验证我们可以比较快速的得到双簇.

参考文献:

- [1] Je'r'myn Besson, C'eline Robardet, Jean-Francois Boulicaut, et al. Constraint-based concept mining and its application to microarray data analysis[J]. Intelligent Data Analysis, 2005, 9(1): 59-82.
- [2] Wille R. Restructuring lattice theory: an approach based on hierarchies of concepts[J]. Ordered sets, 1982: 445-470.
- [3] Ng R, Lakshmanan L, Han J, et al. Exploratory mining and pruning optimizations of constrained associations rules [C] // Proceedings ACM SIGMOD' 98, Washington, ACM Press, 1998:13-24.
- [4] Ruggero G Pensa, Celine Robardet, Jean-Francois Boulicaut. Constraint-driven co-clustering of 0/1 data [M]. USA: Florida, CRC Press LLC, 2008:123-148.

- [5] Abello J, Pardalos P M, Resende M G. Handbook of massive data sets, series: Massive Computing[M]. Boston: Kluwer, 2002.
- [6] Orly Alter, Patrick O Brown, David Botstein. Singular value decomposition for genome-wide expression data processing and modeling[J]. PNAS, 2000, 97 (18): 10101-10106.
- [7] Eckart C, Young G, A principal axis transformation for non-Hermitian matrices[J]. Null Amer. Math. Soc., 1939(45):118-121.
- [8] Bagyamani J, Thangavel K, Rathipriya R. Biological significance of gene expression data using similarity based biclustering algorithm[J]. International Journal of Biometrics and Bioinformatics, 2011, 4 (6): 201-216.

作者简介:



徐晓华 男,博士.研究方向为数据挖掘与机器学习.

席艳秋 女,硕士研究生.研究方向为数据挖掘与机器学习.
潘舟金 男,硕士研究生.研究方向为数据挖掘与机器学习.
陆林 男,硕士研究生.研究方向为数据挖掘与机器学习.
陈峻 男,教授.研究方向为人工智能与并行计算.