시간 기반 마이크로어레이에서 스케일링 및 쉬프팅 패턴을 찾는 새로운 방법

A Novel Method for Finding Scaling-and-Shifting patterns in Time-based Microarray

이동현(Dong-Hyun Lee)[1] 안재균(Jae-Gyoon Ahn)[1] 윤영미(Young-Mi Yoon)[1,2]

노홍찬(Hong-Chan Roh)[1] 박상현(Sang-Hyun Park)[3]

요 약

시간 기반 마이크로어레이 데이터는 유전자 집합의 발현 정도를 일정한 시간 간격으로 측정하여 수치화한 마이크로어레이 데이터를 뜻한다. 시간 기반 마이크로어레이 데이터를 기반으로 다른 유전자 집합을 활성(activation)화 시키거나 억제(inhibition)시키는 유전자 집합을 찾아냄으로써, 유전자 기능 네트워크를 효과적으로 구축 할 수 있다. 본 논문에서는 유전자의 증감 값을 특정 구간에 대하여 표준화하고 이를 통해 클러스터링과 클러스터 간 관계 도출을 동시에 해결 할 수 있는 알고리즘인 PRCluster (Past Relative Cluster)방법을 제시한다. PRCluster 를 검증하기 위해 효모 유전자의 시간 기반 마이크로어레이에서 클러스터를 생성 하고 이 클러스터를 바탕으로 활성 관계 또는 억제 관계를 도출하였다. 실험 결과 찾아낸 클러스터 내 두 객체 간에는 선형적인 함수 관계가 있었다. 또한 마이크로어레이 데이터에 존재하는 오차를 허용함과 동시에 적절한 유사도를 지닌 클러스터 간 활성 혹은 억제 관계를 도출 함을 확인할 수 있었다.

주제어: 마이크로어레이 분석, 클러스터링, 데이터 마이닝, 유전자 기능 네트워크

Abstract

Time-based microarray data is gene expression data generated by measuring the expression value of genes having a certain time span. By means of finding the gene set which activates or inhibits another gene set, the gene function network can be discovered. In this paper, we propose the PR cluster method that clusters the set of genes and mines relations between the found clusters by standardizing gene expression and observing the overall pattern. To verify the PR cluster algorithm, we conducted experiments using time-based yeast microarray data. The experimental results demonstrated that the PR cluster algorithm finds clusters in each of which any two objects have a linear functional relation. Also the experimental results show that the PR cluster algorithm can figure out the relation between clusters with proper similarity and can handle the noise that exists in the microarray data.

Key Words: Microarray analysis, Clustering, Data mining, Gene function network

---

*이 논문은 2006 년도 정부(과학기술부)의 재원으로 한국과학재단의 지원을 받아 수행된 연구임

[1] 연세대학교 컴퓨터과학과

[2] 가천의과학대학교 IT학과

[3] 연세대학교 컴퓨터과학과 교수(교신저자)

1. 서론

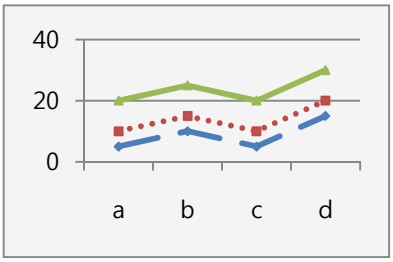
마이크로어레이 데이터는 유전자의 집합의 각 원소가 다양한 조건에서 어느 정도 발현되는지를 측정한 행렬 형태의 데이터이며 일반적으로 행(row)이 유전자의 집합, 열(column)이 조건의 집합으로 구성된다. 시간 기반 마이크로어레이 데이터는 열(column)이 시간 간격인 마이크로어레이 데이터 이다. 다시 말해 시간 기반 마이크로어레이 데이터는 유전자 집합의 각 원소의 발현 값을 특정 시간 간격으로 측정하여 얻은 마이크로어레이 데이터이다.

마이크로어레이 분석의 목적 중 하나는 마이크로어레이에 참여하는 유전자의 기능을 밝히는 것이다. 마이크로어레이의 분석은 크게 두 가지로 나뉜다. 먼저 모든 조건 하에서 유전자의 발현 값을 조사함으로써, 기능적 상관관계를 가지는 유전자를 클러스터링하는 기법이 있다[1][2]. 다른 한 가지는 조건의 부분 집합에서 상관관계를 가지는 유전자를 클러스터링하는 기법으로, 이러한 기법을 바이클러스터링이라 한다[3]. 바이클러스터링은 모든 조건에서 특정 기능과 관련된 유전자의 발현 정도를 관찰할 수 있는 것은 아니라는 점에서 전자의 분석 기법에 비해서 보다 유의한 유전자 집합을 클러스터링할 수 있음이 밝혀졌다[1][4][5][6][7].

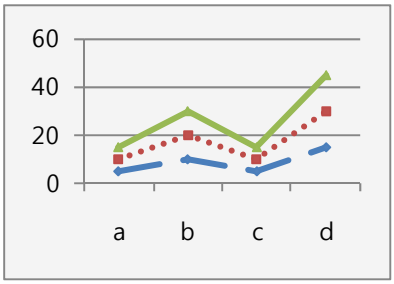
1.1 본 연구의 목적

1.1.1 클러스터 패턴의 종류

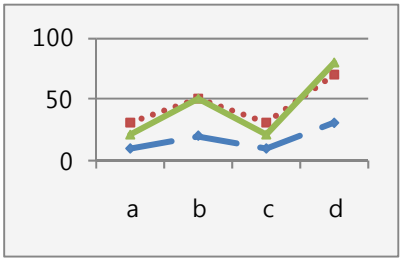
일반적으로 클러스터 패턴은 다음과 같이 세 가지 형태로 분류된다.



<그림 1> Shifting 패턴



<그림 2> Scaling 패턴



<그림 3> Shifting 패턴과 Scaling 패턴이 혼합된 패턴

Shifting 패턴은 각 열의 객체 값 간에 $y = x + b$의 관계가 존재하며 Scaling 패턴은 각 열의 객체 값 간에 $y = ax$ 의 관계가 존재한다.

바이클러스터에 관한 많은 연구는 Shifting 패턴을 찾는데 주력 하였다. Scaling 패턴을 이루는 각 열의 객체 값에 *log* 를 취하면 Shift 패턴을 이루기 때문이다[3].

그러나 Shifting 패턴과 Scaling 패턴이 동시에 존재하는 패턴, 즉 그림 3과 같이 각 열의 객체 값 간에 $y = ax + b$의 관계가 존재하는 패턴의 경우 각 객체 값에 *log*를 취하면 Shift 된 값에 *log*처리가 되므로 아무런 패턴을 이루지 않게 된다.

1.1.2 유전자 집합 간 활성 혹은 억제 관계

단순히 같은 기능을 하는 유전자 집합을 찾아내는 것만으로는 유전자 집합 간의 상관관계를 찾아내기 어렵다[8]. 같은 기능을 하는 유전자 집합을 유추하는 바이클러스터의 각 열(column)은 연속적이지 않으며 열 간의 선후관계 또한 존재하지 않기 때문이다. 따라서 유전자 집합 간 상관관계를 알아내기 위해 시간 기반 마이크로어레이를 대상으로 한 클러스터링 방법이 연구되어 왔다 [8][9].

시간 기반 마이크로어레이의 분석은 그 방법상 기존의 바이클러스터링 기법과 유사하나, 두 가지의 차이점을 지닌다. 먼저, 바이클러스터링이 열(column)의 임의의 집합에서 비슷한 패턴을 보이는 유전자 집합을 클러스터링 하는데 반해 시간 기반 마이크로어레이 분석 기법은 열의 연속된 집합에 대하여 비슷한 패턴을 보이는 유전자 집합을 클러스터링 한다. 또한 바이클러스터를 찾는 문제에 있어 클러스터링 된 유전자 집합 간의 관계를 고려하지 않으나 시간 기반 마이크로어레이 분석 기법은 서로 다른 시간대에서 비슷한 패턴을 보이는 배타적인 유전자 집합 간의 관계를 효과적으로 찾아냄으로써, 유전자 집합 간의 상관관계를 유추해 내는 것 또한 목적으로 한다.

<표 1> 시간 기반 마이크로어레이 예제

	0분	10분	20분	30분	40분
g_1	0.05	0.77	-0.15	0.02	-1.12
g_2	0.21	1.03	0.21	0.28	0.32
g_3	-0.13	0.37	-0.28	0.02	-0.1
g_4	-0.25	0.18	0.77	0.24	0.32
g_5	0.11	0.04	0.75	0.22	0.24
g_6	0.24	0.31	0.95	0.12	0.18
g_7	0.30	0.42	-0.12	0.64	0.56
g_8	0.65	0.55	0.03	0.66	0.55
g_9	0.10	0.24	-0.38	0.47	0.34

표 1은 10분 간격으로 9개의 유전자를 대상으로 발현 값을 조사한 시간 기반 마이크로어레이의 예제이다. 표 1의 유전자 g_1, g_2, g_3의 발현 값은 0분에서 30분 사이에 모두 0.5이상 증가하고, 0.5이상 감소하며, 소폭 증가하는 공통 패턴을 가진다. 반면, 유전자 g_4, g_5, g_6의 발현 값은 10분에서 40분 사이에 모두 0.5이상 증가하고, 0.5이상 감소하며, 소폭 증가하는 공통 패턴을 가진다. 즉, 유전자 집합 {g_1, g_2, g_3}과 {g_4, g_5, g_6}은 10분의 간격을 두고 비슷한 증감 패턴을 지닌다. 이때, 유전자 집합 {g_1, g_2, g_3}은 {g_4, g_5, g_6}를 활성화 시켰다고 유추할 수 있다. 또한, 유전자 g_7, g_8, g_9의 발현값은 10분에서 40분 사이에 모두 0.5이상 감소하고, 0.5이상 증가하며, 소폭 감소하는 공통 패턴을 지닌다. 즉, 유전자 집합 {g_1, g_2, g_3}과 {g_7, g_8, g_9}는 10분의 간격을 두고 정반대의 증감 패턴을 지닌다. 이 경우, 유전자 집합 {g_1, g_2, g_3}은 {g_7, g_8, g_9}의 활성을 억제한다고 유추할 수 있다.

본 연구는 다음 두 가지를 목적으로 한다. 첫째, 시간 기반 마이크로어레이와 같이 데이터의 열(column)에 시간적 순서가 존재하는 행렬 형태 데이터의 클러스터링과 클러스터간 활성 혹은 억제 관계를 도출하는 모델을 제시한다. 특히 두 클러스터의 패턴이 얼마나 유사한지를 수치화 할 수 있는 모델을 제안함으로써 오차를 너무 허용하거나 클러스터에 포함될 객체를 누락시키는 기존 연구의 문제를 해결한다. 둘째, 상기 특성을 지닌 행렬 형태 데이터에서 Shifting 패턴, Scaling 패턴, 그리고 이 두 가지가 결합된 패턴을 동시에 포함하는 클러스터를 찾을 수 있는 알고리즘을 제안한다.

1.2 관련 연구

Shifting 과 Scaling이 혼합된 패턴을 찾는 기법으로 Ben-Dor [4]가 제안한 OPSM이 있다. OPSM은 유전자의 발현 값 기준으로 열

(column)을 정렬한 다음 바이클러스터를 도출함으로써 Shifting 과 Scaling이 혼합된 패턴도 찾아낼 수 있는 장점이 있다. 하지만 활성 혹은 억제 관계 하의 클러스터의 경우 해당 구간이 연속적이어야 한다는 제한 조건이 있어 OPSM기법 만으로는 클러스터간 활성화 혹은 억제 관계를 규명할 수 없는 문제가 있다.

시간 기반 마이크로어레이 분석 기법 중 대표적인 것으로, Ji[8]가 제안한 q-cluster가 있다. q-cluster는 동 시간대에서 그 발현 값이 같이 증감하는 유전자의 집합을 클러스터링한 후, 이 유전자 집합의 증가 감소 패턴을 이용해서 유전자 집합에 고유 번호를 부여한다. 이 고유 번호로서 유전자 집합의 증가 감소 패턴을 쉽게 유추할 수 있으므로, 유전자 집합간의 패턴을 효율적으로 비교할 수 있다는 것은 큰 장점이다. 하지만 q-cluster는 유전자 집합을 클러스터링할 때, 단순히 같이 증감하는 유전자들로서 유전자 집합을 만들 뿐, 어느 정도로 증감하는지에 대한 고려를 하지 않는다. 이러한 방식은 마이크로어레이 혹은 시간 기반 마이크로어레이의 특성인 심한 노이즈를 잘 다룰 수 있다는 점에서 장점일 수 있지만, 클러스터링하지 말아야 할 유전자를 포함시키는 오류를 범한다는 단점이 존재한다.

또 다른 시간 기반 마이크로어레이 분석 기법으로 Kim[9]의 replicate를 이용한 클러스터링 알고리즘이 있다. 이 알고리즘은 유전자 집합을 클러스터링하는데 있어서, 특정 유전자의 특정 시간 간격에서의 증감 정도가 해당 유전자의 전체 모양에서 얼마만큼의 비중을 차지하는 지를 나타내는 값을 계산하고 이 값이 유사한 유전자들을 클러스터링한다. 그리고 두 유전자 집합 간의 유사도를 수치 값으로 표현하는 방식으로 두 유전자 집합간의 활성 혹은 억제 관계를 밝혀낸다. 이 알고리즘은 q-cluster와는 반대로, 오차를 거의 허용하지 않는 엄격한 패턴만을 찾기 때문에, 실제로는 같은 범주 안에 드는 유전자를 클러스터에서 제외하는 오류를 범하는 경우가 많다.

1.3 논문의 순서

향후 논문의 순서는 다음과 같다. 2 장에서 *R-transform* 과 이를 적용한 *δ-PR Cluster* 와 *δ-NR Cluster* 를 정의한다. 3 장은 2 장에서 제시한 모델에 기반한 알고리즘에 대한 설명과 클러스터간 활성 혹은 억제 관계를 규명하기 위한 방법을 소개한다. 4 장에서 실제 마이크로어레이 데이터에 본 모델과 알고리즘을 적용할 때의 결과를 보이고 마지막으로 5 장에서 결론과 본 연구의 공헌, 향후 연구 방향을 제시한다.

2. 모델

본 장은 객체 간의 유사성을 판별하기 위한 수학 모델인 *R-transform* (Relative Transform)과 이와 연관된 여타 수학 모델을 기술하는 것을 목적으로 한다. 또한 표 1 과 같이 객체의 집합과 조건의 집합이 각각 행과 열로 표현되는 행렬 형태의 데이터에 적용 시킬 수 있는 변형된 수학 모델을 제시한다.

2.1 *R-transform* (Relative Transform)

R-transform 은 다음과 같이 정의된다.

<Definition1. *R-transform* >

구간 (a,b) 에서 객체 F 에 대한 함수 $f(x)$ 가 존재할 때 $a < k \le b$ 인 $f'(k)$ 에 대한 *R-transform* 은 다음과 같다.

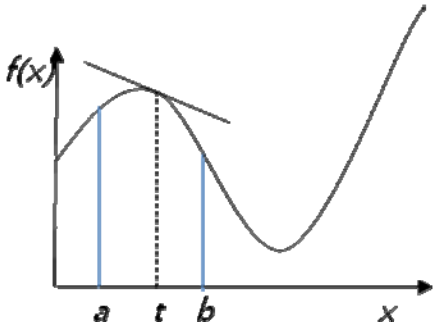
$$Rtrans(f'(k),a,b) = \frac{(b-a)f'(k)}{\int_a^b |f'(x)|\,dx} \qquad (1)$$

R-transform 을 달리 표기하면 다음과 같다.

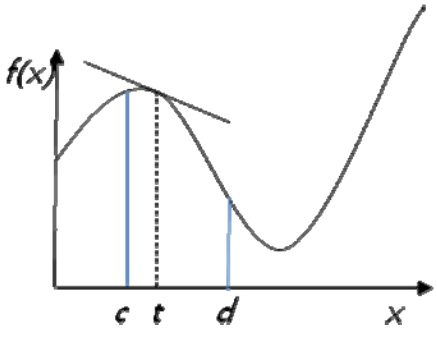
$$Rtrans(f'(k),a,b)=\frac{f'(k)}{\frac{1}{b-a}\int_a^b |f'(x)|dx}$$

$(\int_a^b |f'(x)|dx)/(b-a)$ 는 $f(x)$ 의 구간 (a,b) 의 총 $f'(x)$ 길이를 $1/(b-a)$ 비율로 축소 시킨 값이며 이는 구간 (a,b) 의 총 $f'(x)$ 길이의 평균이다. k 에서의 기울기 $f'(k)$ 는 $lim_{\Delta t\to 0}\frac{f(k+\Delta t)-f(k)}{\Delta t}$ 을 구간 길이 1 로 확장한 값이다. 따라서 $Rtrans(f'(k),a,b)$ 의 기대 값은 $f'(k)$ 가 양수일 경우 1, 음수일 경우 -1 이다.

$Rtrans(f'(k),a,b)$ 은 곡선의 길이를 고려하기 때문에 해당 구간의 전체 특성이 반영된다. 따라서 특정 지점의 기울기라 할지라도 구간을 다르게 설정하면 *R-transform* 값 또한 변한다.



<그림 4> *R-transform* 예시 1



<그림 5> *R-transform* 예시 2

그림 4 와 그림 5 에서 $Rtrans(f'(k),a,b)$ 는 $Rtrans(f'(k),c,d)$ 보다 크다. 구간 (a,b) 의 곡선의 형태가 구간 (c,d) 보다 완만하기 때문이다.

또한 *R-transform* 은 두 함수 간의 유사성 비교에 이용 된다. 구간 (a,b) 에서 $f(x)=p\cdot g(x)+q(p,q$는 실수이며 $p\neq 0)$ 일 때 $p>0$ 에서 $Rtrans(f'(k),a,b)$ = $Rtrans(g'(k),a,b)$ 이며 $p<0$ 에서 $Rtrans(f'(k),a,b)+Rtrans(g'(k),a,b)=0$ 이다.

< lemma1. 두 *R-transform* 의 관계>

$a<k\leq b$ 를 만족하는 구간 (a,b) 에서 $f(x)=p\cdot g(x)+q(p,q$는 실수이며 $p\neq 0)$ 일 때

$p>0$

$Rtrans(f'(k),a,b)=Rtrans(g'(k),a,b)$

$p<0$

$Rtrans(f'(k),a,b)+Rtrans(g'(k),a,b)=0$

Proof)

$f(x)=p\cdot g(x)+q$ 이므로

$f'(x)=p\cdot g'(x)$ 이며

$\int_a^b |f'(x)|dx=|p|\cdot\int_a^b |g'(x)|dx$ 이다.

상기 식을 $Rtrans(f'(k),a,b)$ 에 대입한다.

$p>0$

$$\frac{(b-a)f'(k)}{\int_a^b |f'(x)|dx}=\frac{(b-a)p\cdot g'(k)}{|p|\cdot\int_a^b |g'(x)|dx}=\frac{(b-a)g'(k)}{\int_a^b |g'(x)|dx}$$

$p<0$

$$\frac{(b-a)f'(k)}{\int_a^b |f'(x)|dx}=\frac{(b-a)p\cdot g'(k)}{|p|\cdot\int_a^b |g'(x)|dx}$$

$$= -\frac{(b-a)g'(k)}{\int_a^b |\, g'(x)\,|\, dx}$$

□

2.2 시간 기반 *R-transform*

R-transform 은 연속된 구간에서의 패턴 간 유사성을 측정하므로 2 장에서 언급한 시간적 연속성이 있는 행렬 형태 데이터의 특성을 반영하고 있다. 하지만 다음에 설명할 시간적 연속성이 있는 행렬 형태 데이터의 특성으로 인해 이에 적합한 변형된 *R-transform* 이 필요하다.

어떠한 객체가 지점 t 에서 $f(t)$ 라는 값을 가질 때 이 값에 영향을 미친 모든 요소는 t 이전에 발생한다. t 이후의 모든 구간에서의 요소는 t 의 영향을 받을 뿐 영향을 주지 않는 관계가 성립된다. $a < k < b$ 일 경우, 즉 k 보다 a 가 먼저 발생하였고 b 는 나중에 발생 할 때 $Rtrans(f'(k), a, b)$ 는 k 에 영향을 주는 구간과 k 의 영향을 받는 구간을 동시에 포함하게 된다. 따라서 *R-transform*의 구간 중 k 가 영향을 주는 구간과 영향을 받는 구간을 분리해야 한다.

<Definition2. *PR-transform*>

$a < k$ 를 만족하는 구간 (a, k) 에서 함수 $f(x)$ 가 존재할 때 $f'(k)$ 에 대한 *PR-transform* (Past Relative Transform)은 다음과 같이 정의된다.

$$PRtrans(f'(k), a) = \frac{(k-a)f'(k)}{\int_a^k |\, f'(x)\,|\, dx} \qquad (2)$$

<Definition3. *NR-transform*>

$k < b$ 를 만족하는 구간 (k, b) 에서 함수 $f(x)$ 가 존재할 때 $f'(k)$ 에 대한 *NR-transform* (Next Relative Transform)은 다음과 같이 정의된다.

$$NRtrans(f'(k), b) = \frac{(b-k)f'(k)}{\int_k^b |\, f'(x)\,|\, dx} \qquad (3)$$

PR-transform 과 *NR-transform* 의 기대 값은 *R-transform* 과 같이 $f'(k)$ 가 양수일 경우 1 이며 음수일 경우 -1 이다. 만약 구간 (a, k) 에 전반적으로 영향을 끼친 요인이 지점 k 에도 영향을 끼치면 $PRtrans(f'(k), a)$ 의 절대 값이 1 에 근접하며 구간 (a, k) 에 전반적으로 영향을 끼친 요인 외의 변수가 지점 k 에서 개입 했다면 $PRtrans(f'(k), a)$ 의 절대 값이 1 보다 크거나 작다.

PR-transform 과 *NR-transform* 은 그 차이가 구간을 구분하는 방식에만 있다. 따라서 모델의 수학적 정의와 알고리즘 설명의 중복을 피하기 위해 앞으로 모델 및 알고리즘과 관련한 설명은 *PR-transform* 위주로 진행된다.

2.3 시간 기반 *R-transform* 을 이용한 클러스터링

어떠한 객체 집합 내 원소의 일정 구간에서의 *PR-transform* 값이 모두 유사하다면 이 객체집합의 원소는 동일한 변인에 의하여 그 값이 변하고 있다고 볼 수 있다. *PR-transform* 은 열(column)간에 시간적 순서가 존재하는 데이터에서의 클러스터링에 이용된다.

< Definition4. *δ-PRCluster* >

다음을 만족할 경우 객체의 집합 $G = \{g_1, g_2, ..., g_n\}$ 를 구간 (a, k) 에서 *δ-PR Cluster* (*δ*- Past Relative Cluster)라 정의 한다.

구간 (a, k) 에서 G 의 각 원소에 대한 함수 $g_1(x), g_2(x), ..., g_n(x)$ 가 존재 할 때 $a < h \le k$ 를 만족하는 임의의 h 에 대하여

$g_i, g_j \in G$ 일 때

$$|PRtrans(g'_i(h), a) - PRtrans(g'_j(h), a)| \le \delta$$

δ-PR Cluster 는 특정 구간에 대한 객체 내 원소의 *PR-transform* 값 간의 차가 δ 를 넘지 않는 객체들의 집합이다. 즉 0-*PRCluster* 의 경우에는 클러스터 내 임의의 두 객체 간에 $y = px + q$ 의 관계가 존재하며 δ이 작을수록 클러스터 내 객체 간에 $y = px + q$ 에 근접한 관계가 존재한다.

객체 집합 $G = \{g_1, g_2, ..., g_n\}$ 가 구간 (a, k) 에서 *δ-PR Cluster*일 때 구간 $(a, k+\Delta)$ 에서 *δ-PR Cluster*를 이루는 객체 집합 G' 는 다음 두 가지에 의하여 도출 될 수 있다.

먼저 구간 $(a, k+\Delta)$ 에서 *δ-PR Cluster* 를 이루는 객체 집합 G' 는 G 의 부분 집합 내에 존재한다. G 의 원소 이외의 객체는 이미 구간 (a, k) 에서 G 와 *δ-PR Cluster* 를 이루지 않기 때문이다.

한편 구간 $(a, k+\Delta)$ 에서 *δ-PR Cluster* 인 객체 집합 G' 는 정의 4 에 의하여 다음을 만족해야 한다.

$G' = \{g_1, g_2, ..., g_m\}$ 일 때 $a < h \le k + \Delta$ 를 만족하는 임의의 h 에 대하여

$g_i, g_j \in G'$ 일 때

$$|PRtrans(g'_i(h), a) - PRtrans(g'_j(h), a)| \le \delta$$

그런데 이미 (a, k) 에서 G' 는 *δ-PR Cluster* 를 이루므로 $k < h \le k + \Delta$ 에서 상기 조건이 만족하면 G' 는 구간 $(a, k+\Delta)$ 에서 *δ-PR Cluster* 이다.

< Lemma2. *δ-PR Cluster* 의 확장 >

객체 집합 $G = \{g_1, g_2, ..., g_n\}$ 가 구간 (a, k) 에서 *δ-PR Cluster* 라 하자. 어떠한 객체 집합 G' 가 다음 조건을 만족할 경우 G' 는 구간 $(a, k+\Delta)$ 에서 *δ-PR Cluster* 이다.

$G' \subset G$ 이고 $g_i, g_j \in G'$ 일 때

$k < h \le k + \Delta$ 인 임의의 h 에 대하여

$$|PRtrans(g'_i(h), a) - PRtrans(g'_j(h), a)| \le \delta$$

2.4 행렬 데이터에의 적용

먼저 *R-transform, PR-transform, δ-PRCluster* 를 행렬 형태 데이터에 적용할 때 공통적으로 사용되는 기호를 소개한다.

<표 2> 행렬 데이터와 관련된 기호

기호	설명
g	특정 객체
$G=\{g_1, g_2, \cdots, g_n\}$	객체의 집합
$T = \{t_1, t_2, \cdots, t_m\}$	조건(열)의 집합
p_i	t_i와 t_{i+1} 사이의 구간
$P = \{p_1, p_2, \cdots, p_{m-1}\}$	T 에 대한 구간의 집합
$[p_a, p_b]$	$\{p_a, p_{a+1}, \cdots, p_b\}$의 약식표현
g'_b	객체 g 의 구간 p_b에서의 증감 값
g'_{ab}	객체의 집합 G 의 원소 g_a의 구간 p_b에서의 증감 값
$U = \begin{bmatrix} g'_{11} & \cdots & g'_{1m-1} \\ \vdots & \ddots & \vdots \\ g'_{n1} & \cdots & g'_{nm-1} \end{bmatrix}$	객체의 집합 G 와 구간의 집합 P에 대한 행렬 데이터

<Definition5. *Discrete R-transform*>

구간 집합 $P = \{p_a, p_{a+1}, ..., p_b\}$ 에서 객체 g 의 증감 값이 $\{g'_a, g'_{a+1}, ..., g'_b\}$ 로 표현될 때 $p_k \in P$ 인 구간 p_k 에서의 객체의 증감 값 g'_k 대한 *R-transform* 은 다음과 같이 정의된다.

$$Rtrans(g'_k, a, b) = \frac{(b-a+1)g'_k}{\sum_{i=a}^{b} |g'_i|} \quad (4)$$

$\sum_{i=a}^{b} |g'_i| = 0$ 일 경우 $Rtrans(g'_k, a, b)$ 는 NAN(Not A Number)로 표기한다.

<Definition6. *Discrete PR-transform*>

구간 집합 $P=\{p_a, p_{a+1},..., p_k\}$ 에서 객체 g 의 증감 값이 $\{g'_a, g'_{a+1},..., g'_k\}$로 표현 될 때 g'_k에 대한 *PR-transform* 은 다음과 같이 정의된다.

$$PRtrans(g'_k, a) = \frac{(k-a+1)g'_k}{\sum_{i=a}^{k} |g'_i|} \quad (5)$$

$\sum_{i=a}^{k} |g'_i| = 0$ 일 경우 $PRtrans(g'_k, a)$ 는 NAN(Not A Number)로 표기한다.

<Definition 7. *Discrete δ-PR Cluster*>

조건 집합 $P=\{p_a, p_{a+1},..., p_b\}$ 에서 객체 집합 $G=\{g_1, g_2,..., g_n\}$에 대하여 다음을 만족할 경우 G 는 *δ-PR Cluster*이다.

$p_k \in P$, $g_i, g_j \in G$ 일 때
$|PRtrans(g'_{ik}, a) - PRtrans(g'_{jk}, a)| \le \delta$

<Lemma 3. Discrete *δ-PR Cluster* 의 확장>

조건 집합 $P=\{p_a, p_{a+1},..., p_b\}$ 에서 객체 집합 $G=\{g_1, g_2,..., g_n\}$ 가 *δ-PR Cluster* 라 하자. $P'=\{p_a, p_{a+1},..., p_b,..., p_c\}$ 에서 G' 가 다음을 만족할 경우 G'는 *δ-PR Cluster* 이다.

$G' \subset G$ 이고 $g_i, g_j \in G'$ 일 때
$p_k \in \{p_{b+1},..., p_c\}$인 p_k에 대하여
$|PRtrans(g'_{ik}, a) - PRtrans(g'_{jk}, a)| \le \delta$

3. 알고리즘

이 장에서는 행렬 형태의 데이터에 존재하는 *δ-PR Cluster* 를 찾는 알고리즘에 대하여 기술한다. 아울러 찾아낸 클러스터간에 활성 혹은 억제 관계를 도출할 수 있는 방식을 제시한다.

3.1 *δ-PR Cluster*의 확장

객체 집합 G 가 구간 $[p_a, p_b]$에서 *δ-PR Cluster* 라 하자. 보조 정리 3 의 개념을 축소하여 구간 $[p_a, p_{b+1}]$에서 *δ-PR Cluster* 인 객체 집합을 구할 수 있으면 이를 이용하여 $b<c$ 인 구간 $[p_a, p_c]$ 에서의 *δ-PR Cluster* 또한 구할 수 있다. G 를 기반으로 $[p_a, p_{b+1}]$ 에서 *δ-PR Cluster* 인 객체 집합 G'를 구하고 다시 G'로부터 $[p_a, p_{b+2}]$에서 *δ-PR Cluster* 를 구할 수 있으며 이와 같은 과정을 반복하면 $[p_a, p_c]$ 에서의 *δ-PR Cluster*를 구할 수 있기 때문이다. .

구간 $[p_a, p_{b+1}]$에서 *δ-PR Cluster* 인 객체 집합 G'을 구하는 방법, 다시 말해 $G' \subset G$ 이고 $g_i, g_j \in G'$ 일 때 $|PRtrans(g'_{ib+1}, a) - PRtrans(g'_{jb+1}, a)| \le \delta$ 를 만족하는 G' 를 구하는 문제는 어떠한 숫자의 집합에서 최대 값과 최소 값의 차이가 δ보다 작은 집합을 찾는 문제로 환원된다.

Wang 은 [5]에서 *MDS*(Maximum Dimension Set)을 이용하여 어떠한 숫자의 집합에서 원소의 최대 값과 최소 값의 차이가 특정 값보다 작은 집합을 찾는 문제를 해결하였다. *MDS* 알고리즘은 객체 값을 내림차순으로 정렬한 후 최대 값과 최소 값의 차이가 일정 수준을 넘지 않는 집합을 찾는다. 특히 찾아낸 *MDS* 는 최대 값과 최소 값의 차가 일정 수준을 넘지 않으면서 그 객체의 개수가 최대인 성질을 지닌다. 가령 숫자의 집합 {12,7,5,4,3,1,0,-4} 에서 최대 값과 최소값의 차가 4 이하인 4-*MDS* 는 {7,5,4,3} ,{5,4,3,1}, {4,3,1,0},{3,1,0}이다.

본 연구 역시 G 로부터 $[p_a, p_{b+1}]$에서 *δ-PR Cluster*인 객체 집합 G'을 구할 때 *MDS*를 적용한다. 먼저 객체 G 내 각 원소의 p_{b+1} 에 대한 *PR-transform*을 구하고 이를 내림차순으로 정렬한다. 본 알고리즘은 이 정렬된 숫자의

집합에서 *MDS*를 찾는다. 다만 [5]에서 제시한 방법과 다른 점으로 기존 *MDS*알고리즘은 *MDS*간 overlapping을 허용하지만 본 알고리즘은 *PR-transform*의 집합에 대하여 *MDS*를 구할 때 overlapping을 허용하지 않는다. 원소의 증감 값 간에 편차가 심한 객체의 집합일지라도 원소의 *PR-transform* 간 편차는 적다. 따라서 이 경우 overlapping을 허용하면 매우 많은 수의 중첩된 클러스터가 생성되기 때문이다.

표 3 은 구간 $[p_4, p_5]$에서 0.3-*PR Cluster* 를 이루는 객체 집합 $G = \{g_1, g_4, g_5, g_9, g_{11}\}$ 를 나타낸 표다. 표에서 각 칸은 특정 객체의 해당 구간에서의 증감 값을 나타낸다.

<표 3> 0.3-*PR Cluster* 의 예시

	p_4	p_5	p_6
g_1	4	-4	8
g_4	3	-3	4
g_5	5	-6	22
g_9	11	-9	18
g_{11}	6	-6	6

표 4 는 구간 $[p_4, p_6]$ 에서 0.3-*PR Cluster* 를 이루는 객체 집합을 찾기 위해 p_6에서의 *PR-transform* 값을 기준으로 객체를 정렬한 결과다. *PR-transform* 은 소수점 두 번째 자리에서 반올림 하였다.

<표 4> 정렬된 *PR-transform* 의 예시

	각 객체의 p_6에서의 *PR-transform* 값
g_5	2.0
g_1	1.5
g_9	1.4
g_4	1.2
g_{11}	1.0

<표 4>에서 알 수 있듯 p_6에서 도출 되는 0.3-*MDS* 는 $\{g_1, g_9, g_4\}$ 이며 따라서 구간 $[p_4, p_6]$에서 0.3-*PR Cluster* 를 이루는 객체 집합은 $\{g_1, g_9, g_4\}$이다.

3.2 전체 알고리즘

구간$[p_a, p_b]$에서 δ-*PR Cluster*인 집합을 기반으로 $[p_a, p_{b+1}]$ 에서 δ-*PR Cluster*를 도출하는 방법을 전장에서 소개하였다. 이 방식을 이용하여 시작 구간이 p_k 인 모든 δ-*PR Cluster* 를 찾는 알고리즘을 제시한다.

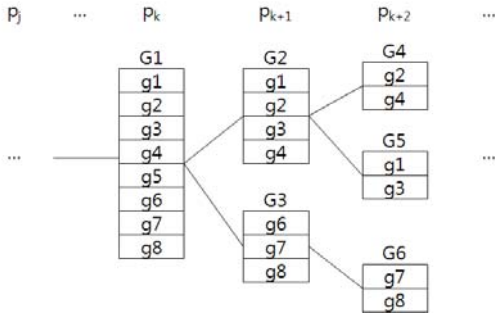
<Algorithm1. Finding δ-*PR Cluster*>
Input: 원소의 개수가 n 개, 구간의 개수가 m-1 인 데이터 행렬 U, 시작구간 p_k, 클러스터를 위한 δ값
Output: 시작구간이 p_k 인 모든 δ-*PR Cluster* 를 저장한 리스트 Result

Variables:
반복문을 위한 i
객체의 집합을 저장하는 리스트 Q1, Q2, Result

```
1. add G to Q1
2. for i from k to m-1
3.     while Q1 is not empty
4.         X←remove first element from Q1
5.         Q2←all δ-MDS at [p_k, p_i] from X
6.     add all elements in Q2 to Result
7.     add all elements in Q2 to Q1
8.     remove all elements in Q2
9. return Result
```

상기 알고리즘은 어떠한 구간$[p_k, p_i]$ 의 δ-*PR Cluster* 를 이전 구간 $[p_k, p_{i-1}]$의 δ-*PR Cluster* 를 기반으로 도출하고 이를 구간 $[p_k, p_k]$ 에서 마지막 구간 $[p_k, p_{m-1}]$ 까지 반복하는 형태로 구성되어 있다(lines2~8). 이전 구간 $[p_k, p_{i-1}]$의 모든 δ-*PR Cluster*는 리스트 Q1 에 저장되어 있으며

이 리스트의 모든 원소에 대하여 *δ-PR Cluster* 를 도출한다(lines3~5). 이전 구간으로부터 도출된 현재 구간 $[p_k, p_i]$ 의 모든 *δ-PR Cluster* 는 리스트 Q2 에 저장한다(line5). 이렇게 생성된 현재 구간 $[p_k, p_i]$ 의 *δ-PR Cluster* 는 다음 구간 $[p_k, p_{i+1}]$ 의 *δ-PR Cluster* 를 도출하는데 필요하므로 Q2 의 모든 원소를 Q1 에 저장한다(line7). Q2 의 모든 원소는 결과값에 해당되므로 Q2 의 모든 원소를 리스트 Result 에 삽입한다(line6).



<그림 6> *δ-PR Cluster* 알고리즘 예시

그림 6 은 *δ-PR Cluster* 를 찾는 과정을 나타내고 있다. G 의 부분 집합 G1 은 구간 $[p_k, p_k]$ 에서 *δ-PR Cluster* 이다. 구간을 $[p_k, p_{k+1}]$ 로 확장 할 때 G1 의 객체 중 $[p_k, p_{k+1}]$ 에서 *δ-PR Cluster* 를 이루는 객체의 집합 G2 와 G3 가 생성되며 이후의 과정 역시 이와 동일하다.

상기 알고리즘은 시작구간 $[p_k, p_k]$ 에서 마지막 구간 $[p_k, p_{m-1}]$ 까지 *δ-PR Cluster* 인 클러스터를 찾는 알고리즘 이다. 따라서 전체 구간집합 *P* 의 첫 원소인 p_1 에서 마지막 원소 p_{m-1} 까지 각 구간을 초기 시작점으로 설정한 후 상기 알고리즘을 반복 수행함으로써 전체 구간 집합의 각 원소를 시작점으로 하는 모든 *δ-PR Cluster* 를 구할 수 있다.

3.3 시간 복잡도

객체의 수가 k 개인 *δ-PR Cluster* 에서 다음 구간으로 확장하기 위해 *MDS* 를 찾을 때 소요되는 실행 시간은 $O(k)$ 이다.

한편 특정 구간에서 다음 구간으로 확장되는 *δ-PR Cluster* 내 객체의 총 개수는 n 이다. 특정 구간에서 다음 구간으로 확장되는 여러 *δ-PR Cluster* 의 원소 개수를 $k_1, k_2, ..., k_i$ 라 할 때 다음의 관계가 존재한다.

$$k_1 + k_2 + ... + k_i = n$$
$$O(k_1) + O(k_2) + ... + O(k_i) = O(n)$$

즉 전체 객체 집합 G 의 특정 구간에서의 *δ-PR Cluster* 를 찾는데 필요한 실행 시간은 $O(n)$ 이다. 초기 시작점을 기준으로 *δ-PR Cluster* 를 찾기 위해 상기 과정이 최대 m 번 반복되며 전체 구간에서의 *δ-PR Cluster* 를 찾기 위해 상기 과정을 다시 m 번 반복하므로 전체 구간에서의 *δ-PR Cluster* 를 찾기 위해 $O(m^2 n)$ 의 계산 량이 필요하다.

3.4 *δ-PR Cluster* 간 활성 혹은 억제 관계의 도출

시간적 순서가 존재하는 데이터 에서 도출한 클러스터간에는 활성 혹은 억제 관계가 있을 수 있다. 따라서 *δ-PR Cluster* 간 에도 활성 혹은 억제 관계가 있을 수 있으며 두 *δ-PR Cluster* 간의 활성 혹은 억제 관계는 다음과 같은 접근법을 통해 판단할 수 있다.

δ-PR Cluster 의 각 구간별 증감 값의 평균을 증감 값으로 갖는 가상의 객체는 *δ-PR Cluster* 를 대표한다. 즉 두 *δ-PR Cluster* 간의 활성 혹은 억제 관계를 판단하는 문제는 두 *δ-PR Cluster* 의 가상의 대표 객체를 구하고 이 두 객체간의 유사성을 판단하는 문제로

치환된다. 만약 두 객체 간에 억제 관계가 있다면 두 객체가 다시 *δ-PR Cluster* 를 형성한다. 물론 두 대표 객체 간 유사성을 판단하기 위한 δ 값은 클러스터를 생성할 때의 δ 값과 다르게 설정해 줄 수 있다. 두 객체 간에 억제 관계는 한 객체의 부호를 바꾼 후 두 객체가 활성 관계에 있음을 보이면 된다[8].

<표 5 > *δ-PR Cluster* 의 대표 객체 예시

	p_1	p_2	p_3
g_1	2.2	-6.1	5.8
g_2	2.5	-7.3	6.4
g_3	4.0	-10.2	8.9
대표 객체	2.9	-7.9	7.0

두 *δ-PR Cluster* 의 활성 혹은 억제 관계를 명확히 판단하기 위해서는 두 *δ-PR Cluster* 내부에 존재하는 활성 혹은 억제 관계를 고려해야 한다. 예를 들어 어떠한 *δ-PR Cluster* X 의 구간이 $[p_3, p_9]$ 이고 다른 *δ-PR Cluster* Y 의 구간이 $[p_5, p_{10}]$ 일 때 두 *δ-PR Cluster* 의 활성 혹은 억제 관계가 X 의 $[p_4, p_8]$ 과 Y 의 $[p_6, p_{10}]$ 에 존재할 수 있다. 따라서 두 클러스터의 각 구간의 부분 집합간의 활성 혹은 억제 관계를 고려해야 한다[8].

두 *δ-PR Cluster* 의 활성 혹은 억제 관계를 명확히 판단하기 위한 전체 과정은 다음과 같다. *δ-PR Cluster* X 의 구간이 $[p_a, p_b]$ 이고 다른 *δ-PR Cluster* Y 의 구간이 $[p_c, p_d]$ 일 때 먼저 X 의 대표 객체 x 와 Y 의 대표 객체 y 를 산출한다. $[p_a, p_b]$ 내 모든 부분 구간과 $[p_c, p_d]$ 의 모든 부분 구간 중 구간 길이가 같은 모든 쌍에 대하여 x 와 y 가 활성 혹은 억제 관계인지를 판단한다.

3.5 클러스터링을 위한 pruning

다음은 클러스터를 더욱 효율적으로 하기 위해 적용한 pruning 정책이다.

1. *δ-PR Cluster*내 모든 객체의 *PR-transform* 값의 부호는 동일하도록 유지한다. *δ-MDS*(Maximum Dimension Set)을 생성 할 때 해당 *δ*-MDS내 객체 중 부호가 다른 객체를 제거하였다. 시간 기반 마이크로어레이에서의 클러스터링에 있어 부호를 이용하는 [8]을 참고하였다.
2. 최소 참여 객체 수: *δ-PR Cluster*로 인정 할 수 있는 객체의 최소 개수이다. 예를 들어 최소 참여 객체 수가 20이라면 유사한 증감패턴을 보이더라도 객체 수가 20개 미만인 *δ-PR Cluster* 는 결과에서 제외하였다.
3. 최소 구간 길이: *δ-PR Cluster*로 인정할 수 있는 구간의 최소 길이다. 가령 최소 구간 길이가 5라면 유사한 증감패턴을 보일지라도 그 구간의 길이가 5 미만인 *δ-PR Cluster* 는 결과에서 제외하였다.

클러스터링 도중 최소 참여 객체 수 미만인 *δ-PR Cluster*가 도출 되었을 경우 하위 *δ-PR Cluster*역시 최소 참여 객체 수 이하다. 또한 전체 구간의 개수가 정해져 있으므로 클러스터링을 진행하여도 도출되는 *δ-PR Cluster*의 구간 길이가 최소 구간 길이를 넘지 못하는 경우가 발생하게 된다. 따라서 최소 참여 객체 수와 최소 구간 길이의 값을 조정하여 클러스터링의 소요 시간을 조절 할 수 있다.

4. 실험

4.1 실험 데이터 및 실험 환경

본 논문에서 사용한 시간 기반 마이크로어

레이는 Tavazoie[10]가 발표한 것으로서 효모(yeast)의 2884개의 유전자가 2번의 세포 사이클이 일어날 동안의 17개의 구간에서 어느 정도 발현된 것인가를 측정한 것이다.

알고리즘 구현 언어는 JAVA며, 2.34GHz Intel CPU 및 2.0GB RAM 사양의 데스크톱에서 클러스터링을 수행하였다.

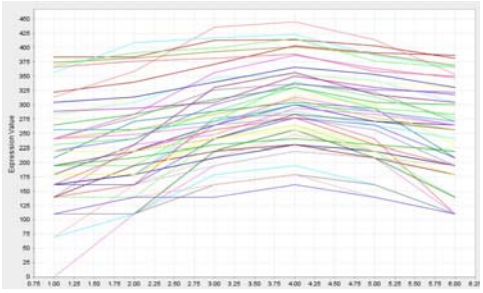
4.2 실험 결과

실험에 쓰인 사용자 지정 파라미터는 다음과 같다.

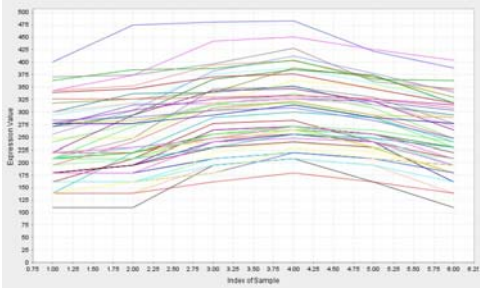
<표 6 > 실험에 사용된 파라미터

클러스터를 위한 δ	0.1
활성 혹은 억제 관계 규명을 위한 δ	0.2
최소 참여 객체 수	30
최소 구간 길이	4

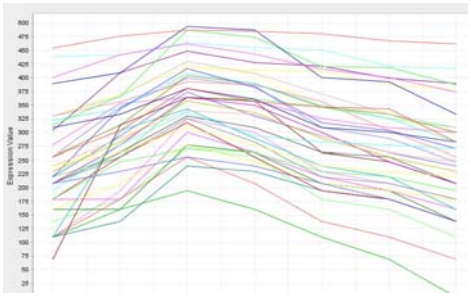
상기 사용자 지정 파라미터가 주어질 때 435개의 0.1-*PR Cluster*와 3081개의 활성 혹은 억제 관계를 도출하였으며 총 소요 시간은 5.2초였다. 다음은 도출된 0.1-*PR Cluster* 중 일부다.



<그림7> 구간 [2,8]에서의 클러스터 9



<그림8> 구간 [9,15]에서의 클러스터 40



<그림9> 구간 [1,8]에서의 클러스터 5

각 클러스터 내 두 유전자 간에 $f(x) = p \cdot g(x) + q(p, q$는 실수이며 $p \neq 0)$의 관계가 나타나 있다. 대표적인 예로 클러스터 5의 각 유전자의 패턴을 보면 Shifting 패턴과 Scaling 패턴, 그리고 두 패턴이 혼합된 형태의 패턴이 있음을 알 수 있다.

찾아낸 클러스터의 생물학적 특성을 측정하기 위하여 실험에서 사용한 효모, 즉 Saccharomyces cerevisiae의 유전자 온톨로지(Gene ontology)데이터 베이스[11]를 이용하였다. 다음은 각 클러스터에 속한 유전자의 유전자 온톨로지에 의거한 생물학적 특성을 나타낸 표이다.

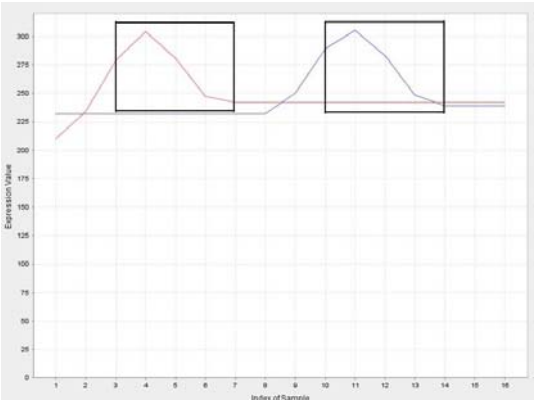
<표 7> 세 클러스터의 생물학적 특성결과

클러스터	P-값	생물학적 기능
9	1.5e-06	sulfur metabolism sulphur metabolism
	3.1e-06	sulfur amino acid transport sulphur amino acid transport
	6.2e-06	sulfur amino acid transporter activity sulphur amino acid transporter activity
40	2.8e-06	DNA metabolism
	1.2e-05	mitotic sister chromatid cohesion
	2.5e-05	chromosome
5	6.7e-09	DNA replication DNA biosynthesis DNA synthesis

	9.9e-09	DNA repair
	1.8e-08	chromosome

상기 표에 나타나 있듯 본 실험에서 찾아낸 클러스터를 통해 실제로 생물학적 특성을 지닌 유전자 집단을 알 수 있다. 특히 찾아낸 클러스터 내 유전자 집단이 각기 수행하는 기능이 비슷함을 알 수 있다. 가령 클러스터 5내에는 DNA 복제와 DNA 복구라는 서로 유사한 기능을 하는 유전자 집단이 존재한다.

다음은 클러스터 (9,40)간 활성 혹은 억제 관계를 나타낸 그림이다. 가시성을 위해 클러스터의 패턴을 대표 객체의 패턴으로 대체하였다. 각 클러스터의 사각형안의 구간 간에 활성 혹은 억제 관계가 존재한다. 실험에 있어 클러스터 간 관계규명을 위한 δ 값으로 0.2를 지정하였기 때문에 두 사각형 안의 구간 은 0.2-*PR Cluster* 를 이룬다. 즉 δ 값을 사용자가 직접 조절함으로써 높은 수준의 유사도를 보임 과 동시에 적절한 수준의 오차를 허용하는 클러스터 관계를 도출할 수 있으며 이로써 기존 시간 기반 마이크로어레이 분석 기법[8][9]의 문제를 해결할 수 있다.



<그림 10> 클러스터 9과 40의 활성 관계

클러스터 9의 구간 [3,7] 에서의 패턴과 클러스터 40의 구간 [10,14]에서의 패턴 간에 유사한 증감이 나타나므로 클러스터 9과 40의 일부 구간에서 활성 관계가 있음을 알 수 있다. 표 7에 나타나 있듯 클러스터 9와 40 각각이 물질대사와 관련된 기능을 수행한다는 점을 고려할 때 본 실험의 결과는 클러스터 9와 40간에 실제로 활성화 관계가 존재할 수 있는 가능성을 시사한다.

5. 결론 및 향후 연구 일정

본 논문은 특정 객체 패턴의 각 부분을 표준화 할 수 있는 수식인 *R-transform*을 정의하고 *R-transform*을 이용한 클러스터링 기법인 δ*-PR Cluster*을 제시하였으며 클러스터간 활성 혹은 억제 관계 역시 동일한 방식으로 해결 할 수 있음을 보였다. 시간 기반 마이크로어레이에서의 클러스터링 결과 기존 연구에서 찾아내기 어려웠던 Shifting 패턴과 Scaling 패턴이 결합된 패턴을 효율적으로 찾아 내었다. 또한 활성화 혹은 억제 관계를 판명함에 있어 클러스터간 부호를 비교하거나 패턴의 모양을 엄격히 구분하는 기존 연구와 달리 두 클러스터의 유사도를 직접 정할 수 있어 클러스터 간 활성 또는 억제 관계를 효과적으로 밝힐 수 있었다.

향후 연구 계획은 세 가지다. 첫째, 상기 실험에 있어 총 3500여 개의 결과가 형성되었다. 실제 생물학에 적용 시키기에는 너무 많은 실험 결과이기 때문에 결과 수를 줄이면서도 생물학적 효용성을 훼손시키지 않는 방법에 대한 추가 연구가 필요하다. 둘째, 다수의 마이크로어레이에서 유전자 집합 간 활성 혹은 억제 관계를 찾아내고, 이러한 관계로부터 유전자 제어 조절 네트워크를 구축하는 기법에 대해서 연구하고자 한다. 셋째, *R-transform*을 이용한 클러스터링은 열(column)간에 시간적 순서가 존재하는 행렬 데이터에 한정된 방법이므로 일반적인 행렬 데이터에서 전체 객체 집합의 부분 집합과 전체 열 집합의 부분 집합의 클러스터, 즉 바이클러스터를 찾아낼 수 있도록 *R-transform*을 개량할 것이다.

6. 참고 문헌

[1] S. C. Madeira and A. L. Oliveira, "Biclustering Algorithms for Biological Data Analysis: A Surve0y," IEEE/ACM Trans. Computational Biology and Bioinformatics, vol. 1, no. 1, pp. 24-45, 2004.

[2] A. Prelic, S. Bleuler, P. Zimmermann, A. Wille, P. Bhlmann, W. Gruissem, L. Hennig, L. Thiele, and E. Zitzler, "A systematic comparison and evaluation of biclustering methods for gene expression data," Bioinformatics, vol. 22, no. 9, pp. 1122-1129, 2006.

[3] Y. Cheng and G.M. Church, "Biclustering of Expression Data," in Proc. 8th Int'l Conf. Intelligent Systems for Molecular Biology, pp. 93-103, 2000.

[4] A. Ben-Dor, B. Chor, R. Karp, and Z. Yakhini, "Discovering local structure in gene expression data: The order–preserving submatrix problem," in Proc. 6th Int'l Conf. Computacional Biology, pp. 49–57, 2002.

[5] H. Wang, W. Wang, J. Yang and P. S. Yu, "Clustering by Pattern Similarity in Large Data Sets," in Proc. ACM SIGMOD Int'l. Conf. Management of Data, pp. 394-405, 2002.

[6] L. Zhao and M. J. Zaki, "triCluster: An Effective Algorithm for Mining Coherent Clusters in 3D Microarray Data," in Proc. ACM SIGMOD Int'l. Conf. on Management of data, pp. 694–705, 2005.

[7] X. Xu, Y. Lu, A. K. H. Tung and W. Wang, "Mining Shifting-and-Scaling Co-Regulation Patterns on Gene Expression Profiles," in Proc. 22nd IEEE Int'l. Conf. on Data Engineering, pp. 89-99, 2006.

[8] L. Ji and K. Tan, "Identifying time-lagged gene clusters using gene expression data, " Bioinformatics, Vol. 21, No. 4, pp. 509-516, 2005.

[9] J. Kim and J. H. Kim, "Difference-based clustering of short time-course microarray data with replicates," BMC Bioinformatics, Vol. 8, Issue. 1, No. 253, 2007.

[10] S. Tavazoie, J. D. Hughes, M. J. Campbell, R. J. Cho, and G. M. Church, "Systematic determination of genetic network architecture," Nat. Genet., Vol.3, pp.281–285, 1999.

[11]http://llama.med.harvard.edu/cgi/func/funcassociate

이동현
2008.3: 연세대학교 컴퓨터과학과 (공학사)
2008.3~: 연세대학교 컴퓨터과학과(석사과정)
관심 분야: 바이오인포메틱스, 데이터 마이닝, SSD
Email : ldh@cs.yonsei.ac.kr

안재균
2006.2: 연세대학교 컴퓨터과학과 (공학사)
2007.9~: 연세대학교 컴퓨터과학과 (석사과정)
관심 분야: 바이오인포메틱스, 데이터마이닝, 데이터베이스 보안
Email : ajk@cs.yonsei.ac.kr

윤영미
1981.2: 서울대학교 자연과학대학 (학사)
1983.6: 오하이오 주립대학 수학과 (학사)
1987.3: 스탠포드 대학교 컴퓨터과학과 (공학석사)
1993.5: IntelliGenetics Inc., Mountainview, California, Software Engineer
1995.2~: 가천의과학대학교 부교수
2008.8: 연세대학교 컴퓨터과학과 (공학박사)
관심분야: 데이터베이스 시스템, 데이터 마이닝, 바이오인포메틱스
Email : amyyoon@cs.yonsei.ac.kr

노 홍 찬
2006.2: 연세대학교 컴퓨터과학과 (공학사)
2008.2: 연세대학교 컴퓨터과학과 (공학석사)

2008.3~: 연세대학교 컴퓨터과학과 (박사과정)
관심 분야: 플래쉬메모리 인덱스, SSD, 데이터 마이닝
Email : fallsmal@cs.yonsei.ac.kr

박 상 현
1989.2 : 서울대학교 컴퓨터공학과 (공학사)
1991.2 : 서울대학교 컴퓨터공학과 (공학석사)
2001.2 : UCLA 대학교 전산학과 (공학박사)
2001.2 ~ 2002.6 : IBM T. J Watson Research Center Post-Doctoral Fellow.
2002.8 ~ 2003.8 : 포항공과대학교 컴퓨터공학과 조교수
2003.9 ~ 2006.8 : 연세대학교 컴퓨터과학과 조교수
2006.9 ~ 현재 : 연세대학교 컴퓨터과학과 부교수
관심분야 : 데이터베이스 보안, 데이터 마이닝, 바이오인포매틱스, XML
E-mail : sanghyun@cs.yonsei.ac.kr