DOI: 10.3745/KIPSTD.2008.15-D.6.767

순위 비교를 기반으로 하는 다양한 유전자 개수로 이루어진 암 분류 결정 규칙의 생성

윤 영 미[†] · 변 상 재[††] · 박 상 현[†††]

요 약

마이크로어레이 기술은 최근 실험적 분자생물학 분야에서 활발히 사용되고 있는 기술이다. 마이크로어레이 데이터는 한 번의 실험으로 수 만개의 유전자에 대한 발현값을 얻을 수 있으므로, 여러 질병의 발현형질을 연구하는데 매우 유용하게 사용된다. 마이크로어레이 데이터의 문제점은 참여하는 유전자의 수에 비해 참여하는 샘플(생물조직샘플)의 수가 매우 적고, 분류분석 기법을 사용하여 얻어진 분류자의 해석이 어렵다는 점이다.

본 연구에서는 위의 문제점을 해결하고자, 샘플 내 순위를 이용하여 동일한 생물학적 목적으로 수행된 공개 마이크로어레이 데이터를 통합하고, 순위 비교를 기반으로 하는 다양한 유전자 개수로 이루어진 암 분류 결정 규칙들로 이루어진 분류자를 제안한다. 본 분류자는 k개의 규칙으로 이루어진 앙상블 방법을 기반으로 하며, 하나의 규칙은 최대N개의 유전자, 관련유전자간의 순위비교 관계식, 판별클래스로 이루어져 있다. 하나의 규칙에 참여하는 유전자의 수를 다양하게 함으로써 좀더 신뢰성 높은 분류자를 생성할 수 있다. 또한 본 분류자는 생물학적 해석이 용이하며, 분류자를 구성하는 유전자를 명확히 식별할 수 있고, 총 개수가 많지 않으므로 임상환경에서의 사용가능성도 생각해 볼 수 있다.

키워드 : 데이터 마이닝, 분류분석, 지식기반 데이터 마이닝, 마이크로어레이 데이터 분류 분석

Generating Rank-Comparison Decision Rules with Variable Number of Genes for Cancer Classification

Youngmi Yoon[†] · Sangjay Bien[††] · Sanghyun Park[†††]

ABSTRACT

Microarray technology is extensively being used in experimental molecular biology field. Microarray experiments generate quantitative expression measurements for thousands of genes simultaneously, which is useful for the phenotype classification of many diseases. One of the two major problems in microarray data classification is that the number of genes exceeds the number of tissue samples. The other problem is that current methods generate classifiers that are accurate but difficult to interpret.

Our paper addresses these two problems. We performed a direct integration of individual microarrays with same biological objectives by transforming an expression value into a rank value within a sample and generated rank-comparison decision rules with variable number of genes for cancer classification. Our classifier is an ensemble method which has k top scoring decision rules. Each rule contains a number of genes, a relationship among involved genes, and a class label. Current classifiers which are also ensemble methods consist of k top scoring decision rules. However these classifiers fix the number of genes in each rule as a pair or a triple. In this paper we generalized the number of genes involved in each rule. The number of genes in each rule is in the range of 2 to N respectively. Generalizing the number of genes increases the robustness and the reliability of the classifier for the class prediction of an independent sample. Also our classifier is readily interpretable, accurate with small number of genes, and shed a possibility of the use in a clinical setting.

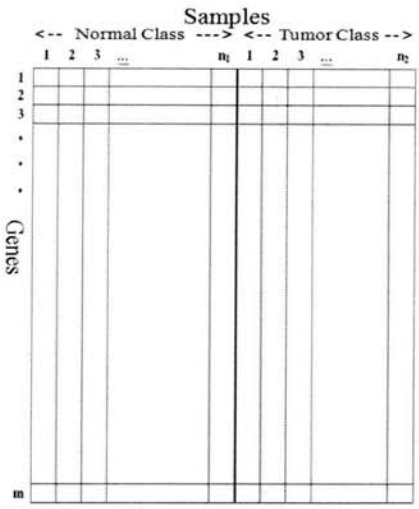
Keywords : Data Mining, Classification, Knowledge-Based Data Mining, Microarray Data Analysis, Microarray Data Classification

1. 서 론

마이크로어레이 기술의 발달로 한 실험에서 대량의 유전자의 발현값을 총체적으로 측정할 수 있게 되었다. 마이크로어레이는 작은 고형체 기판 위에 염기서열을 알고 있는 수 만개의 DNA를 고밀도로 집적한 것이다. 마이크로어레이 데이터는 아래 (그림 1)과 같은 형태를 갖는다. 각각의 행은 하나의 유전자를, 각각의 열은 하나의 샘플을, 하나의 셀 값은 특정 유전자의 특정 샘플에서의 발현값을 의미한다. 하나의 샘플은 유전자 집합으로 이루어지며, 각 샘플은 클래스표지로서 "정상", 또는 "암"을 갖는다.

마이크로어레이 기술을 이용하여, 유전자 발현 프로파일

※ 이 논문은 2006년도 정부(과학기술부)의 재원으로 한국과학재단의 지원을 받아 수행된 연구임(No. R01-2006-000-11106-0).
† 종신회원 : 가천의과학대학교 IT학과 부교수
†† 준 회 원 : 서울대학교 생물정보학전공 석사과정
††† 종신회원 : 연세대학교 컴퓨터과학과 부교수(교신저자)
논문접수 : 2008년 6월 3일
수 정 일 : 1차 2008년 11월 17일
심사완료 : 2008년 11월 17일



(그림 1) 마이크로어레이 데이터세트

에 내재된, 알려지지 않은 생물학적인 지식을 찾아내기 위한 중요한 방법 중의 하나는 유전자 발현 프로파일에 대한 분류분석(Classification)을 수행하는 것이다. 샘플 집합이 트레이닝 데이터세트로 주어졌을 때 유전자 발현 값의 함수형태로서, 클래스속성에 대한 모델을 밝히는 과정이 바로 분류분석이다[8]. 마이크로어레이 데이터는 암질병의 발현형(Phenotype)의 분류분석에 사용되는 유용한 도구이다. 마이크로어레이 분류분석의 두 가지 제약사항은 아래와 같다. 첫째, 통계적으로 유의한 결과를 얻을 수 있을 만큼의 충분한 샘플수가 존재하지 않는다는 점[4, 10]과 둘째, 기존 방법으로 얻어진 분류자가 비교적 정확한 결과는 주지만 해석하기가 용이하지 않다는 점이다.

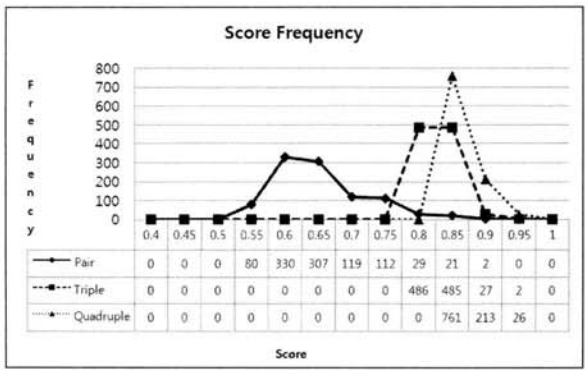
마이크로어레이에 참여한 유전자 중에서 실질적으로 암분류에 영향을 미치는 유전자의 수는 매우 제한적이고, 대다수의 유전자는 암분류와 관련이 없는 잡음(noise)유전자다[12]. 따라서 인포머티브 유전자만을 뽑아서 축소된 유전자 집합을 이용하여 암 발현형(Phenotype) 분류분석에 사용할 수 있다[2]. 대부분의 기존 논문에서 인포머티브 유전자의 갯수를 50에서 100개로 고정한다[6].

동일한 생물학적 목적으로 수행된 마이크로어레이 데이터를 통합하여 샘플의 수를 확대하고 인포머티브 유전자를 선택하여 유전자 차원을 축소시키는 방법은 저자의 기존 방법을 활용하였다[18]. 동일저자는 샘플단위로 유전자의 발현값을 샘플 내 순위 값으로 치환하는 방식으로 독립적으로 생성된 마이크로어레이 데이터를 통합하고 각 유전자 별로 클래스가 완벽하게 분리되는 정도를 측정하여 인포머티브 유전자를 추출하였다.

일반적으로 분류분석은 트레이닝 집합(Training Set)에서 분류자를 찾아 독립 테스트데이터에 적용한다. 트레이닝집합 데이터에 특정한 분포를 가정하고 분류자를 만드는 모수적 방법과, 분포에 대한 가정을 하지 않는 비모수적 분류방법이 있다. 대표적으로 많이 사용되는 비모수적 방법이 SVM[7, 9], CART[3, 13] 등의 방법이 있다. SVM은 기계학습 알고리즘에 바탕을 둔 것으로 두 그룹(클래스)을 분리시키는 분류 초평면을 찾는 방법이다. SVM은 기존의 모수적 방법 보다 확장성이 좋고 모 집단에 대한 가정이 없다. 마이크로어레이 자료로부터 SVM을 이용하여 샘플을 분류할 수 있지만 유전자 변수를 선택하는 방법이 취약하다. 또한 커널과 같이 복잡한 함수를 사용함으로써 최종 분류자의 설명력이 떨어지고[1], 실험적으로 여러 가지 종류의 파라미터 조정을 필요로 하기 때문에 다소 복잡하다는 단점이 있다. CART는 의사결정트리에 기반한 것으로 각 노드의 분리변수는 유전자를 의미한다. 앙상블방법으로 k-Nearest Neighbor (k-NN) [5, 17]가 있다. k-NN은 새로운 샘플에 대하여 학습 데이터 개체 중에서 거리기반으로 유사한 것들을 선택하여 샘플의 클래스를 분류하는 알고리즘이다. 그러나 k-NN 알고리즘은 클래스를 구분하는 함수가 명시적으로 제시되지 않으며, 모든 유전자에 동일한 가중치를 부여하였을 경우 좋은 성능을 제공하지 못한다는 단점을 가지고 있다.

다른 클래스 분류 방법 중 데이터에 따라 처리되는 기계학습 방법으로 Tan이 제안한 k-TSP(k-Top Scoring Pair) [15] 방법이 있다. TSP는 가장 높은 점수를 갖는 유전자의 쌍(Pair)을 찾는 알고리즘이다. 모든 유전자 쌍 (G_i, G_j)에 대하여, "$G_i < G_j$" 관계가 두 클래스에서 나타나는 상대 빈도를 각각 구하여 그 차이를 계산하여 점수 함수로 사용한다. 이 점수가 높을수록 그 유전자 쌍은 두 클래스를 잘 구별한다고 말할 수 있으며 점수가 가장 큰 k개의 유전자 쌍이 k-TSP 분류자로 사용된다. TSP의 경우 단지 두 개의 유전자가 분류규칙이 되기 때문에 생물학적 해석의 용이함은 있으나 학습집합데이터를 약간만 변동시켜도 TSP 분류자 자체가 변할 수 있다. k-TSP 연구에서는 인포머티브 유전자를 먼저 추출하는 단계 없이 바로 클래스 분류자를 찾는 작업을 수행하며 전체 유전자가 분류규칙 생성에 관여하게 되므로, 데이터를 통합함에 따라 계산 량의 증가를 초래한다. 기존의 k-TSP방식을 확장하여 분류규칙에 참여하는 유전자의 수를 세 개(Triple)로 한정하고 모든 샘플에 대해 해당 유전자 조합의 값의 대소 관계를 분석하여 두 클래스를 가장 잘 구분 짓는 관계를 찾아내는 방법으로 저자의 k-TST (Top Scoring Triple) [18]의 방법이 있다. 규칙의 정확도를 위하여 높은 점수를 가지고 있는 유전자 조합 중 상위 k개의 규칙을 분류 규칙으로 분리해 내고, 각 규칙에 따라 독립된 샘플의 클래스를 판정하여 그 결과를 통합한다. 이 때 유전자의 샘플 내 순위 값이 활용된다. 유전자 수를 늘림으로써 하나의 규칙에 참여하는 유전자의 수를 일반화 하고자 하는 시도의 첫 단계라고 볼 수 있다.

본 논문에서는 하나의 규칙에 참여하는 유전자의 수를 일반화하는 k-TSN(Top Scoring N) 모델을 제안한다. (그림 2)는 페어(Pair)규칙, 트리플(Triple)규칙, 쿼드루플(Quadruple) 규칙에서의 점수빈도를 나타낸다. 점수는 규칙이 "정상"샘플 집합에서 나타나는 확률과 "암"샘플 집합에서 나타나는 확률의 차이 값이다. 높은 점수를 갖는 규칙은 더 식별력이 높은 분류규칙을 의미 한다. Latulippe [11] 데이터를 트레이



(그림 2) 규칙을 구성하는 유전자수에 따른 점수 빈도

닝 데이터 세트로 사용하고 규칙에 참여하는 유전자수를 각기 2, 3, 4개로 고정시키고, 각 상위 1000개의 점수를 구하여 빈도 그래프를 보면 유전자수가 증가 할 수록 더 높은 점수를 갖는 빈도가 커짐을 알 수 있다. 높은 점수는 "정상", "암" 클래스를 더 잘 구분하는 것을 의미하므로 참여 유전자수의 일반화가 분류정확도를 높일 수 있을 것이다.

k-TSN 분류자는 k개의 결정 규칙으로 이루어져 있으며, 각 규칙은 2개에서 N개 유전자의 순위 비교 관계식과 클래스표지로 이루어진다.

기존의 k-TSP, k-TST방법은 모든 가능한 유전자 조합의 규칙을 생성하여 각 규칙에 대한 점수를 모두 계산하여 상위 k개의 규칙을 선택하는 방식이다. 시간복잡도는 생성되는 규칙의 수에 비례한다. 유전자의 수가 n일때 기존의 방식을 k-TSN에 직접 적용한다고 가정 했을 경우, k-TSN의 예상 시간 복잡도는 $O(n^n)$(표 1)가 될 것이므로, 이는 실행 가능한 시간이 아니다. 기존의 방법과 달리 본 논문에서 제안하는 방법은 모든 가능한 유전자 조합을 생성하지 않는다. 짧은 규칙(참여유전자수가 적은 규칙)을 이용하여, 생성 가능한 긴 규칙(참여유전자수가 많은 규칙)의 기대 점수를 계산한 후 실질적으로 규칙의 결합여부를 결정하는 방식이다. 이 방식은 계산시간과 메모리공간의 사용량을 절대적으로 줄여준다. 휴리스틱스를 사용하여 불필요한 규칙의 결합을 줄여서 많은 수의 규칙이 빠른 시간안에 k개의 상위 암 분류 결정규칙으로 수렴하게 하고, 정분류율을 높였다.

논문의 구성은 다음과 같다. 2장에서는 암 분류결정 규칙 생성과 관련된 정의, 알고리즘 설명, 휴리스틱을 기술하고 3장에서는 실험을 통한 실행시간 분석과 정분류율을 비교하고 4장에서는 논문에 대한 결론을 맺는다.

2. 암 분류 결정규칙의 생성

본 논문에서 제안하는 분류자는 트레이닝 데이터 세트를 이용하여 k개의 암 분류 결정 규칙을 생성하여, 독립된 샘플의 클래스 표지를 판정한다. 이 장에서는 유전자 발현 마이크로어레이 데이터에서 결정규칙 선정을 위한 점수에 대한 정의를 내리고, 효율적인 규칙 조합 프로세스를 설명하고, 상위점수 리스트로의 빠른 수렴을 위한 휴리스틱, 알고리즘 요약을 보여준다.

2.1. 암 분류 결정 규칙

유전자의 발현 값은 실수 값으로 주어진다. 이 값은 생물 실험실에서 진행하는 마이크로어레이 실험에 따라 스케일이 달라진다. 따라서 발현 값 자체 보다는 샘플의 클래스 표지 판정과 더 직접적으로 관련이 있는 것은 한 샘플 내에서의 상대적 순위 값이다. 본 논문에서 하나의 암 분류 결정규칙은 아래와 같이 정의된다.

1) 최대 N개의 유전자 2) 관련유전자간의 순위 비교 관계식 3) 관계식을 만족했을 때의 클래스 표지.

암분류결정규칙 = {관계식 : $G_1 > G_2 > G_3 > \ldots > G_n$;
클래스표지 : "암"}

관계식에 참여하는 유전자의 수는 2개에서 N개 이다.

2.2. 점수

각 규칙에 대하여 점수가 계산되고 트레이닝 데이터 세트를 이용하여 분별력 상위 집합의 규칙이 생성된다. 점수는 아래와 같이 정의된다.

$$S_N = NormalHit \;/\; NormalCount$$
$$S_T = TumorHit \;/\; TumorCount$$
$$Score = \left\{ \begin{array}{l} S_N - S_T \;\; (\text{클래스표지:"정상"}) \\ S_T - S_N \;\; (\text{클래스표지:"암"}) \end{array} \right\},$$

*NormalCount(TumorCount)*는 트레이닝 데이터 세트에 존재하는 "정상" 샘플("암" 샘플)의 수, *NormalHit(TumorHit)*은 "정상" 샘플("암" 샘플) 집합 중에서 관계식을 만족하는 샘플의 수를 의미한다. 높은 점수는 높은 클래스 분별력을 의미한다. 점수는 0 과 1 사이의 범위에 있다. S_N("정상 점수")는 트레이닝 데이터 세트에 존재하는 "정상"샘플 중에서 관계식을 만족하는 샘플의 비율이고, S_T("암 점수")는 "암" 샘플 중에서 관계식을 만족하는 샘플의 비율이다. 점수는 S_N과 S_T값의 차이 이다. S_N값이 S_T값 보다 더 큰 값을 갖는 관계식은 "정상" 클래스를 클래스 표지로 보유하게 된다. S_T값이 S_N값 보다 더 큰 값을 갖는 관계식은 "암" 클래스를 클래스 표지로 보유하게 된다.

〈표 1〉 생성규칙의 수와 예상 시간복잡도

	k-TSP	k-TST	k-TSN
가능한 유전자 조합 규칙수	${}_nP_2$	${}_nP_3$	${}_nP_2 + {}_nP_3 + \cdots + {}_nP_n$
예상 시간 복잡도	$O(n^2)$	$O(n^3)$	$O(n^n)$

2.3. k-TSN 분류자

<표 2>에서와 같이 최종적으로 k개의 상위 점수를 갖는 규칙의 집합이 분류자를 형성하게 된다. 최적의 k값은 각각의 트레이닝 데이터 세트에 LOOCV(Leave One Out Cross Validation)를 이용하여 찾는다.

독립샘플의 클래스표지를 판정하는 과정은 아래와 같다. 샘플이 규칙의 관계식을 만족하면 규칙의 클래스 표지를, 만족하지 못한다면 규칙의 클래스 표지의 상반 클래스표지를 할당한다. 이렇게 k개의 암분류 결정규칙을 동일 비중으로 적용하여 다수를 얻은 클래스를 최종클래스로 판정한다. 대다수 투표 방법은 복수 개 규칙의 판정 결과를 결합시켜 다수로 나타난 클래스표지를 그 샘플의 최종 클래스표지로 판정 하는데 많이 사용되는 방식이다. 동수가 있을 수 있으므로 k를 홀수로 지정한다. 알고리즘의 성능 평가를 위하여 테스트 데이터 세트에 있는 모든 샘플의 판정 클래스표지와 실질 클래스표지를 비교한다. 정분류율은 아래와 같이 정의된다.

$$\text{정분류율} = \frac{\text{정확하게 판정한 샘플의 수}}{\text{전체 샘플의 수}}$$

2.4 분류규칙의 생성

유전자의 모든 조합을 고려한 규칙의 생성과, 규칙에 대한 점수 계산은 오랜 계산 시간과 메모리 공간을 요구한다. 이미 점수가 계산된 짧은(참여 유전자의 수가 적은) 규칙을 조합하여 새로이 생성이 가능한 긴(참여 유전자의 수가 많은) 규칙의 기대 점수를 계산하다. 이 기대 점수가 최근 규칙 리스트의 최소 점수 보다 작은 경우에는 새로운 규칙의 실질 점수는 계산하지 않는다.

먼저 향후 조합의 요소로 사용될 2-유전자 규칙을 생성한다. 이 요소 규칙을 사용하여 유전자의 수가 3에서 N이 되는 긴 규칙을 형성해 나간다. 2-유전자 규칙 중에서 k개의 상위 점수를 가진 규칙이 초기 k-규칙리스트를 구성한다. 기대 점수의 최대값이 k-규칙리스트의 최소 점수 보다 큰 경우만, 긴 규칙을 생성하여 실질 점수를 계산하고, k-규칙리스트를 갱신한다.

<표 3>과 <표 4>를 이용하여 규칙 조합프로세스를 설명한다. 규칙 A, B는 모두 클래스표지가 "정상"인 규칙으로 새로운, 긴 규칙인 C를 조합하기 위한 요소 규칙으로 사용된다.

<표 4>는 조합하고자 하는 긴 규칙 C의 기대 점수의 최대값과 최소값을 보여준다.

먼저 규칙 C의 S_N 점수와 S_T 점수를 예측한 후에 이를 근거로 규칙 C의 점수를 예측한다. 규칙 C의 점수를 예측하기 위해 S_N의 범위와 S_T의 범위를 예측해 볼 수 있다. 규칙 C의 S_N은 "G1<G2"와 "G2<G3"을 동시에 만족해야 한다. "G1<G2"의 S_N은 규칙 A의 S_N인 p와 같고, "G2<G3"의 S_N은 q이므로 두 요소 규칙의 관계식을 동시에 만족하는 "G1<G2<G3"의 S_N은 p와 q보다 커질 수 없다(그림 3).

S_T는 TumorHit / TumorCount이므로 그 값은 0 이상이지만 1보다 커질 수는 없다. 또한 r과 s가 둘 다 큰 수일 경우에만 (그림 4)와 같이 r과 s를 동시에 만족하는 샘플들이 어느 정도 존재함을 예상할 수 있다. 규칙 A의 S_T 점수인 r의 값이 0.7, 규칙 B의 S_T 점수인 s의 값이 0.5이라 할 때, 규칙 C의 예측 S_T 점수는 (0.7 + 0.5 -1) 인 0.2이다. r과 s의 합이 1보다 큰 경우(두 요소 규칙을 같이 만족하는 샘플이 있는 경우)의 예측 S_T 점수의 최소값은 $(r + s - 1)$ 이고 두 요소 규칙을 만족하는 샘플이 없는 경우는 물론 0 이다. 간단한 표기를 위하여 예측 S_T 점수의 최소값을 $Max(r + s - 1, 0)$로 표기한다. 따라서 규칙C의 점수는 최대 $Min(p, q) - Max(r+s-1, 0)$ 의 값을 가질 수 있다.

규칙 C의 예측 최대값은 아래와 같다:

$$\text{Score}_{max} = Min(p, q) - Max(r + s - 1, 0).$$

〈표 2〉 k-암분류결정규칙

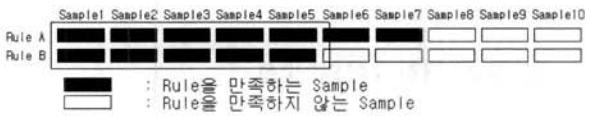
순위	규칙	점수
1	{$G_1 < G_2 < G_3$; "정상"}	0.9
2	{$G_6 < G_3$; "암"}	0.87
:	:	:
k	{$G_4 < G_1 < G_7$; "정상"}	0.76

〈표 3〉 조합할 두 개의 요소 규칙

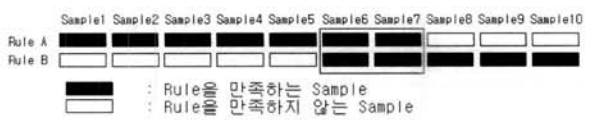
규칙	A	B
관계식	$G_1 < G_2$	$G_2 < G_3$
클래스표지	"정상"	"정상"
S_N	p	q
S_T	r	s
점수	$p-r$	$q-s$

〈표 4〉 조합될 새로운 규칙

규칙	C	
관계식	$G_1 < G_2 < G_3$	
클래스표지	"정상"	
S_N(기대값)	최소값	최대값
	$Max(p+q-1, 0)$	$Min(p, q)$
S_T(기대값)	최소값	최대값
	$Max(r+s-1, 0)$	$Min(r, s)$
점수(기대값)	$S_N - S_T$	



(그림 3) 예측 SN 최대값



(그림 4) 예측 ST 최소값

이 값이 k-규칙 리스트의 최소 점수보다 크다면 규칙 C는 조합해볼 가치가 있으므로 조합 후 실질 점수를 계산한다. 만약 실질 점수가 k-규칙 리스트의 최소 점수보다 크다면 k-규칙 리스트는 갱신된다. 만약 그렇지 않더라도 규칙 C가 다른 규칙과 조합했을 때 더 높은 점수를 얻게 될 수도 있는데, 이때 최대값은 실질 S_N 이다. 만약 S_N마저 k-규칙 리스트의 가장 작은 점수보다 작다면 그 조합은 무의미하므로 버린다. 조합에 참여한 규칙이 그 외의 모든 규칙과 한 번씩 조합을 시도해봤다면, 그 규칙은 더 이상 앞으로 있을 추가적인 조합에 사용될 필요가 없으므로 저장공간에서 삭제한다. 조합이 진행될수록 k-규칙 리스트는 높은 점수 규칙들로 수렴되고, 요소 리스트에는 낮은 점수의 규칙들만 남게 된다. 조합-갱신과정을 반복한 후 더 이상 조합할 요소 규칙이 없거나 조합해도 좋은 점수가 나올 수 없으면 k-규칙은 완성되고, 독립테스트데이터를 이용한 판별정확도 측정에 들어갈 수 있다.

2.5. 휴리스틱(Heuristics)

k-규칙 리스트를 좀 더 빠르게 높은 점수를 갖는 규칙들로 수렴시켜, 의미 없는 조합 횟수를 줄이고, 분류 성능을 보다 높일 수 있는 방법을 아래와 같이 고려하였다.

- 동일클래스 판별 규칙끼리 조합한다. 서로 다른 클래스 표지를 보유한 규칙 사이의 조합은 높은 점수를 얻기 힘들다. "정상클래스판별" 규칙이 유의 하다면 S_N은 높고 S_T는 낮을 것이다. "암클래스판별"규칙은 그 반대이다. 서로 다른 클래스 규칙을 조합하면 예상 S_N 의 최대 점수는 *(Min(p, q))*, 예상 S_T 의 최대 점수는 *(Min(r, s))*로 조합 후에 생기는 규칙은 S_N과 S_T의 최대값이 모두 낮게 나올 것이다. S_N이 높은 규칙끼리 조합하면 높은 점수의 "정상클래스판별" 규칙을 얻기 쉽고, S_T가 높은 규칙끼리 조합하면 높은 점수의 "암클래스판별" 규칙을 얻기 쉬울 것이다. S_N 점수로 정렬된 요소리스트와 S_T 점수로 정렬된 요소리스트를 따로 유지한다.
- "정상클래스판별" 규칙과 "암클래스판별" 규칙의 수를 균형 있게 유지한다. 일반적으로 마이크로어레이 데이터는 정상세포 샘플의 수가 적고 그에 비해 암세포 샘플의 수가 많다. 이런 불균형한 데이터를 트레이닝 데이터로 사용할 경우 S_N이 S_T 보다 높게 나오기 쉽기 때문에, 클래스가 "정상"인 규칙만 뽑힐 가능성이 크다. "암분류결정규칙 = {관계식 : G1 > G2 > G3 > G4 ; 클래스표지 : "정상" }"과 같은 규칙은 독립 샘플을 "정상"으로 판정함에 있어서는 엄격하고 "암"으로 판정하기 쉽기 때문에, 분류자안에 "정상클래스판별" 규칙이 다수 뽑힐 경우 독립 샘플데이터의 정확도 측정 시 암 샘플에 대해서는 아주 잘 맞추지만 정상샘플에 대해서는 그렇지 못할 것이다. 이를 방지하기 위해 점수의 차이는 다소 감수하더라도 k-규칙 리스트에는 "정상클래스판별" 규칙과 "암클래스판별" 규칙이 균형 있게 선택될 필요가 있다.

2.6. 알고리즘요약

이 장에서는 다양한 유전자 개수로 이루어진 암 분류 결정규칙의 생성 알고리즘을 요약한다. 주 알고리즘인 k-TSN 알고리즘(그림 5)은 *MakePairRule()*와 *Join()* 알고리즘을 호출한다. *MakePairRule*()은 모든 경우의 두 개의 유전자 쌍에 대해 점수를 계산한다. 핵심 알고리즘인 *Join* (그림 6)은 새롭게 조합하려고 하는 긴 규칙의 점수의 범위를 예상하여, 실질 점수의 계산 여부를 결정한다. 예상 점수가 최근

```
k  TSN Algorithm

Input: a Learning Dataset(LS),  k
Output: k  Rule List

1.        PairRule ← MakePairRule(LS).
2.        Initialize 2 Empty Arrays: NormalPairRules and
        TumorPairRules.
3.        Repeat for each rule R in PairRule
4.        If R is Normal Rule Then insert R into NormalPairRules.
5.        Else insert R into TumorPairRules.
 End
6.        NormalRules←Join(NormalPairRules,k/2,Normal).
7.        TumorRules←Join(TumorPairRules,k/2,Tumor).
8.        Return the result of 6 and 7.
```

(그림 5) k-TSN 알고리즘

```
Join Algorithm

Input: PairRuleList, k(number of rules to be selected),
     Rule Type(Normal or Tumor)
Output: k-Rule List

1.        k-Rule List ← Initialize an Empty Array.
2.        Repeat for k times
3.          Select the best (High Score) Pair Rule.
4.          Insert the Rule into k-Rule List.
5.          Remove the Rule from PairRuleList.
          End
6.        minScore ← the lowest Score from k-Rule List.
7.        If Rule Type = Normal then PrimaryScore ← NS,
 SecondaryScore ← TS.
8.        Else PrimaryScore ← TS, SecondaryScore ← NS.
9.        ElementList ← Sort PairRuleList by PrimaryScore.
10.       While ElementList is not empty
11.         Remove Rules from ElementList that have   lower PrimaryScore
than minScore.
12.         If ElementList is empty then return k-Rule List.
13.         HighRules ← Select the Rules
   (possibly multiple) that have highest
    PrimaryScore from ElementList.
14.          LowRules ← Select all Rules from ElementList except the
rules in HighRules.
15.           Delete Rules in 13 from ElementList.
16.           For each HR in HighRules
17.             For each LR in LowRules
18.               If ChkAddable(HR,LR)=false then goto17
19.               UpperBound = min(HR.PrimaryScore,
      LR.PrimaryScore).
20.               LowerBound = max(HR.SecondaryScore +
      LR.SecondaryScore-1, 0).
21.               If UpperBound-LowerBound < minScore
then  goto 17.
22.               JoinR ← Combine(HR, LR).
23.               If JoinR.Score > minScore then
24.                 Insert JoinR into k-Rule List.
25.                 Erase a rule that has the lowest Score from k-Rule List.
26.                 Update minScore.
                  End
27.               If JoinR.PrimaryScore > minScore
28.                  Insert JoinR into ElementList.
              End
                End // For each LR in LowRules
    End // For each HR in HighRules
 End // While ElementList is not empty
29.       Return k-Rule List.
```

(그림 6) Join 알고리즘

k-규칙리스트의 최소 점수 보다 크다면 실질 점수를 계산하고, 작다면 실질 계산을 생략한다. 계산된 실질 점수가 최근 k-규칙리스트의 최소 점수 보다 크다면, k-규칙 리스트를 갱신한다. 또한 계산된 실질 S_N(정상판별규칙의 조합시)이 정상 k-규칙 리스트의 최소 점수 보다 크다면 요소 리스트에 삽입되어 향후 조합프로세스에 사용된다. *Join()* 알고리즘에서 호출되는 *ChkAddable()* 알고리즘은 유전자로 본 두 규칙의 조합가능성을 조사하며, *Combine()* 알고리즘은 실질적으로 요소규칙들을 연결하여 긴 규칙을 생성함과 동시에 실질 점수를 계산한다.

3. 실험 및 결과

3.1 실험 환경

이 장 에서는 전립선암 마이크로어레이데이터를 이용하여 본 알고리즘의 실행시간과 분류 정확도를 비교하였다. 비교 대상 알고리즘으로는 k-TSP와 저자의 선행연구인 k-TST [18] 이다. 저자의 선행연구에서 k-TST를 SVM과 같은 다른 기존의 방법과 광범위한 비교를 하였으므로 논 논문에서는 주로 k-TST, k-TSP와 본 알고리즘과의 비교에 집중한다. k-TSP는 Tan[15]에서 제공하는 실행파일을 사용하였고, k-TST는 선행연구의 실행파일을 사용하였다. 사용된 마이크로어레이 데이터는 전립선암 세포를 이용한 동일한 생물학적 목적의 실험 데이터로서, 논문의 저자를 통하여 개별적으로 수집하였다. 플래트폼은 Affymetrix HG_95AV2 [19] 이다. 편의를 위하여 사용된 마이크로어레이 데이터를 각 저자의 성(Singh [14], Welsh [16] and LaTulippe [11])으로 언급한다. 사용된 마이크로어레이 데이터의 내역을 아래 <표 5>에 표시하였다.

3.2. LOOCV를 이용한 최적의 k선택

최적의 k값은 트레이닝 데이터 별로 다르다<표 6>. k값을 5에서 15로 증가시켜 나가면서 기본적인 LOOCV(Leave One Out Cross Validation)을 이용하여 최적의 k값을 찾는다. k-TSP와 k-TST를 위한 최적의 k값은 [18]을 사용하였다.

〈표 5〉 전립선암 마이크로어레이 데이터

데이터	유전자 수	정상 샘플의수	암 샘플의수	총 샘플수
Welsh	12626	9	24	33
LaTulippe	12626	3	23	26
Singh	12600	50	52	102

〈표 6〉 최적 k 값과 LOOCV정확도

트레이닝 데이터 세트	최적 k	정확도(%)
Welsh	9	100
Latulippe	13	100
Singh	9	92.16
Welsh+Latulippe	7	98.31
Welsh+Singh	5	91.85
Latulippe+Singh	9	90.63

3.3. 실행시간 분석

이 장에서는 k-TSP, k-TST, k-TSN의 실행시간을 비교분석한다. k-TST와 k-TSN은 첫 단계로 전체 유전자로 부터 1%의 인포머티브유전자를 먼저 추출하여 이 인포머티브 유전자 만을 암 분류 결정규칙의 생성에 사용한다. 대부분의 논문들에서 약 50에서 100개 [6] 사이의 인포머티브 유전자를 사용하는 것으로 언급되어 있다. 본 알고리즘에서는 총 유전자로 부터 120개(1%)의 인포머티브 유전자를 추출하여 분류분석에 활용하였다.

<표 7>은 전립선암 마이크로어레이 데이터의 실행시간을 보여준다. k-TSN이 다른 두 개의 알고리즘 보다 실행시간이 짧다는 것을 보여 준다. k-TSN알고리즘에서 상위 k-규칙 리스트의 수렴속도는 트레이닝 데이터 세트에 따라 달라진다. Latulippe같이 전체 샘플의 수가 적고, 정상샘플의 수와 암샘플의 수가 매우 비대칭인 트레이닝 데이터에서는, 상위 분류 규칙을 생성하는 데 긴 시간이 소요된다. Latulippe 데이터는 26개의 총 샘플에서 3개의 정상샘플, 23개의 암샘플로 이루어져 있다. 하지만 Singh과 같이 샘플의 수가 크고, 정상샘플과 암샘플의 수가 균형을 이루는 경우에는 상위 분류 규칙을 생성하는 데 적은 시간이 소요된다. 위의 <표 5>에서 보여주는 Singh 데이터의 총샘플의 수는 102개 이고 정상샘플의 수가 50, 암샘플의 수가 52개로 구성되어 있다. 트레이닝 데이터로서 Singh이 사용된 경우가 Latulippe이 트레이닝데이터로 사용된 경우보다 실행시간이 적게 소요되었다. Singh에 대하여 23.782와 24.063초가 소요된 반면 Latulippe의 경우는 105.515와 105.718초가 소요되었다. 또한 <표 7>은 순위를 이용한 통합 트레이닝 데이터에 대한 실행시간도 보여준다. k-TST의 경우 마이크로어레이 데이터의 통합에 의하여, 샘플 수가 증가된 경우 샐행시간도 증가된 반면 k-TSN의 경우에는, 통합이 샘플수의 증가와 정상샘플과 암샘플수의 균형을 가져오는 효과가 생길 수 있으므로 빠르게 상위 암분류 규칙 리스트로 수렴하는 경향을 보인다. k-TSP와 비교해 볼 때, k-TSN은 가장 좋은 경우는 20배, 가장 좋지 않은 경우에도 4배 정도 빠르다. k-TST와 비교해 볼 때, k-TSN은 가장 좋은 경우는 20배, 가장 좋지 않은 경우에도 1.2배 정도 빠르다.

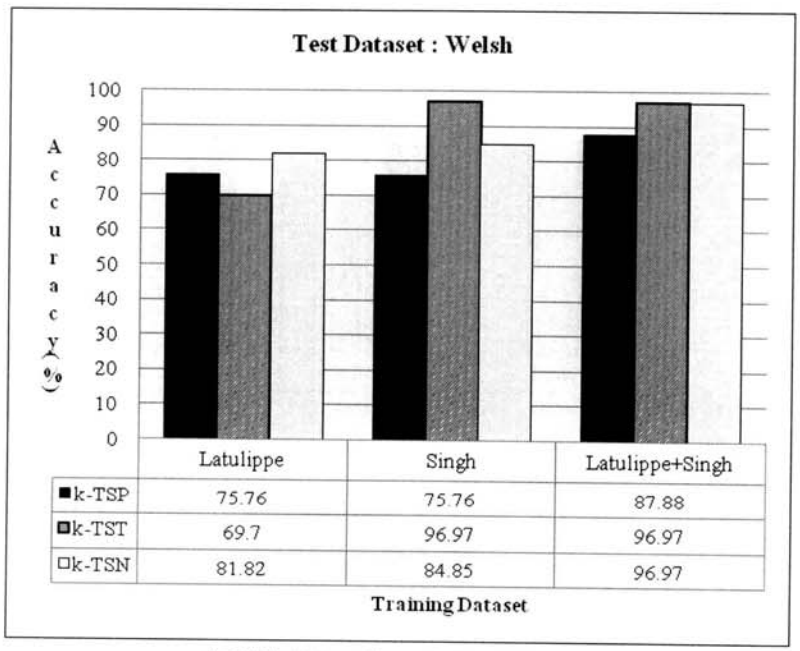
3.4 정분류율

이 장에서는 트레이닝 데이터를 이용하여 얻어진 암분류규칙을, 트레이닝데이터와는 독립적인 테스트데이터에 적용하여 k-TSP, k-TST와 성능을 비교한다. 정분류율은 2.3 장에 정의 되어 있다. Welsh, Latulippe, 통합Welsh+Latulippe같이 샘플수가 적고, 암과 정상샘플의 수가 매우 차이가 많이 나는 데이터는 트레이닝 데이터로 적합한 데이터가 아니다. Welsh의 총 샘플 수는 33, Latulipe의 총샘플 수는 26이다.

이들 데이터가 트레이닝데이터로 사용된 경우 k-TSP, k-TST, k-TSN의 정분류율은 (그림 9)와 같이 대체적으로

〈표 7〉 전립선암 마이크로어레이데이터를 이용한 k-TSN, k-TST, k-TSN의 실행시간 분석(단위: 초)

테스트 데이터	트레이닝 데이터 세트	k-TSN			k-TST			k-TSP
		Informative Gene Selection	Classification	Total	Informative Gene Selection	Classification	Total	Total
Singh	Welsh	3.437	52.672	56.109	3.422	156.641	160.063	242.156
	Latulippe	2.718	102.297	105.515	2.765	133.891	136.656	459.422
	Welsh + Latulippe	5.797	12.703	18.500	6.203	287.578	293.781	318.703
Welsh	Singh	9.844	13.938	23.782	9.516	442.469	451.985	511.218
	Latulippe	2.750	102.968	105.718	2.828	134.093	136.921	459.015
	Singh + Latulippe	12.219	14.406	26.625	11.953	561.75	573.703	614.797
Latulippe	Singh	9.782	14.281	24.063	9.656	442.171	451.827	510.515
	Welsh	3.453	52.922	56.375	3.672	156.750	160.422	241.047
	Singh + Welsh	13.047	13.704	26.751	12.937	584.703	597.64	628.968



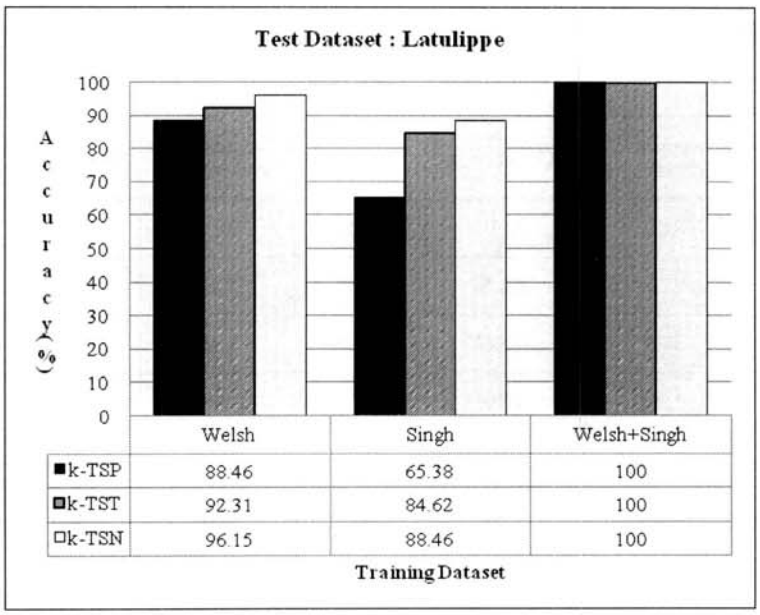
(그림 7) Welsh 테스트데이터의 정분류율

높지 않다. Singh, 통합Singh+Welsh, 통합Singh+Latulippe과 같이 샘플 수가 큰 데이터가 트레이닝 데이터로 사용된 경우(그림 7, 8), k-TSN의 정분류율은 k-TSP보다 높고, k-TST의 정분류율에 필적하는 결과를 보인다.

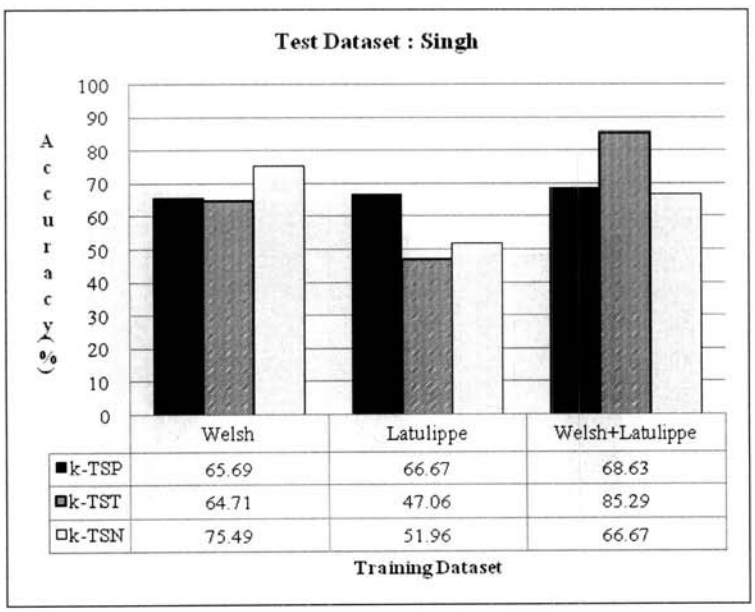
큰 샘플 크기(100개이상)를 가진 트레이닝데이터에 대하여 k-TSN은 k-TSP와 비교하여 최대 11%의 향상을, k-TST에 필적하는 결과를 보인다. 전립선암 마이크로어레이 데이터를 이용한 9개의 실험 중, 7개의 실험에서 k-TSN이 k-TST 보다 더 높거나 동일한 정분류율을 제공한다.

4. 결 론

본 논문에서는 마이크로어레이 데이터를 이용하여, 분류 규칙에 참여하는 유전자의 수를 일반화하는, 새로운 암 분류 방법을 제안하고, 실험을 통하여 기존의 방법인 k-TSP, k-TST와 실행시간과 정분류율을 비교하였다. 본 논문에서



(그림 8) Latulippe 테스트데이터의 정분류율



(그림 9) Singh 테스트데이터의 정분류율

제안하는 암분류자는 생물학적 해석이 용이하며, 데이터에 따라 처리되는 앙상블 기계 학습 방법이다. 본 알고리즘에서는 긴 규칙(참여 유전자의 수가 많은)을 생성할 때 점수가 이미 계산된 기존의 짧은 규칙을 이용하여 예상 점수를 계산한 후에 실질 계산 여부를 결정한다. 이 점이 기존의 앙상블 방법인 k-TSP, k-TST와 다른 점이다. 기존의 방법에서는 모든 가능한 유전자 조합에 대하여 모든 실질 계산을 수행한다. 본 논문에서 제안하는 방법이 계산시간과 메모리공간을 줄여준다.

본 논문에서는 단독 마이크로어레이 데이터 세트와 순위 기반 통합 마이크로어레이 데이터 세트가 모두 트레이닝 데이터 세트로 사용되었다. 기존의 방법과 실행시간과 정분류율을 비교하였다. 기존의 방법인 k-TSP와 비교하였을 때 9개의 실험 중에서 가장 좋은 경우는 20배, 가장 좋지 않은 경우에도 4배가 빠르다. 기존의 방법인 k-TST와 비교하였을 때 9개의 실험 중에서 가장 좋은 경우는 20배, 가장 좋지 않은 경우에도 1.2배가 빠르다. 트레이닝 데이터 세트가 마이크로어레이 통합에 의해 샘플 수가 증가 했을 때 특히 빠르다. 또한 트레이닝 데이터 세트의 샘플 수가 100이상으로 충분할 때 본 논문의 정분율이 k-TSP와 비교했을 때 11% 증가했고, k-TST에는 필적한다.

본 논문이 제안하는 분류자는 빠른 실행시간, 높은 정분

류율, 생물학적으로 직관적인 해석을 제공하며, 규칙에 참여하는 유전자들을 명확히 구분할 수 있으며, 규칙에 참여하는 유전자의 총수가 크지 않으므로 임상환경에서의 사용 가능성도 생각해 볼 수 있다. 마이크로어레이 데이터의 또 다른 대표적인 플래트폼으로 cDNA가 있다. 향후 cDNA방식으로 얻어진 마이크로어레이 데이터에 본 논문의 알고리즘를 적용한 성능비교를 계획하고 있다.

참 고 문 헌

[1] 서울대학교 통계학과 생물정보통계연구실, "마이크로어레이 자료의 통계적분석," 자유아카데미, 2005.

[2] M. Banerjee, S. Mitra, and H. Banka, "Evolutionary Rough Feature Selection in Gene Expression Data," *IEEE Transactions on Systems, Man, and Cybernetics-Part C*, Vol.37, pp.622-636, 2007.

[3] L. Breiman, J.H. Friedman, R.A. Olshen, and C.J. Stone, "Classification and Regression Tree," Champmans & Hall, 1984.

[4] C. Campbell, S. Mukherjee, P. Tamayo, S. Rogers, R. Rifkin, A. Engle, T. R. Golub, J. Mesirov, "Estimating Dataset Size Requirements for Classifying DNA Microarray Data," *Journal of Computational Biology*, Vol.10, pp.119-142, 2003.

[5] S. Dudoit and J. Fridlyand, "Classication in microarray experiments," Statistical Analysis of Gene Expression Microarray Data, Chapman and Hall, 2003.

[6] T. R. Golub, D. K. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J. P. Mesirov, H. Collier, M. L. Loh, J. R. Downing, M. A. Caligiuri, "Molecular classification of Cancer: class discovery and class prediction by gene expression monitoring," *Science*, Vol.286, pp.531-537, 1999.

[7] I. Guyon, J. Weston, S. Barnhill, V. Vapnik, "Gene selection for cancer classification using support vector machines," *Machine Learning*, Vol.46, pp.389-422, 2002.

[8] J. Han, M. Kamber, *Data Mining: Concepts and Techniques Second Edition*. San Francisco :Morgan Kaufmann, 2006.

[9] T. Joachims, *Learning to Classify Text Using Support Vector Machines: Methods, Theory, and Algorithms*. Kluwer Academic Publishers, 2003. http://svmlight.joachims.org/

[10] Y. Lai, B. Adam, R. Podolsky, J. She, "A mixture model approach to the tests of concordance and discordance between two large-scale experiments with two-sample groups," *Bioinformatics*, Vol.23, pp.1243-1250, 2007.

[11] E. LaTulippe, J. Satagopan, A. Smith, H. Scher, P. Scardino, V. Reuter, "Comprehensive gene expression analysis of prostate Cancer reveals distinct transcriptional programs associated with metastatic disease.," *Cancer Research*, Vol.62 pp.4499-4506, 2002.

[12] Y. Lu, J. Han, "Cancer classification using gene expression data," *Information Systems*, Vol.28, pp.243-268, 2003.

[13] J. R. Quinlan, *C4.5: Programs for Machine Learning*. San Francisco: Morgan Kaufmann Publishers Inc, 1993.

[14] D. Singh, P. G. Febbo, K. Ross, D. G. Jackson, J. Manola, C. Ladd, "Gene expression correlates of clinical prostate Cancer behavior," *Cancer Cell*, Vol. 1, pp.203-209, 2002.

[15] A. Tan, D. Naiman, L. Xu, R. Winslow, D. Geman, "Simple decision rules for classifying human Cancers from gene expression profiles," *Bioinformatics*, Vol. 21, pp.3896-3904, 2005.

[16] J. B. Welsh, L. M. Sapinoso, A. I. Su, S. G. Kern, J. Wang-Rodriguez, C. A. Moskaluk, "Analysis of gene expression identifies candidate markers and pharmacological targets in prostate Cancer," *Cancer Research*, Vol.61, pp.5974-5978, 2001.

[17] E. Wit, J. McClure, *Statistics for Microarrays: Design, Analysis and Inference*. NJ: John Wiley & Sons Inc., 2004.

[18] Y. Yoon, J. Lee, S. Park, S. Bien, H. C. Chung, S. Y. Rha, "Direct integration of microarrays for selecting informative genes and phenotype classification," *Information Sciences*, Vol.178, pp.88-105, 2008.

[19] http://www.affymetrix.com/index.affx

윤 영 미

e-mail : ymyoon@gachon.ac.kr

1981년 서울대학교 자연과학대학(학사)

1981~1983년 오하이오 주립대학 수학과 (학사수료)

1987년 스탠포드대학교 컴퓨터과학과(석사)

1987~1993년 IntelliGenetics Inc., Mountainview, California, Software Engineer

1995년~현 재 가천의과학대학교 IT학과 부교수

2008년 연세대학교 컴퓨터과학과(박사)

관심분야 : 데이터베이스 시스템, 데이터 마이닝, 바이오인포매틱스

변 상 재

e-mail : sayaya@snu.ac.kr

2007년 연세대학교 컴퓨터과학(학사)

2007년~현 재 서울대학교생물정보학 전공 석사과정

관심분야 : 데이터 마이닝, 생물정보학

박 상 현

e-mail : sanghyun@cs.yonsei.ac.kr

1989년 서울대학교 컴퓨터공학과(학사)

1991년 서울대학교 컴퓨터공학과(석사)

2001년 UCLA대학교 전산학과(박사)

1991~1996년 대우통신 연구원

2001~2002년 IBM T. J Watson Research Center Post-Doctoral Fellow

2002~2003년 포항공과대학교 컴퓨터공학과 조교수

2003~2006년 연세대학교 컴퓨터과학과 조교수

2006~현 재 연세대학교 컴퓨터과학과 부교수

관심분야 : 데이터베이스 보안, 데이터 마이닝, 바이오인포매틱스, XML