

분류시스템의 분류 규칙 발견을 위한 유전자 알고리즘[†]

(Genetic Algorithm to find Classification Rule for Classifier Systems)

김 대 희*, 박 상 호*

(Dae-Hee Kim, Sahng Ho Park)

요 약 분류시스템은 현재의 유용한 규칙들로부터 새로운 규칙들을 만들어 가기 위해 학습하는 규칙 기반 시스템이다. 본 논문에서는 방대한 데이터베이스에서 유용한 정보를 얻는 분류시스템의 분류 규칙 발견을 위한 유전자 알고리즘을 제안하였다. 제안된 방법을 자동차 보험 문제에 적용하여 제안된 유전자 알고리즘 기반 분류시스템의 성능을 평가하였다.

핵심주제어 : 유전자 알고리즘, 분류 시스템, 분류 규칙

Abstract A Classifier System is a system based on rules to invent new rules from the present useful ones. In this paper, Genetic Algorithms are proposed to find good classification rule of Classifier System which can extract useful information from huge database. The proposed scheme is applied to the real problems such as the car insurance problem to evaluate the performance of Genetic Algorithm based classifier systems.

Key Words : Genetic Algorithm, Classifier System, Classification Rule

1. 서 론

광 컴퓨터와 네트워크 기술의 발전으로 엄청난 양의 정보가 생성되고 있으며, 정보의 무제한적인 증가는 우리가 원하는 자료를 찾아내는 일을 보다 어렵게 만들고 있다. 따라서 축적된 자료를 효과적으로 관리하여 필요한 지식으로 가공할 수 있는가의 여부가 중요한 관심사가 되고 있고, 데이터베이스에 축적한 수많은 데이터에서 원하는 정보를 체계적으로 분류할 수 있는가에 관심이 모아지고 있

다. 정보를 분류한다는 것은 복잡성을 가진 수많은 데이터에서 일정한 패턴, 즉 규칙을 찾아내는 기술이다. 분류시스템(Classifier System)[1]은 현재의 유용한 규칙들로부터 새로운 규칙들을 만들어 가기 위해 점수할당과 규칙발견을 통해 학습하는 규칙 기반의 시스템이다. 학습에 의한 경험은 끊임없는 새로움과 복잡성 속에서 관련 있는 규칙들이 존재하면 미래의 행동을 좌우할 수 있다. 이러한 학습시스템에 적합한 알고리즘이 유전자 알고리즘(Genetic Algorithm)이다. 유전자 알고리즘은 탐색공간이 크거나 복잡성을 가진 문제를 해결하는데 유용하여 유전자 알고리즘을 이용한 분류시스템은 데이터 마이닝[2,3,4]에도 적용이 가능하다.

[†] 이 논문은 2001학년도 안동대학교 해외파견 연구비 지원에 의하여 연구되었음.

* 안동대학교 정보통신공학과

본 논문에서는 데이터베이스에서 유용한 정보를 얻는데 이용되는 분류시스템의 분류 규칙 발견을 위한 유전자 알고리즘을 제안한다. 유전자 알고리즘의 연산자 개발 및 연산자에 따른 유전자 알고리즘의 성능을 평가하여 학습 방법을 제안한다. 제안된 방법의 우수성을 보이기 위하여 자동차 보험 문제를 통하여 실제 문제에 적용됨을 보인다. 본 논문의 구성은 1장 서론에 이어, 2장에서는 유전자 알고리즘의 기본 개념을 설명한다. 3장에서는 분류 규칙 발견시스템에 대하여 설명하며, 4장에서는 분류 규칙 발견시스템을 위한 유전자 알고리즘을 제안하고, 5장에서는 제안된 알고리즘을 이용한 분류 규칙 발견 시스템의 성능을 평가한다. 마지막으로 6장에서는 연구 내용을 요약하고 결론을 맺는다.

2. 유전자 알고리즘(Genetic Algorithm)

유전자 알고리즘은 선택적 도태나 돌연변이 같은 생물 진화의 원리로부터 착안한 알고리즘으로서 확률적 탐색이나 학습 그리고 최적화를 위한 기법[5,6,7,8]이다. 일반적인 유전자 알고리즘의 수행 과정은 다음과 같다[5,9,10].

- (1) 문제에 맞는 염색체의 구조와 유전자 조작을 정의하고, 적합도(fitness) 함수를 설계한다.
- (2) 초기 모집단(Population)을 생성한다.
- (3) 모집단 각 개체의 적합도를 평가한다.
- (4) 적합도가 높은 유전자가 새로운 개체를 생성할 확률이 높아지도록 개체들을 선택한다.
- (5) 모집단의 선택된 개체들에 대해 유전자 조작을 수행함으로써 새로운 모집단을 생성한다.
- (6) 적합도가 높은 유전자들을 계속 확장시켜 나감으로서 원하는 해에 수렴하도록 위 3~5과정을 반복한다.

유전자 알고리즘의 연산자에는 재생산, 교배, 돌연변이 등이 있다. 재생산은 우수한 유전자들이 다음 세대에 많이 나타나도록 하는 연산자로서 적합도에 따라 개체들을 선택하여, 선택된 개체 수만큼의 우수한 유전자를 다음 세대에 생성하도록 한다. 교배는 두 부모의 염색체를 조합하여 바꾸어 두 개의 자식의 염색체를 만드는 유전자 조작이다. 선택된 부모의 우수한 유전자를 각 자식들에게 유전

되도록 한다. 돌연변이는 유전자를 일정한 확률로 변화시키는 조작이다. 돌연변이는 부모 유전자에게 없는 속성을 다음 세대에 도입하는 역할을 한다. 돌연변이는 국소 탐색(local minima)에서 벗어나기 위해서 필요하며, 돌연변이가 없는 경우에는 초기 유전자 조합 이외의 공간을 탐색할 수 없어 초기 조합에 적절한 해가 없을 경우 원하는 해를 구할 수 없는 경우가 있을 수도 있다.

3. 분류 규칙 발견시스템

3.1 분류 규칙 발견시스템의 구성

분류 규칙 발견 시스템은 내부의 모든 규칙들이 동일한 단순 형태를 갖는 메시지 전달방식의 규칙 기반 시스템이다. 분류 규칙 발견시스템(Classifier System)의 구성은 그림 1과 같이 일반적으로 세 단계의 활동으로 구별할 수 있다. 첫 번째 단계는 실행 시스템(Performance System)으로 외부환경과 직접 상호 작용하여 규칙을 생성하는 부분이다. 일반적으로 실행 시스템의 규칙들은 다양한 유용성이 있고, 그들 중 몇몇 혹은 심지어 대부분이 잘못된 것일 수 있으므로 규칙평가가 필요하다. 두 번째 단계는 규칙 중 어떤 것이 효율적인지를 결정하는 점수 할당 시스템(Credit Assignment System)이다. 활동의 세 번째 단계는 규칙 발견 시스템(Rule Discovery System)이다.

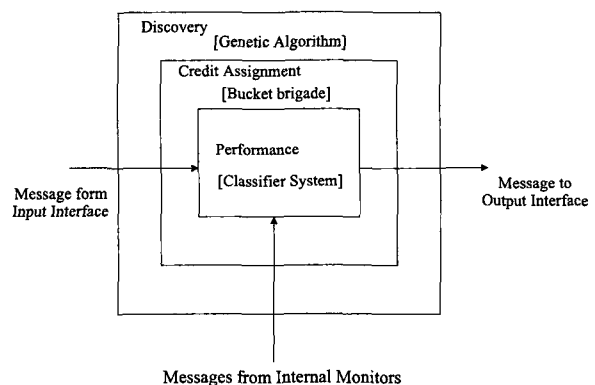


그림 1. 분류 규칙 발견 시스템의 일반적인 구성

3.2 실행 시스템(Performance System)

실행 시스템은 입력 인터페이스로부터 입력받은

메시지와 실행 시스템 내부에 포함하고 있는 메시지 리스트에 기록되어 있는 규칙의 조건과의 매칭 여부를 조사한다. 실행 시스템은 구성은 그림 2에서 볼 수 있듯이 4개의 기본 부분으로 구성된다. 메시지(Message)는 0과 1의 조합으로 구성되는 문자열이며, 규칙(Rule)의 가장 단순한 형태는 k-비트의 문자열이다. 분류자(classifier) 리스트는 실행 규칙을 기록하는 규칙 베이스로 규칙은 “조건 / 행동”의 형태이다. 조건 부분은 어떤 종류의 메시지가 규칙을 만족하는지를 명시하고 행동 부분은 규칙이 만족되었을 때 메시지가 어떻게 보내지는가를 명시한다. 행동은 $\{0,1\}^k$ 로 구성된 k-bit 문자열의 집합이며, 조건은 $\{0,1,\#\}^k$ 로 구성된 문자열의 집합이다. 여기서 “#”은 “don't care”를 나타낸다.

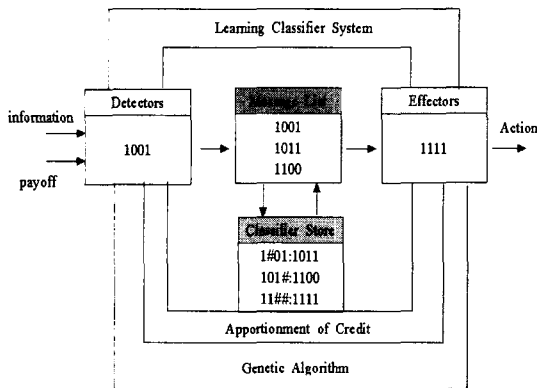


그림 2. 실행 시스템의 구성

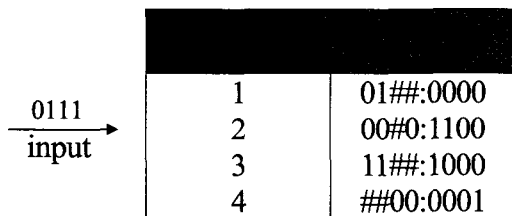


그림 3. 실행 시스템의 수행 과정

그림 3을 통해 실행 시스템의 수행 과정을 볼 수 있다. 입력 인터페이스로부터 메시지 “0111”이 입력되면 분류자(Classifier)의 조건을 만족하는지를 검사한다. 1번 조건과 일치하므로 “0000” 행동을 출력한다. 다음 단계에선 “0000”과 만족하는 조

건을 검사하여 2번 조건과 일치하는 “1100”을 출력한다. 이러한 과정을 반복하여 마지막에는 “0001”을 출력하게 된다.

3.3 점수 할당 시스템(Credit Assignment System)

주어진 시간에 활동하는 규칙들 중 어떤 규칙이 가장 효율적인가를 결정하기 위한 점수 할당 방법은 버킷 집단 알고리즘(Bucket Brigade Algorithm)을 사용한다. 알고리즘을 구현하기 위해, 각 메시지는 강도라고 불리는 양을 할당받고, 버킷 집단 알고리즘은 시스템에 대한 메시지의 전체적인 유용성을 반영하기 위해 강도를 조절한다. 각 시간 단계에서, 각 만족된 메시지는 강도에 기반 하여 입찰하고, 단지 가장 높은 값을 부른 메시지들만이 다음 시간 단계에 메시지 리스트에 기록된다. 입찰가격은 아래 식 과 같이 정의된다.

$$B(C, t) = bR(C)s(C, t) \quad (1)$$

여기서 $s(C, t)$ 는 시간 t 에서 메시지 C 의 강도이며, $R(C)$ 은 한정성으로, C 의 조건 부분에 있는 # 아닌 것의 개수를 조건 부분의 문자열의 길이로 나눈 것과 같으며, b 는 1보다 상당히 적은 상수이다. 입찰 가격의 크기는 분류자가 행동부분에 의하여 명세 되는 메시지를 새로운 메시지 리스트에 붙일 확률을 결정한다.

입찰에서 승리한 메시지 C 가 그것의 메시지를 메시지 리스트에 놓았을 때 메시지는 입찰 가격 $B(C, t)$ 의 양 만큼 강도 $s(C, t)$ 를 줄임으로서 특권을 지불한다.

$$s(C, t+1) = s(C, t) - B(C, t) \quad (2)$$

이 승리자(일치된 메시지들을 보낸 메시지들 $\{C'\}$)는 입찰 가격의 양만큼 그들의 강도가 증가한다.

$$s(C', t+1) = s(C', t) + aB(C, t) \quad (3)$$

여기서 $a = 1/(\text{no. of members of } \{C'\})$ 이다.

4. 유전자알고리즘을 이용한 규칙발견시스템

규칙 발견 시스템은 유전자 알고리즘을 이용하여 규칙을 발견하는 기능을 수행한다. 분류 규칙 발견시스템의 수행 중 점수 할당 문제에서 얻은 강도 $s(C, t)$ 를 적합도 함수로, 메시지 리스트의 일부를 개체군(Population)으로 보고 유전자 알고리즘을 이용하여 메시지 리스트를 바꾸고 새로운 규칙을 생성한다.

4.1 유전자 알고리즘의 학습방법

분류 규칙 발견시스템에서 학습은 1단계 실행시스템과 2단계 버킷 집단 알고리즘을 이용해 규칙 평가를 수행한 후 3단계 유전자 알고리즘을 이용하여 새로운 규칙을 발견해 가는 과정이다. 유전자 알고리즘의 기본적인 수행주기는 다음과 같다.

유전자 알고리즘이 그림 4와 같은 과정을 통하여 분류자를 만든다. 알고리즘은 알파벳 $\{1, 0, \#\}$ 위에 M 개의 문자열 $\{C_1, C_2, \dots, C_M\}$ 들의 집합 $B(t)$ 에서 할당된 강도 $s(C_j, t)$ 를 가지면서 다음과 같이 동작한다.

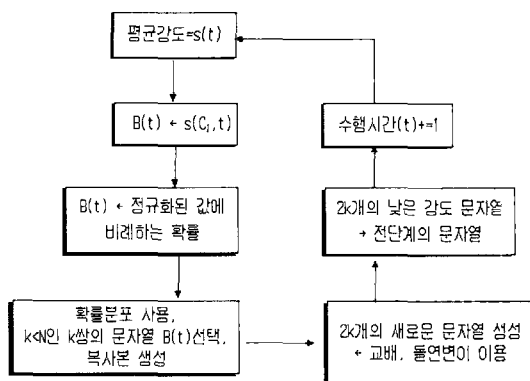


그림 4. 유전자 알고리즘이 분류자를 만드는 과정

- (1) $B(t)$ 에 있는 문자열의 평균 강도 $s(t)$ 을 계산하고, $B(t)$ 에 있는 각 문자열 C_j 에 정규화 된 값 $s(C_j, t)/s(t)$ 을 할당한다.
- (2) $B(t)$ 에 있는 문자열에 정규화 된 값에 비례하는 확률을 할당한다. 그리고, 이 확률분포를 사용하여 $n \ll M$ 인, n 쌍의

문자열을 $B(t)$ 에서 선택하고 복사본을 만든다. 본 논문에서는 n 쌍의 문자열을 선택하기 위해 문자열을 $B(t)$ 에 10%에서 50%까지로 각각 선택하였다.

- (3) 교배와 가능한 다른 유전연산자들을 복사된 각 쌍에 적용하고, $2n$ 개의 새로운 문자열을 만든다.
- (4) $B(t)$ 에 있는 $2n$ 개의 낮은 강도의 문자열을 3단계에서 새로 생성된 새로운 문자열로 대체한다.
- (5) 알고리즘의 다음 사용을 위해 t 를 $t+1$ 로 놓고 1 단계부터 반복한다.

유전자 알고리즘은 위의 과정을 반복하여 최적의 분류 규칙을 찾기 위해 학습해 간다.

4.2 교배 연산자

교배는 강도 $s(t)$ 값이 약한 문자열을 교체하기 위해 강도 $s(t)$ 이 강한 문자열을 부모 염색체로 선택하고, 선택된 부모 염색체를 조합하여 강도 $s(t)$ 가 강한 자식 염색체를 만든다. 교배는 선택된 문자열의 쌍에 교배 위치(Crossover Point)를 그림 5와 같이 4가지 형태의 제한된 교배 방법(Crossover Type)을 적용하였다.

- (1) 교배 방법 1 : 문자열 전체에서 임의로 $1 \leq i \leq k$ 가 되는 위치 i 을 잡는다. 여기서 k 는 문자열의 전체 길이 이고, 위치 i 는 교배 위치이다. 교배는 두 문자열에서 교배 위치의 왼쪽 부분을 교환한다.
- (2) 교배 방법 2 : 문자열에서 $1 \leq i \leq k - M_zone$ 가 되는 위치 i 을 임의로 잡는다. 위치 i 는 교배 위치이다. 교배는 i 번째 위치에서 M_zone 시작 전 위치까지의 부분을 교환한다. 즉 $1 \leq i \leq k - M_zone$ 와 M_zone 사이의 부분을 교환한다.
- (3) 교배 방법 3 : 교배 방법2에서와 같이 교배 위치를 잡고, 교배는 두 문자열에서 교배 위치의 왼쪽 부분을 교환한다.

(4) 교배 방법 4 : 문자열에서 $1 \leq i \leq k - M_zone$ 가 되는 임의의 두 지점 i 와 $i+1$ 을 잡고, 교배는 두 지점의 가운데 부분을 교환한다.

교배 방법 1, 2, 3은 일점 교배이며, 교배 방법 4는 복수점 교배이다. 본 논문에서는 확률적 규칙을 이용하기 위해 교배 확률은 각각 10%, 20%, 30%, 40%, 50%로 적용하였다.

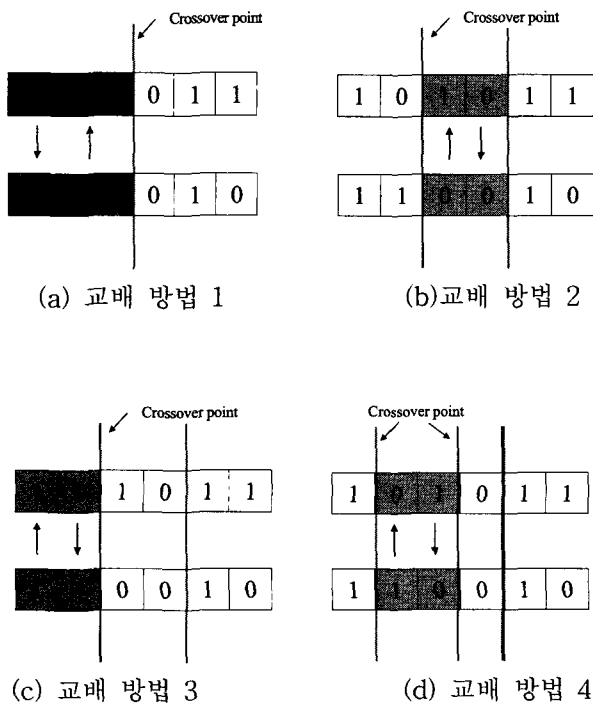


그림 5. 교배 방법 1, 2, 3, 4

5. 실험 및 고찰

분류시스템의 분류 규칙 발견을 위한 유전자 알고리즘의 성능을 자동차보험 문제에 적용하여 제안된 방법의 성능을 평가하였다. 실험에서는 자동차 보험 문제에서 사용되는 고객의 위험도를 평가하는 항목의 집합을 염색체로 구성하였다. 염색체란 유전자 알고리즘에서 하나의 해를 나타내는 개체이며, 이러한 해들의 집합을 개체군(Population)이라고 한다.

자동차 보험 문제는 실제 문제에 적용됨을 보이

기 위해 고안하였다. 자동차 보험회사에서 새로 가입하려는 고객을 평가하여 사고율이 높은 고객은 아주 비싼 보험료를 내거나 가입 자체가 불가능하도록 하고 있다. 실험에서는 자동차 보험에 새로 가입하려는 고객의 위험도를 평가하는 항목의 집합을 메시지로 구성하였다. 항목의 집합은 8개로 구성되며 다음과 같은 특성을 가지고 있다.

- $d_1 = \begin{cases} 1, & \text{장기간 무사고로 보험료 싼 운전자,} \\ 0, & \text{그렇지 않은 경우;} \end{cases}$
- $d_2 = \begin{cases} 1, & \text{21세이하 또는 60세이상운전자,} \\ 0, & \text{그렇지 않은 경우;} \end{cases}$
- $d_3 = \begin{cases} 1, & \text{삼년동안 두차례 이상 사고자} \\ 0, & \text{그렇지 않은 경우;} \end{cases}$
- $d_4 = \begin{cases} 1, & \text{중대과실 사고 경력자(10개항목),} \\ 0, & \text{그렇지 않은 경우;} \end{cases}$
- $d_5 = \begin{cases} 1, & \text{차종: 업무용,} \\ 0, & \text{차종: 개인출퇴근용;} \end{cases}$
- $d_6 = \begin{cases} 1, & \text{스포츠카, 지프, 중기계, 엠블런스 등 특수차량,} \\ 0, & \text{일반승용차;} \end{cases}$
- $d_7 = \begin{cases} 1, & \text{가입경력(운전경력) 삼년이하운전자,} \\ 0, & \text{그렇지 않은 경우;} \end{cases}$
- $d_8 = \begin{cases} 1, & \text{손해율이 높은 지역 운전자,} \\ 0, & \text{그렇지 않은 경우;} \end{cases}$

“행동”이 가지는 특성은 다음과 같다.

$$\text{행동} = \begin{cases} 00, & \text{보험 가입 거부,} \\ 01, & \text{보험료 인상,} \\ 10, & \text{보험료 동결,} \\ 11, & \text{보험료 인하.} \end{cases}$$

이 문제에서 규칙을 구현한 분류자는 다음의 형식을 갖는다.

01##11#0 / 01

여기서 “조건”은 “01##11#0”이 되며 “행동”은 “01”이 된다. 위 분류자가 의미하는 것은 ‘21세 이하의 운전자가 스포츠카를 운전하면 다른 조건에 상관없이 보험료는 인상된다는 행동을 나타낸다. 본 실험에 사용된 염색체는 그림 6과 같으며, 개체군은 그림 7과 같다. 실험을 위해 필요한 변수들의 값은 표 1과 같다. M_zone 은 실험을 위해 설계된 자동차 보험 문제에서는 그림 6의 항목의 집합 중 d_1 , d_8 을 지정하였다. 항목 d_1 과 d_8 은 전체 문자열에의 값이 변화하여도 커다란 의미를 주지 않는다. 각 교배 방법에 따른 i 의 범위와

M_zone 의 범위는 그림 8과 같다. 실험은 각각의 교배 방법에 따른 교배 확률 값을 다르게 적용하여 실험하였다. 실험 1은 교배 방법 1, 실험 2는 교배 방법 2, 실험 3은 교배 방법 3, 실험 4는 교배 방법 4를 사용하였으며 교배 확률은 각 실험에서 10%, 20%, 30%, 40%, 50%로 하여 실험하였다.

1	0	0	1	1	0	0	1
d1	d2	d3	d4	d5	d6	d7	d8

그림 6. 염색체의 디자인(자동차 보험 문제)

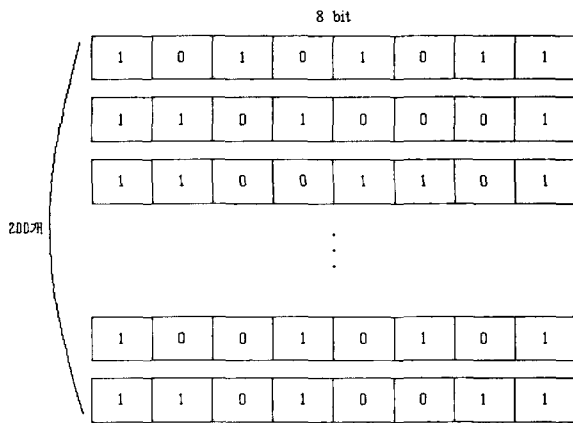


그림 7. 자동차 보험 문제의 개체군

표 1. 자동차 보험 문제의 실험 파라미터

	값
number of classifiers	200
probability of mutation	0.02
probability of crossover	1.0
number of iterations	2,000

1	2	...	k-1	k	
1	0	1	0	1	1
i 의 범위					

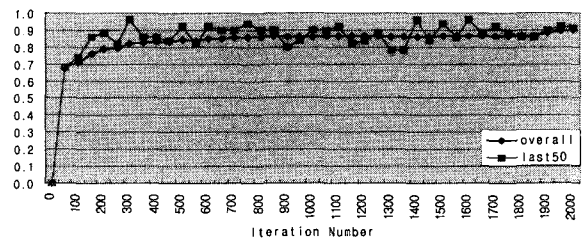
(a) 교배 방법 1에서 i 의 범위

1	2	...	k-1	k	
1	0	1	0	1	1
i 의 범위				M_zone 의 범위	

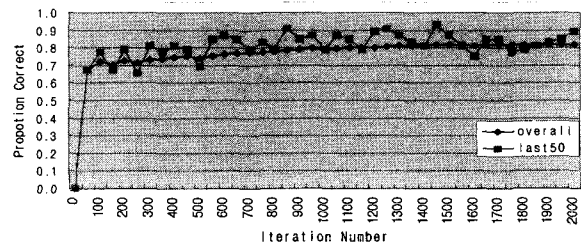
(b) 교배 방법 2, 3, 4에서 i 의 범위

그림 8. 교배방법별 i 의 범위

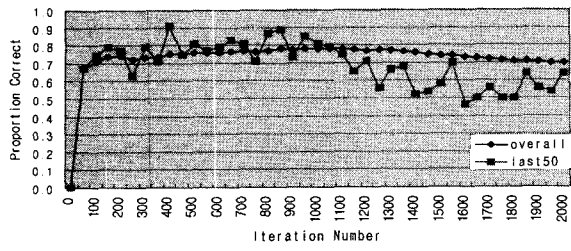
교배방법 1에 의한 실험결과는 그림 9와 같다. 여기서 가로축은 교배횟수이며 세로축은 개체군 중에서 주어진 문제에 대한 정확한 규칙들의 비율을 나타내어 연산횟수에 따른 정확한 규칙의 비율의 관계를 나타내었고 각 연산 횟수에 대한 전체 평균과 최종 50회의 평균값을 나타내었다.



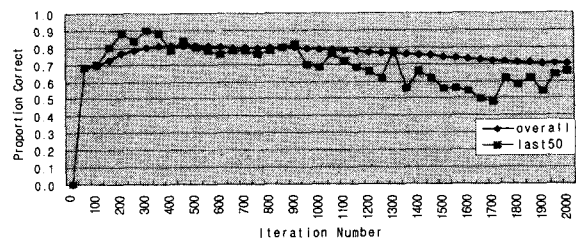
(a) 교배 확률 10%



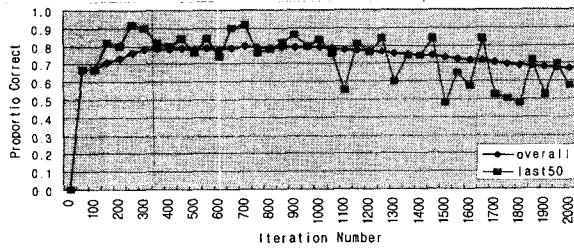
(b) 교배 확률 20%



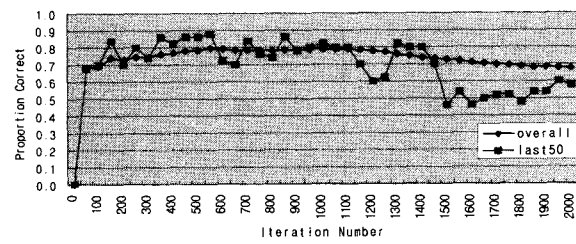
(c) 교배 확률 30%



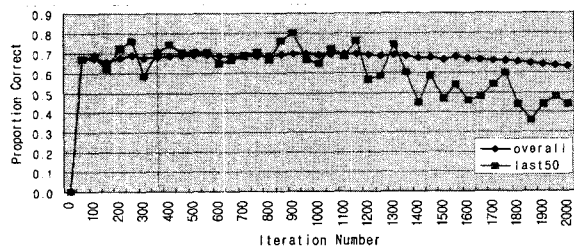
(b) 교배 확률 20%



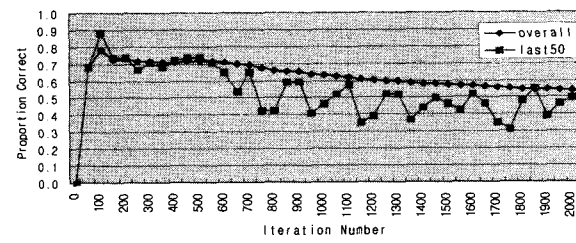
(d) 교배 확률 40%



(c) 교배 확률 30%



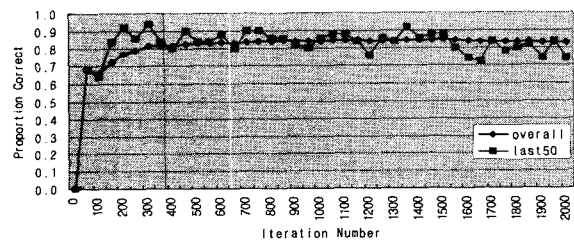
(e) 교배 확률 50%



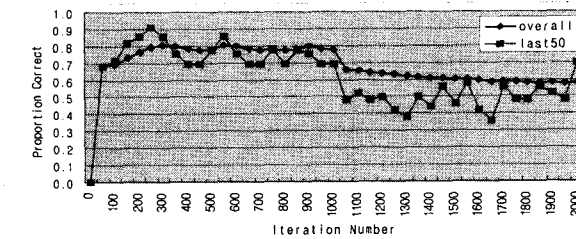
(d) 교배 확률 40%

그림 9. 교배 방법 1의 실험 결과

교배방법 2에 따른 실험 결과는 그림 10과 같다.



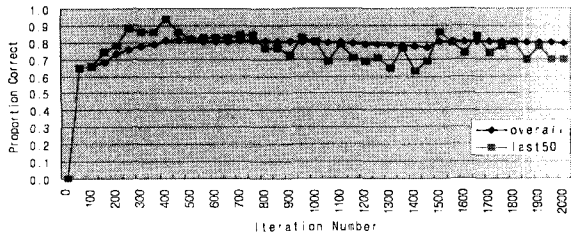
(a) 교배 확률 10%



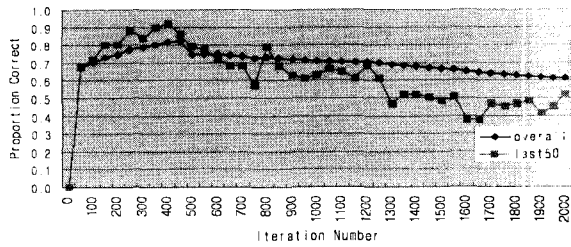
(e) 교배 확률 50%

그림 10. 교배 방법 2의 실험 결과

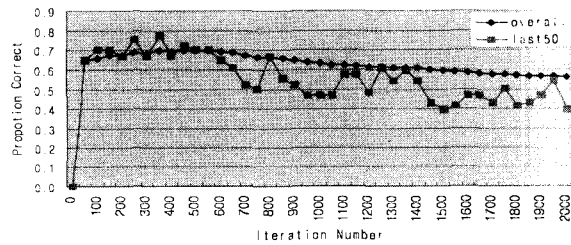
교배방법 3에 의한 실험 결과는 그림 11과 같다.



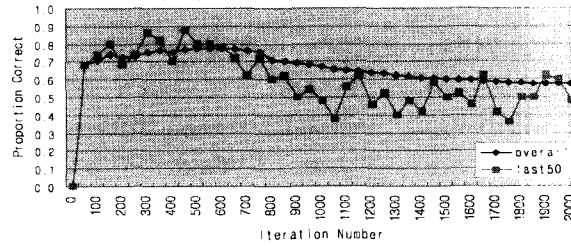
(a) 교배 확률 10%



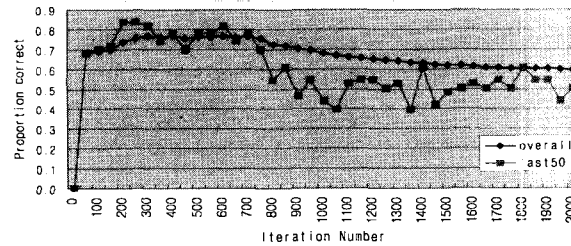
(b) 교배 확률 20%



(c) 교배 확률 30%



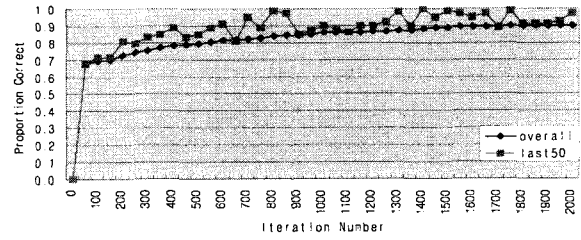
(d) 교배 확률 40%



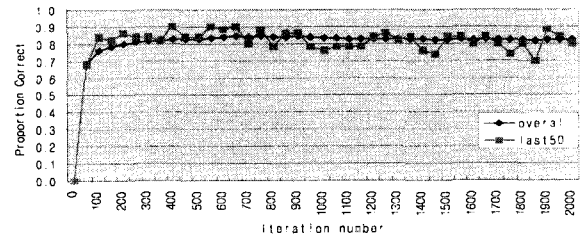
(e) 교배 확률 50%

그림 11. 교배 방법 3의 실험 결과

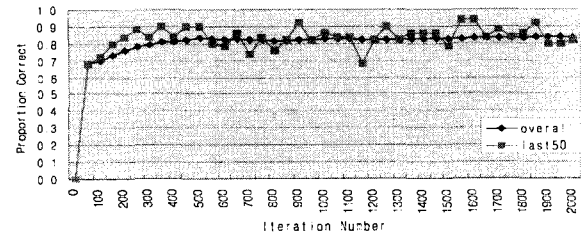
교배방법 4에 의한 실험 결과는 그림 12와 같다.



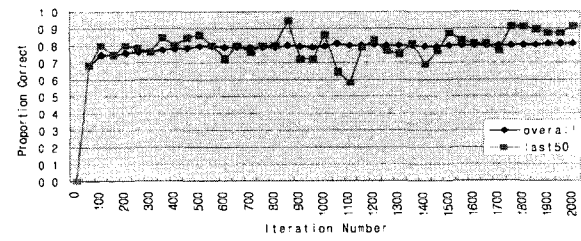
(a) 교배 확률 10%



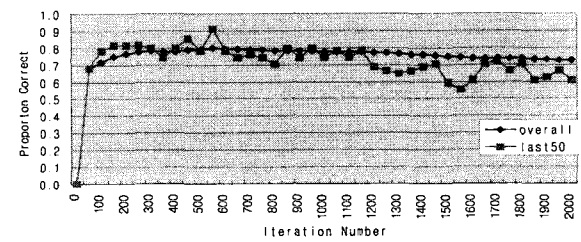
(b) 교배 확률 20%



(c) 교배 확률 30%



(d) 교배 확률 40%



(e) 교배 확률 50%

그림 12. 교배 방법 4의 실험 결과

실험결과에 의하면 각 교배방법에서 300회 이하의 적은 교배 횟수로 90%이상의 개체가 정확한 규칙을 나타내고 있어 제안한 방법은 빠른 시간 내에 분류시스템에 분류규칙을 발견해 줄 수 있음을 알 수 있다. 교배방법에 따른 성능은 다음과 같다. 교배 방법 1, 4에서는 교배 확률 값이 커질수록 성능이 저하됨을 보이고 있으며, 실행횟수가 증가함에 따라 성능 저하의 폭은 그리 크지 않았다. 교배 방법 2, 3에서도 교배 확률 값이 커질수록 성능이 저하됨을 보이고 있지만, 실행횟수가 1,000회가 넘으면서 성능 저하의 폭은 상당히 커짐을 보였다. 교배 방법 1에서 교배 확률이 10%, 20%일 때는 수행횟수가 증가함에 따라 성능은 좋아졌지만, 30%, 40%, 50%일 때는 1,100회 이상 수행하면서 부터 성능이 저하되고 있다. 교배 방법 2에서는 교배 확률이 10%일 경우를 제외하고 나머지 경우에는 일정 수행횟수 이후 성능이 점점 저하됨을 보이고 있다. 교배 방법 3에서도 교배 방법 2에서와 같이 교배 확률이 10%일 경우를 제외하고 나머지 경우에는 일정 수행횟수 이후 성능이 점점 저하됨을 보이고 있다. 교배 방법 4에서는 교배 확률이 10%~40%경우에는 수행횟수가 증가함에 따라 성능도 좋아졌다. 특히, 1,000회 이상 수행하였을 때 부터 일정이상의 성능을 유지하고 있다. 주어진 두 문제에서 교배 방법 4를 사용하고 교배 확률 10% 일 때 성능이 가장 뛰어났으며. 교배 방법 4, 교배 방법 1, 교배 방법 2, 교배 방법 3순으로 좋은 성능을 보이고 있다.

6. 결 론

본 논문에서는 분류 규칙 발견을 위한 유전자 알고리즘 학습방법을 제안하였다. 분류 규칙 발견 시스템의 1단계인 실행시스템에서 생성한 규칙들을 생성하고 2단계인 점수할당 시스템에서 버킷 알고리즘을 이용하여 규칙평가를 하고, 3단계인 규칙 발견 시스템에서 평가된 규칙들 중에서 유전자 알고리즘을 이용하여 규칙을 발견하였다. 유전자 알고리즘의 연산자를 개발하였으며, 제안된 방법을 자동차 보험문제에 적용하여 제안된 교배 방법들과 교배 확률에 따른 유전자 알고리즘의 학습방법의 성능을 평가하였다.

제안된 분류 규칙 발견시스템은 자동차 보험문

제를 통하여 실제 문제에 적용될 수 있음을 보였다.

참 고 문 헌

- [1] D. E. Goldberg, Genetic Algorithms in Search : Optimization & Machine Learning, Addison wesley, 1989.
- [2] M. Berry and G. Linoff, Data Mining Techniques, Wiley, 1997.
- [3] Breiman, J. H. Friedman, R. A. Olshen, and C. J. Dtone, Classification and Regression Trees, Chapman & Hall, 1984.
- [4] J.Han and M. Kamber, "Data Mining: Concepts and Techniques",Morgan Kaufmann, 2000.
- [5] John H. Holland, Adaptation in Natural and Artificial Systems, Ann Arbor : The University of Michigan Press, 1975.
- [6] David Beasley, David R. Bull, and Ralph R. Martin "An Overview of Genetic Algorithms : Part1, Fundamentals," University Computing, Vol. 15(2), pp. 58-69, Inter University Committee on Computing, 1993.
- [7] David Beasley, David R. Bull, and Ralph R. Martin "An Overview of Genetic Algorithms : Part 2, Research Topics," University Computing, Vol. 15(4), pp. 170-181, 1993.
- [8] G. Seront and H. Bersini, "A new GA-local search hybrid for continuous optimization based on multi level single linkage clustering", Proc. of GECCO-2000, pp. 90-95, 2000.
- [9] Z. Michalewicz, Genetic Algorithm + Data Structures = Evolution Programs, 2nd Ed., Springer Verlag, 1994.
- [10] John R. Koza, Genetic Programming : On the Programming of Computers by Means of Natural Selection, Cambridge, MA, MIT Press, 1992.



김 대 희 (Dae-Hee Kim)

1999년 한국방송통신대학교 컴퓨터
과학과 이학사

2002년 안동대학교 컴퓨터공학과
공학석사

2002년 3월 ~ 현재 안동대학교 정보통신공학과 박사
과정

(관심분야 : 임베디드 시스템, 무선네트워크 등)



박 상 호 (Sahng Ho Park)

1979년 경북대학교 전자공학과
공학사

1981년 영남대학교 대학원 전자
공학과 공학석사

1989년 Syracuse University, Computer Engineering,
MS

1995년 State University of New York at Buffalo,
Electrical Engineering, Ph.D.

1996년 3월 ~ 현재 안동대학교 정보통신공학과 부
교수

(관심분야 : 이동통신, 멀티미디어통신, 무선네트워크)