미리 순서가 매겨진 학습 데이타를 이용한 효과적인 증가학습

(Efficient Incremental Learning using the Preordered Training Data)

이 선 영 † 방 승 양 ††

(Sunyoung Lee) (Sung-Yang Bang)

요 약 증가학습은 점진적으로 학습 데이타를 늘려가며 신경망을 학습시킴으로써 일반적으로 학습시간을 단축시킬 뿐만 아니라 신경망의 일반화 성능을 향상시킨다. 그러나, 기존의 증가학습은 학습 데이타를 선정하는 과정에서 데이타의 중요도를 반복적으로 평가한다. 본 논문에서는 분류 문제의 경우 학습이 시작되기 전에 데이타의 중요도를 한 번만 평가한다. 제안된 방법에서는 분류 문제의 경우 클래스 경계에 가까운 데이타일수록 그 데이타의 중요도가 높다고 보고 이러한 데이타를 선택하는 방법을 제시한다. 두 가지 합성 데이타와 실세계 데이타의 실험을 통해 제안된 방법이 기존의 방법보다 학습 시간을 단축시키며 일반화 성능을 향상시킴을 보인다.

Abstract Incremental learning generally reduces training time and increases the generalization of a neural network by selecting training data incrementally during the training. However, the existing methods of incremental learning repeatedly evaluate the importance of training data every time they select additional data. In this paper, an incremental learning algorithm is proposed for pattern classification problems. It evaluates the importance of each piece of data only once before starting the training. The importance of the data depends on how close they are to the decision boundary. The current paper presents an algorithm which orders the data according to their distance to the decision boundary by using clustering. Experimental results of two artificial and real world classification problems show that this proposed incremental learning method significantly reduces the size of the training set without decreasing generalization performance.

1. 서 론

인공신경망(artificial neural network, ANN)은 인간의 뇌에서 일어나는 정보처리 작용을 모사하는 계산 모델이다. 인공신경망 모델의 예로는 다층 퍼셉트론 신경망(multi-layer perceptron network)이 있다. 신경망의 학습은 Rumelhart, Hinton 그리고 Williams 등에 의해 개발된 오류 역전파(backpropagation, BP)학습 방법을 통해 이루어진다[1]. 오류 역전파 알고리즘은 gradient descent 방법으로 국부적 최적해(Local minima)에 도달하거나 수렴 속도가 느리다는 단점이 있다. 이러한 오류 역전파 알고리즘의 단점을 개선하여, 신경망의 학습속도 및 일반화 성능(generalization performance)을 향상시키기 위한 방법으로는 초기 가중치 및 학습율 등을 설정하는 방법, 최적의 신경망의 구조를 찾는 방법등이 있다. 이 외에 신경망의 수렴 속도를 빠르게 하고 일반화 성능을 향상시키기 위한 또 다른 접근 방식이 능동학습(active learning)이다. 능동학습은 주어진 학습 데이타가 충분히 많을 경우 학습에 도움이 되는 중요한 데이타를 선택하여 학습에 사용함으로써, 학습 시간을 단축시키면서 동시에 일반화 성능을 향상시킨다[2]. 능동학습과는 달리 주어진 모든 데이타를 학습에 사용하는 기존의 학습 방법을 수동학습(passive learning)이라 한다.

능동학습에서 가장 중요한 부분은 '어떻게 학습에 도

† 비 회 원 : 포항공과대학교 컴퓨터공학과
pulpiri@nova.postech.ac.kr
†† 종신회원 : 포항공과대학교 컴퓨터공학과 교수
sybang@postech.ac.kr
논문접수 : 1999년 2월 18일
심사완료 : 1999년 11월 30일

움이 되는 중요한 데이타를 선택하느냐' 하는 것이다. 기존의 능동학습에서 학습 데이타의 중요도를 평가하기 위한 방법은 현재까지 선택된 데이타로 학습한 신경망을 이용한다. 이러한 동적인 지식을 이용하여 데이타의 중요도를 평가하므로 학습 데이타의 중요도를 반복적으로 평가해야 한다.

본 논문에서는 패턴 분류 문제의 경우 학습 데이타의 중요도를 신경망의 학습이 시작되기 전에 미리 평가하는 새로운 능동학습 방법을 제시한다. 제시된 방법으로 학습에 도움이 되는 중요한 데이타를 선택하여 학습 데이타를 축소함으로써 일반화 성능이 향상됨을 보인다. 본 논문에서는 신경망의 학습을 위해 결정경계(decision boundary)에 가까운 데이타를 선택한다. 이때, 결정경계에 가까운 데이타를 선택하기 위해 측정되는 데이타의 중요도는 신경망 학습이 시작되기 전에 미리 측정된다.

본 논문의 구성은 다음과 같다. 2장에서는 능동학습과 관련된 기존의 연구들을 살펴보고 3장에서는 본 논문에서 제안한 결정경계에 가까운 데이타를 선택하는 증가학습 알고리즘을 제시한다. 4장에서는 제시한 알고리즘으로 실험한 결과를 분석하고 5장에서는 전체적인 내용을 요약, 결론을 맺고 향후 연구 방향에 대해 기술하고자 한다.

2. 능동 학습

능동학습은 주어진 학습 데이타가 잡음이나 데이타의 중복으로 인하여 각 데이타들이 학습에 기여하는 정도가 다를 경우 학습에 중요한 데이타만을 선택하여 학습하는 방법이다. Lange와 Männer는 신경망의 일반화 성능이 데이타의 수가 증가함에 따라 향상되지 않고, 오히려 최적의 데이타 집합으로 학습한 신경망의 일반화 성능이 향상됨을 보였다[14]. 이렇게 학습에 도움이 되는 최적의 데이타 집합을 찾아내어 학습함으로써 학습시간을 단축시키고 일반화 성능을 향상시키는 것이 능동학습이다[8,9,14,15].

신경망은 입출력 쌍으로 이루어진 다음 학습 데이타 집합에 대해 학습을 수행한다.

$$D=\{(x_1,y_1),(x_2,y_2),\cdots,(x_N,y_N)\},\quad X_i=(x_i,y_i),\quad |D|=N \tag{1}$$

신경망에서의 학습은 수식(2)의 에러함수를 최소화하는 방향으로 가중치를 수정하는 비선형 최적화(nonlinear optimization) 문제를 해결하는 것이다.

$$E(D\,|\,w)=\sum_{i=1}^{N}E(y_i\,|\,x_i,w) \tag{2}$$

능동학습은 크기 N인 전체 학습 데이타 집합에 대하여 학습을 수행하는 것이 아니라 초기 학습 데이타 집합 D_0에서 학습을 시작하여 학습 데이타 집합을 점점 증가시켜 학습을 진행한다. 이것은 수식(2)의 오류함수를 직접 최소화하기보다는 수식(3)과 같은 방향으로 오류함수를 최소화한다.

$$E(D_0\,|\,w)\Rightarrow E(D_1\,|\,w)\Rightarrow\cdots\Rightarrow E(D_i\,|\,w) \tag{3}$$

이때, $|D_i|$ 는 i번째 학습 데이타 집합의 크기를 나타내며 다음 관계를 만족한다.

$$|D_0|\lhd|D_1|\lhd\cdots\lhd|D_s|=|D| \tag{4}$$

수식(3)과 수식(4)에서 알 수 있듯이 초기에는 소수의 학습 데이타로 학습을 수행한다. 신경망이 만족할 만한 일반화 성능에 도달하면 학습을 끝내고, 그렇지 못한 경우에는 학습에 사용할 다음 데이타를 선택한다. 이때, 학습 데이타는 각각의 능동학습에서 제시한 방법으로 데이타의 중요도를 평가하여 선택한다.

능동학습은 학습에 사용하기 위해 선택된 데이타를 후보 데이타 집합(candidate training set)에서 제거하는지의 여부에 따라 증가학습(incremental learning)과 선택학습(selective learning)으로 나뉘어 진다. 증가학습에서는 신경망 학습을 위해 선택된 데이타들이 후보 데이타 집합에서 제거된다. 따라서, 학습과정이 진행됨에 따라 학습에 사용되는 데이타 집합은 증가하고 후보 데이타 집합은 감소하게 된다. 이와는 달리 선택학습은 학습 데이타를 후보 데이타 집합에서 선택한 후 선택된 데이타를 후보 데이타 집합에서 제거하지 않는다. 따라서, 각 데이타 선택 단계에서 모든 데이타들이 선택될 동등한 기회를 가지게 된다.

현재까지 제안된 증가학습 방법에는 다음과 같은 방법들이 있다. Cloete와 Ludik은 Delta Training 방법[5]과 Increased Complexity Training 방법[6,7]을 제안하였다. Zhang과 Röbel은 현재까지 학습된 신경망에서의 에러가 가장 큰 데이타들을 선택하여 분류 문제 뿐 아니라 함수 추정 문제에도 사용할 수 있는 능동학습을 제안하였다[8,9]. Cohn은 신경망의 MSE(mean square error)의 기대치를 최소화하는 학습 데이타를 선택하는 방법인 Optimal Experiment Design을 고안하였다[10]. Plutowski와 White 그리고 Sung와 Niyogi은 ISB(integrated squared bias)를 최소화하는 학습데이

타를 선택하는 방법을 제시하였다[11,12]. 또한, Englbrecht와 Cloete는 학습 데이타의 입력 벡터를 아주 조금 변경시켰을 때 출력 벡터의 변화가 얼마나 민감한지를 측정하는 Sensitivity Analysis를 이용하는 방법을 제안하였다[13]. 이러한 방법들은 학습 데이타를 선택하기 위해 데이타의 중요도를 반복적으로 평가해야 한다는 단점이 있다. 본 논문에서는 패턴 분류 문제의 경우 반복적인 계산을 하지 않고 결정경계에 가까운 데이타를 선택하는 증가학습 방법을 제안한다.

3. 제안 방법

패턴 분류 문제의 경우, 신경망에서의 학습은 클래스를 분류하는 경계면 즉, 결정경계를 찾는 과정이다. 따라서, 신경망의 학습에 도움이 될 가능성이 있는 데이타들은 결정경계에 존재하는 데이타들이다[3,4,13]. 이미 많은 연구에서 결정경계의 데이타가 학습에 중요한 데이타임을 보여왔다. Cohn은 학습 데이타를 임의로 선택하여 학습시키는 것 보다 결정경계에 가까운 데이타를 선택하여 학습시켰을 경우 신경망의 일반화 성능이 향상됨을 보였다[15]. 또, Engelbrecht는 입력의 아주 작은 변화에 출력이 달라지는 데이타가 결정경계에 존재하는 데이타라는 점을 이용하여 이러한 데이타를 선택하는 증가학습 알고리즘을 제안하였다[13].

만약 학습 데이타를 결정경계에 가까운 순서대로 번호를 부여할 수 있다면 데이타의 중요도를 반복적으로 평가하는 비용을 줄일 수 있을 것이다. 본 논문에서는 결정경계에 가까운 데이타를 순서대로 나열하는 방법을 제안하고자 한다.

3.1 학습 데이타의 중요도

신경망에서의 결정경계는 출력노드가 동일한 출력 값을 갖는 데이타들이 이루는 궤적으로 다음과 같이 정의할 수 있다.

$$\{X \mid F(X) = W\phi(X) + b = 0\} \tag{5}$$

$F(X)$는 분류함수(discriminant function)이며 W는 d차원 벡터이다. $\phi(X)$는 임의의 비선형 함수(nonlinear function)이며 b는 bias를 나타낸다. 이때, 결정경계 F(X)는 d차원의 입력공간에서 d-1차원의 hyperplane에 대응된다.

그림1에서처럼 데이타 X에서 결정경계까지의 거리 l은 다음과 같이 나타낼 수 있다.

$$l = \frac{|F(X)|}{\|w\|} \tag{6}$$

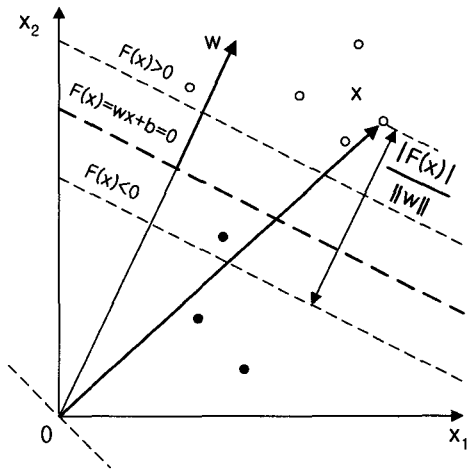


그림 1 2차원 입력 공간 (X_1,X_2) 에서 F(X)=0 에 대응되는 선형 결정경계

결정경계에 가까운 데이타를 선택한다는 것은 학습 데이타들이 수식(7)의 관계를 만족할 때, X_1부터 차례로 선택한다는 의미이다.

$$\frac{|F(X_1)|}{\|w\|} < \frac{|F(X_2)|}{\|w\|} < \cdots < \frac{|F(X_N)|}{\|w\|}, X_i \in D \tag{7}$$

그러나, 문제는 신경망 학습의 목표가 바로 결정경계를 찾는 것이므로 위의 $F(X)$를 알지 못한다는 것이다. 따라서, $F(X)$를 모르면서 $F(X)$에 가까울 가능성이 있는 데이타를 선택하는 방법에 대하여 알아본다. 앞으로 논문 전체에 걸쳐 "결정경계에 가까운"은 "결정경계에 가까울 가능성이 있는"을 뜻한다.

3.2 클러스터화 방법(Clustering)의 이용

클래스를 분류하기 위한 결정경계는 클래스들의 경계에 속한다[그림2참조]. 앞에서도 언급했듯이 결정경계를 알지 못하기 때문에 결정경계에 가까운 데이타를 찾을 수 없다. 그러나 클래스들의 경계에 가까운 데이타는 다음과 같이 찾을 수 있다.

우선 데이타를 클래스별로 임의의 개수의 클러스터로 클러스터화한다. 데이타를 클래스별로 하므로 클러스터의 경계로 클래스의 경계를 근사할 수 있다. 즉, 클래스간의 결정경계는 서로 다른 클래스의 클러스터로 이루어진 클러스터 쌍의 경계에 포함된다.

단순하고 알기 쉬운 예가 그림3에 있다. 그림3을 살펴보면 클래스1은 g_{11}, g_{12}의 두 개의 클러스터로 구성되고 클래스2도 g_{21}, g_{22}의 두 개의 클러스터로 구성된다. 따라서, 클래스1과 클래스2의 경계는 그림과 같은 4개의

클러스터 쌍의 경계로 근사할 수 있다. 일반적으로, 클래스간의 결정경계에 가까운 데이타를 선택하는 것은 가능한 모든 클러스터 쌍에 대하여 클러스터의 경계에 가까운 데이타를 선택하는 것으로 근사할 수 있다.

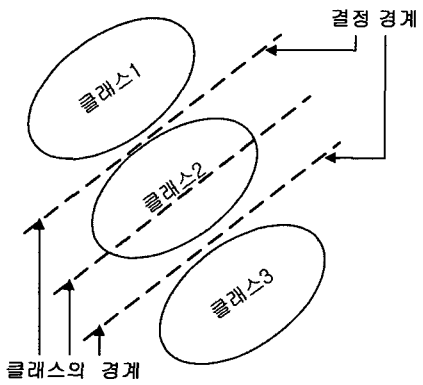


그림 2 결정경계와 클래스의 경계

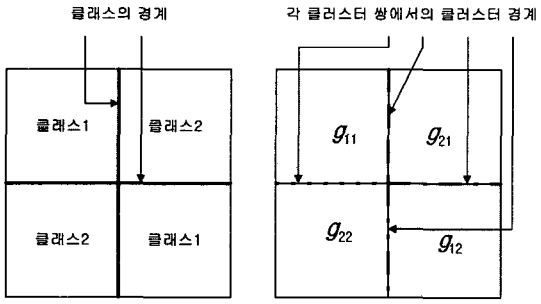


그림 3 클래스의 경계와 클러스터의 경계

데이타 선택은 다음의 가능한 모든 클러스터 쌍에 대하여 동시에 이루어진다.

클러스터 쌍 : g_{ui} 와 g_{vj}

g_{ui} : 클래스 u의 임의의 클러스터 $1 \le i \le n_u$

g_{vj} : 클래스 v의 임의의 클러스터 $1 \le j \le n_v$

n_u : 클래스 u의 클러스터 개수

n_v : 클래스 v의 클러스터 개수

클러스터 쌍 g_{ui} 와 g_{vj} 에 대하여 클러스터의 경계에 가까운 데이타는 다음과 같은 방법으로 선택한다. g_{ui}의 평균 벡터에 가장 가까운 데이타를 μ_{ui}라하고 g_{vj}의 평균 벡터에 가장 가까운 데이타를 μ_{vj} 라 한다. g_{ui}의 데이타 중 클러스터의 경계에 가장 가까운 데이타는 일반적으로 μ_{vj}에 가장 가까운 데이타가 된다. 따라서 g_{ui}에서 데이타를 선택하는 경우 μ_{vj}에 가장 가까운 데이타부터 차례로 선택한다. 물론 이와 같은 일반론을 따르지 않는 경우는 얼마든지 찾을 수 있다. 가령 그림4에서처럼 클래스2의 평균 벡터에 가장 가까운 데이타를 x 라고 하자. 이 경우 클래스1의 데이타 a, b 중 x에 더 가까운 것은 b이다. 그러나 결정경계에 더 가까운 것은 a이다. 그러나, 우리는 이러한 경우는 일반적으로 일어나지 않고 일어난다 하더라도 드물다고 보는 것이다. 마찬가지로 g_{vj}의 데이타 중 클러스터의 경계에 가장 가까운 데이타는 일반적으로 μ_{ui}에 가장 가까운 데이타가 된다. 따라서, g_{vj}에서 데이타를 선택하는 경우 μ_{ui}에 가장 가까운 데이타부터 차례로 선택한다.

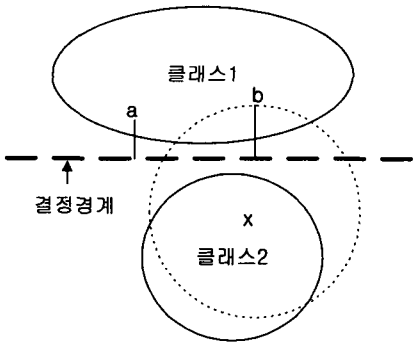


그림 4 제시한 방법으로 클래스의 경계에 가장 가까운 데이타를 선택할 수 없는 경우

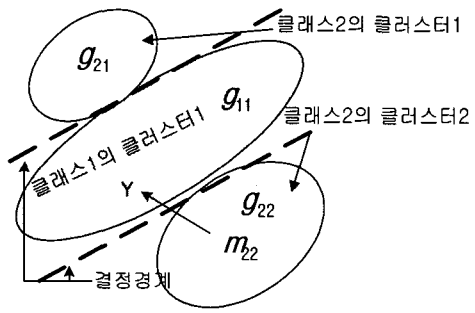


그림 5 클러스터 쌍에서 결정경계에 가까운 데이타의 선택

예를 들어, 그림5의 경우를 살펴보도록 하자. 클래스 1은 하나의 클러스터로 클래스 2는 두 개의 클러스터로 클러스터화 될 것이다. 클래스 1과 클래스 2가 이루는 결정경계는 클러스터 쌍 g_{11}과g_{21}, g_{11}과g_{22}가 이루는 경계에 포함된다. 따라서, 결정경계에 가까운 데이타를 선

택하기 위해, 클러스터 쌍의 경계에 가까운 데이타를 선택한다. 결정경계에 가까운 데이타는 다음과 같은 방법으로 선택한다.

μ_{11}을 g_{11}에 속하는 데이타 중, g_{11}의 평균 벡터에 가장 가까운 학습 데이타라고 하고, μ_{22}를 g_{22}에 속하는 데이타 중, g_{22}의 평균 벡터에 가장 가까운 학습 데이타라고 하자. g_{11}에서는 μ_{22}에 가장 가까운 데이타 Y부터 선택한다. 만약 g_{22}에서 데이타를 선택한다면, g_{11}의 μ_{11}에 가장 가까운 데이타를 선택하면 된다.

3.3 제안된 증가학습 알고리즘

3.2절에서 설명했듯이, 클래스 u의 임의의 클러스터 g_{ui}에서 데이타를 선택할 때, 결정경계에 가까울 가능성이 있는 데이타는 클래스 v($\neq$u)의 클러스터 g_{vj}의 평균 벡터에 가장 가까운 학습 데이타 μ_{vj}와의 거리가 가까운 데이타이다. 이 데이타를 수식(8)로 근사하여 찾는다. 학습 데이타는 이미 클래스별로 클러스터화되어 있다고 하자. 이때 클래스 u의 임의의 클러스터 g_{ui}와 클래스 v의 임의의 클러스터 g_{vj}에 대하여 다음의 척도를 계산하여 이 값이 작은 데이타부터 차례로 선택한다.

$$I(X_{ui}) = | X_{ui} - \mu_{vj} |$$
$$X_{ui} \in g_{ui}$$
$$\mu_{vj} \text{는 } g_{vj} \text{의평균벡터에가장가까운학습데이터} \quad (8)$$

논문에서 제안한 방법은 초기에 임의의 데이타에 대하여 학습을 시작하지 않고, 각 클러스터의 평균 벡터에 가장 가까운 데이타부터 학습을 진행한다. 즉, 신경망이 초기에 주어진 학습 데이타를 대표하는 데이타들을 학습하게 된다. 따라서, 초기 데이타 선택 단계에서 어느 정도의 학습 성능을 나타낼 수 있으므로 기존의 증가학습보다 데이타 선택 단계를 감소시킬 수 있다. 또한, 데이타의 중요도를 데이타 선택단계에서 계산하지 않고, 데이타를 clustering한 후 학습이 시작되기 전에 미리 결정할 수 있으므로 학습동안 데이타 선택을 위한 반복적인 계산과정을 필요로 하지 않는다.

데이타의 중요도를 평가하기 위한 알고리즘은 다음과 같다.

g_{ui} : 클래스 u의 클러스터 i
g_{vj} : 클래스 v의 클러스터 j
*step*1 : 클래스별로학습데이터를클러스터화한다;
*step*2 : 각클러스터에대하여평균벡터에가장가까운학습데이터를구한다.
*step*3 : 각클러스터쌍에속해있는학습데이터 X_{ui}의 중요도 $I(X_{ui})$를평가;
$I(X_{ui}) = | X_{ui} - \mu_{vj} |$
$X_{ui} \in g_{ui}$
μ_{vj}는 g_{vj}의평균벡터에가장가까운학습데이터

제안한 데이타의 중요도를 평가하는 방법을 이용한 증가학습 알고리즘은 다음과 같다.

D_i : i번째학습데이터선택단계에서의학습데이터집합
C_i : i번째학습데이터선택단계에서의후보데이터집합
*step*1 : D_i를모든클러스터의평균벡터에가장가까운데이터들로초기화한다.
$C_i = D - D_i$
*step*2 : D_i로신경망을학습한다.
*step*3 : *if* $E(D_i | w) \le \varepsilon$ or C_i *is empty then*
학습을끝낸다.
else
각 클러스터 쌍에서 $I(X_{ui})$값이 작은 λ개의학습데이터선택한다.
선택된 학습데이터를 D_i에 추가한다.
$C_i = D - D_i$
goto step2;

4. 실험 및 결과 분석

4.1 실험 환경

실험에서는 합성 데이타와 실세계 데이타에 대하여 기존의 증가학습 모델 중 Zhang의 증가학습[8]과 모든 데이타를 학습에 사용하는 수동학습 방법을 비교하였다. 실험 조건은 데이타 개수를 제외한 모든 파라미터(신경망 구조, 학습율 등)를 동일하게 설정하여 실험하였다. 실험에서 사용한 신경망은 하나의 은닉층을 갖는 3층 퍼셉트론 신경망이며 Sun SPARC 20(Solaris 2.5)에서 실험하였다. 학습 알고리즘은 오류 역전파 알고리즘을 사용하였고 실험 결과는 5번 수행하여 평균을 구하였다. 학습의 종료 조건은 Zhang의 방법에서 사용한 현재까지의 학습에러가 지정된 성능 레벨에 도달할 경우 학습을 끝내는 방법을 사용하였다[8].

4.2 합성 데이타의 이진 분류 문제

Circle in Square 이진 분류 문제와 XOR 이진 분류 문제에 대하여 실험하였다. 신경망 구조는 2-3-1, 학습율은 0.1로 설정하였다.

4.2.1 Circle in Square 이진 분류 문제

Circle in Square 이진 분류 문제는 반경이 0.4인 원 영역과 원 밖의 사각영역의 두 클래스를 분류하는 문제이다[그림6 참조].

$$class = \begin{cases} 0 & \text{if } \sqrt{x^2 + y^2} \le 0.4 \\ 1 & \text{otherwise} \end{cases}$$

입력은 이차원이며 400개의 학습 데이타를 [-1:1] 범위에서 균등(uniform) 한 분포를 갖도록 생성시켰으며,

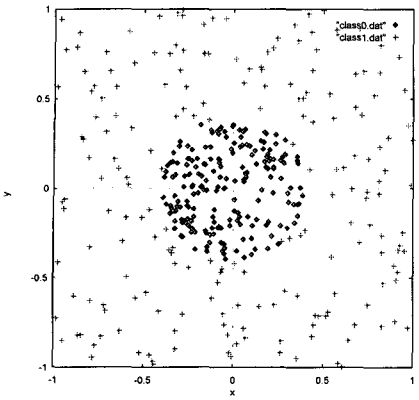
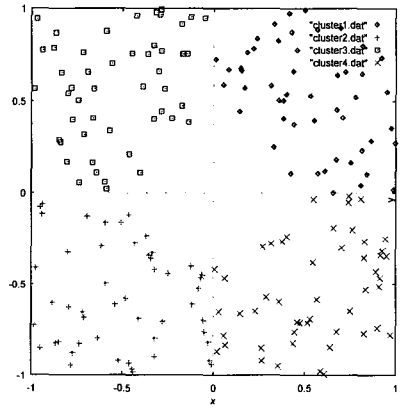


그림 6 Circle in Square 이진 분류 문제



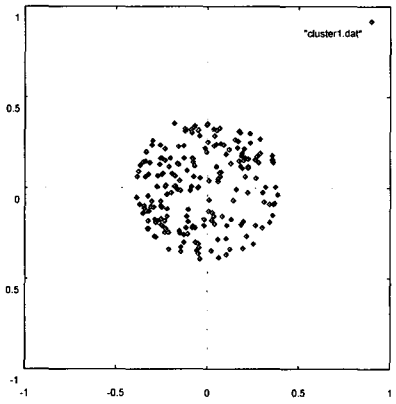


그림 7 클래스 0 와 클래스 1 의 클러스터화 한 결과

검증(validation)을 위해 [-1:1] 범위를 0.1 간격으로 나눈 격자(grid) 데이타를 441개 생성시켜 사용하였다. 그림7은 클래스별로 클서스터화 한 모습이다. 클래스 1은 4개의 클러스터로 클러스터화하였으며 클래스 0는 1개의 클러스터로 클러스터화하였다. 각 학습 데이타 선택 단계에서는 16개씩 선택하여 학습하였다.

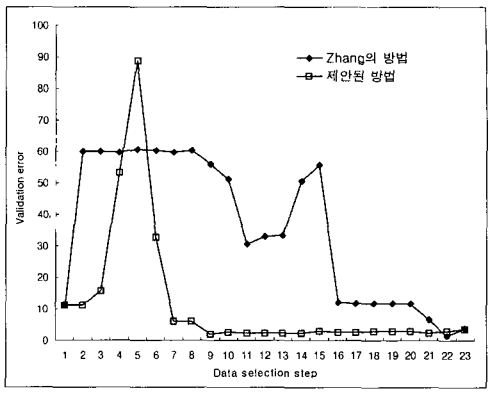


그림 8 검증 에러(Square in Circle 이진 분류 문제)

그림8은 Zhang의 증가학습과 논문에서 제안한 증가학습에 대하여 각 데이타 선택 단계에서의 검증 에러를 나타낸 것이다.

그림9와 그림10은 Zhang의 방법과 논문에서 제안한 방법이 수동학습과 비슷한 성능을 보일 때, 데이타를 선택한 모습이다.

표1은 수동학습과 비슷한 일반화 성능을 보일 때의 Zhang의 방법 그리고 논문에서 제안한 방법의 검증 에러를 나타낸다. 실험 결과를 살펴보면 논문에서 제안한 방법은 결정경계에 가까운 데이타를 효과적으로 선택하여 기존의 증가학습인 Zhang의 방법보다 더 적은 수의 데

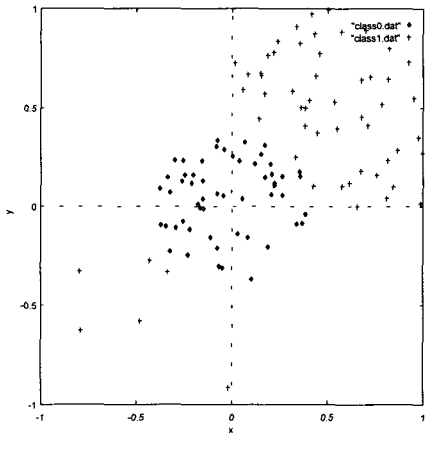


그림 9 선택된 데이타(Zhang의 방법)

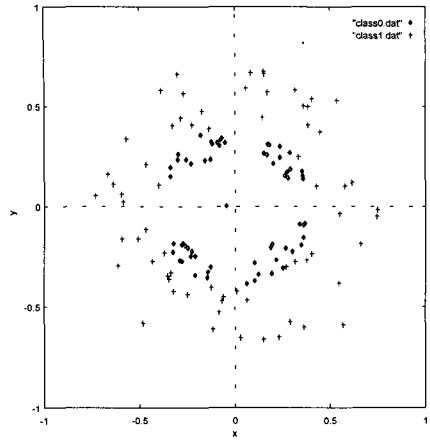


그림 10 선택된 데이타(제안된 방법)

표 1 Circle in Square 이진 분류 문제의 일반화 성능

데이타 선택 방법	검증 에러(%) / 사용된 데이타 개수
수동학습	1.81 / 400
Zhang의 방법	5.21 / 336
제안된 방법	2.2 / 133

이타만으로 더 좋은 일반화 성능을 보임을 알 수 있다.

4.2.2 XOR 이진 분류 문제

XOR 이진 분류 문제에서는 200개의 학습 데이타를 [-1,1] 사이의 값으로 균등한 분포를 갖도록 생성시켰다.

그림12는 클래스별로 클러스터화 한 결과이다. 두 클래스 모두 두 개의 클러스터로 클러스터화하였으며, 각

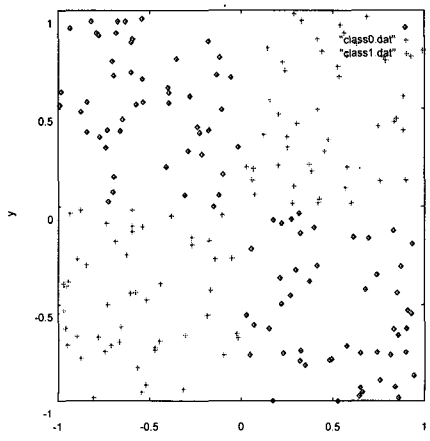


그림 11 XOR 이진 분류 문제

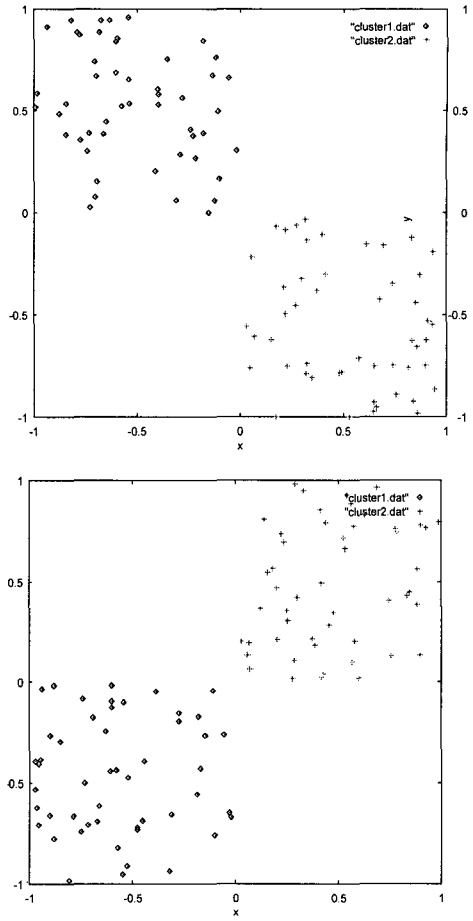


그림 12 클래스 0 와 클래스 1의 클러스터화 한 결과

데이타 선택 단계마다 16개씩의 데이타를 선택하여 학습에 사용하였다. 그림13은 Zhang의 증가학습과 논문에서 제안한 증가학습에 대하여 각 데이타 선택 단계에서의 검증 에러를 나타낸 것이다. 그림14와 그림15는 Zhang의 방법과 논문에서 제안한 방법이 수동학습과 비슷한 성능을 보일 때, 데이타를 선택한 모습이다. 표2는 수동학습과 비슷한 일반화 성능을 보일 때의 Zhang의 방법 그리고 논문에서 제안한 방법의 검증 에러를 나타낸다.

XOR이진 분류 문제도 Circle in Square이진 분류 문제와 마찬가지로 제안된 방법이 적은 수의 데이타를 학습에 사용하여 일반화 성능이 향상됨을 알 수 있다.

합성 데이타의 실험 결과를 살펴보면, 신경망의 일반화 성능은 학습에 도움이 되는 최적의 데이타 집합으로

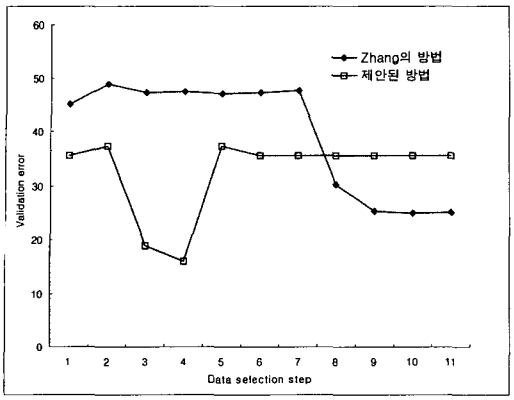


그림 13 검증 에러(XOR 이진 분류 문제)

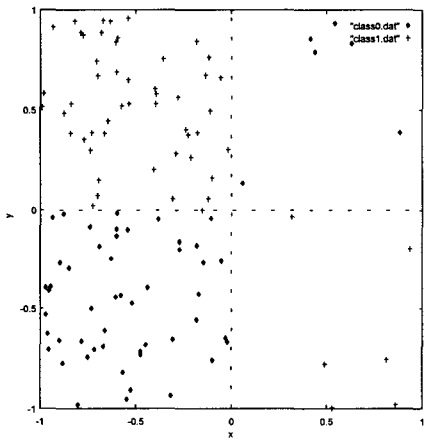


그림 14 선택된 데이타(Zhang의 방법)

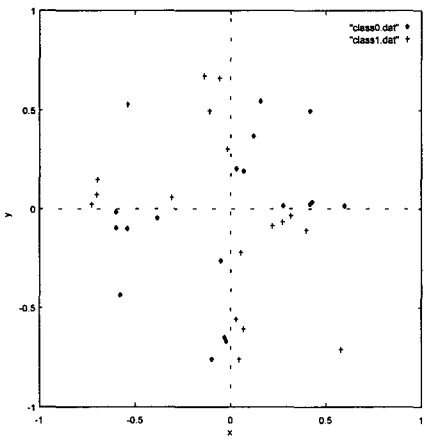


그림 15 선택된 데이타(제안된 방법)

학습할 경우 최대가 되며, 학습 데이타 집합이 커지더라도 일반화 성능이 향상되지 않는 것을 알 수 있다.

표 2 XOR 이진 분류 문제의 일반화 성능

데이타 선택 방법	검증 에러(%) / 사용된 데이타 개수
수동학습	35.6 / 200
Zhang의 방법	26.3 / 160
제안된 방법	18.8 / 36

4.3 필기숫자 인식 문제

실험에서 사용한 필기 숫자 데이타는 UCI repository 데이타베이스[18]를 이용하였다. 필기 숫자 이미지는 32×32 비트맵 이미지이다. 학습을 위해 32×32 비트맵이미지를 4×4 마스크를 사용하여 8×8 입력 행렬로 만들었다. 학습에 사용한 총 데이타 개수는 3823이고 validation을 위해 1797개의 데이타를 사용하였다.

실험에 사용한 신경망 구조는 64-40-10이며 학습율은 0.1로 설정하였다. 수동학습과 논문에서 제안한 방법으로 실험하였다. 10개의 클래스를 각각 3개의 클러스터로 클러스터화 하여 데이타의 중요도를 평가하였으며 각 데이타 선택 단계마다 200개씩의 데이타를 선택하였다.

표 3 필기숫자 인식 문제의 일반화 성능

데이타 선택 방법	검증 에러(%) / 사용된 데이타 개수
수동학습	3.2 / 3823
제안된 방법	1.6 / 1980

표3에서 알 수 있듯이 논문에서 제안된 방법이 더 적은 수의 데이타를 학습에 사용하여 수동학습보다 약 2%정도 일반화 성능이 향상되었다.

4.4 제안한 방법의 성능평가 및 기존 방법과의 비교 분석

4.4절에서는 제안한 알고리즘의 학습 데이타를 선택하는 과정을 살펴보고, 계산 복잡도를 기존의 방법과 비교한다. 표기상의 편의를 위해 기존의 증가학습을 줄여서 IL(incremental learning)로, 본 논문에서 제안한 클러스터화 방법을 이용한 증가학습을 IC(incremental learning using clustering)로 표기한다.

IC는 초기에 각 클러스터의 평균 벡터에서 가장 가까운 학습 데이타를 학습한다. 따라서, 초기에 학습 데

이타의 분포를 어느 정도 학습하게 되므로, 학습 데이타 선택 단계를 감소시킬 수 있다. 기존의 증가학습은 초기에 학습 데이타를 임의로 선택한다. 다음 학습 데이타의 선택은 현재까지 선택된 데이타로 학습한 신경망의 정보를 이용한다. 따라서, 앞으로 선택될 데이타는 이전 단계에서 선택된 데이타에 의존하게 된다. 만약, 초기에 outlier와 같은 학습에 도움이 되지 않는 데이타가 선택된다면, 신경망을 원하는 성능에 도달하게 하기 위한 데이타 선택 단계가 더욱 늘어나게 될 것이다.

IL과 IC의 계산 복잡도를 살펴보면 다음과 같다. 데이타 집합의 크기를 N으로 표기하고, s번째 데이타 선택 단계 이후의 데이타 집합의 크기를 N_s로 표기한다.

$$N_s = N_0 + \lambda \cdot s \tag{9}$$

N_0는 초기에 선택된 데이타 집합의 크기이고 λ는 각 데이타 선택단계에서의 학습데이타 집합의 증가치를 말한다. 이때, 최대 데이타 선택 단계의 수는 다음 수식으로 나타낼 수 있다.

$$s_m = \left\lceil \frac{N - N_0}{\lambda} \right\rceil, \quad \lceil x \rceil : ceiling\ function \tag{10}$$

D_{Ns}는 s번째 데이타 선택 단계이후의 학습 데이타 집합을 나타내고, t_s는 s번째 학습 데이타 선택 단계에서 학습 데이타 집합 D_{Ns}로 학습할 경우의 총 epoch수이다. K는 신경망에서 수정할 가중치의 수이다. C는 각 증가학습에서 사용한 데이타의 중요도를 평가하기 위한 비용을 나타낸다. 이때, IL과 IC의 계산 복잡도를 다음과 같다. 각각의 계산 복잡도에서 사용하는 방법에 따라 t_s와 s_m은 달라질 수 있다.

• IL의 계산 복잡도

$$T_{IL}(D_N) = \sum_{s=0}^{s_m} T(D_{N_s}) = \sum_{s=0}^{s_m} \left(\sum_{t=1}^{t_s} \sum_{p=1}^{N_s} K + C \right) \tag{11}$$

• IC의 계산 복잡도

$$T_{IC}(D_N) = C + \sum_{s=0}^{s_m} T(D_{N_s}) = C + \sum_{s=0}^{s_m} \sum_{t=1}^{t_s} \sum_{p=1}^{N_s} K \tag{12}$$

IC에서 학습 데이타를 선택하기 위한 비용은 학습 데이타를 클러스터화하기 위한 비용과 데이타 중요도를 평가하기 위한 척도를 계산하는 비용으로 이루어진다. 이때, 데이타의 중요도를 평가하기 위한 계산 복잡도는 O(N)이며 학습 데이타의 클러스터화를 위한 계산 복잡도는 k-means 클러스터화 방법을 사용하는 경우 O(PN)이다. P는 클러스터화를 위해 데이타를 보여주는 횟수이다. 따라서, 데이타의 중요도를 평가하기 위한 비용은 클러스터화에 필요한 비용으로 요약될 수 있다.

IL방법의 경우 데이타의 중요도를 계산하기 위한 계산 복잡도를 살펴보면 다음과 같다. Engelbrecht는 입력 데이타를 아주 조금 변경시켰을 경우 출력의 변화를 나타내는 sensitivity행렬을 구하여 sensitivity가 높은 데이타를 선택하는 방법을 사용하였다[13]. 이때의 계산 복잡도는 $O(s_mNM)$이다. M은 sensitivity 행렬을 구하기 위한 비용이며 이것은 신경망 구조에 비례하여 증가되는 비용이다. Plutowski는 ISB(Integrated squared bias)값을 최대화하는 데이타를 선택한다[11]. 이것은 데이타의 에러 기울기가 전체 데이타의 에러 기울기와 서로 밀접하게 관련되어 있는 데이타를 선택하는 증가학습으로 계산 과정에서 Hessian matrix approximation이 필요하다. 따라서 gradient descent 수정 방법과 동일한 계산 복잡도가 필요하다. 따라서 필요한 계산 복잡도는 $O(s_mT_{BP}(N))$이다. Zhang은 현재 신경망에서의 에러가 가장 큰 데이타를 중요한 데이타로 선택한다[8]. 이때의 계산 복잡도는 $O(s_mNE)$이다. E는 신경망에서의 에러를 구하기 위한 비용으로 신경망의 구조에 비례하여 증가하는 비용이다.

IC 는 Plutowski의 방법보다 데이타를 선택하기 위한 계산 복잡도가 적다. 그러나, clustering을 위한 비용으로 인해 최악의 경우 Zhang의 방법과 Engelbrecht의 방법보다는 데이타를 선택하기 위한 비용이 늘어날 수도 있다. 앞 절의 비교 실험을 살펴보면 IC는 Zhang의 방법보다 효과적으로 결정경계의 데이타를 선택하여 데이타 선택 단계를 감소시킨다. 따라서, Zhang의 방법보다 수렴속도가 빠르며 좋은 일반화 성능을 보임을 알 수 있다.

표 4 데이타의 중요도를 평가하기 위한 계산 복잡도

IC	Engelbrecht	Plutowski	Zhang
O(PN)	$O(s_mNM)$	$O(s_mT_{BP}(N))$	$O(s_mNE)$

IC는 현재까지 학습된 신경망에 대한 어떠한 정보도 이용하지 않는다. 따라서, 신경망의 학습이 시작되기 전에 학습 데이타의 중요도를 알 수 있다는 장점이 있다. 따라서, 주어진 학습 데이타가 적어서 학습이 어려운 경우에 새로운 데이타를 인위적으로 생성하기 위한 방법

에도 효과적으로 이용될 수 있으리라 본다. 또한, 기존의 증가학습과 마찬가지로 주어진 학습 데이타가 잡음이 적고 일반화를 위한 충분한 정보를 가지고 있을 경우 수동학습보다 효과적이다.

모든 증가학습이 그렇듯이 IC가 항상 수동학습보다 일반화 성능이 향상되는 것은 아니다. 만약 주어진 데이타가 드문드문 분포되어 있나 모든 데이타가 이미 학습에 중요한 데이타라면 IC는 모든 데이타를 학습에 사용할 것이다. 이런 경우, IC는 데이타 선택으로 인한 비용으로 인해 수동학습보다 비싼 방법이다. IC에서는 결정경계에 가까운 데이타를 선택하기 위해 임의의 개수의 클러스터로 데이타를 클러스터화하였다. 클러스터의 개수가 많을수록 결정경계에 가까운 데이타를 효과적으로 선택하지만 클러스터 쌍이 늘어나게 됨으로써 계산양이 많아지게 된다. 따라서 적절한 클러스터의 개수를 선택하는 문제는 해결해야 할 문제이다.

5. 결론 및 향후 연구방향

신경망의 학습시간은 신경망의 복잡도와 학습 데이타 집합의 크기에 비례하여 증가한다. 신경망의 학습 목표는 학습에 사용하지 않은 데이타에 대하여도 원하는 답을 내는 일반화 성능을 향상시키는 것이다. 기존의 연구에서 학습 데이타 집합의 크기가 커지더라도 신경망의 일반화 성능이 향상되지 않고 최적의 학습 데이타 집합으로 학습시켰을 경우 일반화 성능이 향상됨을 보였다[14]. 따라서, 학습에 중요한 데이타를 선택하여 선택된 데이타만으로 신경망을 학습 시킨다면 학습 시간을 단축시킬 뿐만 아니라 일반화 성능을 향상시킬 수 있을 것이다. 패턴 분류 문제의 경우 학습은 결정경계를 찾는 것이다. 따라서, 신경망의 학습에 도움이 될 가능성이 있는 데이타는 결정경계에 가까운 데이타들이다[13]. 본 논문에서는 패턴 분류문제의 경우 결정경계에 가까운 데이타를 효과적으로 선택하여 학습하는 증가학습 방법을 제안하였다.

본 논문에서 제안한 방법은 데이타의 중요도를 데이타 선택 단계에서 반복적으로 평가하지 않고 선택 단계 이전에 미리 계산하는 방법이다. 이 방법은 Plutowski의 증가학습 방법보다는 학습 데이타의 중요도를 계산하는 계산 복잡도가 적다. 그러나, Zhang의 방법보다는 데이타 clustering으로 인한 비용으로 인해 데이타 중요도를 평가하는 계산 복잡도가 더 늘어날 수도 있다. 그러나, 실험에서 알 수 있듯이 Zhang의 방법보다 수렴속도와 일반화 성능이 각각 Circle in Square 분류 문제에서는 약 33.6%, 3%, XOR 분류 문제에서는 약 32.1%, 2% 향상됨을 알 수 있다. 또한, 효과적으로 결정경계에 있는 데이타를 선택하는 모습을 볼 수 있다.

본 논문에서 제안한 방법은 결정경계에 가까운 데이타를 선택하기 위해 학습 데이타를 클러스터화한다. 따라서, 주어진 문제에 적합하게 클러스터의 개수를 정하는 방법에 관한 연구가 필요하다. 또한, 본 논문에서 제시한 척도를 발전시켜 학습에 도움이 되는 새로운 데이타를 생성하는 방법도 연구될 수 있으리라 기대된다.

참 고 문 헌

[1] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning Internal Representations by Error Propation," Parallel Distributed Processing, Vol. 1, pp. 318-362, Bradford Books, Cambridge MA, 1986.

[2] C. Cachin, Pedagogical Pattern Selection Strategies, Neural Networks, Vol. 7, No. 1, pp. 175-181, 1994.

[3] S. Ahmad and G. Tesauro, "Scaling and generalization in neural networks : a case study," in Proc. 1988 Connec. Models Summer School, pp. 3-10, 1989.

[4] K. A. Huyser and M. A. Horowitz, "Generalization in connectionist networks that realize Boolean functions," in Proc. 1988 Connec. Models Summer School, pp. 191- 200, 1989.

[5] C. Cachin, Pedagogical Pattern Selection Strategies, Neural Networks, Vol. 7, No. 1, pp. 175-181, 1994.

[6] I. Cloete, J. Ludik. "Increased Complexity Training," IWANN, pp. 267-271, 1993.

[7] J. Ludik, I. Cloete, "Incremental Increased Complexity Training," ESANN, Brussels, Belgium, pp. 161-165, 1994.

[8] B. T. Zhang, Accelerated Learning by Active Example Selection, International Journal of Neural Systems, Vol. 5, No. 1, pp. 67-75, 1994.

[9] A. Röbel, The dynamic pattern selection algorithm: Effective training and controlled generalization of backpropagation neural networks, Technische Universität Berlin, Germany, Tech. Rep., 1994.

[10] D. A. Cohn, "Neural Network Exploration using Optimal Experiment Design," AI Memo 1491, MIT Artificial Intelligence Laboratory, 1994.

[11] M. Plutowski, H. White, "Selecting Concise Training Sets from Clean Data," IEEE Transactions on Neural Netowrks, Vol. 4, No. 2, pp. 305-318, 1993.

[12] K. K. Sung, P. Niyogi, "A Formulation for Active Learning with Applications to Object Detection, AI Memo 1438, MIT Artificial Intelligence Laboratory, 1996.

[13] A. Englbrecht, I. Cloete, Selective Learning using Sensitivity Analysis, IJCNN98, Vol.2, pp.1150-1155, 1998.

[14] R. Lange, R. Männer, "Quantifying a Critical Training Set Size for Generalization and Overfitting using Teacher Neural Network," ICANN, Vol. 1, pp. 497-500, 1994.
[15] D. Cohn, L. Altlas, R. Ladner, Improving Generalization with Active Learning, Machine Learning, Vol. 15, pp. 201-221, 1994.
[16] Chulee Lee, David A. Landgrebe, "Decision boundary Feature Extraction for Neural Networks," IEEE Transactions on Neural Networks, Vol. 8, No. 1, pp 75-83, 1997.
[17] A. K. Jain and R. C. Dubes, "Algorithm for Clustering Data," Englewood Cliffs, NJ07632 : Prentice Hall, 1988.
[18] http://ftp.ics.uci.edu/pub/machine-learning-databases/optdigits/

이 선 영
1996년 이화여자대학교 전자계산학과 졸업. 1999년 포항공과대학교 컴퓨터공학과 석사 졸업.

방 승 양
1966년 일본 Kyoto대학 전기공학 학사. 1969년 서울대학교 전기공학 석사. 1972년 University of Texas 전산학 박사. 1981년 ~ 1984년 한국전자기술 연구소 실장 및 부장. 1984년 ~ 1986년 (주) 유니온 시스템 연구소 소장 역임. 1987년 ~ 현재 포항공과대학 컴퓨터공학과 교수. 관심분야는 신경회로망, 패턴인식 및 Man-machine interface 임