

모멘트의 동적 변환에 의한 Kernel Relaxation의 성능과 RMSE

김은미^{*} · 이배호^{**}

요 약

본 논문에서는 순차적 학습 방법에서의 동적 모멘트를 제안한다. 동적 모멘트에서의 가변적인 모멘트를 이용하여 수렴 속도와 학습 성능을 향상시키며 회귀율에서도 이를 확인할 수 있다. 제안된 학습 방법은 기존의 정적모멘트와는 달리 수렴 정도에 따라 현재의 학습에 과거의 학습률을 달리 반영하는 방법이다. 기존의 정적 상수로 정의된 모멘트가 전체 학습에 동등하게 영향을 주는 반면 제안된 동적모멘트를 이용한 학습 방법은 학습 수행에 따라 동적으로 모멘트를 변경함으로써 수렴 속도와 학습 성능을 효과적으로 제어할 수 있다. 이전의 분류문제와 회귀문제의 분리확인과는 달리 본 논문에서는 제안된 동적모멘트의 성능과 회귀율을 동시에 확인한다. 본 논문에서 사용한 회귀방법은 RMS 오류율을 사용하였으며 제안된 학습방법인 동적모멘트를 SVM(Support Vector Machine)의 순차 학습방법인 KA(Kernel Adatron)과 KR(Kernel Relaxation)에 적용하여 RMS 오류율을 확인하였다. 공정한 학습 성능 평가를 위해 신경망 분류기 표준평가데이터인 SONAR 데이터를 이용하였으며 실험 결과 동적모멘트를 이용한 학습 성능과 수렴 속도 및 RMS 오류율이 정적모멘트를 이용한 학습방법보다 향상되었음을 확인하였다.

Performance and Root Mean Squared Error of Kernel Relaxation by the Dynamic Change of the Moment

Eun-Mi Kim^{*} and Bae-Ho Lee^{**}

ABSTRACT

This paper proposes using dynamic momentum for sequential learning method. Using The dynamic momentum improves convergence speed and performance by the variable momentum, also can identify it in the RMSE(root mean squared error). The proposed method is reflected using variable momentum according to current state. While static momentum is equally influenced on the whole, dynamic momentum algorithm can control the convergence rate and performance. According to the variable change of momentum by training. Unlike former classification and regression problems, this paper confirms both performance and regression rate of the dynamic momentum. Using RMSE(root mean square error), which is one of the regression methods. The proposed dynamic momentum has been applied to the kernel adatron and kernel relaxation as the new sequential learning method of support vector machine presented recently. In order to show the efficiency of the proposed algorithm, SONAR data, the neural network classifier standard evaluation data, are used. The simulation result using the dynamic momentum has a better convergence rate, performance and RMSE than those using the static moment, respectively.

Key words: Dynamic momentum, Root Mean Square, Sequential Learning Method

본 연구는 한국과학재단 지정 전남대학교 고품질전기전자
부품 및 시스템연구센터의 연구비 지원에 의해 연구되었음.

접수일 : 2002년 8월 28일, 완료일 : 2003년 2월 6일

^{*} 준회원, 전남대학교 컴퓨터공학과 대학원생

^{**} 정회원, 전남대학교 전자컴퓨터정보통신공학부 교수

1. 서 론

순차 학습 방법(sequential learning method)은 support vector machine(SVM)의 해를 산출하기 위해 계산되어지는 quadratic programming(QP) 문제를 해결하기 위한 접근방법 중 하나이다[1]. 이때 QP는 입력 패턴의 양이 커질 경우 시간과 공간에 있어서 복잡한 연산과 과도한 메모리를 요구하는 최대의 취약점을 갖는다. 순차 학습 방법인 kernel adatron(KA), kernel adatron using the moment(KAm), kernel relaxation(KR)등은 Vapnik에 의해 제안된 기존의 SVM을 수정 보완한 학습법으로 또 다른 접근방법인 재구성(decomposition)방법보다 직접적인 접근 방법으로 볼 수 있다[1,2].

KA의 학습 방법은 SVM의 비선형 특징 공간에 기존의 신경망 학습 방법을 접목한 방식으로 기울기 강하/상승(gradient descent/ascent)기법을 이용하여 가중치를 학습함으로써 해를 찾는 방법이다[3]. 그러나 이에 대한 체계적인 이론적 전개는 최근 발표된 KR에 의해 완성되었다[4]. 모멘트를 이용한 학습법은 과거의 학습을 현재의 학습에 반영하는 것으로 이때 모멘트는 상수이며 사용자에 의해 정의된다. 커널 공간에서 모멘트를 이용한 방법은 모멘트를 이용한 KA의 학습 성능 향상 기법으로 소개되었다[5].

KR은 KA와 동등한 학습 성능을 가지며 support vectors(SVs)의 수를 효과적으로 줄이는 저밀도 학습법이다[4]. 소개된 순차 학습법은 기존의 선형 판별 함수의 학습법을 커널 공간(kernel space)으로 확장한 것으로 볼 수 있다[4]. 이때 선형 판별 함수의 학습법에서 나타나는 수렴시의 발진 현상은 커널 공간에서도 발생하게 된다. 이러한 발진 현상을 줄여 수렴속도를 향상하는 방법으로는 기존의 선형 판별 함수의 학습법이나 신경망에서 제안된 모멘트를 이용할 수 있다[5]. 이에 본 논문에서는 기존의 상수 모멘트를 정적모멘트(static moment)로 정의하며 본 논문에서 제안한 동적모멘트(dynamic momentum)와 구분한다.

정적 모멘트는 일정기간의 학습 횟수와 학습 성능을 한계적인 비례관계로 둘 경우 임의의 한계점까지 학습 성능은 학습 횟수에 따라 증가하게 되며 수렴은 한계점과 학습 횟수가 균형을 이룬 것으로 설명할 수 있다. 순차 학습법의 발진은 한계점까지의 중간과 선에서 나타나는 현상으로 볼 수 있으며 결국 이러한

한계점까지의 도달 간격을 좁혀 줌으로써 발진을 억제할 수 있게 된다. 이러한 측면에서 볼 때 정적모멘트는 상수로 선언되어 학습에 있어 수렴 정도(학습의 횟수나 성능)에 관계없이 일정하게 영향력을 가짐으로써 평균적인 효과를 기대할 수 있다. 그러나 전체학습의 수렴과 학습정도에 관계없이 동일한 모멘트 값의 영향을 받는 것으로 이는 학습 횟수의 증가와 해의 수렴에 대한 기본원리를 충분히 반영하였다고 할 수 없다. 결국 이상적인 방법은 학습 성능과 학습 횟수에 따라 다른 크기를 갖는 모멘트를 갖는 것이다.

본 논문에서는 순차 학습법에서의 수렴시 발진억제를 위해 과거의 속성값을 반영하는 상수 τ 를 추가하여 수렴 정도에 따라 모멘트의 크기를 조절하여 순차 학습법의 수렴 속도와 학습 성능을 향상시키는 학습 방법을 제안한다.

공정한 학습 성능 평가를 위해 신경망 분류기 표준 평가 데이터인 SONAR 분류 문제를 이용하였으며[6]이며, 회귀문제의 평가를 위해 root mean square error(RMSE)를 사용하였다.

실험 결과 수렴 속도와 학습 성능 면에서 기존의 정적모멘트를 이용한 방법에 비해 향상된 성능과 수렴 속도를 보이는 것을 확인하였으며 RMSE 또한 동적모멘트일 경우 에러율이 $7.3943e-001$ 로 감소되는 것을 확인하였다.

2. 수정된 Support Vector Machine

가중치를 바이어스만큼 확장하여 기존과 달리 가중치와 바이어스를 동시에 학습시킴으로 순차적 학습을 할 수 있다. 입력 공간에서의 가중치 \mathbf{w} 와 바이어스 항 w_0 로 확장한 확장 공간에서 가중치 벡터 $\mathbf{a}=[\mathbf{w}, w_0]$ 를 최소화하면 가중치 벡터 \mathbf{w} 와 바이어스 항 w_0 는 동시에 최소화된다. 이때 $\mathbf{y}=[\mathbf{x}, 1]$ 로 확장된다[7].

분리 문제(separable problem)인 경우 입력값 \mathbf{x}_i 와 목표치 d_i 에 대한 선형판별 함수는 다음 조건을 만족한다[7].

$$\begin{aligned} g(\mathbf{x}_i) &> 0, \quad d_i = 1 \\ g(\mathbf{x}_i) &< 0, \quad d_i = -1. \end{aligned} \quad (2.1)$$

비분리 문제 (nonseparable problem)인 경우 선형 판별 함수는 margin b에 오차 ξ 를 도입하여 표현한다.

$$d_i g(\mathbf{x}_i) \geq b - \xi_i, \quad i=1, 2, \dots, N \text{ with } \xi_i \geq 0. \quad (2.2)$$

여기서 $\xi = 0$ 일 때 패턴으로부터 분리 초평면까지의 최대여유 r 를 갖는다.

가중치 \mathbf{a} 와 오차를 동시에 최소화하는 비용함수 Φ_a 는 아래와 같다[2].

$$\Phi_a(\mathbf{a}, \xi) = \frac{1}{2} \|\mathbf{a}\|^2 + C \sum_{i=1}^N \xi_i. \quad (2.3)$$

여기에서 C는 정규화 인수로 해석할 수 있다. 라그랑지안 승수법을 이용하여 라그랑지안 함수를 구성한다[2].

$$\begin{aligned} L(\mathbf{a}, \xi; \alpha, \beta) &= \frac{\|\mathbf{a}\|^2}{2} + C \sum_{i=1}^N \xi_i \\ &- \sum_{i=1}^N \alpha_i [d_i (\mathbf{a}' \mathbf{y}_i) - b + \xi_i] \\ &- \sum_{i=1}^N \beta_i \xi_i, \end{aligned} \quad \text{with } \alpha_i \geq 0, \beta_i \geq 0, \xi_i \geq 0. \quad (2.4)$$

이때 최적화 문제의 해는 \mathbf{a} 와 b에 관하여 최소, $\alpha_i \geq 0$ 일 때 α_i 에 관하여 최대가 되어야 한다.

쌍대 공간(dual space)에서의 라그랑지안 $L(\alpha)$ 는 아래와 같다[4].

$$L(\alpha) = b \sum_{i=1}^N \alpha_i - \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j d_i d_j \mathbf{y}_i^t \mathbf{y}_j \quad \text{subject to } 0 \leq \alpha_i \leq C. \quad (2.5)$$

기존의 SVM과 비교하면 등식 제한사항이 소거되었다. 따라서 순차 학습방법이 가능하게 되었으며, 두 번째 항의 $\alpha_i d_i$ 를 α_i 로 치환하면 첫 번째 항의 α_i 는 $\alpha_i d_i$ 가 된다.

$$L(\alpha) = b \sum_{i=1}^N \alpha_i d_i - \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j \mathbf{y}_i^t \mathbf{y}_j \quad \text{subject to } 0 \leq \alpha_i d_i \leq C. \quad (2.6)$$

기존의 SVM의 등식 제한조건은 제거되었으며 이

로써 커널 공간에서 가중치 \mathbf{a} 를 학습할 수 있는 순차 학습 알고리즘이 가능하게 된다[4].

3. 순차 학습 방법

3.1 Kernel Adatron

커널은 특징공간의 정확한 형태를 고려하지 않아도 최적 초평면(optimal hyperplane)을 구성하기 위해 사용할 수 있다[1]. 내적 커널의 정의는 다음과 같다.

$$\mathbf{K}(\mathbf{x}, \mathbf{x}_i) = \boldsymbol{\varphi}^T(\mathbf{x}) \boldsymbol{\varphi}(\mathbf{x}_i). \quad (3.1.1)$$

여기 $\boldsymbol{\varphi}(\mathbf{x})$ 는 크기가 $(m+1)$ -by-1로 확장된 특징 벡터이다. 사용할 커널은 머서의 정리(Mercer's theorem)를 만족해야 한다[1]. 실험에서 머서의 정리를 만족하는 아래의 RBF 커널을 사용하였다.

$$\exp\left(-\frac{1}{2\sigma^2} \|\mathbf{x} - \mathbf{x}_i\|^2\right) \quad (3.1.2)$$

이때 σ^2 은 사용자에게 의해 정의될 수 있다. 커널 공간에서 커널 행렬을 패턴 행렬로 커널의 행과 열을 패턴 벡터로 정의하면 라그랑지안 승수 \mathbf{a} 는 패턴의 가중치가 되어 아래와 같이 $\mathbf{g}_k(\mathbf{Y})$ 를 정의한다.

$$\mathbf{g}_k(\mathbf{Y}) \equiv \text{sign}(\mathbf{K} \mathbf{a}) = \mathbf{d}. \quad (3.1.3)$$

이때 커널 판별 함수는 다음과 같다[1].

$$g_k(\mathbf{y}_j) \equiv \mathbf{a}' \mathbf{K}(:, j) = \mathbf{K}(j, :) \mathbf{a} \quad (3.1.4)$$

따라서 입력 공간은 방정식의 수가 미지수의 수보다 많은 과잉결정 구조(overdetermined system)이고 커널공간은 방정식의 수와 미지수의 수가 같은 완전 결정 구조(exactly determined system)가 된다. 결국 커널 공간에서는 비특이 행렬 \mathbf{K} 에 대하여 해가 언제나 존재하게 된다[4]. 따라서 선형판별 함수의 모든 학습 방법을 그대로 커널 공간에 적용할 수 있게 된다. 커널의 최대 강하/상승법은 아래와 같다 [4].

$$\Delta \mathbf{a} = \eta (\mathbf{d} - \mathbf{a}' \mathbf{K}). \quad (3.1.5)$$

위 식을 요소(component)별로 표현하면 다음과

같다[4].

$$\Delta a_j = \eta(d_j - \mathbf{a}^t \mathbf{K}(:,j)). \quad (3.1.6)$$

위 식은 Campbell이 제안한 KA(kernel adatron)이다[3]. 하지만 주어진 패턴 y_j 에 대해 벡터 \mathbf{a} 의 요소 중에서 a_j 만 학습하는 것으로 이 방법은 최대 강하 학습법이 아니라 좌표 강하법이다[4].

3.2 Kernel Relaxation

KR은 주어진 가중치 \mathbf{a} 를 학습하는 방법으로 유도 과정은 다음과 같다[4]. 식(3.1.3)에 의해 error cost function(오류 비용 함수)을 아래와 같이 정의한다[2].

$$E = \xi^2 = \frac{1}{2} \|(\mathbf{d} - \mathbf{K} \mathbf{a})\|^2, \quad (3.2.1)$$

$$\frac{\partial E}{\partial \mathbf{a}} = (\mathbf{d} - \mathbf{K} \mathbf{a})(-\mathbf{K}). \quad (3.2.2)$$

따라서 위 식은 주어진 가중치 \mathbf{a} 를 학습하는 아래의 식으로 변경되며, 신경망에서 델타 규칙(Delta Rule)로 알려져 있다.

$$\Delta \mathbf{a} |_{y_j} = \eta(d_j - \mathbf{a}^t \mathbf{K}(:,j)) \mathbf{K}(:,j). \quad (3.2.3)$$

4. 동적 모멘트를 이용한 학습방법

이미 모멘트를 이용한 SVM의 학습 성능 향상은 KA를 이용한 방법으로 소개되었다[5]. 이 학습방법은 단순히 선형 판별 함수의 학습법에서 제안된 모멘트를 KA알고리즘에 적용한 것으로 각 알고리즘에 대한 최적 해로의 수렴과 성능 향상을 위한 모멘트의 선택에 많은 어려움을 가지며 기존의 정적모멘트를 매우 완만한 곡선 형태로 수렴에 근접해 가는 최대 강하 학습법을 KR에 적용하였을 경우 아래와 같은 두 가지 측면에서의 문제점을 생각해 볼 수 있다.

첫 번째 문제점은 수렴을 위해 필요한 충분히 작은 모멘트를 선언했을 경우로 학습 초기 모멘트의 영향을 거의 받지 못함으로 수렴시간과 성능 면에서 정적모멘트 사용이전에 비해 향상된 변화를 기대할 수 없다는 것이다. 두 번째 문제점은 초기 학습의 속도를 높이기 위해 수렴점에서의 모멘트 보다 큰 모멘

트가 주어졌을 경우이다. 결국 정적모멘트는 초기 선언된 모멘트가 전체학습에서 균등하게 적용됨으로써 KR의 특성인 수렴점에 대해 매우 유연한 곡선으로 근접하는 학습법에 대해서는 수렴속도와 성능 개선에 대한 효과적인 대안이 되지 못한다. 따라서 최적의 모멘트는 정적 모멘트에서 나타난 단점들을 보완하고 장점을 살릴 수 있는 모멘트 스케줄링 방법이 된다. 본 논문에서는 수렴에 따른 가변적인 모멘트를 이용하여 SVM의 학습 성능을 개선하는 새로운 방법을 제안한다. 이러한 스케줄링 방법을 위해 본 논문에서는 학습 횟수와 성능을 한정적 비례관계로 가정한다. 한정적 비례관계란 초기 한정된 구간 내의 학습에서 최초 학습 성능향상을 기대할 수 있는 임의의 학습 회수와 성능과의 관계를 말하며 이때 학습 성능은 학습횟수와 비례관계를 가지게 된다. 따라서 순차 학습법의 발진은 초기 한정적 비례관계의 구간에서 또 다른 한정적 비례관계로 가는 중간 과정에서 나타나는 일련의 현상으로 볼 수 있다. 이러한 일련의 현상들은 다음 한계점까지의 도달 간격을 조정하여 간격을 좁힐 수 있으며 이는 곧 학습에서 발진을 억제하면 성능 개선을 가져온다. 결국 동적 모멘트를 이용한 학습방법은 학습 초기 모멘트의 크기를 크게 하여 수렴 속도를 높였으며 점차 학습 수행 횟수가 많아져 수렴에 가까워질수록 모멘트의 크기는 정밀하게 조정된다. 일반적으로 모멘트 값이 학습률을 증가할 경우 시스템이 국부적인 변화에 덜 민감해 지는 것으로 알려져 있다[7]. 하지만 학습 횟수에 따라 연속적으로 모멘트를 증가시킬 경우 모멘트는 단조 증가 함수가 되어 학습횟수가 매우 많아지게 되며 과거의 학습이 현재의 학습을 능가하게 되어 학습 성능과 수렴 속도에 대해 개선 효과를 가져오지 못하게 된다. 따라서 제안된 동적 모멘트를 이용한 학습법은 조정된 모멘트의 크기가 초기 최대모멘트 크기보다 작은 범위 내에서 조정되게 된다. 결국 KR의 수렴특성에 따라 초기 모멘트 값은 학습 횟수에 따른 증가를 보이며 수렴에 가까워질수록 다시 모멘트의 크기를 최대 모멘트 크기 이내에서 결정되게 함으로써 전체적으로 KR의 수렴속도를 향상시키고 성능을 개선시키는 방법이 된다.

본 논문에서 제안하는 모멘트는 \mathbf{M} 으로 아래와 같이 정의된다.

τ 는 초기 모멘트의 값의 크기를 최소화하기 위해

$$M = \frac{m(k+1)}{\tau}$$

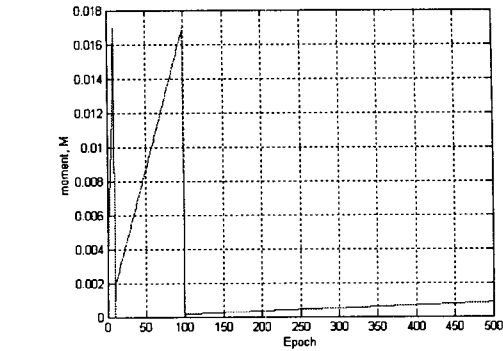
if $M > m$; $M = 0$; $\tau = \tau^2$;

endif (4.1)

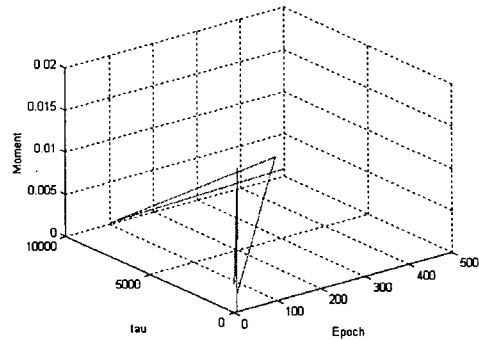
사용한 제어 변수로 예상된 최초 한계치에서의 학습 횟수가 된다. m 은 기존의 상수 모멘트로 동적 모멘트를 이용한 학습 방법에서는 적용될 모멘트의 상한값이 된다. 결국 동적 모멘트 M 은 앞서 말한 것과 같이 학습 횟수 k 에 의해 정의된 모멘트 한계치 m 의 범위 내에서 세밀하게 조정된다. 제한 조건으로 $M < m$ 을 두어 M 이 단조 증가되는 것을 제한함으로써 기존의 상수 모멘트인 m 을 초기 선언된 모멘트 내에서 가변적 변화가 가능하도록 하였다. 이때 최초의 한계치에 대한 학습 횟수인 τ 에 도달 할 경우 $M=m$ 이 되어 최초의 한계치에서 다시 모멘트 값은 초기화되어 현재의 학습을 그대로 반영하게 된다. 결국 M 의 값이 모멘트의 상한값인 m 보다 클 경우 동적모멘트를 다시 0으로 하고 τ 를 τ^2 으로 다음번의 예상된 한계치를 최대로 뒀으로써 예상된 한계치 내에서 다시 모멘트를 증가하면서 학습하게 된다. 이때 처음 정의된 한계치에서의 모멘트의 증가는 $\frac{m(k+1)}{\Delta\tau}$ 의 증가치를 가지며 학습하게 된다. 이러한 학습 방법은 사용자 정의에 의해 정의되는 정적 모멘트로 결정하기 어려운 KR의 유연한 초평면에 대한 모멘트의 값을 지정할 수 있다. 또한 기존의 모멘트 값의 변경에만 의존하던 모멘트의 정의에 예상된 한계치에 대한 학습 횟수를 추가함으로써 보다 적극적인 모멘트 스케줄링을 정의할 수 있다.

그림 1은 학습에 참여하지 않은 단순한 모멘트변화를 나타낸 그래프이다. 그림 1-a은 $\tau=10$ 일 때 모멘트의 변화를 나타낸 것으로 가로축은 학습 횟수, 세로축은 모멘트의 변화를 나타낸다. 그림 1-a에서 $M=0$ 인 부분은 $M=m$ 인 경우이며 이 때 τ 는 각각 10과 100이 되며 다음번 τ 는 10000이 되나 그림 1에서는 제외하였다. 즉, 그림 1-b에서는 τ 와 모멘트의 관계를 보여 주는 그래프로 학습 횟수가 증가함에 τ 의 영향으로 100번째 학습 이후에는 상당히 완만한 경사를 이루며 증가 해가는 것을 볼 수 있다. 결국 이러한 모멘트의 스케줄링 방법은 최대 강하 학습법인 KR의 수렴곡선이 가지는 미세한 진동현상을 줄

여 수렴 속도의 개선을 가져온다. 여기에서 정의되는 모멘트의 값은 기존의 정적모멘트에서 사용하는 수동적인 방법으로써는 그 값을 예측하기가 힘들다. 표 1은 제안된 동적 모멘트를 순차 학습법의 KR에 적용한 전체적인 알고리즘을 나타낸 것이다.



(a) DM



(b) τ 에 따른 DM

그림 1. 모멘트 변화 그래프

5. Root Mean Square Error

회귀문제는 입력 값이 고차원인 특징 영역에서 나타난다. 본 논문에서는 회귀문제의 한 방법인 RMS 오류율을 사용하였다. RMSE는 (Mean Square Error)MSE의 절대값인 표준 에러로 성능에 대한 어려움을 확인함으로써 회귀문제를 해결할 수 있다.

$$RMSE = \sqrt{\frac{1}{T} \sum_{i=1}^T (Y_i^s - Y_i^a)^2} \tag{5.1}$$

$\langle Y_i^s = Y_i$ 의 결과값, $Y_i^a = Y_a$ 의 실제값

$Y_i^s = Y_i^a$ 일 경우 100%의 성능을 갖게 되며 이 때 RMSE는 0이 된다. 즉 RMSE가 작을수록 성능은 향상되어진다.

표 1. Dynamic Momentum Algorithm for KR

```

Given Training Data Y, d
Construct Kernel K

Begin initialize
    a = 0,  $\eta(\cdot)$ ,
    margin b,  $\gamma$  ( $b > \gamma > 0$ ),  $k = 0$ 

do
    /* for each epoch */
    Shuffling the training data
    /* for each training data */

for  $k = 1, \dots, N$ 
    if  $d_k \mathbf{a}^t \mathbf{K}(:, k) \leq \gamma$ 
        if  $d_k (\mathbf{a}(k)|_k + \Delta \mathbf{a}(k)|_k) < 0$   $\alpha_k = 0$ 

    else
         $\mathbf{M} = \frac{m(k+1)}{\tau}$ 

        if  $\mathbf{M} > m$ 
             $\mathbf{M} = 0$ 
             $\tau = \tau^2$ 
        endif
         $\mathbf{a}(k+1) = \mathbf{a}(k) +$ 
             $\eta(k) \frac{d_k b - \mathbf{a}^t \mathbf{K}(:, k)}{\|\mathbf{K}(:, k)\|^2} \mathbf{K}(:, k)$ 
             $+ \mathbf{M} (\mathbf{a}(k) - \mathbf{a}(k-1))$ 
        endif
    else
         $\alpha_k = 0$ 

    end if
end for
.
.
until  $d_k \mathbf{a}^t \mathbf{K}(:, k) > \gamma$  for all  $k$ 
return a
End
    
```

6. 실험 및 결과

6.1 실험 환경

본 실험은 Windows 2000서버를 사용하였으며, 시스템의 구현을 위해 Matlab Ver. 6.1을 사용하였다. 본 실험에서 사용한 데이터는 신경망 분류기 표준 평가 데이터인 SONAR 데이터[6]로 차원의 수는 60차원, 데이터의 총 개수는 208개로 처음 104개를 학습 데이터로 나머지 104개를 테스트 데이터로 사용하여 실험하였다. 실험은 제안된 동적모멘트의 학습 성능과 RMSE를 평가하기 위한 것으로 KA와 KR에 각각 정적·동적모멘트를 적용하여 학습성능과 RMSE를 확인하였다. KA에 사용된 최적 정적모멘트의 값과 동적모멘트의 값은 0에서 2까지 0.1씩 증가시키며 각 모멘트 값에 대해 매회 50번씩 학습한 결과 중 가장 우수한 학습성능을 갖는 모멘트 값을 확인하였으며 KR의 최적 모멘트 값의 산출도 KA와 동일하게 모멘트를 변화시켰다. 이하 실험에서 정적 모멘트 값과 동적모멘트 값은 실험 결과 최적의 분류 성능을 보인 KA에서는 3.0e-3과 2.6e-1, KR에서는 4.0e-3와 1.7e-2를 적용하였다.

6.2 실험 결과

본 실험은 RBF함수에서 사용자 정의 인수 σ 값의 변화에 따른 학습 성능을 알아보기 위해 σ 를 0.1에서 0.9까지 0.1씩 변화시켰으며 각 Epoch당 학습 성능을 알아보기 위해 Epoch를 50번에서 200번까지 변화시켜 실험하였다. 또한 동적 모멘트 사용 이전과 동적 모멘트의 사용 후에 대한 최적의 학습 성능을 알아보기 위해 최적의 모멘트를 이용하여 σ 각각에 대한 학습성능을 비교하였다. 이하 모멘트 사용하지 않은 것을 NM(not moment), 정적모멘트를 적용한 것을 SM(static moment), 동적모멘트를 적용한 것을 DM(dynamic momentum)라 칭한다. 표 2는 KA에 모멘트를 사용하지 않은 경우와 최적의 모멘트를 사용한 기존의 정적모멘트 및 본 논문에서 제안한 동적 모멘트 각각에 대한 학습성능과 그때의 RMS 오류율을 비교한 것이다. 전체적인 실험은 σ 의 변화에 따른 학습 성능을 비교해 보는 것으로 σ 의 변화를 0.1~0.9까지 변화시킨 결과 중 NM, DM, SM의 성능이 가장 좋았던 σ 값이 0.3일 경우만을 나타내었다.

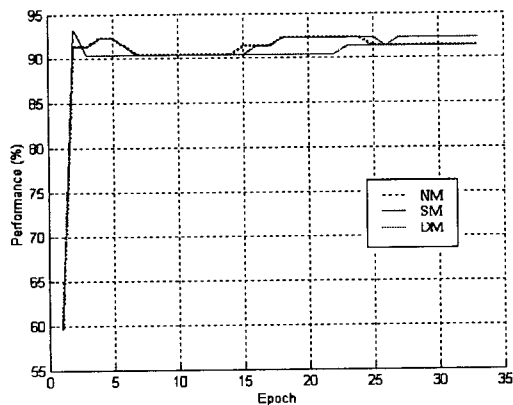
표 2. KA에서 σ 에 대한 학습 성과와 RMSE 비교

(단위: %)

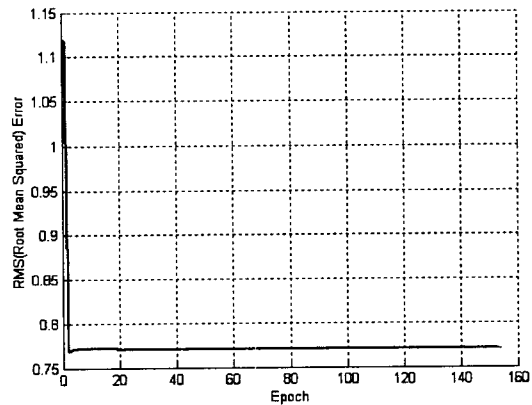
| σ | | Epoch | 50 | 100 | 150 | 200 | 수렴 | % |
|----------|----------------------|-------|-------------|-------------|-------------|-------------|----|-------------|
| | | | | | | | | |
| 0.3 | NM(Training/Testing) | | 100 / 91.34 | 100 / 91.34 | 100 / 91.34 | 100 / 91.34 | 33 | 100 / 91.34 |
| | RMS Error | | 7.7059e-001 | 0.7706e-001 | 0.7706e-001 | 0.7706e-001 | | 0.7706e-001 |
| | DM(Training/Testing) | | 100 / 92.30 | 100 / 92.30 | 100 / 92.30 | 100 / 92.30 | 33 | 100 / 92.30 |
| | RMS Error | | 7.7104e-001 | 7.7104e-001 | 7.7104e-001 | 7.7104e-001 | | 7.7000e-001 |
| | SM(Training/Testing) | | 100 / 91.34 | 100 / 91.34 | 100 / 91.34 | 100 / 91.34 | 33 | 100 / 91.34 |
| | RMS Error | | 7.7061e-001 | 7.7061e-001 | 7.7061e-001 | 7.7061e-001 | | 7.7000e-001 |

실험 결과 훈련 데이터(training data)의 경우, 수렴 시 가장 높은 학습 성능은 거의 100%로 나타났으며 실험 데이터(testing data)의 경우, 모멘트를 사용하지 않을 때 수렴 시 가장 높은 학습 성능은 $\sigma=0.3$ 에서 91.34%, 정적모멘트를 사용했을 때 91.34%, 동적모멘트를 사용했을 때 92.30%로 동적모멘트 적용시 학습 성능의 향상을 알 수 있었다. 그러나 RMS 에러율은 그림 2-b처럼 모멘트 적용 전과 모멘트 적용 후의 다소 적은 차이를 보일 뿐 거의 같은 곡선을 나타냈다. 이는 KA의 빠른 수렴 속도에 따른 문제점으로 사료된다. 이에 KR과 같은 유연한 초평면을 갖으며 잦은 발진 현상을 보이는 relaxation에서 확실한 평가가 가능한가를 알아보기로 한다. 표 3은 KR에서의 NM, SM, DM과 그 때의 RMSE를 비교한 것이다. 실험은 KA와 동일하게 하였으며 실험 결과 모멘트를 사용하지 않은 경우 수렴 시 가장 높은 학습

성능은 $\sigma=0.3$ 일때 93.26%, 정적모멘트를 사용했을 때 $\sigma=0.4$ 에서 최적 성능이 94.23%, 동적모멘트를 사용했을 때 $\sigma=0.4$ 에서 96.15%로 나타났다. 두 실험 결과 DM>SM>NM순으로 전체적으로 모멘트를 이용하여 학습하였을 경우 향상된 학습성능을 보였으며 정적모멘트보다 본 논문에서 제안한 동적모멘트가 우수한 것으로 나타났다. 그러나 실험에서 모멘트를 이용하지 않은 KR의 학습 성능에서 알 수 있듯이 σ 의 조정만으로는 성능 향상에 한계가 있음을 알 수 있다. 그림 3은 $\sigma=0.4$ 에서의 Epoch의 변화에 따른 학습성과 RMSE를 그래프로 나타낸 것이다. 모멘트를 사용하지 않을 때 Epoch 30에서 학습성능이 높게 나왔으나 이것은 학습 중에 나타나는 발진에 불과하며, 정적모멘트를 이용하여 학습 성능의 향상을 가져왔으나 수렴속도에는 약 160정도로 크게 변화가 없는 것을 알 수 있다. 수렴에 근접해 갈수록 동적모멘



(a) 학습 성능



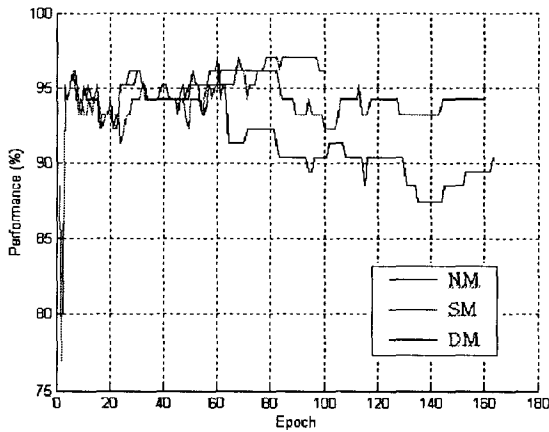
(b) RMSE

그림 2. KA에 적용한 NM, SM, DM의 학습 성과와 RMSE

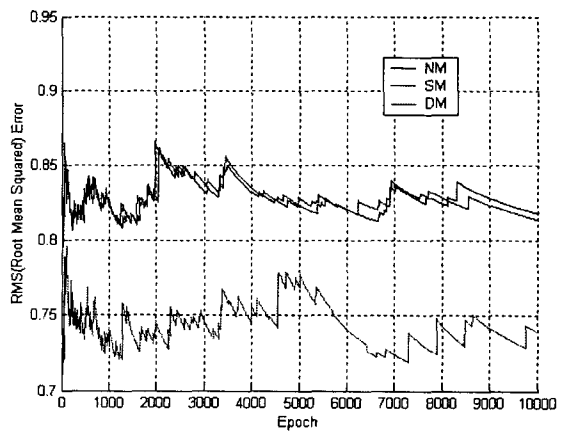
표 3. KR에서 σ 에 대한 학습 성과와 RMSE 비교

(단위: %)

| σ | | Epoch | 50 | 100 | 150 | 200 | 수렴 | % |
|----------|----------------------|-------|-------------|---------------|---------------|---------------|-----|---------------|
| | | | | | | | | |
| 0.3 | NM(Training/Testing) | | 100 / 93.26 | 100 / 93.26 | 100 / 93.26 | 100 / 93.26 | 34 | 100 / 93.26 |
| | RMS Error | | 8.2414e-001 | 8.2414e-001 | 8.2414e-001 | 8.2414e-001 | | 8.2414e-001 |
| | DM(Training/Testing) | | 100 / 93.26 | 100 / 92.26 | 100 / 92.26 | 100 / 92.26 | 34 | 100 / 92.26 |
| | RMS Error | | 8.2742e-001 | 8.2742e-001 | 8.2742e-001 | 8.2742e-001 | | 8.2742e-001 |
| | SM(Training/Testing) | | 100 / 93.26 | 100 / 92.26 | 100 / 92.26 | 100 / 92.26 | 34 | 100 / 92.26 |
| | RMS Error | | 8.2749e-001 | 8.2749e-001 | 8.2749e-001 | 8.2749e-001 | | 8.2749e-001 |
| 0.4 | NM(Training/Testing) | | 100 / 95.19 | 98.07 / 90.38 | 99.03 / 88.46 | 99.03 / 90.38 | 163 | 99.03 / 90.38 |
| | RMS Error | | 7.2678e-001 | 7.9300e-001 | 7.5783e-001 | 7.5352e-001 | | 7.5352e-001 |
| | DM(Training/Testing) | | 100 / 95.19 | 100 / 96.15 | 100 / 96.15 | 100 / 96.15 | 100 | 100 / 96.15 |
| | RMS Error | | 7.2587e-001 | 7.2587e-001 | 7.2587e-001 | 7.2587e-001 | | 7.2587e-001 |
| | SM(Training/Testing) | | 100 / 95.19 | 99.03 / 92.30 | 100 / 94.23 | 100 / 94.23 | 160 | 100 / 94.23 |
| | RMS Error | | 7.2883e-001 | 7.2883e-001 | 7.2883e-001 | 7.2883e-001 | | 7.2883e-001 |



(a) 학습 성능



(b) RMSE

그림 3. KR에 적용한 NM, SM, DM의 학습 성과와 RMSE

트를 사용하였을 때의 학습성능이 Epoch의 증가에 따라 향상되는 것을 볼 수 있었다. 즉, 동적모멘트를 이용한 학습방법이 정적모멘트를 이용한 학습방법보다 우수한 학습 성능을 보였으며 수렴속도가 두 배 정도 빨라진 것을 알 수 있다. 이와 마찬가지로 회귀현상에서도 표 3과 그림 3-b에서 나타나는 것처럼 모멘트 사용전이나 기존의 정적모멘트 사용보다 동적모멘트의 적용시 RMS 오류율이 상당히 감소됨을 확인할 수 있다. 결과적으로 수렴점에 매우 유연한 초평면을 갖는 KR의 학습에 있어서 정적모멘트를 이용한 방법보다 동적모멘트를 이용한 학습방법

이 학습성과 수렴 속도 향상에 효과적인 방법이 됨을 RMSE를 통해 확인할 수 있다.

7. 결 론

본 논문에서는 SVM의 순차 학습방법에 있어서 수렴을 개선을 위해 동적 모멘트를 이용한 학습 방법을 제안하였다. 제안된 학습 방법은 수렴에 따른 모멘트의 값을 동적으로 변경하는 방법이다. 제안된 동적모멘트를 이용한 학습법을 SVM의 순차학습방법인 KA와 KR을 이용하여 실험하였다. 실험 결과 KA

와 KR가 수렴에 이를 경우 RMS 에러율은 수렴되지 않았지만 회귀율의 향상을 볼 수 있었다. 보다 정확한 결과 산출을 위해 Epoch을 10,000회 까지 실험한 결과 RMS 에러율이 모멘트 사용 전과 정적 모멘트의 사용의 유사함과 동적모멘트의 상당한 감소를 볼 수 있었다. 이로써 다시 한번 동적모멘트의 우수성을 확인할 수 있었다. 그러나 제안된 동적모멘트를 이용한 학습법에서의 τ 값이 성능 향상에 직접적인 영향을 미치는가에 대해서 유동적인 τ 값의 변화에 따른 연구가 필요하다.

참 고 문 헌

[1] J. C. Platt, "Fast training of support vector machines using sequential minimal optimization," *Advances in Kernel Methods- Support Vector Learning*, B. Scholkopf, C. Burges, A. Smola, editors, pp. 185-208, MIT-Press, 1998.

[2] S. Haykin, *Neural Networks, A comprehensive Foundation Second Edition* by Prentice-Hall, Inc, 1999.

[3] T. T Friess, N. Cristianini, C. Campbell, "The Kernel-Adatron Algorithm: a Fast and Simple Learning Procedure for Support Vector Machines," in Shavlik, J., ed., *Machine Learning: Proceedings of the Fifteenth International Conference*, Morgan Kaufmann Publishers, San Francisco, CA 1998.

[4] 류재홍, 정종철 "커널 이완절차에 의한 커널 공간의 저밀도 표현 학습," *한국퍼지 및 지능시스템학회*, 2001년도 추계 학술대회 학술 발표 논문집, Vol. 11, No. 2, pp. 60-64, 2001.12.

[5] 조용현, "모멘트를 이용한 Support Vector Machines의 학습성능개선," *한국정보처리학회 논문지* 제 7권 제 5호, pp. 1446-1455, 5, 2000

[6] Y. Li et al. . "The relaxed online maximum margin algorithm," In *Advances in NIPS 13*, 1999.

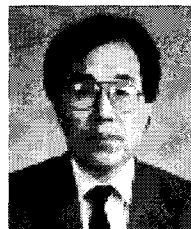
[7] R. O. Duda, P. E. Hart, D. G. Stork, *Pattern Classification, Second Edition* by John Wiley & Sons, Inc, 2001.



김 은 미

2001년 여수대학교 컴퓨터공학과 (공학사)
2001년~2003년 전남대학교 대학원 컴퓨터공학과(공학석사)

관심분야: 신경망, 패턴인식, 영상처리, 전문가시스템 등
E-mail : em1342@hotmail.com



이 배 호

1978년 한양대학교 전자공학(공학사)
1980년 한국과학기술원 전기 및 전자공학(공학석사)
1993년 미국 University of Missouri, Columbia(공학박사)

1980년~1983년 국방과학연구소
1993년~현재 전남대학교 전자컴퓨터정보통신공학부, 정보통신연구소 부교수

관심분야: 영상처리, 컴퓨터 비전, 인공지능, 패턴인식, 신경회로망 등
E-mail : bhlee@chonnam.ac.kr

교 신 저 자

김 은 미 500-757 광주광역시 북구 용봉동 전남대학교 컴퓨터공학과 인공지능연구실(공 6-719)