Ramp 임계 함수를 적용한 적응 학습 알고리즘의 수렴성

박소희 · 조제황
동신대학교 전기전자공학부

Convergence Properties of a Adaptive Learning Algorithm Employing a Ramp Threshold Function

So-Hee,Park and Che-Hwang,Cho
Dongshin University, Dept. of Electrical and Electronic Engineering
hee023@hanmail.net, chcho@white.dongshinu.ac.kr

요 약

적응 학습 알고리즘으로 가중치를 변화시키는 단층 신경망의 출력부에 Ramp 임계 함수를 적용하여 입력이 zero-mean Gaussian random vector인 경우 가중치의 stationary point를 구하고, 적응 학습 알고리즘을 유도한다.

I. 서 론

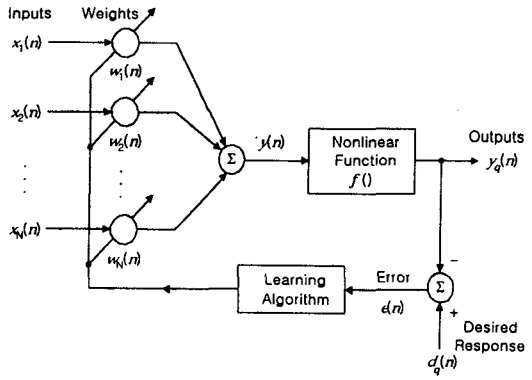

그림 1. 비선형 함수를 갖는 적응 학습 시스템

단층 퍼셉트론(Single-layer Perceptron)[1,2]이나 적응 선형 뉴론(ADALINE ; Adaptive Linear Neuron)[3]은 그림 1과 같이 N 개의 적응 가중치(Adaptive Weight)와 하나의 합산 노드(Summing Node)로 구성된다. 중간 단계의 출력 $y(n)$은 입력 $x_1(n)$, $x_2(n)$, $\cdots x_N(n)$ 와 가중치 $w_1(n)$, $w_2(n)$, $\cdots w_N(n)$ 의 내적으로 표현되고, $y(n)$는 비선형함수 $f()$에 의해 최종 출력 $y_q(n)$ 로 변환된다. 비선형 함수는 일반적으로 최소값과 최대값을 각각 -1 과 +1 로 제한하는 함수이며, hard limiter, ramp, sigmoid 등이 대표적인 함수이다. 오차 $e(n)$은 요구된 출력 $d_q(n)$와 계산된 출력 $y_q(n)$의 차로써 알고리즘의 입력이며, 그 결과는 가중치를 변화시킨다. 즉 알고리즘은 오차 $e(n)$이 감소되도록 가중치를 변화시킨다. 본 논문에서는 비선형 함수를 ramp 함수로 사용하는 경우, 가중치의 수렴 결과인 stationary point를 구하고 가중치를 수렴시키는 알고리즘을 얻고자 한다.

II. 본 론

1. 적응 학습 알고리즘의 유도

그림 1에 주어진 적응 시스템에서 n번째 가중치와 zero-mean Gaussian 분포를 갖는 입력을 transpose를 의미하는 T를 사용하여 행렬로 표현하면 각각 다음과 같다.

$$W(n)=[w_1(n), w_2(n), \cdots, w_N(n)]^T \quad (1)$$

$$X(n)=[x_1(n), x_2(n), \cdots, x_N(n)]^T \quad (2)$$

이 때 중간 단계의 출력 $y(n)$은 다음과 같이 표현할 수 있다.

$$y(n)=W^T(n)X(n)=X(n)^T W(n) \quad (3)$$

그림 1에 주어진 것과 같이 계산된 출력 $y_q(n)$은 그림 2에 주어진 비선형 함수 $f()$의 결과로써 중간 단계의 출력 $y(n)$을 입력으로 하며, 따라서 다음과 같이 표현할 수 있다.

$$y_q(n)=f(y(n)) \quad (4)$$

오차 $e(n)$은 그림 1에 주어진 것과 같이 요구된 출력 $d_q(n)$와 계산된 출력 $y_q(n)$의 차로써 다음과 같다.

$$e(n)=d_q(n)-y_q(n) \quad (5)$$

그림 1에 주어진 것과 같은 적응 시스템의 가중치를 학습(Learning)시키는 알고리즘은 일반적으로 다음과 같이 나타낼 수 있다.

$$W(n+1)=W(n)+\mu(-\nabla_n) \quad (6)$$

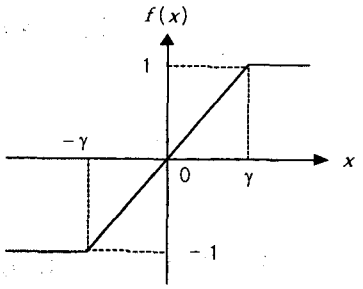

그림 2. ramp 임계함수

μ 는 수렴 속도를 결정짓는 step size로써 양의 실수이며, ∇_n 은 오차 $e(n)$의 제곱에 대한 가중치 $W(n)$의 변화율로써 학습반복수 n 이 증가함에 따라 감소하고, 따라서 다음과 같이 표현된다.

$$\nabla_n=\frac{\partial e^2(n)}{\partial W(n)}=2e(n)\frac{\partial e(n)}{\partial W(n)} \quad (7)$$

식(4)와 식(5)를 이용하여 식(7)의 미분 항을 구하기 위해 chain rule을 적용하면 다음과 같은 결과를 얻을 수 있다.

$$\frac{\partial e(n)}{\partial W(n)}=-g(y(n))X(n) \quad (8)$$

여기서 $g(y(n))$은 $\frac{\partial f(y(n))}{\partial y(n)}$ 과 동일한 표현이며, 그림 2에 주어진 ramp 임계함수를 변수에 대해 미분한 결과로써 그림 3과 같이 주어진다.

식(7)과 식(8)을 식(6)에 대입할 때 그림 1에 주어진 적응시스템의 출력함수를 그림 2와 같은 ramp 임계함수로 사용하는 경우 다음과 같은 학습 알고리즘을 얻을 수 있다.

$$W(n+1)=W(n)+2\mu e(n)g(y(n))X(n) \quad (9)$$

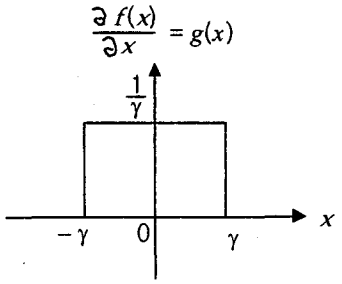

그림 3. ramp 임계함수의 미분

그림 3에서 알 수 있는 바와 같이 $|y(n)| > \gamma$ 인 경우 $g(y(n))=0$ 이고, 이 결과를 식(9)에 대입할 때 $W(n+1)=W(n)$ 이므로 더 이상 학습이 진행되지 않음을 의미한다. 따라서 아직 수렴점에 도달하지 않고 학습이 진행 중에 있을 때 식(9)는 $-\gamma \le y(n) \le \gamma$ 에서 의미가 있으므로 다음과 같이 수정할 수 있다.

$$W(n+1)=W(n)+\frac{2}{\gamma}\mu e(n)X(n) \quad \text{for } -\gamma \le y(n) \le \gamma \tag{10}$$

2. 알고리즘의 stationary point

식(10)의 기대치를 구하면 다음과 같다.

$$E[W(n+1)]=E[W(n)]+\frac{2}{\gamma}\mu E[e(n)X(n)] \quad \text{for } -\gamma \le y(n) \le \gamma \tag{11}$$

학습이 완료되어 수렴된 경우의 가중치 행렬을 stationary point 라하고 W_* 로 표시하는 경우, $E[W(n+1)] = E[W(n)] = W_*$ 이므로 식(11)은 다음과 같이 정리된다.

$$E[e_*(n)X(n)]=0, \text{for } -\gamma \le y(n) \le \gamma \tag{12}$$

식(12)는 수렴점에서 오차 $e_*(n)$와 입력 $X(n)$이 통계적으로 직교함을 나타낸다. 식(12)에 식(5)와 식(4)를 대입하면, 다음과 같은 식을 얻을 수 있다.

$$E[X(n)d_q(n)]=E[X(n)f(y(n))] \quad \text{for } -\gamma \le y(n) \le \gamma \tag{13}$$

식(13)의 좌변은 $X(n)$와 $d_q(n)$의 cross correlation으로써 다음과 같이 P 로 표시한다.

$$E[X(n)d_q(n)]=P \tag{14}$$

우변은 Price 정리를 이용하여 다음과 같은 결과를 유도할 수 있다.

$$E[X(n)f(y_*(n))]=\xi W_* R \tag{15}$$

여기서 R 은 auto-correlation $E[X^T(n)X(n)]$이고, ξ 은 다음과 같다.

$$\xi=\frac{1}{\gamma}\, erf\left(\frac{\gamma}{\sqrt{2}\sigma}\right) \tag{16}$$

여기서 γ 는 그림 2에 주어진 ramp 임계함수의 파라매터이고, σ^2 는 $y_*(n)$ 의 variance로써 zero-mean 인 경우, 다음과 같이 주어진다.

$$\sigma^2=E[y_*^2(n)]=W_*^T R W_* \tag{17}$$

식(13)에 식(14)와 식(15)를 대입하여 stationary point를 구하면 다음과 같다.

$$W_* = \frac{1}{\xi}R^{-1}P \tag{18}$$

만약 $\gamma \geq 3 \cdot \sqrt{2}\sigma$ 인 경우 식(16)에서 $\xi = \frac{1}{\gamma}$ 이므로 식(18)은 다음과 같이 정리된다.

$$W_* = \gamma R^{-1} P \qquad (19)$$

참고 문헌

[1] R.P.Lippmann,"An introduction to computing with neural nets," IEEE ASSP Mag., vol.4, pp.4-22, Apr. 1987.

[2] J.J.Shynk and S.Roy,"Convergence properties and stationary points of a perceptron learning algorithm," Proc. IEEE, vol.78, pp.1599-1606, Oct. 1990.

[3] B.Widrow, R.G.Winter, and R.A.Baxter,"Layered neural nets for pattern recognition," IEEE Trans. Acoust. Speech, Sig. Proc., vol.36, pp.1109-1118, July 1988.