밀도에 기반한 퍼지 서포트 벡터 머신을 이용한 멀티 카데고리에서의 패턴 분류

Density based Fuzzy Support Vector Machines for multicategory Pattern Classification

박종훈, 최병인 ,이정훈
한양대학교 전자전기제어계측공학과

Jong-Hoon Park, Byung-In Choi and Frank Chung-Hoon Rhee
School of Electrical Engineering Computer Science, Hanyang University
E-mail : {jhpark2, bichoi, frhee}@fuzzy.hanyang.ac.kr

ABSTRACT

본 논문은 multiclass 문제에서 기존에 나와 있는 fuzzy support vector mahchines 이 decision boundary 를 설정하는데 있어 모든 훈련 데이터에 대해서 바람직한 decision boundary 를 만들지 못하므로 그러한 경우를 예로 제시한다. 그리고 그에 대한 개선점으로 밀도를 이용해 decision boundary 를 조정하여 기존 FSVM 의 decision boundary 보다 더 타당한 decision boundary 를 설정하는 것을 보인다.

Key Words : Fuzzy support vector machines, multiclass, density

1. 서 론

Support vector machines(SVMs) 은 Vapnik 에 의해 제안된 통계적 학습 이론에 기반을 두고 있다. SVM 은 많은 응용에서 전통적인 학습 기구들 보다 높은 수행성을 보여줬고 분류 문제를 해결하는 강력한 수단으로 소개되어 왔다. SVM 은 입력 데이터들을 고차원의 특징 공간으로 사영(mapping)하고, 이 공간에서 두 클래스 사이의 마진이 최대가 되는 optimal separating hyperplane 을 찾는다. 그러나 두 클래스가 아닌 여러개의 클래스로 문제가 확장될 때는 4가지 타입의 방법들이 사용된다[1, 2, 3]. 그 중 pairwise support vector machines 은 k 개의 class problem 을 $k(k-1)/2$ 개의 two-class problem 으로 바꾸어준다. 그러나 이 방법은 분류되지 않는 영역이 생긴다. 그래서 이것을 해결하기 위해 membership functions 을 이용하여 푸는 방법이 제안 되었다[4]. 그러나 membership function 을 이용한 방법도 완벽하지는 않다. 최적 hyperplane 을 결정하는 support vector 가 변하지 않는 한 어떠한 훈련 데이터가 들어와도 두 클래스를 분류하는 optimal separating hyperplane 는 변하지 않는다는 문제점이 있다. 본 논문의 목적은 fuzzy support vector machines 이 가지는 문제점을 해결하기 위해 밀도를 고려해줘 decision boundary 를 조정하고자 한다.

본 논문의 나머지는 다음과 같이 구성되어 있다. 2절에서는 SVM 의 이론을 설명하고 3절에서는 membership function을 이용한 FSVM 을 설명한다. 그리고 4절에선 밀도를 이용한 decision boundary 의 조정을 설명하고 5절에선 실험을 통한 성능을 보여주고 6절에서는 결론을 짓는다.

2. Support Vector Machines

선형적으로 분리 가능한 두 데이터 집합에 대하여, SVM 은 그 평면에서 가장 가까운 support vector 들 간의 거리(margin)가 최대이고 두 데이터 집합을 분할하는 최대 마진 다차원 평면을 찾아 두 데이터 집합을 최적으로 분할한다. 주어진 l 개의 패턴 쌍 $(\mathbf{x}_i, y_i)$ 들에 대하여, 최대 마진 hyperplane $\langle \mathbf{w} \cdot \mathbf{x}_i \rangle + b$ 는 다음의quadratic programming problem 을 푸는 것에 의해서 얻어질 수 있다.

$$\begin{aligned} &\text{minimize}_{\mathbf{w}, b} && \langle \mathbf{w} \cdot \mathbf{w} \rangle, \\ &\text{subject to} && y_i(\langle \mathbf{w} \cdot \mathbf{x}_i \rangle + b) \geq 1, \\ & && i = 1, \ldots, l, \end{aligned} \tag{2.1}$$

여기서 $\mathbf{w}, b$ 와 $y_i = 1$는 각각 가중치 벡터 (weight vector), 바이어스(bias), 그리고 클래스 라벨을 나타낸다. 식(2.1)은 다음의 목적 함수 $W(\alpha)$ 를 최대화로 단순화할 수 있다.

$$\begin{aligned} \text{maximize} \quad W(\alpha) &= \sum_{i=1}^{l} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{l} y_i y_j \alpha_i \alpha_j \langle \mathbf{x}_i \cdot \mathbf{x}_j \rangle, \\ \text{subject to} \quad & \sum_{i=1}^{l} y_i \alpha_i = 0, \ \alpha_i \geq 0, \\ & i = 1, \ldots, l \end{aligned} \tag{2.2}$$

여기서 α 는 Largrange multiplier 를 나타낸다. 위의 2차 최적화 문제를 푸는 것으로 최적의 가중치 벡터 $\mathbf{w}^*$와 bias b^*를 다음과 같이 얻을 수 있다.

$$\mathbf{w}^* = \sum_{i=1}^{l} y_i \alpha_i \mathbf{x}_i, \tag{2.3}$$

$$b^* = \frac{\max\limits_{y_i=-1}\left(\left\langle \mathbf{w}^* \cdot \mathbf{x}_i \right\rangle\right) + \min\limits_{y_i=1}\left(\left\langle \mathbf{w}^* \cdot \mathbf{x}_i \right\rangle\right)}{2} \tag{2.4}$$

여기서, $\mathbf{w}^*$는 마진이 $1/\|\mathbf{w}^*\|$ 인 maximum margin hyperplane 을 형성한다. 선형 분리가 불가능한 데이터 집합에 대해서는 분리 불가능한 정도를 나타내는 변수 ξ_i(slack variable) 를 위의 최적화 문제에 첨가하는 것이 요구되고 다음과 같이 표현될 수 있다.

$$\xi_i = \max(0, \gamma - y_i(\langle \mathbf{w} \bullet \mathbf{x}_i \rangle + b)) \tag{2.5}$$

식 (2.2)에 ξ_i 를 포함하면 다음과 같은 2차 최적화 문제를 얻을 수 있다.

$$\begin{aligned} \text{maximize} \quad W(\alpha) &= \sum_{i=1}^{l} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{l} y_i y_j \alpha_i \alpha_j \langle \mathbf{x}_i \cdot \mathbf{x}_j \rangle, \\ \text{subject to} \quad & \sum_{i=1}^{l} y_i \alpha_i = 0, \\ & C > \alpha_i \geq 0, \ i = 1, \ldots, l \end{aligned} \tag{2.6}$$

여기서 C는 양의 정수이다. 식(2.6)이 식(2.2)에서 단지 α 에 상한 값 C를 첨가한 것과 같으므로, 식(2.6)의 해답은 식(2.3), (2.4) 와 같다. 그리고 비선형적으로 분리가 불가능한 경우 데이터 공간의 패턴을 높은 차원의 속성공간으로 변환시켜주는 변환함수 $\phi(\bullet)$ 를 이용하여 커널함수 K 를 통해 다음과 같이 2차최적화 문제를 표현할 수 있다.

$$\phi(\mathbf{x}_i) \bullet \phi(\mathbf{x}_j) = K(\mathbf{x}_i \bullet \mathbf{x}_j)$$

$$\begin{aligned} \text{maximize} \quad W(\alpha) &= \sum_{i=1}^{l} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{l} y_i y_j \alpha_i \alpha_j K(\mathbf{x}_i \bullet \mathbf{x}_j), \\ \text{subject to} \quad & \sum_{i=1}^{l} y_i \alpha_i = 0, \\ & C > \alpha_i > 0, \quad i = 1, \ldots, l, \end{aligned} \tag{2.7}$$

3. Fuzzy Support Vector Machines

분류되지 않는 영역을 해결하기 위해 fuzzy membership functions 을 도입한다[4]. 이것은 클래스 i 에 대해서 클래스 j 의 optimal separating hyperplane 의 직각방향으로 membership functions $m_{ij}(\mathbf{x})$ 정의한다. 여기서 $D_{ij}(\mathbf{x}) = \mathbf{w}_{ij}^{t}\mathbf{x} + b_{ij}$ 이고 $D_{ij} = -D_{ji}$이다.

$$m_{ij}(\mathbf{x}) = \begin{cases} 1 & \text{for } D_{ij}(\mathbf{x}) \geq 1, \\ D_{ij}(\mathbf{x}) & \text{otherwise} \end{cases} \tag{3.1}$$

여기서는 음의 membership degree 도 허용해준다. minimum operator 를 사용하여 클래스 i의 membership function 을 정의할 수 있다.

$$m_i(\mathbf{x}) = \min_{j=1,\ldots,k} m_{ij}(\mathbf{x}) \tag{3.2}$$

식(3.2)를 사용하여 data $\mathbf{x}$ 는 식(3.3)에 의해 분류된다.

$$\operatorname*{argmax}_{i=1,\ldots,k} m_i(\mathbf{x}) \tag{3.3}$$

그래서 minimum operator 에 의해 분류되지 않는 영역은 아래 그림처럼 조정될 수 있다.[4]

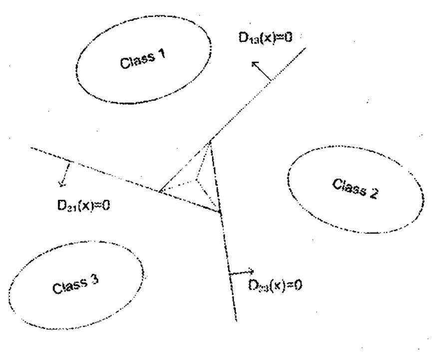


Fig1. Decision boundary with membership functions

4. Density based Fuzzy Support Vector Machines

Fuzzy support vector machines 에선 분류될 수 없는 영역은 두 클래스의 optimal separating hyperplane 에 의해 형성되는데 그것은 결국 서포트 벡터의 위치에 의한 것이라고 말할 수 있다. 그런데 퍼지 서포트 벡터 머신[4]은 이 서포트 벡터의 위치가 변하지 않으면 어떤 입력 데이터가 들어오더라도 두 클래스의 optimal separating hyperplane 은 변하지 않으며 decision boundary 도 변하지 않는다는 문제점이 있다. 이것을 해결하기 위해 본 논문에선 입력 데이터의 밀도분포를 통해 decision boundary 를 조정하려 한다. 각 클래스의 훈련 데이터 주변 밀도를 추정함으로서 분류되지 않는 영역에서 각 클래스의 밀도가 decision doundary 에 영향을 미치도록 하였다. 각 클래스의 훈련 데이터 주변의 밀도추정 방법은 parzen-window 방법[5]을 사용한다.

$\mathbf{x}_n$의 밀도를 추정하기 위해 $\mathbf{x}_n$을 포함하는 영역 R_n을 형성한다. 이때 V_n을 R_n의 부피라고 하고, k_n을 R_n안의 데이터 수라고 한다면 밀도 추정치 $p_n(\mathbf{x})$는 다음과 같은 식을 가진다.

$$p_n(\mathbf{x}) = \frac{k_n/n}{V_n} \tag{4.1}$$

Parzen windows 는 영역 R_n을 고정시키고, 영역안에 존재하는 데이터의 개수 k_n을 계산하여 밀도추정치 $p_n(\mathbf{x})$ 를 얻는 방법이다.

$$p_n(\mathbf{x}) = \frac{1}{n}\sum_{i=1}^{n}\frac{1}{V_n}\varphi\left(\frac{\mathbf{x}-\mathbf{x}_i}{h_n}\right) \tag{4.2}$$

h_n : 다차원 공간의 선분길이

$V_n = h_n^d$: 다차원 공간의 부피

$p_n(\mathbf{x})$는 h_n 에 따라 값에 따라 h_n 이 작으면 밀도분포가 완만해지고 h_n가 커지면 밀도분포가 날카로워진다. 또한 밀도분포를 normalize 시키기 위해 $p_n(\mathbf{x})$ 의 최대값으로 나누어 주어야 한다.

식(4.2)를 이용하여 두 클래스의 optimal separating hyperplane D_{ij}를 새롭게 정의한다.

$$D_{ij}(\mathbf{x}) = \alpha(\mathbf{w}_{ij}^t\mathbf{x} + b_{ij}) + (1-\alpha)(p_{n_i}(\mathbf{x}) - p_{n_j}(\mathbf{x})) \tag{4.3}$$

여기서 α 는 밀도와 마진 사이의 중요도를 주는 파라미터이다. 식(4.3)을 식(3.1)에 대입하여 새로운 $m_{ij}(\mathbf{x})$ 를 얻을 수 있다. 이를 통해 클래스 i 에 대한 minimum operator 로 membership function 을 얻을 수 있다.

$$m_i(\mathbf{x}) = \min_{j=1,\ldots,k} m_{ij}(\mathbf{x}) \tag{4.4}$$

식(4.4)를 이용하여 data $\mathbf{x}$ 는 식(4.5)에 의해 분류된다.

$$\underset{i=1,\ldots,k}{\operatorname{argmax}}\, m_i(\mathbf{x}) \tag{4.5}$$

5. 실험 결과

본 논문에서 제안한 방법의 타당성을 보이기 위하여 기존 FSVM 의 decision boundary 설정이 좋지 못한 예를 위에서 제안한 방법으로 시뮬레이션 한 결과를 보이도록 하겠다.

class 가 3개이고 각 훈련 데이터가 98, 93, 64개씩인 경우를 위에 제안한 알고리즘에 의해 실험하였다. 데이터가 서포트 벡터가 변하지 않는 범위에서 분포하고 있다. 기존방법에 의한 decision boundary 는 아래 그림과 같다.

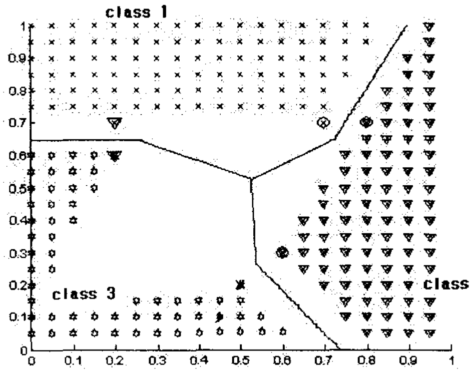
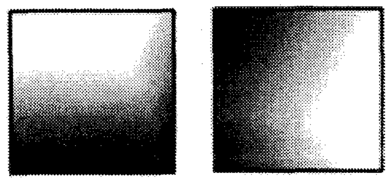
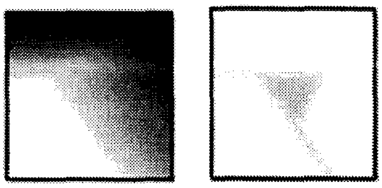


Fig 2. FSVM decision boundary for 3-class

class3 의 데이터 분포는 나머지 class 처럼 고르지 못하고 가운데 부분에 데이터가 존재하지 않는다. 따라서 가운데 그만큼 밀도의 영향을 더 받게 된다. 다음 그림들은 기존의 FSVM 방법, 밀도, 제안한 방법의 min, max operator 적용단계를 gray level 로 표현했다.

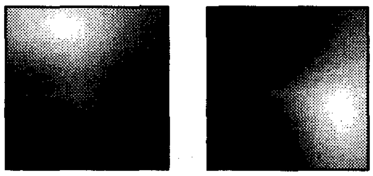


(a) (b)

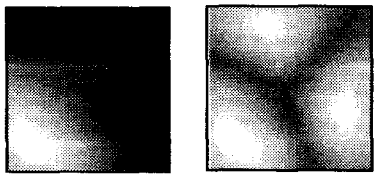


(c) (d)

Fig 3. 기존 FSVM (a) m_1 (b) m_2 (c) m_3 (d) max

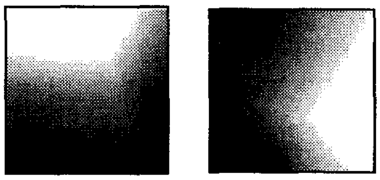


(a) (b)

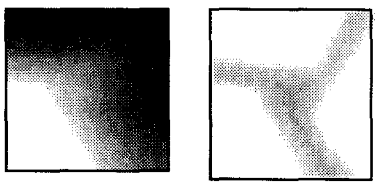


(c) (d)

Fig 4. 밀도 (a) m_1 (b) m_2 (c) m_3 (d) max



(a) (b)



(c) (d)

Fig 5. FSVM+밀도 (a) m1 (b) m2 (c) m3 (d) m4

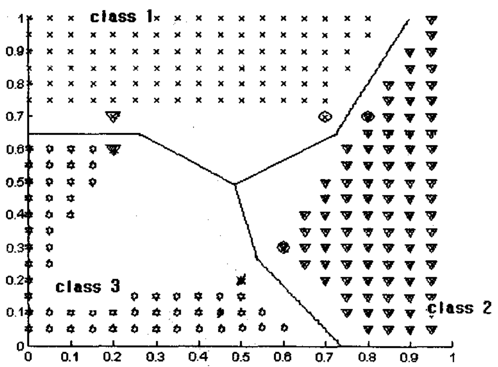


Fig 6. FSVM+밀도 를 이용한 decision boundary

이 실험은 윈도우 시이즈를 0.2 로 하였고 α 값을 0.3 으로 하여 D_{ij} 가 밀도에 더 영향을 받도록 하였다. 윈도우 사이즈가 작게 되면 분류되지 않는 영역에 데이터가 존재하지 않기 때문에 밀도의 영향이 줄어들어 기존 FSVM 의 decision boundary 와 똑같은 결과를 가져온다. α 값의 변화를 주었을 decision boundary 변화가 심하였다. 이는 밀도와 마진 중 어떤 것에 더 중요도를 두느냐에 따라 decision boundary 가 변화되는 것으로 α 값이 작으면 밀도의 영향을 많이 받아 decision boundary 가 많이 변화되고 α 가 크면 밀도의 영향을 덜 받아 기존의 FSVM decision boundary 와 유사하게 형성된다.

6. 결 론

본 논문이 제안한 알고리즘은 기존 FSVM 이 서포트 벡터가 변하지 않는 이상 어떤 훈련 데이터에 대해서도 decision boundary 가 똑같은 문제가 있어 decision boundary 를 형성하는 separating hyperplane 에 parzen-window 를 적용하여 각 클래스의 훈련 데이터의 밀도 분포가 decision boundary 에 영향을 미치게 하였다. 데이터의 밀도를 통해 decision boundary 를 조정하여 훈련데이터에 대한 더 정확한 decision boundary 를 형성하였다.

감사의 글 : 본 연구는 한국과학기술원 영상정보특화센터를 통한 국방과학연구소의 연구비 지원으로 수행되었습니다.

참 고 문 헌

[1] Vapnik, V. (1998). *Statistical learning theory*. New York: Wiley

[2] Kreßel, U. H.-G (1999). Parirwise classification and support vector machines. In B. Scholkopf, C. J. C. Burges, & A. J. smola (Eds.), *Advances in kenel methods: Support vector learning* (pp. 255-268). Cambridge, MA: MIT Press.

[3] Dietterich, T.G., &Bakiri, G. (1995). Solving multiclass learning problems via error-correcting output codes. *Journal of Artificial Intelligence Research*, 2, 263-286

[4] Daisuke Tsujinishi, Shigeo Abe (2003). Fuzzy least squares support vector machines for multiclass problems. *Neural Networks*, 16. 785-792.

[5] Richard O. Duda, Peter E. Hart, David G. Stork (2000) *pattern classification.*: Wiley