13

기계학습과 통계학

서울대학교 ▌김 용 대

1. 서 론

통계학적 방법론에서 가장 기본이 되는 것은 탐색적 자료분석이다. 이는 분석자가 자료를 시각화하고 정리하는 과정이며, 이를 통해서 발견된 유용한 정보를 통계추론을 통해서 검증하는 일련의 과정을 거친다. 그러나 20세기말부터 시작된 컴퓨터 혁명으로 인하여 이러한 전통적 통계학방법론이 적용될 수 없는 새로운 자료들이 생성되기 시작하였다. **특히 자료의 수와 변수의 수의 급격한 증가로 대변되는 거대자료의 출현은 탐색적 자료분석의 효율성을 크게 떨어뜨렸으며, 거대 자료에 대한 통계학적 방법론의 새로운 패러다임을 요구하게 되었다. 이러한 맥락에서 통계학자들이 기계학습 방법론에 대하여 관심을 갖기 시작하였다.** 특히 관심 있는 변수의 증가는 "다차원 저주"라는 현상을 만들었으며, 이로 인하여 전통적인 통계학 방법론들은 새로운 유형의 자료(다차원 거대자료)의 분석에 사용될 수가 없게 되었다.

기계학습 분야는 그 시작이 인간의 참여를 최소화하는 방법론의 개발이었으며, 다차원 거대자료의 출현과 더불어 다양한 종류의 알고리즘들이 제안되었다. 하지만 자료가 커지면서 이를 분석하는 기계학습 알고리즘들의 복잡성도 같이 빠른 속도로 증가하고 있다. 그리고 이러한 알고리즘의 복잡성의 증가는 알고리즘의 원리에 대한 이해를 어렵게 하고 있으며, 새로운 유형의 자료에 적합한 새로운 알고리즘의 개발을 어렵게 하고 있다. 이러한 맥락에서 **기계학습 분야에서는 여러 가지 알고리즘들의 원리를 이해할 수 있는 새로운 인식방법을 필요로 하며, 통계학적 사고는 매우 중요한 사고의 도구로 사용되고 있다.**

이러한 학문적 조류에서 기계학습과 통계학의 만남은 자연스러운 것이며, 각자의 분야에서 수십 년간 쌓아온 지식을 서로 교류하면서 시너지 효과를 극대화하고 있다. **본 논문에서는 최근에 목격되고 있는 기계학습 방법론과 통계학적 방법론의 결합과 이를 통한 새로운 이론 및 방법론의 창출에 대하여 살펴보고자 한다.**

기계학습 분야 중에서 통계학이 가장 활발하게 적용되고 연구되고 있는 분야는 지도학습 분야(supervised learning)이다. 지도학습 분야란 자료가 입력과 출력으로 구성되어 있으며, 자료분석의 목적은 입력을 이용하여 출력을 예측할 수 있는 예측모형의 구축에 있다. 지도학습 방법론은 구글(google)과 같은 검색엔진, 의학진단(medical diagnostic), 바이오인포메틱스(bioinformatics) 그리고 자연어 언어 처리(natural language processing), 부정 신용카드 적발 문제, 주식시장 분석, 음성과 숫자인식(speech and handwriting recognition), 자동화 사물인식(object recognition in computer vision)와 로봇운동(robot locomotion) 등 매우 다양한 분야에서 사용되어지고 있다.

지도학습 방법론 중에서 20세기말에 두 개의 중요한 알고리즘이 개발되는데, 하나는 SVM(Support Vector Machine)이고 다른 하나는 부스팅(boosting)이다. 이 두 방법론은 전산학 전공자에 의해서 개발되었으며, 많은 실증적 연구를 통하여 예측력 측면에서 기존의 기계학습 방법론(의사결정나무, neural network)을 질적으로 향상시켰음이 밝혀졌다. 실증적 연구 이후에 많은 연구자들에 의해서 이 두 개의 알고리즘이 왜 예측력을 급격하게 향상시켰는가에 대한 연구가 시작되었으며, 이 연구에 통계학자들이 많은 기여를 하고 있다. **본 논문에서는 SVM과 부스팅 알고리즘을 통계학적 사고로 이해하는 방법에 대하여 살펴보고, 이를 통하여 어떻게 새로운 알고리즘들이 현재 개발되고 있는지에 대하여 살펴보고자 한다.**

2. SVM과 부스팅

이 절에서는 대표적인 알고리즘인 SVM과 부스팅에 대해 살펴보고자 한다. 먼저 지도학습 방법론에 필요한 용어정리와 함께 두 개의 알고리즘에 대해서 살펴보고자 한다.

2.1 용어 정리

자료의 수가 n인 학습집합(learning set)을 $LS=\{(x_1, y_1), (x_2, y_2), \cdots, (x_n, y_n)\in R^p\times \pm 1\}$로 표기하자. 분류자 $d(x)$는 $R^p\rightarrow \pm 1$인 함수이다. 즉, 입력 변수 $x\in R^p$과 출력변수 $y\in\{-1, 1\}$사이의 관계식을 구성하는 함수이다.

일반적으로 분류자 $d(x)$는 그 특성상 연속함수가 아니고, 따라서 분류자를 직접 구하는 것은 매우 어렵다. 이러한 문제를 해결하는 방법으로는 입력변수 공간 R^p에서 실수로 대응되는 함수 $f(x)$를 추정한 후, 분류자 $\mathrm{sgn}(f(x))$를 구한다. 여기서, 함수 $f(x)$를 편의상 분류함수라 부르기로 한다.

주어진 분류함수의 성능을 측정하는 방법으로는 오분류율을 사용하는데, 오분류율은

$$\Pr(\mathrm{sgn}(f(X))\neq Y) = \mathrm{E}(\mathrm{I}(\mathrm{sgn}(f(X))\neq Y$$

로 정의되면, 0-1 위험율(0-1 risk)이라고도 한다. 여기서 $\mathrm{sgn}(z)$는 만약 $z>0$이면, 1 그리고$z<0$이면 -1로 정의된다.

2.2 SVM

SVM은 처음 Boser, Guyon과 Vapnik에 의해 1992년도 COLT(Computational Learning Theory) 학회에 처음 소개되었다. 이후 SVM은 픽셀(pixel)을 입력변수로 사용하는 숫자인식(handwriting recognition) 문제에서, 예측력의 우수성이 두각을 드러내어 널리 사용되기 시작하였다.

Vapnik의 SVM의 모태는 perceptron 알고리즘으로, 그 역사는 1960년대로 거슬러 올라간다[1][2]. Perceptron 알고리즘은 선형분류함수(즉 $f(x)=b+wx$로 표현)를 반복적으로 추정하는 방법이다. Perceptron 방법의 문제점은 구하여진 분류모형에 대한 이해가 쉽지 않다는 것이다. 또한, 자료가 선형으로 분류가 되지 않는 경우에는 알고리즘이 수렴하지 않을 수도 있다. SVM은 perceptron의 이러한 문제를 해결하였다고 평가를 받는데, 먼저 주어진 자료에서 최적의 분류모형을 정의한 후, 이 최적의 분류모형을 찾는다. 여기서, **SVM의 가장 두드러진 특징은 최적의 분류모형을 정의하는 방법이다. 이를 위하여 마진(margin)이라는 개념이 도입되는데, 마진이란 주어진 자료가 분류경계(decision boundary)에서 떨어진 거리로 정의할 수 있다. SVM은 주어진 자료들의 마진의 최소값을 최대로 하는 분류경계를 최적분류모형으로 정의한다.** 즉, 주어진 분류모형의 성능을 최소마진으로 측정하는 것이다.

SVM의 아이디어를 수식을 나타내면, 다음의 최적화문제를 푸는 것이 된다.

$$\mathrm{minimize}_w \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}\xi_i \qquad (1)$$

subject to $y_i(\langle w, x_i\rangle + \beta_0)\geq 1-\xi_i,\ \xi_i\geq 0$.

여기서 C는 두 항의 상대적인 비를 조절하는 사용자 사전 입력 모수(tuning parameter)이다. Vapnik은 (1)의 문제를 quadratic programing을 이용하여 풀 수 있음을 보였으며, 또한 그 해가

$$f(x) = \langle w, x\rangle + \beta_0 = \sum_{i=1}^{n}\alpha_i y_i \langle x_i, x\rangle + \beta_0 \qquad (2)$$

으로 주어짐을 밝혔다[1]. (2)식에서 주목해야할 사항은 최적분류모형이 입력변수사이의 내적만으로 나타낼 수 있다는 것인데, 이러한 성질을 이용하여 Vapnik은 SVM을 비선형 분류경계의 추정으로 다음과 같이 확장할 수 있음을 보였다[1]. 먼저, 입력변수 x의 비선형 대응 $\phi(x)$를 생각하고 분류모형 $f(x)$가 $\phi(x)$의 선형함수라고 가정한다. 즉, $f(x) = b+w\phi(x)$이다. 이를 (2)식에 대입하면, 최적분류모형은

$$f(x) = \sum_{i=1}^{n}\alpha_i y_i \langle \phi(x_i), \phi(x)\rangle + \beta_0 \qquad (3)$$

로 주어진다. 여기서 중요한 사실은 최적분류모형이 $\langle\phi(x), \phi(y)\rangle$에 만 의존한다는 것이다. 따라서, $\phi(x)$를 정하는 문제는 $\langle\phi(x), \phi(y)\rangle$을 정하는 문제와 같아지게 된다. 여기서, $K(x, y) = \langle\phi(x), \phi(y)\rangle$를 커널이라고 부르며, 많이 쓰이는 커널로는 RBF 커널 $K(x, y) = \exp(-\|x-y\|^2/\tau)$ 등이 있다. 식 (3)에서 알 수 있는 SVM의 중요한 성질 중의 하나는, 분류모형의 모수가 $\phi(x)$의 차원에 의존하지 않고 자료의 수에만 의존한다는 것이다. 이와 같은 이유로, SVM은 다차원저주를 훌륭히 극복한 방법으로 평가 받는다.

2.3 부스팅

부스팅의 기본아이디어는 여러 개의 나쁘지 않은 분류모형을 결합하여 아주 좋은 분류모형을 만드는 것이다. 이를 다음의 유명한 예를 통해 소개한다. 경주마를 통해 수익을 최대화하는 프로그램을 만들기를 원하는 겜블러가 있다고 하자. 그 프로그램은 최근까지의 말 경주들의 결과(경주마들의 우승회수, 경주마의 배당금 등)를 바탕으로 말 경주의 우승자를 매우 정확하게 예측하는 것이 우선일 것이다. 그런 프로그

램을 만들기 위해서는 전문 겜블러의 자문을 필요로 하며, 그 전문 겜블러가 제안하는 최선의 배당 공식(thumb of rule)은 아마도 예를 들면, 최근 가장 많은 경주를 우승한 말 혹은 가장 많은 배당을 받은 말 등이 될 것이다. 그러나 하나의 배당 공식만을 사용한 단일 분류자는 부정확하거나 그 예측의 변동이 큰 단점이 있다. 또한 전문 겜블러가 여러 가지 최선의 공식을 제안한다면, 겜블러 입장에서는 어떻게 이들을 취합하여 최선의 이익을 올리는 규칙을 만들 것인지 무척이나 고민일 것이다. 이러한 상황에서 부스팅은 다소 예측력의 성능은 좋지 않은 규칙일지라도 이들을 결합함으로써 높은 성능을 발휘하는 효과적인 방법을 만들게 한다.

이러한 부스팅 아이디어를 실제 자료분석에 사용할 수 있도록 개량한 알고리즘인 AdaBoost가 Schapire와 Freund에 의해 제안된다[3]. Adaboost 알고리즘을 표 1에 소개하였다. **Adaboost 알고리즘의 기본아이디어는 여러 개의 분류모형을 만들기 위하여 연속적으로 자료의 가중치를 조절한다. 이때, 흥미로운 사실 하나는 가중치를 조절하는 방법인데, 정분류된 자료의 가중치는 줄이고 오분류된 자료의 가중치는 늘리는 방법을 사용하고 있다.**

Adaboost는 원래 학습에러(traning error)을 빨리 줄이는 방법으로 개발된 것이다. 따라서 초창기에는 언뜻 이 알고리즘은 분류함수를 과적합(overfitting)하기 쉬운 단점이 있을 것처럼 보였다. 그러나 이후 부스팅의 우수한 예측력이 여러 문헌들에 의해 실증적으로 보고되어졌다. 실제로 여러 학자들이 우려한 과적합 현상은 잘 일어나지 않으며, 오히려 학습에러가 0이 되어도 예측에러가 계속 감소하는 경향이 보고되었다.

3. Regularized empirical risk minimization

SVM과 부스팅은 그 기본 아이디어나 개발 동기가 완전히 다르다. 하지만 이렇게 보기에는 완전히 다른 두개의 알고리즘들을 통계학에서 오래전부터 연구되어졌던 regularized empirical risk minimization(RERM) 원리를 이용하여 이해할 수 있다.

3.1. General set-up

손실함수 $l(\cdot)$는 R에서 양수로 가는 감소함수이다. 지도분류방법의 목적은 주어진 손실함수 $l(\cdot)$에 대해 목적함수, 모집단의 위험(risk) $E(l(Yf(X)))$를 최소화하는 해를 찾는 것이다.

표 1 Adaboost(Adaptive boosting) 알고리즘

1. 각 자료들에 대한 가중치 $\{w_1^{(1)}, \cdots, w_n^{(1)}\}$을 모두 $1/n$으로 초기화한다.
2. For $m = 1, \cdots, M$
   (a) 자료의 가중치 $\{w_1^{(m)}, \cdots, w_n^{(m)}\}$을 이용하여 분류자 $G_m(x)$를 구함.
   (b) 오차 $\varepsilon_m = \sum_{i=1}^{n} w_i^{(m)} I(G_m(x_i) \neq y_i)$ 계산.
   (c) m번째 분류자 G_m의 중요도 $a_m = \frac{1}{2} ln(\frac{1-\varepsilon_m}{\varepsilon_m})$ 계산.
   (d) $w_i^{(m+1)} \leftarrow w_i^{(m)} \cdot \exp[a_m(2I(y_i \neq G_m(x_i)-1))]$로 갱신.
3. 최종 분류자 $f(x) = \text{sign}(\sum_{m=1}^{M} a_m G_m(x))$

그러나 **(X, Y)에 대한 결합분포를 모르기 때문에 자료로부터 구한 경험위험(empirical risk)을 최소화하는 해를 구하게 된다. 이런 일련의 과정을 ERM 원리(empirical risk minimization principle)이라고 한다.** 즉, 주어진 학습집합 LS에 대해, 경험위험 $\sum_{i=1}^{n} l(y_i f(x_i))$이 최소화하는 해를 찾는 것이다. 그러나 **대개 ERM은 입력변수의 차원 p가 클 때 수치적으로 불안정한 해를 도출한다고 알려져 있다. 또한 과적합의 위험을 안고 있어서 이로부터 구한 해가 예측력의 측면에서 좋지 않을 때가 많다. 이러한 단점을 해결하는 하나의 방안은 목적함수를 regularization함으로써 해에 제약을 가하는 것이다. 즉, 해의 가능한 영역을 줄여주는(shrink) 것이다. 더 구체적으로 regularized ERM (RERM)은 다음의 식으로 표현할 수 있다.**

$$\frac{1}{n}\sum_{i=1}^{n} l(y_i, f(x_i)) + \lambda J(f) \qquad (4)$$

여기서 $J(f)$는 벌점(penalty) 함수이며, λ는 regularization과 관련된 인자로서 함수 f의 복잡도를 제어한다. 예를 들면, f가 선형함수 $f(x) = \beta_0 + \beta^T x$이라면 $J(f) = \|\beta\|_2^2$를 예를 들 수 있겠다. 이 경우, (4)문제는 통계학에서 자주 사용되어지는 릿지회귀모형이 된다[4].

3.2 SVM

2.2절의 식 (1)은 식 (4)에서 손실함수로

$$l_H(z) = \max(1 - z, 0) = (1 - z)_+$$

을, 그리고 벌점함수로 $J(f) = \|\beta\|_2^2$ 사용하는 RERM 방법과 동일함을 알 수 있다[5]. 이때, $l_H(z)$를 경첩 손실함수(hinge loss function)라고 부른다.

비선형분류모형을 추정하는 식 (3)은 reproducing

kernel Hilbert space(RKHS)이론을 이용하여 설명할 수 있다[6]. 주어진 커널 $K(x, y)$로부터 생성되는 RKHS공간을 H_K라 하고, 다음의 RERM최적화 문제를 생각하자.

$$\sum_{i=1}^{n} l_H(yf(x_i)) + \lambda \|f\|_{H_K}^2 \qquad (5)$$

Kimeldorf와 Wahba에 의하여 식 (5)를 최소로 하는 f는 $f(x) = \sum_{i=1}^{n} \alpha_i K(x_i, x) l_H(z)$로 표현된다는 것이 증명되었는데, 이는 식 (3)과 같음을 쉽게 알 수 있다[7]. 즉, SVM은 주어진 커널에 대하여, 이에 상응하는 RKHS에서 RERM을 이용한 분류모형을 추정하는 문제로 이해할 수 있다.

3.3 부스팅

부스팅의 Adaboost 알고리즘은 Friedman을 통해 통계적 관점에서 설명되어졌다[8]. 그는 Adaboost 알고리즘은 손실함수로 기하(exponential) 손실 $l(z) = \exp(-z)$를 사용한 전진적인 단계별 가법 모형(forward stagewise additive model)과 같음을 보였다. 이 알고리즘은 표 2에 있다. 이를 확인하기 위하여 f는 다음과 같은 가법모형으로 두자.

$$f(x) = \beta_m G_m(x), \quad G_m \in \{-1, 1\}$$

이제 매 반복마다 표 2의 (a) 식을 최소화하는 가중치 β_m과 m단계의 분류자 G_m을 찾는다. 그런 다음 $f_m(x) = f_{m-1}(x) + \beta_m G_m(x)$으로 갱신된다. 이를 다음과 같이 재표현할 수가 있는데, $w_i^{(m)} = \exp(-y_i f_{m-1}(x_i))$라고 두면 표 2의 (a)식은 다음과 같다.

$$(\beta_m, G_m) = \arg\min_{\beta, G} \sum_{i=1}^{n} w_i^{(m)} \exp[-y_i \beta G(x)]$$

이제, G_m과 β_m을 찾기만 하면 된다. 위 식에서 $\beta > 0$인 상황에서는 경험위험을 최소화하는 G는 다음의 식이다.

$$G_m = \arg\min_G \sum_{i=1}^{n} w_i^{(m)} I(y_i \neq G(x_i))$$

표 2 전진적 단계별 가법 모형

1. $f_0(x) = 0$로 초기화한다.
2. For $m = 1, \cdots, M$
   (a) $(b_m, G_m) = \arg\min_{b, G} \sum_{i=1}^{n} l[y_i(f_{m-1}(x_i) + bG(x))]$
   (b) $f_m(x) = f_{m-1}(x) + b_m G_m(x)$
3. $f(x) = \mathrm{sgn}\left(\sum_{m=1}^{M} a_m G_m(x)\right)$

이제 β를 구하기 위해서는 다음의 관계식을 살펴볼 필요가 있다.

$$\begin{aligned}
&\sum_{i=1}^{n} w_i^{(m)} \exp[-y_i \beta G(x)] \\
&= \exp(-\beta) \sum_{y_i = G(x_i)} w_i^{(m)} + \exp(\beta) \sum_{y_i \neq G(x_i)} w_i^{(m)} \\
&= (\exp(\beta) - \exp(-\beta)) \sum_{i=1}^{n} w_i^{(m)} I(y_i \neq G(x_i)) \\
&\quad + \exp(-\beta) \sum_{i=1}^{n} w_i^{(m)}.
\end{aligned}$$

또한, $\varepsilon_m = \sum_{i=1}^{n} w_i^{(m)} I(G_m(x_i) \neq y_i)$이라 놓으면 G_m이 주어진 상황에서 표 2의 (a) 식을 최소화하는 β는 아래와 같이 성립한다.

$$\beta = 1/2 \ln[(1-\varepsilon_m)/\varepsilon_m]$$

이제 $f_m(x) = f_{m-1}(x) + b_m G_m(x)$과 $-y_i G_m(x_i) = 2I(y_i \neq G_m(x_i)) - 1$ 관계식을 이용하면 새로 갱신되는 가중치는

$$w_i^{(m+1)} = w_i^{(m)} \exp[\beta_m (2I(y_i \neq G_m(x_i)) - 1]$$

을 만족한다. 그러므로 표 1에서 제시한 Adaboost 알고리즘의 가중치 갱신공식과 일치한다.

AdaBoost의 과적합 문제를 해결하기 위하여 regularized 부스팅 알고리즘들이 개발되었다. 그 중에서 가장 널리 사용되는 것이 Mason 등에 의해서 개발된 L_1부스팅인데, Adaboost와 매우 유사하나 regularization을 위하여 $\sum_{k=1}^{M} |\beta_k| < \lambda$라는 제약조건하에서 점진적 가법모형알고리즘을 적용한다[9].

4. Extension: Choice of loss functions

RERM을 통한 SVM과 부스팅의 이해는 이 두가지의 알고리즘을 다양하게 확장할 수 있는 길을 제공한다. 그 중 한 가지가 손실함수를 바꿔서 새로운 알고리즘을 만드는 것이다.

먼저, 본래 분류문제로 돌아가자. 분류문제에서의 0-1 손실함수는

$$l(yf(x)) = I(y \neq \mathrm{sgn}(f(x)))$$

로 정의된다. 그러나 RERM 문제에서 실제적으로 0-1 손실함수를 이용하여 해를 구하기가 매우 힘들어 계산학적으로 NP 문제가 된다. 이 문제를 극복하는 방안으로 0-1 손실함수보다 큰 볼록한 대리 손실함수(convex surrogated loss function) l을 생각하고, RERM 최소화하는 문제는 0-1손실함수 대신에 다른

l을 사용하여 해를 구하게 된다. 이를 위하여, SVM에서는 경첩손실함수를, 부스팅에서는 기하손실함수를 사용하였다.

먼저, SVM에서 사용된 경첩손실함수를 살펴보자. Lin에 의해서 0-1손실함수보다 큰 볼록한 손실함수 중 0-1손실함수에 가장 가까운 손실함수가 경첩손실함수임이 밝혀졌다[10]. 이러한 사실로부터 SVM의 우수한 예측력을 설명하려는 시도가 Zhang, Bartlett 등에 의해서 많은 연구가 되고 있다[11][12].

경첩함수보다 0-1손실함수에 더 근접한 손실함수를 생각할 수 있으며, Ψ-학습방법(Ψ-learning)은 Shen 등 에 의해 제안되어졌다[13]. Ψ-학습방법은 손실함수로써 다음의 $l_\Psi(\cdot)$를 고려하였다.

$$l_\Psi(z) = (1 - z)_+ - (-z)_+ .$$

이 손실함수는 경첩손실함수에 비하여 0-1 손실함수에 더 가깝지만, 볼록함수가 아닌 단점이 있다. 그러나 경첩함수보다 더 촘촘한 성질로 인하여 해가 더 빨리 수렴하며, 경첩손실함수를 쓰는 SVM에 비해 이상점(outlier)의 영향력을 극소화할 수 있게 되었다. 그러나 경첩손실함수처럼 손실함수가 볼록함수이면 대역적(global) 해를 보장받을 수 있지만, 볼록함수가 아닌 경우에는 국소적(local) 최소해가 되는 단점이 있다. 따라서 앞으로의 연구과제는 비볼록함수에 대해서 최적화방법에 관한 기계 학습론적 연구가 필요하다.

부스팅에 사용된 기하손실함수 대신에 Friedman (2001)은 음의 이항분포 로그가능도함수(negative binomial log likelihood)를 이용하였다. 즉, $f(x) = \log[p(x)/(1-p(x)]/2$ 로짓모형에 대해,

$$l(yf) = \log(1+\exp(-2yf)) \qquad (6)$$

을 이용하여 확률추정이 가능한 모형을 제시하였다. 여기서, $p(x) = \Pr(Y=1|x)$이다. 로짓손실함수 (6)은 기하손실함수에 비하여 0-1손실함수에 가까우며, 실증적으로 기하손실함수에 비하여 우수한 예측력을 제공한다는 것이 입증되었다.

5. 결 론

이제까지 기계학습분야에서 제안되어 대표적인 분류방법으로 자리매김한 SVM과 부스팅의 간단한 소개와 통계학 관점에서 이 방법들에 대한 학자들의 해석을 덧붙였다.

본 논문에서 다루지 않은 여러 연구 분야로는 다양한 종류의 regularization 방법의 개발, 서로 다른 손실함수들 사이의 장단점에 대한 연구, 비볼록 최적화 문제, 다범주 분류문제에서의 좋은 손실함수의 개발 등, 아직 풀리지 않은 문제들이 산적해 있으며, 수학적 이론과 프로그램 능력이 뛰어난 젊은 과학자들을 기다리고 있다.

참고문헌

[1] Vapnik, V., Statistical learning theory, Wiley, New York, 1998.

[2] Rosenblatt, F., "The Perceptron: A probabilistic model for information storage and organization in the brain", Psychological Review, Vol. 65, 386-408, 1958.

[3] Freund, Y and Schapire, R., "Experiments with a new boosting algorithm", In Machine Learning: Proceedings of the Thirteenth International Conference., pp.148-156, 1996.

[4] Hoerl, A.E. and Kennard, R. Ridge regression: Biased estimation for nonorthogonal problems. Technometrics, Vol. 12, 55-67, 1970.

[5] Wahba, G., "Support vector machines, reproducing kernel Hilbert spaces and the randomized GACV", in Schoelkopf, B., Burges, C. and Smola, A. eds, 'Advances in Kernel Methods Support Vector Learning', MIT Press, pp. 69-88, 1999.

[6] Wahba, G., Spline Models for Observational Data, Society for Industrial and Applied Mathematics, 1990.

[7] Kimeldorf, G.S. and Wahba, G. Some results on Tchebycheffian spline functions. Journal of Mathematical Analysis and Applications, Vol. 33, 82-95, 1971.

[8] Friedman, "Greedy Function Approximation: A Gradient Boosting Machine", The Annals of Statistics, Vol. 29, No. 5, pp. 1189-1232, 2001.

[9] Mason, L., Baxter, J., Bartlett, P. L. and Frean, M. Functional gradient techniques for combining hypotheses. In A. J. Smola, P. L. Bartlett, B. Scholkopf, and D. Schuurmans, editors, Advances in Large Margin Classifiers, 221-246. MIT press, Cambridge, 2000.

[10] Lin, Y., "Support vector machines and the

bayes rule in classification", Data Mining and Knowledge Discovery, Vol. 6, No. 3 , pp.259-275, 2002.

[11] Zhang, T., "Statistical behavior and consistency of classification methods based on convex risk minimization" (with discussion), The Annals of Statistics, Vol. 32, pp.56-85, 2004

[12] Bartlett, P.L., Jordan, M. I. and McAuliffe, J.D., "Convexity, classification, and risk bounds", Journal of the American Statistical Association, Vol. 101, No. 473, pp.138-156, 2006.

[13] Shen, X., Tseng, G.C., Zhang, X. and Wong, W.H., "On Y-learning", Journal of the American Statistical Association, Vol. 98, No. 463, pp.724-734, 2003.

김 용 대

1991. 2 서울대학교 계산통계학(학사)
1993. 2 서울대학교 계산통계학(석사)
1997.12 미국 오하이오 주립대학교 통계학(박사)
1997.10～1999. 2 미국 보건성(NIH) 연구원
1999. 2～2001. 8 한국외국어대학교 정보통계학과 조교수
2001. 9～2004. 2 이화여자대학교 통계학과 조교수
2004. 3～현재 서울대학교 통계학과 부교수
관심분야 : Classification, Bayesian method, Model selection, Bioinformatics
E-mail : ydkim903@snu.ac.kr