

## 데이터 마이닝에서 배깅, 부스팅, SVM 분류 알고리즘 비교 분석\*

이영섭<sup>1)</sup> 오현정<sup>2)</sup> 김미경<sup>3)</sup>

### 요약

데이터 마이닝에서 데이터를 효율적으로 분류하고자 할 때 많이 사용하고 있는 알고리즘을 실제 자료에 적용시켜 분류성능을 비교하였다. 분류자 생성기법으로는 의사결정나무기법 중의 하나인 CART, 배깅과 부스팅 알고리즘을 CART 모형에 결합한 분류자, 그리고 SVM 분류자를 비교하였다. CART는 결과 해석이 쉬운 장점을 가지고 있지만 데이터에 따라 생성된 분류자가 다양하여 불안정하다는 단점을 가지고 있다. 따라서 이러한 CART의 단점을 보완한 배깅 또는 부스팅 알고리즘과의 결합을 통해 분류자를 생성하고 그 성능에 대해 평가하였다. 또한 최근 들어 분류성능을 인정받고 있는 SVM의 분류성능과도 비교?평가하였다. 각 기법에 의한 분류 결과를 가지고 의사결정나무를 형성하여 자료가 가지는 데이터의 특성에 따른 분류 성능을 알아보았다. 그 결과 데이터의 결측치가 없고 관측값의 수가 적은 경우는 SVM의 분류성능이 뛰어남을 알 수 있고, 관측값의 수가 많을 때에는 부스팅 알고리즘의 분류성능이 뛰어났으며, 데이터의 결측치가 존재하는 경우는 배깅의 분류성능이 뛰어남을 알 수 있었다.

주요용어: 데이터마이닝, 의사결정나무, 배깅, 부스팅, 알고리즘 비교, CART, SVM .

### 1. 서론

지난 수십 년 간 데이터의 양은 기하급수적으로 증가하고 있으며 이로 인해 우리가 원하는 정보를 찾아내는 일은 더욱 어려워지고 있다. 데이터마이닝 기법은 이러한 상황에서 다양한 분석 방법을 제시하여 데이터에 적합한 모델을 설정하여 원하는 정보를 얻어내는데 도움을 준다.

데이터마이닝 기법을 이용하여 분류문제를 해결하고자 할 때 중요한 문제는 주어진 데이터를 이용하여 목표변수를 가장 잘 예측할 수 있는 분류자를 형성하는 것이다. 실제로 데이터 마이닝에서는 다양한 분류기법을 제시하고 있으며, 우리는 이러한 분류기법을 이용하여 서로 다른 특징을 가지는 분류그룹의 특성을 효과적으로 파악 할 수 있다. 그러나 여

\* 본 연구는 한국과학재단 목적기초연구 (과제번호: R01-2004-000-10689-0) 지원으로 수행되었음.

1) (교신저자) (100-715) 서울시 중구 펜동 3가26, 동국대학교 통계학과 교수

E-mail : yung@dongguk.edu

2) (130-071) 서울시 동대문구 용두동 한신빌딩 4층 DNI컨설팅, 연구원

E-mail : ulgi@dni.co.kr

3) (100-715) 서울시 중구 펜동 3가26, 동국대학교 통계학과 대학원

E-mail : kmk@dongguk.edu

러 가지 분류 기법 중 데이터의 특성에 맞는 효과적인 분류방법을 결정하기 위해서 많은 시행착오를 겪어야 한다. 더욱이 주어진 데이터 특성에 맞는 효율적인 분류방법을 선택하지 못할 경우 많은 시간과 노력을 낭비하게 되며 얻어진 결과 또한 신뢰성을 잃게 된다.

따라서 본 연구에서는 데이터 마이닝에서 사용되고 있는 여러 가지 분류 알고리즘을 사용하여 데이터의 특성이나 패턴에 맞는 효과적인 분류 알고리즘을 제시하고자 한다. 이렇게 함으로써 모델 형성 과정에서 걸리는 시간과 노력을 최소화 할 수 있을 것이다. 데이터 마이닝에서 제 공하는 분류 알고리즘 중 배깅과 부스팅 알고리즘에 대한 평가는 이미 이영섭 등(2003)에 의해서 실시되었다. 비교적 결과해석이 쉬워 데이터마이닝의 분류문제를 해결하기 위해 많이 사용하는 CART는 데이터에 따라 생성된 분류자가 다양하여 불안정하다는 단점을 가지고 있다. 이러한 단점은 재표본 기법에 의한 배깅 또는 부스팅 알고리즘과의 결합을 통해 그 성능이 개선된다. 그러나 재표본 기법에 의존한 알고리즘은 데이터의 속성에 따라 분류자를 생성하는데 많은 시간이 걸린다. 또한 이상치나 특이값 등이 존재하여 데이터가 잘 정제되지 않은 경우 지나치게 그 데이터에만 의존하여 엉뚱한 결과를 생성할 수 있다는 단점을 가지고 있다. 따라서 본 연구에서는 최근 들어 뛰어난 분류성능으로 평가받고 있는 SVM(support vector machine)알고리즘을 배깅과 부스팅 알고리즘을 CART에 결합시켜 얻은 분류자와 비교해 보고자 한다. 알고리즘에 대한 보다 자세한 설명은 참고 문헌을 참조하기로 하며 여기에서는 간략하게 설명하고자 한다.

## 2. 의사결정나무

의사결정나무는 의사결정규칙(decision rule)을 나무구조로 도표화하여 분류(classification)와 예측(prediction)을 수행하는 분석방법이다. 이 방법은 분류 또는 예측의 과정이 나무구조에 의한 추론규칙에 의해서 표현되기 때문에 다른 방법들에 비해서 연구자가 그 과정을 쉽게 이해하고 설명 할 수 있다는 장점을 가진다. 분류 의사결정 나무의 알고리즘은 분리 기준에 따라 Kass(1980)가 제안한 CHAID(chi-squared automatic interaction detection), Breiman 등(1984)이 제안한 CART(classification and regression trees), Quinlan(1993)이 제안한 C4.5가 있다.

본 연구에서는 여러 분류 기법들의 성능을 비교하기 위해 기존의 의사결정나무 알고리즘 중에서 가장 널리 쓰이는 방법인 CART 알고리즘을 사용하였다. CART는 의사결정나무를 형성할 때 이진분리를 하며, 분리할 때 지니지수를 가장 감소시키는 입력변수를 찾고, 입력변수의 분리 기준점을 정한다. 선택된 입력 변수와 그것의 분리 기준점에 따라 생성된 노드들에 다시 같은 방법으로 반복함으로써 나무 구조를 형성해 나간다. 이렇게 형성된 완전모형에서 과대적합(overfitting)을 방지하고 부적절한 추론규칙을 가지고 있는 노드를 가지치기(pruning)하여 최적의 의사결정나무를 얻는다. CART에 대한 보다 자세한 설명은 Breiman 등(1984)을 참조하면 된다.

CART는 결과 해석이 쉬운 장점을 가지고 있지만 데이터에 따라서 생성된 분류자가 다양하여 불안정하다는 단점을 가지고 있다. 이러한 CART의 단점을 보완하여 보다 안정된 의사결정나무 모형을 구축하고자 재표본 방법을 결합한 배깅 또는 부스팅 알고리즘이 개

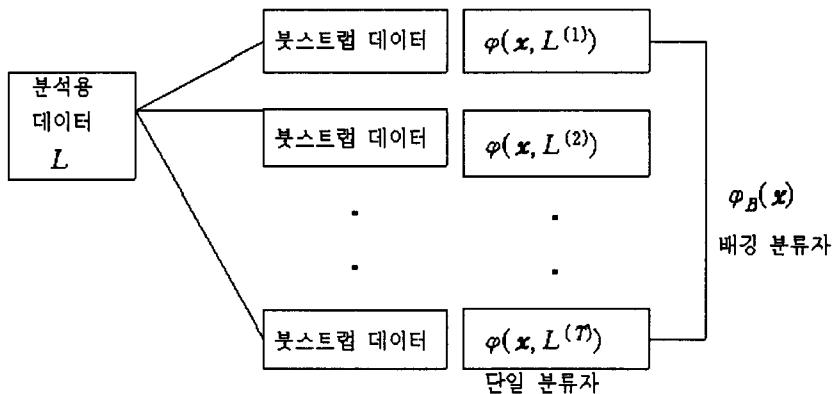


그림 3.1: 배깅 알고리즘

발되었다.

### 3. 배깅 분류자

데이터가 조금이라도 바뀐 상태에서 분류자의 변동성이 큰 경우에는 예측자의 변동성을 감소시키고자 붓스트랩(bootstrap) 방법을 통해 분류자를 얻을 수 있다. 이러한 방법을 배깅(bagging) 알고리즘이라 하며, 배깅은 붓스트랩 방법(Efron과 Tibshirani, 1993)을 이용한 양상을 기법으로 Breiman(1996)에 의해 소개되었다.

배깅 알고리즘 과정은 그림 3.1에 나와 있는 것과 같다. 우선 모집단으로부터 추출된 분석용 데이터(training data set)  $L$ 에서 복원 단순 임의추출에 의해 붓스트랩 분석용 데이터를 생성한다. 이러한 방법을  $T$ 번 반복하여  $T$ 개의 붓스트랩 분석용 데이터를 생성한다. 각각의 붓스트랩 분석용 데이터에 적합한 분류 알고리즘(예를 들면 의사결정나무)를 적용하여 단일분류자  $\varphi(\mathbf{x}, L^t)$ 를 형성하여  $T$ 개의 단일 분류자 집합  $\varphi(\mathbf{x}, L^t), t = 1, 2, \dots, T$ 을 얻는다. 이러한 단일 분류자를 결합하는 방법에는 목표 변수가 연속형 일 때 평균(average), 범주형일 때는 다중 투표(majority vote)를 사용한다. 이렇게 결합되어 형성된 분류자를 배깅 분류자 ( $\varphi_B(\mathbf{x})$ )라 한다.

Breiman(1996)에 의하면 분석용 데이터가 불안정(unstable)하다면, 배깅 분류자의 결합을 통해서 분류 성능이 향상되어진다. 그러나 분석용 데이터가 안정적(stable)이라면, 배깅 과정을 통해서 얻어진 배깅 분류자는 분석용 데이터에서 얻어진 단일 분류자와 비슷하다.

### 4. 부스팅 분류자

붓스트랩 재표본 방법은 각각의 붓스트랩 표본이 서로 독립적이기 때문에 생성된 분류자끼리의 연관성이 없다. 그러나 일반적으로 분류 문제는 잘못 분류된 개체에 더 관심을 가

$$\text{최종 분류자} \quad H(x) = \text{sign}\left[ \sum_{t=1}^T \alpha_t h_t(x) \right]$$

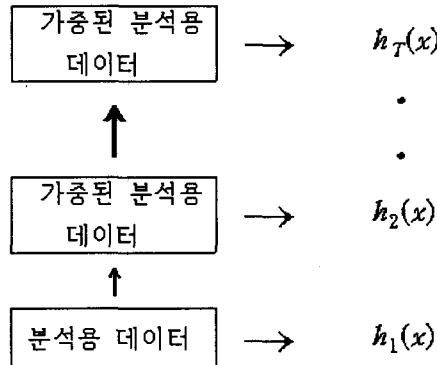


그림 4.1: 부스팅 알고리즘

지고 이들을 더 잘 분류하는 것에 그 목적을 둔다. 이러한 관점에서 개발되어진 재표본 방법이 부스팅(boosting) 알고리즘이다.

부스팅 알고리즘의 생성 배경은 기계학습(machine learning) 연구에서 이루어진다. 기계학습 연구의 이론적인 시초는 Valiant(1984)의 PAC(probably approximately correct) 학습 모델(learning model)이며, Kearns과 Valiant(1994)는 처음으로 약 분류(weak classification) 알고리즘을 제안하였다. Schapire(1990)는 약 분류자들은 부스팅을 통해 강 분류자로 변형 할 수 있음을 보였으며, 그 후에 Freund(1995)는 Schapire 알고리즘의 결점을 보완하여 더 효율적인 부스팅 알고리즘을 제안하였다.

부스팅 알고리즘은 그림 4.1과 같이 간단히 나타낼 수 있다. 부스팅 방법의 초점은 분류자를 순차적으로 생성하고, 분석용 데이터는 이전의 분류자의 수행을 바탕으로 하며 추출된 각 관측값을 사용한다는 것이다. 처음 분석용 데이터 관측값의 가중치는 동일한 상태에서 시작하며, 분류자에 의해 오분류된 관측값은 높은 가중치를 주고, 정분류된 관측값은 반대로 낮은 가중치를 부여함으로써 관측값들의 가중치가 재조정된다. 이러한 과정을 통해 가중치가 재조정된 관측값들에 의해 새로운 가중된 분석용 데이터가 형성될 때, 가중 치가 증가한 관측값이 많이 선택되게 함으로써 분류하기 힘든 관측값을 더 잘 분류하도록 한다. 이때 이전의 분석용 데이터에서 가중치가 재조정되어 가중된 분석용 가중 데이터는 식(4.1)과 같이 나타난다.

$$D_{t+1}(i) = \frac{D_t(i) \exp(-\alpha_t y_i h_t(x_i))}{Z_t} \quad i = 1, 2, \dots, n \quad t = 1, 2, \dots, T \quad (4.1)$$

여기서  $i$ 는 관측치를 나타내며  $t$ 는 반복횟수를 나타낸다.  $D_t(i)$ 는  $t$ 시점에서  $i$ 번째 관측 치에 대한 가중치이며  $\alpha_t$ 는  $t$ 단계에서 형성된 데이터 셋의 가중치이다. 또한  $y_i$ 는 각 개체의

목표 변수를 의미한다.  $Z_t$ 는 정규화 요인(normalization factor)이다. 이러한 과정을  $T$ 회 반복하여 각 단계의 약 분류자와 가중치를 선형 결합한 최종 분류자  $H(x) = \text{sign}[\sum_{t=1}^T \alpha_t h_t(x)]$ 를 와 같이 나타낼 수 있다.

## 5. SVM 분류자

재표본 기법에 의존한 알고리즘이 데이터의 속성에 따라 분류자를 생성하는데 많은 시간이 걸린다. 또한 이상치나 특이값이 존재하여 데이터가 잘 정제되지 않은 경우 지나치게 그 데이터에만 의존하여 엉뚱한 결과를 생성 할 수 있다는 단점을 가지고 있다. 이러한 단점을 보완하기 위해 간단하고 명확한 이론적 근거에 바탕을 둔 SVM(support vector machine) 알고리즘에 대해서 알아보고자 한다. SVM은 기본적으로 두 범주를 갖는 관측값들을 분류하는 방법이다. 이 방법은 Vapnik(1979)에 의하여 발표된 바 있으나 최근에 와서야 그 성능을 인정받게 되었으며, Vapnik(1995)과 Burger(1998)에 의해 잘 소개되고 있다. SVM의 목적은 주어진 많은 데이터들을 가능한 멀리 두 개의 집단으로 분리시키는 최적의 초평면(hyperplane)을 찾는 것이다. SVM 분류자는 다음과 같은 과정을 거쳐 형성된다.

$N$ 개의 객체로 이루어진 표본에서,  $i$ 번째 객체를  $p$ 개의 변수로 이루어진 벡터  $\mathbf{x}_i$ 로 표기하고 이에 대응하는 분류된 범주를  $y_i$ 로 표기하자. 범주  $y_i$ 가 1 또는 -1의 두 가지 범주를 갖는다고 하면 두 범주를 구분해 주는 초평면은 무수히 많이 존재한다. 여기서 두 개의 범주를 완전히 구분해 주는 분리 초평면(separating hyperplane)을  $H : \mathbf{w}'\mathbf{x} + b = 0$ 과 같이 나타내자. 그림 5.1에서 이 초평면을 최적 초평면(optimal hyperplane)이라 한다. 이때 분리 초평면에서부터 가장 가까운 범주 1을 가진 객체를 지나는 초평면과 가장 가까운 범주 -1을 가진 객체를 지나는 초평면 사이의 거리는  $2/\|\mathbf{w}\|$ 이다 (여기서  $\mathbf{w}$ 는 단위길이를 갖는 초평면과 직교하는 벡터이다). 이 거리를 마진(margin)이라 한다. 따라서 선형 SVM은 식(5.1)과 같은 최적화문제를 통해 마진을 최대화시키는  $\mathbf{w}$ 와  $b$ 를 찾는 것이 목적이다.

$$\begin{aligned} & \text{Min}_{\mathbf{w}} \frac{\mathbf{w}'\mathbf{w}}{2} \\ & \text{subject to} \quad y_i(\mathbf{w}'\mathbf{x}_i + b) \geq 1 \end{aligned} \quad (5.1)$$

최적화 문제를 풀기 위해 양의 라그랑지 계수를 도입하여 라그랑지 함수를 유도하여  $\mathbf{w}$ 와  $b$ 를 구할 수 있다. SVM은 입력 데이터가 선형 분리가 불가능한 경우에는 커널함수의 사용에 의해 최적의 초평면을 결정하는 문제를 해결하게 해준다. 이것은 특징공간으로 입력벡터를 투영시킴으로써 내적에 대한 계산만을 필요로 하게 되어 고차원일 경우 계산상의 어려움을 덜 수 있다. 이렇게 사용되는 커널 함수로는 RBF 커널, r차 다항커널, Sigmoidal 커널 등이 사용되고 있다(Cristianini와 Shawe-Taylor, 2000). 이렇게 형성된 SVM 분류자는 최적의 분류 평면을 형성하는데 있어서 경험적인 방법이 아닌 명료한 이론적 근거에 기반하여 알고리즘이 간단함을 알 수 있다. 또한 입력 공간의 비선형적인 높은 차수를 특정 공간에 선형적으로 투영하여 해석 할 수 있도록 하여 학습을 성공적으로 수행하는데 미치는 요소들을 규명 할 수 있게 해준다. 그러나 이러한 SVM은 결측치가 존재하는 경우 결측치 대체에 대한 방법이 미흡하기 때문에 결측치가 많이 발생한 자료에 대해서 정확한 분류자

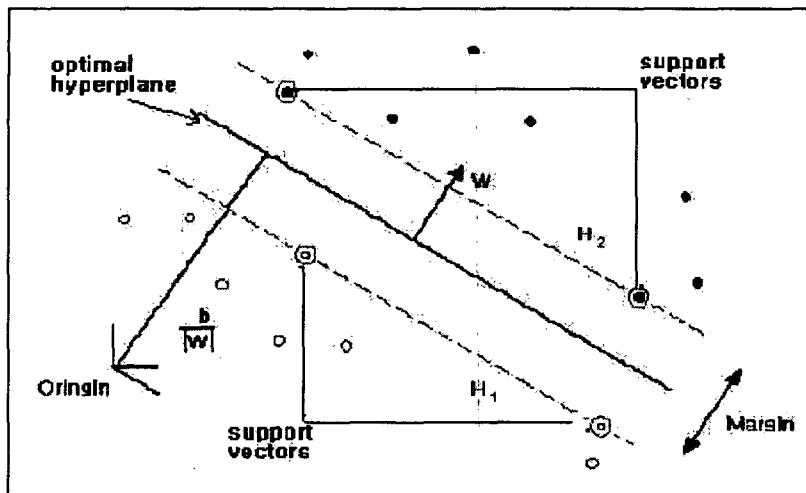


그림 5.1: 최적의 분리 초평면

형성이 어려운 단점이 있다. SVM에 관한 기본적인 개념들은 김현중(2004)을 참고하기 바란다.

그림 5.1은 완전히 두 객체로 오분류 없이 분리 가능한 경우 마진을 최대화하는 최적의 초평면을 찾았을 때 서포트 벡터(support vector)의 위치를 기하학적으로 나타낸 것이다. 각 분리 초평면에 놓이는 객체들이 서포트 벡터이다.

## 6. 실제 데이터를 이용한 알고리즘 비교 및 결과

앞에서 설명한 여러 종류의 분류자는 각기 다른 장·단점을 가지고 있다. 어느 경우에나 항상 우수한 알고리즘은 없으며 데이터의 속성이나 환경에 따라 가장 적합한 분류 알고리즘을 선택하는 것이 필요하다. 따라서 앞에서 설명한 CART, 배깅과 부스팅 알고리즘을 CART에 결합한 분류자와 SVM 알고리즘의 분류 성능을 실제 데이터에 적용하여 데이터 속성에 따른 각 알고리즘의 특성을 비교해보도록 하겠다.

본 연구에서 사용된 데이터는 목표변수가 범주형인 경우만 사용하였고, 각 데이터의 특성에 따라 CART, CART 모형 구축 방법을 이용한 배깅과 부스팅 알고리즘 구축, SVM 알고리즘 중 어느 것이 더 효과적으로 잘 적합 되는지에 대해서 알아보고 그 특성이나 패턴을 찾아보자 한다. 알고리즘의 특성 비교를 위해서 사용된 데이터는 UCI 데이터 저장소에서 추출하였다(Blake 와 Merz (1998)). 각 데이터들 대한 특성값을 관측값의 수, 목표변수의 수, 입력변수의 수, 연속형 변수와 범주형 변수의 수, 결측치의 존재 유무에 따라 표 6.1과 같이 나타내었다. 실험에 사용한 데이터는 총 28개의 실제 데이터이다. 배깅과 부스팅을 이용한 의사 결정나무 형성과정은 SAS프로그램의 Enterprise Miner(SAS/EM)를 사

표 6.1: 데이터의 특성 값

데이터	관 측 값 의 수	목표 변수 수	입력 변수 수	연 속 형 변 수	범 주 형 변 수	결 측 치	데이터	관 측 값 의 수	목표 변수 수	입력 변수 수	연 속 형 변 수	범 주 형 변 수	결 측 치
glass	214	7	9	9	x	x	auto_map	406	5	7	7	x	o
segment	2310	7	19	19	x	x	shuttle	43500	7	9	9	x	x
ionosphere	351	2	34	34	x	x	nursery	10343	4	8	1	7	x
iris	150	3	4	4	x	x	ringnorm	7400	2	20	20	x	x
letter	20000	26	16	16	x	x	twonorm	7400	2	20	20	x	x
austria	690	2	14	6	8	x	sonar	208	2	60	60	x	x
breasts	699	2	9	9	x	o	hepatitis	155	2	19	19	x	o
wine	178	3	13	13	x	x	vechicle	846	4	18	18	x	x
balance-scale	625	3	4	x	4	x	car	1354	4	6	2	4	x
german	1000	2	20	7	13	x	clever	303	2	13	6	7	o
heart	270	2	13	6	7	x	led7	3200	9	7	x	7	x
labor	57	2	16	8	8	o	credit	690	2	15	6	9	o
zoo	101	7	16	1	15	x	vote	435	2	16	x	16	o
bupa	345	2	6	6	x	x	flare	1066	6	12	10	2	x

용하였다. 의사결정 나무의 모든 방법은 SAS/EM에서 제공하는 디플트 옵션을 사용하였다. SVM을 사용하여 두개 이상의 범주를 가진 자료를 분석 할 때 사용되는 방법으로는 일대일(one-against-one)접근법과 일대다(one-against-the other)접근법 두 가지 방법이 있다. Weston과 Watkins(1998) 그리고 Platt 등(2000)에 의해 일대일 접근법이 더 좋은 분류 성능을 가지며 일대다 접근법은 문제가 많은 것으로 지적되고 있다. 따라서 본 논문에서는 SVM의 다중 분류에 있어서 일대일 접근법을 사용하였다. 또한 SVM에서 사용 될 수 있는 여러 가지 커널 함수 중 RBF 커널함수를 사용하여 모형을 형성하였다. 실제로 계산된 서포트 벡터는 커널 함수에 의해 그다지 큰 영향을 받지 않은 것으로 알려져 있다(Saunders,1998).

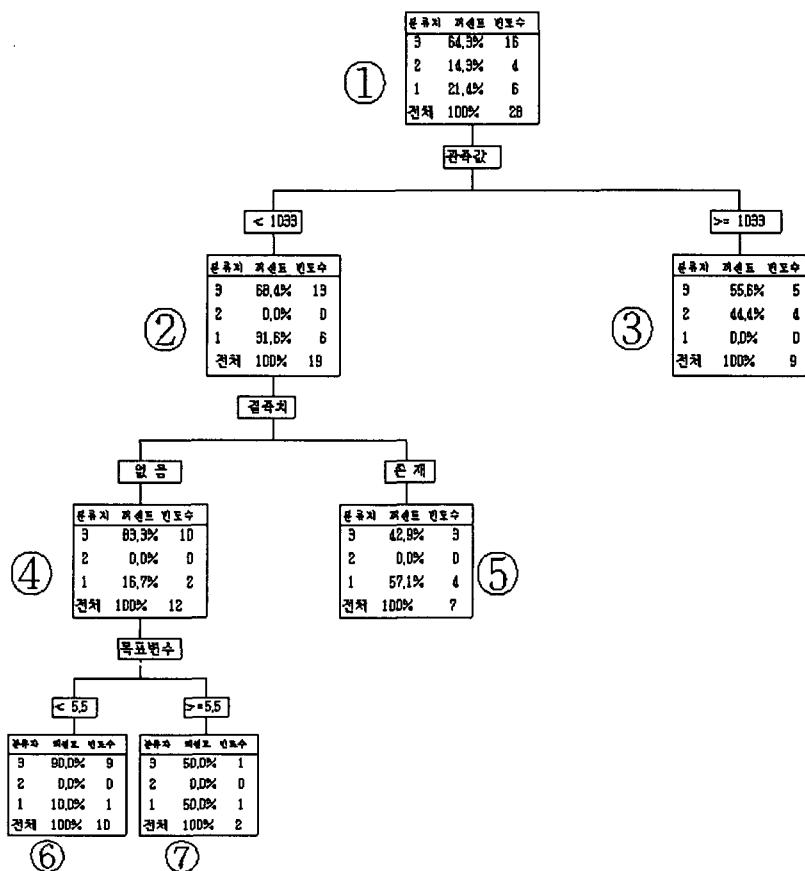
배깅과 부스팅 그리고 SVM의 알고리즘을 비교 분석하기 위한 기준으로는 오분류율을 사용하였다. CART, 배깅, 부스팅 ,SVM 알고리즘의 오분류율은 분석용 데이터로 모형을 형성한 후 평가용(testing) 데이터를 모델에 적용시켜 측정한다. 즉, 모든 데이터에 대해 분석용 데이터를 70%로 평가용 데이터를 30%로 분리한 다음 분석용 데이터로부터 모형을 구축한 후 평가용 데이터에 구축된 모형을 적용시켜 오분류율을 측정하였다. 안정적인 오분류율의 측정을 위해서 두 개의 데이터로 분류 할 때 초기값을 다르게 하여 분석용 데이터와 평가용 데이터로 분리하는 작업을 10번 시행하였고 각 시행마다 모형 구축과 평가를 실시하여 오분류율을 계산하였다. 표 6.2에 나타난 각 데이터들에 대한 오분류율은 각 시행마다 계산된 10개 오분류의 평균이다. Breiman(1996) 및 Optiz와 Maclin(1999)에 의에 배깅과 부스팅의 반복회수는 25번이 될 때까지 오분류율이 개선이 되었지만 그 이후에는 오분류율의 감소에 별 영향을 받지 않는다고 제안되고 있다. 따라서 배깅과 부스팅에 의한 분류

표 6.2: CART, 배깅, 부스팅, SVM의 오분류율 비교

데이터	CART	배깅	부스팅	SVM	데이터	CART	배깅	부스팅	SVM
glass	0.279	<b>0.173</b>	0.200	0.273	townorm	0.194	0.053	0.073	<b>0.021</b>
labor	0.165	<b>0.105</b>	0.183	0.327	iris	0.069	0.064	0.062	<b>0.026</b>
austra	0.173	<b>0.141</b>	0.144	0.335	letter	0.147	0.074	0.039	<b>0.029</b>
credit	0.166	<b>0.136</b>	0.147	0.142	led7	0.273	0.267	0.270	<b>0.265</b>
auto_mpg	0.032	<b>0.030</b>	0.044	0.034	german	0.310	0.251	0.254	<b>0.248</b>
hepa	0.160	<b>0.141</b>	0.217	0.225	wine	0.077	0.052	0.071	<b>0.027</b>
car	0.125	<b>0.098</b>	<b>0.028</b>	0.039	blance-scale	0.197	0.146	0.200	<b>0.102</b>
segment	0.051	0.049	<b>0.010</b>	0.067	breasts	0.059	0.044	0.051	<b>0.035</b>
shuttle	0.003	0.002	<b>0.001</b>	0.002	vechicle	0.302	0.257	0.223	<b>0.160</b>
flare	0.251	0.252	<b>0.237</b>	0.247	heart	0.245	0.200	0.206	<b>0.175</b>
nersery	0.106	0.105	0.037	<b>0.017</b>	sonar	0.295	0.254	0.223	<b>0.176</b>
ringnorm	0.129	0.087	0.123	<b>0.016</b>	zoo	0.127	0.110	0.094	<b>0.043</b>
vote	0.043	0.043	0.043	<b>0.042</b>	clever	0.219	0.164	0.215	<b>0.159</b>
bupa	0.306	0.278	0.276	<b>0.269</b>	ionosphere	0.089	0.081	0.120	<b>0.057</b>

자를 형성하는데 있어서 25번의 반복을 하도록 한다. 표 6.2에서 제시한 28개 데이터의 배깅과 부스팅, SVM의 오분류율을 비교한 결과 18개 데이터에 대해서 SVM의 오분류율이 낮음을 알 수 있었다. 6개의 데이터에 대해서는 배깅의 오분류율이 낮았고 나머지 4개의 데이터에 대해서는 부스팅의 오분류율이 낮음을 알 수 있다. 데이터에 특성에 대한 각 알고리즘의 분류 경향을 살펴보기 위해 각 데이터에서 가장 낮은 오분류율을 가지는 알고리즘을 목표 변수로 정하였다. 그리고 나서 데이터의 특성을 나타내는 목표변수, 관측값의 수와 입력변수에서 범주형 변수와 연속형 변수의 수와 결측치를 독립변수로하여 그림 6.1과 같이 의사결정나무를 형성하였다. 즉, 그림 6.1은 각 자료에서 가장 분류성능이 우수한 알고리즘을 목표변수로 두고 의사결정나무를 형성한 결과이다.

그림 6.1의 의사결정나무의 ④노드와 ⑥노드 자료의 대부분이 SVM 자료로 나타남을 알 수 있다. 따라서 관측자료의 수가 작고 결측치가 존재하지 않으면서 목표변수의 수가 6개 미만인 경우는 SVM 알고리즘의 분류성능이 뛰어남을 알 수 있다. 이러한 사실로부터 SVM 분류자는 결측치에 영향을 많이 받는다는 것을 알 수 있었다. 또한 분류자 생성이 관측 자료들의 내적 계산으로 이루어지므로 관측값의 수가 적은 경우 내적 계산이 적어져 더 유용하게 분류 문제를 해결하고 있음을 알 수 있었다. 그림 6.1을 살펴보면 ①번 노드의 부스팅 자료 총 4개 모두가 ③의 노드로 분류되었음을 알 수 있다. 따라서 관측자료들의 수가 많은 경우는 부스팅 알고리즘의 성능이 뛰어남을 알 수 있다. 부스팅 알고리즘의 경우 잘 분류되지 않는 관찰치들에 초점을 맞추어 분류기능을 향상시키므로 더 복잡한 데이터에서 분류성이 향상되고 있음을 알 수 있다. 또한 ⑤번 노드에서 전체 자료의 57%가 배깅 자료로 나타났으므로 결측치가 존재하는 경우 배깅 알고리즘이 선호되고 있음을 알 수 있다.



\*1: 배깅 분류자 , 2: 부스팅 분류자 , 3: SVM 분류자

그림 6.1: 알고리즘 비교를 위한 의사결정나무

## 7. 결론 및 향후과제

본 논문에서는 데이터 마이닝에서 사용되고 있는 여러 가지 분류기법들을 데이터의 특성에 따른 성능을 비교해보았다. 비교대상은 CART, CART 분류자에 배깅과 부스팅 알고리즘을 결합시켜 생성한 분류자와, SVM 분류자를 사용하였다. 그 결과 총 28개 실제 데이터에 대해서 18개의 데이터에 의해서 SVM의 분류성능이 뛰어남을 알 수 있었고, 나머지 6개, 4개의 데이터에 대해서는 각각 배깅과 부스팅 알고리즘의 분류성능이 뛰어남을 알 수 있었다. 이 결과를 바탕으로 데이터의 특성에 따라 의사결정나무를 만든 결과 관측 자료의 수가 많지 않고 결측치가 존재하지 않는 경우 SVM의 분류성능이 뛰어남을 알 수 있었고, 관측 자료의 수가 많은 경우는 부스팅 알고리즘의 성능이 뛰어남을 알 수 있었다. 결측치가 존재하는 경우는 배깅이 다른 두 개의 알고리즘 보다 분류성능이 향상됨을 알 수 있었다.

따라서 이러한 결론을 바탕으로 데이터의 특성을 파악하고 분류자를 결정하게 된다면 시간과 노력을 줄일 수 있을 뿐만 아니라 더 효과적인 결론을 유도 할 수 있을 것이다.

그러나 28개의 데이터에 대한 결과를 각 분류 알고리즘의 특성을 일반화시키는 것은 다소 무리가 있을 수 있다. 향후에는 더 많은 데이터와 그의 특성들을 이용하여 보다 일반화된 분류 알고리즘의 특성이나 패턴을 조사하는 것이 필요하다. 또한 오분류율이 아니라 ROC 곡선이나 이익 도표 또는 사전확률에 따른 비용함수등을 고려한 오분류율등의 다른 측정 기준으로 각 알고리즘을 비교 분석하는 것을 향후 과제로 남겨두고자 한다.

### 참고문헌

- 김현중(2004). Support Vector Machine의 이론과 응용, 한국통계학회, <추계 학술발표회 논문집>, 1-1.
- 이영섭, 오현정(2003). 데이터 마이닝에서 배깅과 부스팅 알고리즘 비교 분석, 한국통계학회, <춘계 학술발표회 논문집>, 97-102.
- Blake, C.L. and Merz, C.J. (1998). UCI Repository of machine learning databases [<http://www.ics.uci.edu/~mlearn/MLRepository.html>]. Irvine, CA: University of California, Department of Information and Computer Science.
- Breiman, L.(1996). Bagging predictor, *Machine Learning*, **26**, 123-140.
- Breiman, L., Friedman, J. H. and Olshen, R. A. and Stone C. J. (1984). *Classification and Regression Trees*, Chapman and Hall.
- Burger, C. J. C(1998). A tutorial on support vector machines for pattern recognition, *Bell Laboratories, Lucent Technologies*.
- Cristianini, N. and Shawe-Taylor, J.(2000). *Support Vector Machines and Other Kernel-based Learning Methods*, Cambridge University Press.
- Efron, B. and Tibshirani, R. (1993). *An Introduction to the Bootstrap*, Chapman and Hall.
- Freund, Y. (1995). Boosting a weak learning algorithm by majority, *Information and Computation*, **121**, 256-285.
- Kass, G.V. (1980). An exploratory technique for investigating large quantities of categorical data, *Applied Statistics*, 119-127.
- Kearns M. and Valiant. L,G (1994). Cryptographic limitations on learning boolean formulae and finite automata, *Joural of the Association for Computing Machinery*, **41**, 67-95.
- Optiz D. and Maclin R.A.(1999). Popular ensemble methods: An empirical study, *Journal of Artificial Intelligence Research*, **11**, 169-198.
- Platt J. and Cristianini, N. and Shawe-Taylor, J.(2000). Large margin DAGs for multiclass classification, *Advances in Neural Information Processing Systems*, **12**, 547-553.
- Quinlan, J.R. (1993), C4.5, *Programs for Machined Learning*, Morgan Kaufmann, San Mateo.
- Saunders, C. (1998), Support vector machine user manual, RHUL Technical Report.
- Schapire, R. (1990). The strength of weak learnability, *Machine Learning*, **5**, 197-227.
- Weston, J. and Watkins C. (1998). Multi-class support vector machines, *Technical Report CSD-TR-98-04*, Royal Holloway.
- Valiant, L.C. (1984). A theory of the learnable, *Communication of the ACM*, **27**, 1134-1142.

- Vapnik, V. (1979). *Estimation of Dependences Based on Empirical Data*, Nauka. (English translation Springer Verlag, 1982)
- Vapnik, V. (1995). *The Nature of Statistical Learning Theory*, Springer Verlag.

[ 2004년 7월 접수, 2004년 12월 채택 ]

## An Empirical Comparison of Bagging, Boosting and Support Vector Machine Classifiers in Data Mining\*

Yung-Seop Lee<sup>1)</sup> Hyun-Joung Oh<sup>2)</sup> Mee-Kyung Kim<sup>3)</sup>

### ABSTRACT

The goal of this paper is to compare classification performances and to find a better classifier based on the characteristics of data. The compared methods are CART with two ensemble algorithms, bagging or boosting and SVM. In the empirical study of twenty-eight data sets, we found that SVM has smaller error rate than the other methods in most of data sets. When comparing bagging, boosting and SVM based on the characteristics of data, SVM algorithm is suitable to the data with small numbers of observation and no missing values. On the other hand, boosting algorithm is suitable to the data with number of observation and bagging algorithm is suitable to the data with missing values.

*Keywords:* Bagging; Boosting; Data mining; Decision trees; Empirical comparison; CART; SVM

---

\* This work was supported by grant No. R01-2004-000-10689-0 from the Basic Research Program of the Korea Science & Engineering Foundation.

1) (corresponding author) Professor, Dept. of Statistics, Dongguk University, 26, 3-Ga, Phil-dong, Chung-gu, Seoul 100-715, Korea

E-mail: yung@dongguk.edu

2) Researcher, DNI consulting, yougdo-dong, Dongdaemun-gu, Seoul 130-071, Korea  
E-mail: ulgi@dni.co.kr

3) M.S., Dept. of Statistics, Dongguk University, 26, 3-Ga, Phil-dong, Chung-gu, Seoul 100-715, Korea  
E-mail: kmk@dongguk.edu