Journal of the Korean Data & Information Science Society
2014, **25**(6), 1283–1291

http://dx.doi.org/10.7465/jkdi.2014.25.6.1283
한국데이터정보과학회지

전진적 단계 알고리즘을 이용한 대용량 데이터와 순차적 배치 데이터의 분류[†]

윤영주[1]

[1]대전대학교 비즈니스정보통계학과

접수 2014년 8월 28일, 수정 2014년 9월 16일, 게재확정 2014년 10월 2일

요 약

본 논문에서는 대용량이거나 시간에 따라 순차적으로 들어오는 데이터의 분류를 위한 전진적 단계 알고리즘을 제안한다. Adaboost 알고리즘은 노이즈가 있는 데이터에 대하여 성능이 떨어지는 것으로 알려져 있다. 이를 해결하기 위한 한 가지 방법으로 전진적 단계 선형 회귀 방법을 사용한다. 대용량 데이터나 순차적 배치 데이터의 경우에도 이러한 상황을 극복하기 위해 전진적 단계 알고리즘 방법을 적용한 방법을 제안한다. 모의실험과 실제 자료 분석을 통해 제안된 알고리즘이 좋은 성능을 보임을 알 수 있었다.

주요용어: 개념 변화, 대용량 데이터, 순차적 배치 데이터, 앙상블 방법, 전진적 단계 알고리즘.

1. 서론

현대는 빅데이터 (big data) 시대이다. 많은 분야에서 대용량의 데이터가 시시각각 수집되고 있다. 이런 경우 시간이 흐르면 자료에 내재되어 있던 종속변수와 예측변수들 간의 관계들이 변화가 있기 마련이다. 이러한 경우를 개념 변화 (concept drift)라 부른다. 개념 변화는 한꺼번에 일어날 수도 있고 점진적으로 일어날 수도 있다. 이때 개념 변화를 알아채지 못하고 기존의 모형을 그대로 사용할 경우 그 성능은 떨어질 수밖에 없다. 따라서 개념 변화에 빠르게 조정할 수 있는 알고리즘이 필요하게 된다 (Street와 Kim, 2001).

Wang 등 (2003)은 환경 변화를 반영할 수 있는 단일 분류자 (single classifier)보다는 정확도나 효율성, 편이성의 측면에서 앙상블 (ensemble) 방법이 좀 더 좋은 방법이라 주장했다. Kuncheva (2004)는 러닝 시간이 주요 목적이 아니고 정확도가 중요하다면 앙상블 방법이 자연스러운 해결책이 될 것이라 주장했다. SEA (streaming ensemble algorithm; Street와 Kim, 2001) 방법은 배깅 (bagging; Breiman, 1996) 형태의 알고리즘으로 앙상블을 구성할 때 균일한 가중치를 사용하였다. 하지만 Yoon (2010)은 모의실험 등을 통해 SEA 방법은 개념의 변화에 느리게 반응함을 보였으며 이에 Adaboost와 Arc X-4를 이용한 방법을 제안하여 더 좋은 성능이 나타냄을 보였다. Yoon (2010)이 제안한 두 방법은 정확도의 측면에서는 크게 차이를 보이지 않았으므로 본 논문에서는 Adaboost방법을 이용한 방법과 비교하였다.

개념 변화가 없고 자료를 한꺼번에 이용하는 경우 부스팅 (boosting)은 예측의 정확도를 향상시키기 위한 앙상블 방법 중 가장 좋은 방법 중 하나로 알려져 있다. C4.5 (Quinlan, 1993)이나 CART

[†] 이 논문은 2012학년도 대전대학교 신진교수학술연구비에 의해 지원되었음
[1] (300-716) 대전광역시 동구 대학로 62, 대전대학교 비즈니스정보통계학과, 조교수. E-mail: yoonyj@dju.kr

(Breiman 등, 1984)를 이용한 AdaBoost (Freund와 Schapire, 1997)와 Arc-x4 (Breiman, 1998)이 널리 알려져 있는 방법이다. 부스팅 알고리즘은 이전 분류자에 의해 오분류된 개체에 더 집중하여 순차적으로 분류자를 생성하고 그렇게 생성된 "약한 (weak)" 분류자들을 결합하여 "강한 (strong)" 분류자를 만들어 내는데 있다. AdaBoost는 가중치를 부여하여 분류자들을 결합하고 Arc-x4는 동일한 가중치로 분류자들을 결합하는 방법이다. Yoon (2010)에서는 부스팅 방법을 기초로 하여 대용량 데이터나 순차적 데이터를 위한 새로운 앙상블 알고리즘을 소개하였다. 이 알고리즘은 각 시점의 데이터 배치에 대하여, 이전 시점에 만들어진 분류자들로부터 계산된 가중치를 이용하여 그 시점의 분류자를 만들어 내고 이렇게 만들어진 분류자들을 결합하여 새로운 데이터를 예측하는 방법이다. 이 방법은 SEA 방법보다 개념변화에 빠르게 적응하고 정확도 측면에서도 좋은 성능을 보였다 (Yoon, 2010). 그러나 Adaboost 알고리즘은 종속변수에 노이즈가 존재하는 경우 그 성능이 떨어지는 것으로 알려져 있다 (Dietterich, 2000). 개념 변화가 있는 경우 순차적 배치 데이터에서 개념 변화가 일어날 경우 독립변수와 종속변수 관계가 기존 (이전 시점의) 배치와 다르게 되므로 이를 종속변수에 노이즈가 생긴 것으로 생각해 볼 수 있다. 따라서 순차적 배치 데이터에 적용되는 부스팅 알고리즘도 성능이 어느 정도 영향이 있을 것이라 예상해 볼 수 있다. Adaboost의 이러한 단점을 극복하기 위한 한 방법은 정규화 (regularization) 기법이다. 정규화 기법은 LASSO (Tibshirani, 1996)와 같은 방법을 직접 구현하여야 하나 자료가 많거나 분류자가 많은 경우 쉽지 않기 때문에 LASSO를 간단하게 근사할 수 있는 전진적 단계 선형 회귀법 (forward stagewise linear regression)을 이용할 수 있다 (Hastie 등, 2001). 최근에는 Kim 등 (2012)이 커널 능형 회귀 (kernel ridge regression)을 이용하여 전진적 단계 선형 회귀법으로 적합시킨 FSKRF (forward stagewise kernel ridge regression) 기법을 제안하기도 하였다. 정규화 기법을 근사시킬수 있는 전진적 단계 알고리즘을 Yoon (2010)이 제안한 방법에 적용시키면 이러한 단점을 극복할 수 있을 것이라 기대할 수 있다.

본 논문은 다음과 같이 구성되어 있다. 2절에서는 Yoon (2010)에서 제안된 AdaBoost와 Arc-x4 알고리즘을 대용량 데이터나 순차적 데이터에 적용시킨 부스팅 알고리즘을 간략하게 소개한다. 2절에서 소개된 기존 방법의 약점을 극복하기 위한 정규화 기법은 3절에서 제안한다. 4절에서는 제안된 알고리즘과 기존의 부스팅 방법과 비교를 위해 모의실험과 실제 자료를 이용하여 비교한다. 제안된 방법은 기존의 방법보다 좋은 성능을 보임을 알 수 있다. 마지막으로 5절에서는 본 논문의 결과에 대한 요약 및 결론을 서술한다.

2. 부스팅 개념을 이용한 순차적 앙상블 방법

이번 절에서는 Yoon (2010)에서 소개된 순차적 데이터에 이용할 수 있는 부스팅 알고리즘에 기초한 부스팅 앙상블 방법을 간단히 소개하도록 하겠다. 대용량 데이터의 경우 데이터를 T개로 분할하여 적절하게 순서를 정하여 마치 순차적으로 데이터가 들어오는 것처럼 생각하여 알고리즘을 적용할 수 있으므로 생략하도록 한다. t번째 데이터셋을 $\{(x_{t,1}, y_{t,1}), \cdots, (x_{t,n_t}, y_{t,n_t})\}$이라 하자. 여기서 $x_{t,j}$는 예측 변수(predictors)로 이루어진 벡터이며 $y_{t,j}$는 그룹을 나타내는 변수로 1 또는 -1의 값을 갖는다. 이 알고리즘에서 순차적으로 t시점의 개별 분류자 h_t를 만드는데 필요한 시점 데이터에 대한 가중치는, $t-1$시점까지 만들어진 개별 분류자 $h_1, \cdots, h_{t-1}$를 부스팅 알고리즘에서의 개별 분류자처럼 간주하고 이 분류자들을 이용하여 만들며, 이 가중치로 가중 오류 (weighted error)를 최소로 하는 h_t를 만들어 낸다. 데이터를 한꺼번에 이용하는 기존의 부스팅 방법과 차이점은 매시점마다 가중치와 부스팅 앙상블을 만들어 낼 때의 개별 분류자 h_t들에 대한 계수가 달라진다는 것이다. 즉 각 시점에서의 부스팅 앙상블이 기존에 만들어진 개별 분류자들을 토대로 새롭게 만들어 진다는 점에서 차이가 있다. 이는 시간이 흘러감에 따라 올 수 있는 변화에도 대응할 수 있다는 점에서 장점을 갖는다고 볼 수 있다. 다만 개별 분

류자를 기억하고 있어야 하므로 개별 분류자는 되도록 간단한 것을 사용한다. 자세한 내용과 알고리즘은 Yoon (2010)에 나와 있다.

3. 전진적 단계 알고리즘 개념을 이용한 순차적 앙상블 방법

Yoon (2010)의 부스팅 알고리즘은 각 개별분류자의 계수를 업데이트할 때 각 개별분류자의 가중 오류의 함수 (예 : Adaboost의 경우 $\ln(\frac{1-\epsilon_j}{\epsilon_j})$에 비례)로 업데이트를 한다. Dietterich (2000)은 이러한 부스팅 방법은 종속변수 y에 노이즈가 존재하면 그 노이즈에 의해 다른 앙상블 방법에 비해 성능이 떨어짐을 보였다. 개념 변화는 이전 시점 배치들의 관점에서 보면 종속변수에 노이즈가 생기는 것으로 생각해 볼 수 있으므로 기존의 부스팅 방법이 약점을 가질 수도 있을 것이다. 또한 Hastie 등 (2001) 많은 연구자들에 의하면 Adaboost는 매 단계마다 최적의 해를 찾는 알고리즘으로 노이즈가 존재하는 소수의 자료를 잘 적합시키려고 한다. 그러다 보면 다수의 자료에 대한 성능이 떨어질 수도 있으므로 그 대안으로 부스팅 알고리즘 상의 각 분류자에 대한 계수의 절대값을 축소하여 업데이트하는 정규화 기법을 적용하는 것이 성능이 더 좋을 수 있다 하였다 (Hastie 등, 2001). 이런 정규화는 LASSO (Tibshirani, 1996)와 같은 방법을 직접 구현하여야 하나 자료가 많거나 분류자가 많은 경우 쉽지 않기 때문에 LASSO를 간단하게 근사할 수 있는 전진적 단계 선형 회귀법 (forward stagewise linear regression) 혹은 ϵ-부스팅 알고리즘을 이용하는 것이 위의 문제를 해결할 수 있는 방법이 될 수 있다 (Hastie 등, 2001). 이를 Yoon (2010)의 알고리즘에 적용시키면 아래와 같은 알고리즘을 만들 수 있다.

1. T : 데이터 배치의 수, $Mval$: 부스팅 회수
2. $t = 1, \cdots, T$에 대해 다음을 반복한다.
   (1) $t = 1$이면, $h_1 = \arg\max_h \sum_i y_{1,i} h(x_{1,i})$ 이며 $\lambda_1 = 1$.
   (2) $t > 1$이면,
      (a) $\beta_t^{(1)} = \mathbf{0}$: 길이가 $t-1$인 벡터
      (b) $m_i = (y_{t,i} h_1(x_{t,i}), \cdots, y_{t,i} h_{t-1}(x_{t,i}))^T$.
      (c) $M = (m_1, m_2, \cdots, m_{n_t})^T$.
      (d) $j = 1, \cdots, Mval$에 대해 다음을 반복한다.
         (ㄱ) $i = 1, \cdots, n_t$에 대해 $d_{j,i} = \exp((-M\beta_t^{(j)})_i) / \sum_k \exp((-M\beta_t^{(j)})_k)$, 여기서 $(\mathbf{a})_p$는 $\mathbf{a}$의 p번째 원소 (element)를 뜻한다.
         (ㄴ) $k_j = \arg\max_k (d_j^T M)_k$, 여기서 $d_j = (d_{j,1}, \cdots, d_{j,n_t})^T$이다.
         (ㄷ) $r_j = (d_j^T M)_{k_j}$, $\epsilon_j = (1 - r_j)/2$.
         (ㄹ) $\alpha_j = \frac{1}{2} \ln(\frac{1-\epsilon_j}{\epsilon_j})$.
         (ㅂ) $\beta_t^{(j+1)} = \beta_t^{(j)} + \epsilon \times sign(\alpha_j) e_{k_j}$, 여기서 e_{k_j} 는 k_j번째 원소가 1인 단위 벡터이며 ϵ는 미리 주어진 값이다.
      (e) $d_{t,i} = \exp((-M\beta_t^{(Mval+1)})_i) / \sum_k \exp((-M\beta_t^{(Mval+1)})_k)$.
      (f) $h_t = \arg\max_h \sum_i d_{t,i} y_{t,i} h(x_{t,i})$
      (g) $\beta = \frac{1}{2} \ln(\frac{1-\epsilon_t}{\epsilon_t})$, 여기서 ϵ_t는 h_t의 가중오류이다.
      (h) $\lambda_t = (\beta_t^{(Mval+1)T}, \beta)^T$.
   (3) $H_t(x) = sign\left(\frac{h(x)^T \lambda_t}{||\lambda_t||_1}\right)$, 여기서 $h(x) = (h_1(x), \cdots, h_t(x))^T$이다.

Yoon (2010)의 알고리즘과 가장 큰 차이는 알고리즘에서 2-(2)-(d)-(ㅂ) 부분이다. 각 분류자의 계수를 업데이트할 때 가중오류의 함수인 α_j만큼 더하거나 빼는 것이 아니라 미리 정한 ϵ만큼 (예를 들면 0.01) 더하거나 빼는 것으로 정규화의 효과를 얻게 하는 것이다. 또한 부스팅의 회수 ($Mval$)을 다르게 하여 BIC나 GCV와 같은 기준으로 최적의 부스팅 회수를 구하는 것도 또 하나의 차이이다.

4. 모의실험과 실제자료 분석

4.1. 모의실험

Yoon (2010)의 알고리즘인 부스팅 방법과의 비교를 위해 다음과 같은 모의실험을 실시하였다. 전체적인 모의실험 세팅은 Yoon (2010)을 참고하였다.

4.1.1. 데이터셋

비교를 위해 Yoon (2010)에서와 같은 데이터 셋을 사용하였다.

1. Sphere 데이터 : 샘플 $(\mathbf{x}, y)$는 3차원의 서로 독립인 $\mathbf{x} = (x_1, x_2, x_3)^T$ 을 갖는다. 단 $x_i \in [0, 1]$, $i = 1, 2, 3$ 이다. 기하학적으로 보면 샘플은 3차원 입방체(cube)에 위치하고 있다. 실제 그룹의 경계는 다음과 같이 정의되는 구이다.

$$B(x) = \sum_{i=1}^{3}(x_i - c_i)^2 - r^2 = 0.$$

   여기서 $c = (c_1, c_2, c_3)$는 구의 중심이며, r은 반지름이다. $B(x) \leq 0$이면 $y = 1$이고 $B(x) > 0$이면 $y = -1$이다. 이 데이터 셋은 예측변수들이 모두 연속형이고 그룹 경계가 비선형이기 때문에 학습 (learning)이 쉽지 않다.

2. Twonorm 데이터 : 이 데이터 셋은 20차원의 2 그룹 데이터이다. 각 그룹은 단위 공분산 행렬을 갖는 다변량 정규분표에서 생성된다. 그룹 1은 평균이 $(a, a, \cdots, a)$이며 그룹 2는 평균이 $(-a, -a, \cdots, -a)$이다.

Sphere 데이터의 경우, 개념의 변화가 없는 경우 (no concept drift)는 구의 중심 c가 변하지 않게 데이터를 생성하며 개념의 변화가 있으면 구의 중심을 각 차원별로 $\pm\delta$만큼 변화시킨다. 예를 들면 현재 시점 (block)에서 구의 중심이 $c = (0.4, 0.6, 0.5)$이고 $\delta = 0.05$이고 각 차원 이동 부호가 $(+1, -1, -1)$이라면 다음 시점 (block)에서 구의 중심은 $c = (0.45, 0.55, 0.45)$가 된다. 본 논문에서는 시작시점의 구의 중심을 $c = (0.5, 0.5, 0.5)$로 시작하였고 $\delta = 0.2$로 하였다. Twonorm 데이터의 경우, 개념 변화가 없으면 각 그룹의 평균이 고정되며 개념의 변화가 있는 경우에는 평균의 부호가 $r\%$가 변화하도록 하였다. 본 모의실험에서는 $r = 40$을 사용하였다. 각 시점에서의 데이터 셋의 크기는 500이다. 각 시점에서 생성된 앙상블의 정확도를 측정하기 위해 2000개의 테스트 셋을 사용하였다. 50번째 시점까지 데이터를 생성하였고 개념의 변화가 있는 경우 20번째 시점에서 변화가 생기도록 하였다.

4.1.2. 대용량 데이터 분류를 위한 모의실험

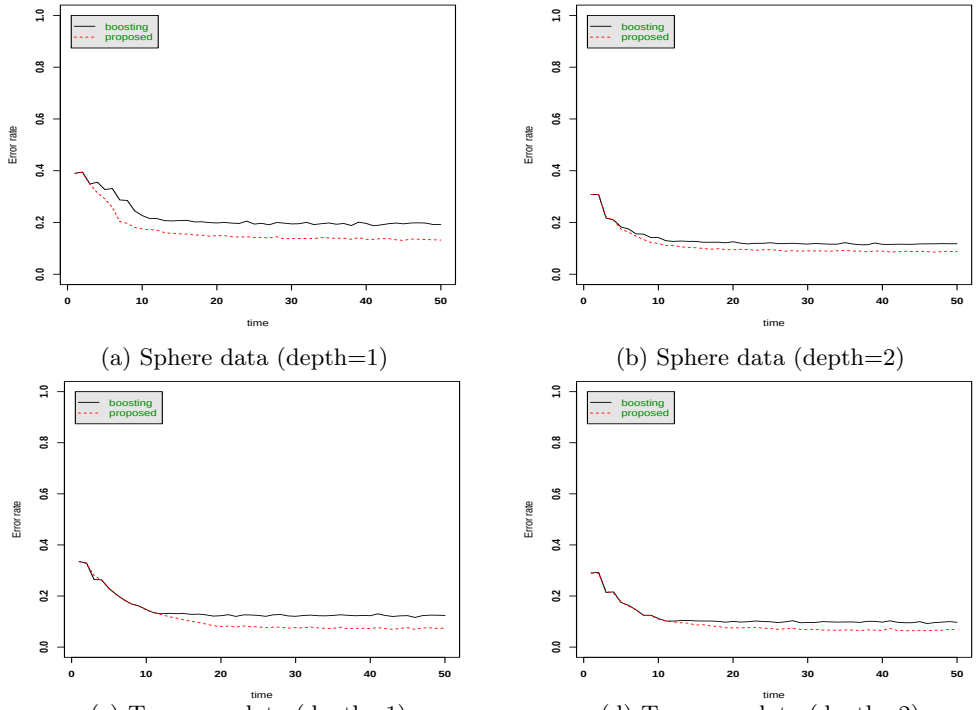
제안된 전진적 단계 알고리즘은 Yoon (2010)의 부스팅 알고리즘처럼 대용량 데이터에도 이용할 수 있다. 대용량 데이터 셋을 상대적으로 작은 숫자의 데이터 셋으로 나누어서, 나누어진 각각의 데이터 셋에서 분류자들을 만들고 이를 이용한 앙상블 방법을 적용하면 가능하다. 이런 방법을 사용한다면 일반적인 배깅이나 부스팅 방법보다 더 빠르면서도 메모리가 적게 드는 결과를 얻을 수 있을 것이다. 비교를 위해 4.1.1의 두 데이터 셋을 이용하였다. 데이터 셋의 크기는 25,000이고 이 데이터를 임의로 50개로 나누었다. 2000개의 테스트 셋을 이용하여 각 방법의 오분류율을 계산하였다.

4.1.3. 개별 분류자와 비교 앙상블 방법

본 모의실험에서는 개별 분류자를 가지치기 (pruning)가 없는 CART를 이용하여 부스팅 알고리즘과 본 논문의 알고리즘을 비교하였다. 각 시점에서 간단한 분류자를 이용해야만 더욱 효과적이므로 깊이 (depth)가 1, 2인 의사결정나무를 생성하였다. 각 시점에서의 평균 오분류율을 계산하기 위해 50번 반복시행 하였으며 각 시점에서 사용하는 개별분류자는 현재시점에서 10시점 이전까지의 개별분류자들로 고정하였다. 제안된 알고리즘의 경우, ϵ은 0.01로 하였다.

4.1.4. 모의실험결과

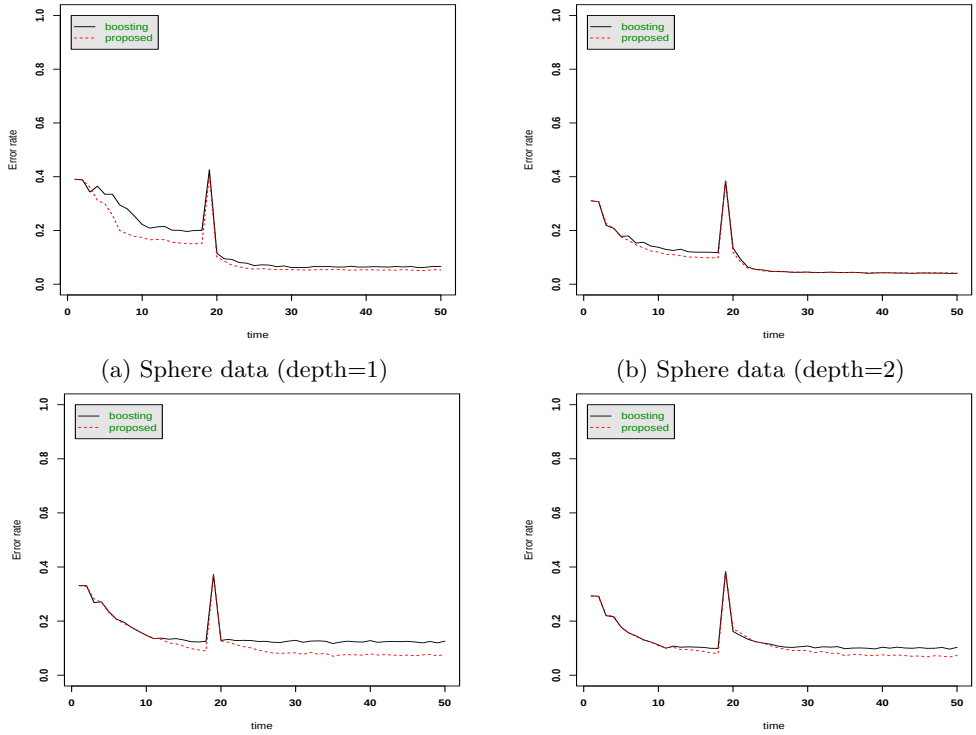
Figure 4.1은 개념의 변화가 없는 경우의 결과이다. Sphere 데이터의 경우 전진적 단계 알고리즘 방법이 좋은 결과를 특히 간단한 분류자 (깊이가 1인 경우)에서 보인다. 그러나 Twonorm 데이터의 경우는 거의 차이가 없다. 4.1.1에서 설명했듯이 Sphere 데이터는 학습하기 힘든 경우이다. 이처럼 분류하기 어려운 경우 제안한 전진적 단계 알고리즘이 더 좋은 결과를 보일 것으로 예상된다. 개념의 변화가 있는 경우의 결과는 Figure 4.2에서 볼 수 있다. 두 알고리즘 모두 변화가 일어난 시점에서 성능이 떨어지다가 곧 빠르게 회복하는 모습을 보인다. 미세하지만 제안된 알고리즘이 좋은 결과를 보이고 있으며 Twonorm 데이터의 경우 개념변화 후 오류율이 제안된 알고리즘이 부스팅 알고리즘에 비해 조금 더 낮음을 알 수 있다. 대용량 데이터의 경우, Table 4.1에 결과 (테스트셋의 오분율의 평균과 표준오차)를 나타내었다. 두 데이터 모두 의사결정나무의 깊이가 1인 경우 (개별 분류자가 간단한 경우) 제안된 알고리즘이 상대적으로 더 좋은 성능을 보였다. 이 결과는 순차적 데이터 경우와 같이 대용량 데이터의 분류의 경우에도 부스팅 알고리즘보다는 제안된 알고리즘이 더 효과적이라는 것을 알려준다.



(a) Sphere data (depth=1) (b) Sphere data (depth=2)

(c) Twonorm data (depth=1) (d) Twonorm data (depth=2)

Figure 4.1 Error rates of no concept drift situations



(a) Sphere data (depth=1) (b) Sphere data (depth=2)

(c) Twonorm data (depth=1) (d) Twonorm data (depth=2)

Figure 4.2 Error rates of concept drift situations

Table 4.1 The simulation results of large scale data

	Sphere data		Twonorm data	
algorithms	Average of test error	s.e of test error	Average of test error	s.e of test error
		depth=1		
Boosting	0.190	0.002	0.119	0.001
Proposed method	0.137	0.002	0.057	0.001
		depth=2		
Boosting	0.120	0.002	0.090	0.001
Proposed method	0.088	0.002	0.057	0.001

4.2. 실제자료 분석

다음의 두 데이터는 대용량 자료의 분류를 위한 제안된 방법의 효과를 측정하기 위해 사용하였다.

- Adult : 이 데이터는 Kohavi (1996)가 여러 가지 분류 방법을 비교하기 위해 사용한 미국 Census Bureau의 자료이다. 나이와 교육 수준, 직업, 성별등 14가지 인구통계학적 특성 등을 기초로 연 50,000달러이상 혹은 이하의 수입을 내는지를 예측하는 문제이다. 이 자료에는 50,000달러 이상의 소득이 23.93%를 차지하고 있으며 총 48,842명의 자료가 있다. 훈련자료 (training data)는 32,561개이며 나머지는 테스트 자료 (test data)로 사용한다.
- Anonymous web browsing : 이 자료는 Microsoft 웹 싸이트를 방문한 32,117명의 Web browsing 특성을 기록한 자료이다. 여기서는 사용자가 방문한 web 페이지를 기초로 하여 “Free down-

loads" 페이지를 방문하는지를 예측하고자 한다. 이 자료에는 "Free downloads"페이지를 방문한 유저가 10,835명 (33.1%)이며, 294개의 이항 예측변수가 있다. 테스트 자료의 크기는 5,000이다.

이 두 자료는 UCI machine learning repository (Bache와 Lichman, 2013)에서 이용할 수 있다. 자료를 임의로 나누어야 하기 때문에 10번 반복 시행하여 각 방법의 오분류율의 평균을 비교하였다. 또한 각 블록 크기가 500과 1000 정도가 되도록 하였으며 개별 분류자는 깊이 (depth)가 1인 CART를 사용하였다. Table 4.2에서는 이 두 자료에 대한 결과 (테스트셋의 오분류율의 평균과 표준오차)를 볼 수 있다. Adult 데이터에서는 제안된 알고리즘 방법이 부스팅 방법보다는 약간 좋거나 비슷한 결과를 보였으나 Anonymous Web browsing 데이터에서는 부스팅 방법이 제안된 알고리즘 방법보다는 같거나 약간 좋은 결과를 보였다.

Table 4.2 The results of real data sets

block size	500		1000	
algorithms	Average of test error	s.e of test error	Average of test error	s.e of test error
		Adult data		
Boosting	0.163	0.003	0.151	0.001
Proposed method	0.157	0.002	0.149	0.001
		Anonymous web browsing data		
Boosting	0.280	0.002	0.277	0.003
Proposed method	0.291	0.005	0.277	0.003

5. 요약 및 결론

Yoon (2010)에서 볼 수 있듯이 부스팅 알고리즘은 앙상블 방법 중 대용량 데이터나 순차적 배치 데이터에 대해 좋은 대안이 될 수 있었다. 그러나 부스팅 알고리즘은 자료에 노이즈가 존재하는 경우 그 성능이 저하된다. 순차적 배치 데이터의 경우, 특히 개념 변화가 존재하는 경우는 자료에 노이즈가 존재하는 경우와 비슷하게 인식될 수 있으므로 부스팅의 성능이 안 좋을 수 있다고 추측된다. 이의 해결책 중 하나는 정규화 기법이다. 본 논문에서 제안된 전진적 단계 알고리즘은 정규화 기법을 근사적으로 구현하고 계산이 비교적 간단한 알고리즘이므로 부스팅의 단점을 어느 정도 해결할 것이라 생각된다. 실제로 본 논문에서는 모의실험과 실제 자료 분석을 통해 제안된 알고리즘이 대용량 데이터나 순차적 배치 데이터에 대해 부스팅보다 좋은 성능을 가질 수 있는 알고리즘이라는 것을 보였다.

References

Bache, K. and Lichman, M. (2013). UCI machine learning repository [`http://archive.ics.uci.edu/ml`]. University of California, School of Information and Computer Science, Irvine, CA.

Breiman, L. (1996). Bagging predictors. *Machine Learning*, **24**, 123-140.

Breiman, L. (1998). Arcing classifiers (with discussion). *Annals of Statistics*, **26**, 801-849.

Breiman, L., Friedman, J., Olshen, R. and Stone, C. (1984). *Classification and regression trees*, Chapman and Hall, New York, NY.

Dietterich, T. G. (2000). An experimental comparison of three methods for constructing ensembles decision trees: bagging, boosting and randomization. *Machine Learning*, **40**, 139-157.

Freund, Y. and Schapire, R. E. (1997). A decision-theoretic generalization of online learning and application to boosting. *Journal of Computer and System Science*, **55**, 119-139.

Hastie, T., Tibshirani, R. and Friedman, J. (2001). *The elements of statistical learning*, Springer-Verlag, New York, NY.

Kim, S. H., Cho, D. H. and Seok, K. H. (2012). Study on the ensemble methods with kernel ridge regression. *Journal of the Korean Data & Information Science Society*, **23**, 375-383.

Kohavi, R. (1996). Scaling up the accuracy of naive-bayes classifiers: A decision-tree hybrid. *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, 202-207.

Kuncheva, L. I. (2004). Classification ensemble for changing environments. *Proceedings of 5th International Workshop on Multiple Classifier systems*, 1-15.

Quinlan, J. R. (1993). *C4.5 : programs for machine learning*, Morgan Kaufmann, San Maeto, CA.

Street, W. N. and Kim, Y. S. (2001). A streaming ensemble algorithm (SEA) for large scale classification. *Proceedings of the 7th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 377-382.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society B*, **58**, 267-288.

Wang, H., Fan, W., Yu, P. S. and Han, J. (2003). Mining concept drifting data streams using ensemble classifiers. *Proceedings of then 9th ACM SIGKDD International Conference on Knowledge discovery and Data Mining*, 226-235.

Yoon, Y. J. (2010). Boosting algorithms for large-scale data and data batch stream (in Korean). *The Korean Journal of Applied Statistics*, **23**, 197-206.

Journal of the Korean Data & Information Science Society
2014, **25**(6), 1283–1291

http://dx.doi.org/10.7465/jkdi.2014.25.6.1283
한국데이터정보과학회지

Classification of large-scale data and data batch stream with forward stagewise algorithm[†]

Young Joo Yoon[1]

[1]Department of Business Information Statistics, Daejeon University

Received 28 August 2014, revised 16 September 2014, accepted 2 October 2014

Abstract

In this paper, we propose forward stagewise algorithm when data are very large or coming in batches sequentially over time. In this situation, ordinary boosting algorithm for large scale data and data batch stream may be greedy and have worse performance with class noise situations. To overcome those and apply to large scale data or data batch stream, we modify the forward stagewise algorithm. This algorithm has better results for both large scale data and data batch stream with or without concept drift on simulated data and real data sets than boosting algorithms.

Keywords: Concept drift, data stream, ensemble method, forward stagewise algorithm, large scale data.

[†] This research is supported by the Daejeon University research fund (2012).
[1] Assistant professor, Department of Business Information Statistics, Daejeon University, Daejeon, 300-716, Korea. E-mail: yoonyj@dju.kr