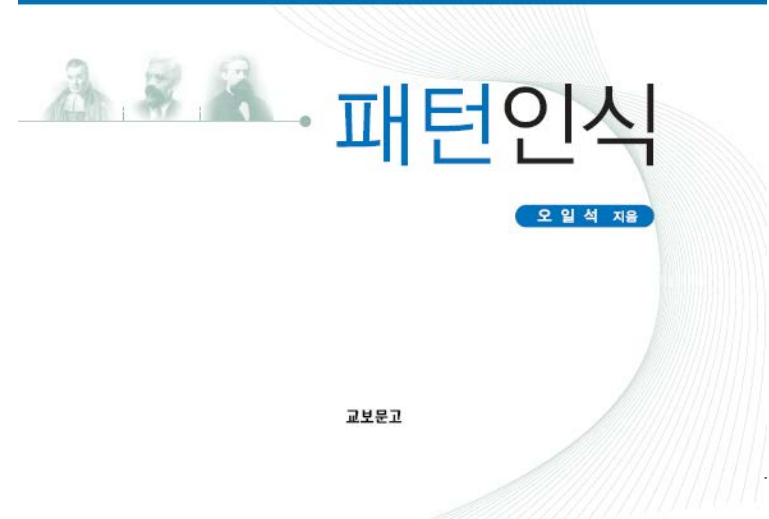


12장. 혼성 모델

오일석, 패턴인식, 교보문고, 2008.



들어가는 말

- 패턴인식에는 아주 많은 다양한 알고리즘 존재
 - 어느 것이 가장 좋은가? 보편적으로 우수한 알고리즘이 존재하는가?
(12.1~12.2 절)
 - 단일 알고리즘의 한계를 인정하고 여러 알고리즘을 결합하는 혼성 모델
(12.3~12.6 절)
 - 여러 알고리즘을 결합하면 가장 좋은 단일 알고리즘보다 좋다는 것이 공통된 의견. 따라서 실용 시스템 구현에서 혼성 모델 많이 사용하는 추세

12.1 알고리즘의 성능 측정

■ 공학적 관점

- 특정 문제가 주어진 상황에서 그 문제를 푸는 가장 우수한 프로그램을 설계하는 것이 목표
- 공학적으로는 ②와 같은 분류기를 선택해야 함

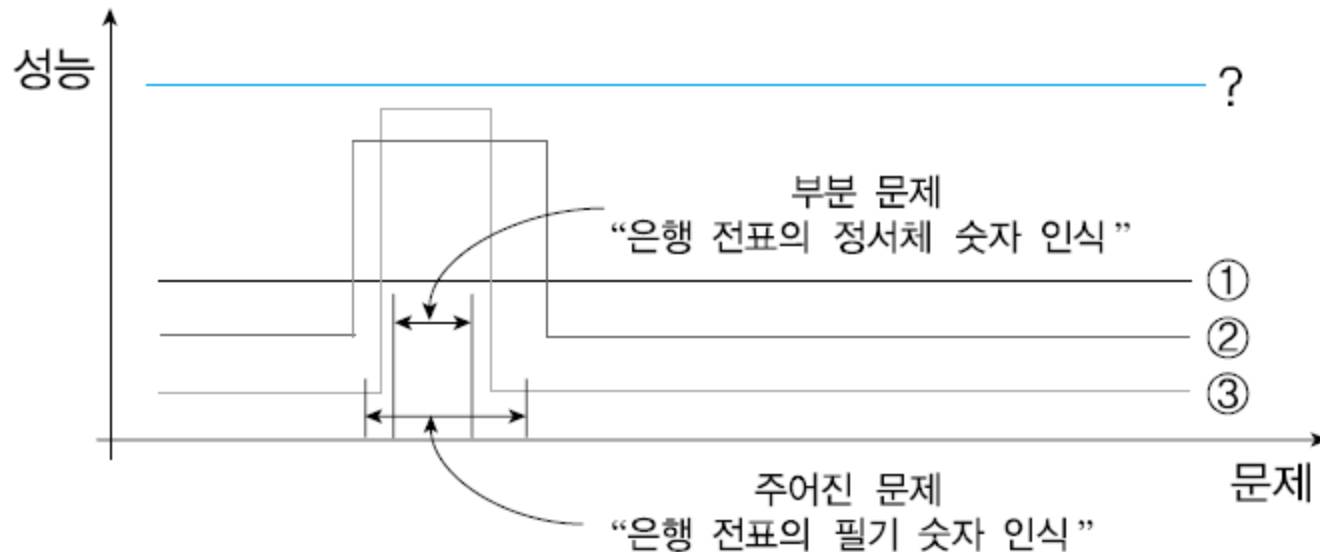


그림 12.1 분류 알고리즘의 성능 특성

12.1 알고리즘의 성능 측정

■ 공짜 점심 없음

- 모든 문제에 대해 다른 모든 알고리즘을 능가하는 알고리즘이 있나? (즉 그림 12.1에서 파란색 성능을 갖는 알고리즘 있나?)
- 없다. ← Wolpert 와 Macready의 공짜 점심 없음 정리

No free lunch라는 용어는 경제학에서 이미 사용하던 것이다. 어떤 특정 정책에 의해 얼핏 보면 이득을 얻는 것 같지만 그것은 한 측면의 이득일 뿐이고 반드시 이면에 다른 측면이 있고 그 측면에서 손해가 발생한다는 것이다.

12.2 재 샘플링에 의한 성능 평가

■ 재 샘플링 기법

- 데이터베이스의 질적 양적 품질은 인식기의 성능을 좌우함
- 분류기 모델을 선택하는데에는 훈련 집합 이외에 별도의 검증 집합 필요
- 현실적인 문제
 - 데이터베이스 수집에는 많은 비용이 든다.
 - 대부분 경우 양이 부족하다.
 - 이런 경우 재 샘플링 기법을 사용하면 성능 측정의 신뢰도를 높일 수 있다.

12.2 재 샘플링에 의한 성능 평가

■ 교차 검증 cross validation

- 훈련 집합을 k 개의 부분 집합으로 등분한 후,
- $k-1$ 개로 훈련하고 나머지 한 개로 테스트. 이런 과정을 서로 다른 부분 집합으로 k 번 수행하여 얻은 성능을 평균함
- 이를 k -겹 교차 검증이라 부름
- $k=N$ 인 경우 (N 은 샘플 개수) 하나 남기기 leave-one-out 또는 잭나이프 jackknife 기법

알고리즘 [12.1] 교차 검증

입력: 분류기 c , 훈련 집합 $X = \{(\mathbf{x}_1, t_1), (\mathbf{x}_2, t_2), \dots, (\mathbf{x}_N, t_N)\}$, k ($2 \leq k \leq N$)

출력: 성능 q

알고리즘:

1. X 를 k 개의 부분 집합으로 등분하고 그들을 X_1, X_2, \dots, X_k 라 한다.
2. **for** ($i = 1$ **to** k) {
3. $X' = \bigcup_{j=1, j \neq i}^k X_j$ 로 c 를 학습시킨다.
4. X_i 로 c 의 성능을 측정하여 q_i 라 한다.
5. }
6. $q = (q_1 + \dots + q_k)/k$;



12.2 재 샘플링에 의한 성능 평가

■ 붓스트랩bootstrap

붓스트랩 기법은 중복을 허용한다. N 개의 샘플을 가진 샘플 집합 X 가 있다고 하자. 집합 X 에서 N 개의 샘플을 임의로 뽑는다. 이때 한번에 하나씩 뽑는데 뽑은 샘플은 다시 집어 넣는다. 따라서 어떤 샘플은 두 번 이상 뽑히고 어떤 샘플은 한 번도 안 뽑힐 수도 있다. 이렇게 얻은 샘플 집합을 X_1 이라 표기하자. 이러한 샘플링 작업을 독립적으로 T 번 수행하여 X_1, X_2, \dots, X_T 를 만든다. 이들을 이용하여 성능을 T 번 측정하고 그 결과를 평균한 값을 최종 성능으로 취한다. 이렇게 구한 값은 통계적으로 보다 신뢰도가 높다고 볼 수 있다.

알고리즘 [12.2]

붓스트랩

입력: 분류기 c , 훈련 집합 $X = \{(\mathbf{x}_1, t_1), (\mathbf{x}_2, t_2), \dots, (\mathbf{x}_N, t_N)\}$, 반복 회수 T

출력: 성능 q

알고리즘:

1. **for** ($t = 1$ **to** T) {
2. X 에서 임의로 N 개의 샘플을 뽑아 X_t 라 한다. 이때 대치를 허용한다.
3. X_t 로 분류기 c 를 학습시킨다.
4. $X - X_t$ 로 c 의 성능을 측정하여 q_t 라 한다.
5. }
6. $q = (q_1 + \dots + q_T) / T$;

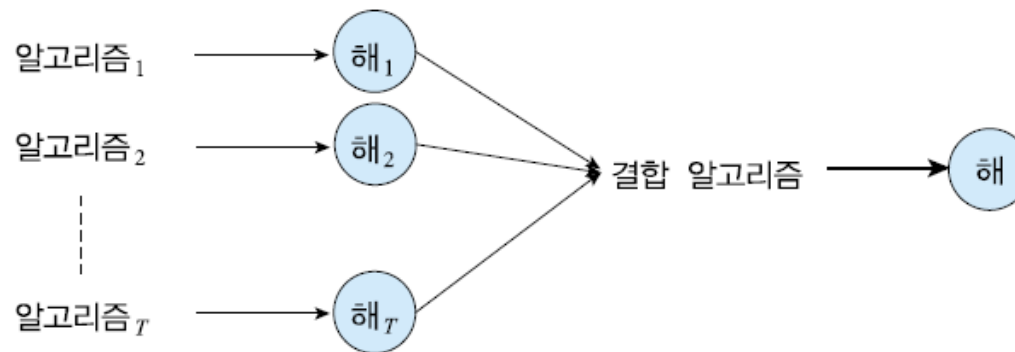


12.3 혼성 모델의 발상

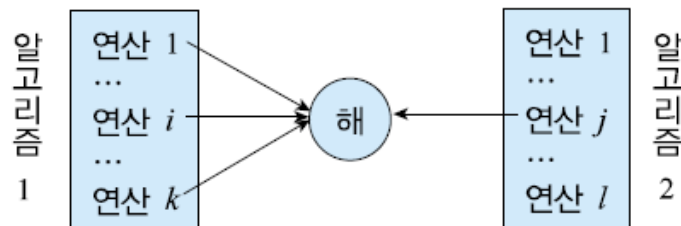
사람들은 어려운 일을 달성하기 위해 협동하는 경우가 많다. 각자가 가진 서로 다른 능력을 하나로 모아 보다 성공적인 결과를 얻기 위해서이다. 패턴 인식에서 혼성 모델은 hybrid model 이러한 아이디어에서 창안한 것이다. 혼성 모델 아이디어를 실현하기 위해서는 여러 질문에 적절한 답을 해야만 한다. 무엇을 결합할 것인지? 어떻게 결합할 것인지? 왜 결합하는지? 결합 알고리즘은 수학적으로 타당 sound 한지? 이러한 문제 의식을 가지고 혼성 모델의 세계로 들어가 보자.

12.3.1 동기

- 사람들의 의사 결정 방식을 모방
 - 여러 전문가의 의견을 듣고 그것을 결합하여 최종 결정을 내린다.
 - 혼성 모델은 이러한 동기에서 출발한다.
- 여러 알고리즘이 협동하는 방식



(a) 알고리즘 결합



(b) 연산 공유

그림 12.3 혼성 모델의 유형

12.3.1 동기

- 세가지 문제를 풀어야 한다.
 - 앙상블 생성: 여러 전문가를 확보하는 일에 해당
 - 앙상블 선택: 전체 전문가 집단에서 보다 유용한 전문가의 선택에 해당 (선택적)
 - 앙상블 결합: 전문가의 의견을 결합하여 최종 의견을 만드는 과정에 해당

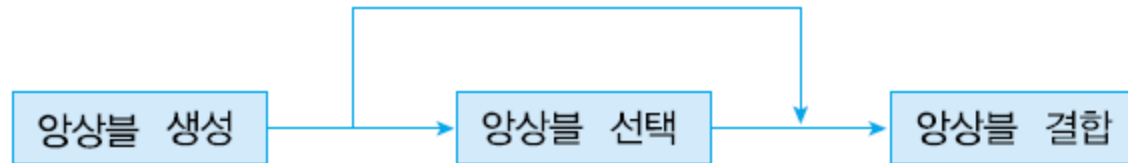


그림 12.4 분류기 앙상블 시스템의 세 가지 문제

12.3.2 이유

패턴 인식 시스템을 설계할 때 앙상블 시스템 도입을 검토하는 것은 이제 특별한 생각이 아니다. 개발 현장에서 기본 요소로 간주하고 있는 것이다. 이러한 인기에 대한 몇 가지 이유가 있다. 이러한 이유를 알면 앙상블 시스템을 보다 잘 이해할 수 있게 된다.

- 혼성 모델을 사용하는 몇 가지 이유
 - 나쁜 운을 피할 수 있다.
 - 성능 향상을 꾀할 수 있다.
 - 데이터 양에 따른 어려움을 극복할 수 있다.
 - 데이터 질에 따른 어려움을 극복할 수 있다.
 - 다중 센서 시스템에서 효과적이다.
 - 결정 경계가 너무 복잡한 경우에 효과적일 수 있다.
 - 점진 학습이 가능하다.

12.4 앙상블 생성

- 여러 방식
 - 재 샘플링 기법: 배깅, 부스팅
 - 서로 다른 분류 알고리즘 사용 (예, MLP, SVM, k-NN 등)
 - 서로 다른 특징 부분 집합 사용
- 앙상블을 구성하는 요소 분류기들의 다양성은 매우 중요하다.

12.4.1 배깅

붓스트랩

■ 붓스트랩을 변형한 기법

□ Bootstrap aggregating

으로 T 개의 샘플 집합 X_1, X_2, \dots, X_T 를 얻었다고 하자. 배깅은 X_t 를 훈련 집합으로 사용하여 분류기 c_t 를 만든다. 이러한 분류기 학습을 $1 \leq t \leq T$ 에 대해 독립적으로 수행하여 T 개의 분류기를 만든다. 이때 분류 알고리즘은 어떤 것이라도 좋다. MLP 일 수도 있고 트리 분류기일 수도 있다.

알고리즘 [12.3]

배깅

입력: 훈련 집합 $X = \{(\mathbf{x}_1, t_1), (\mathbf{x}_2, t_2), \dots, (\mathbf{x}_N, t_N)\}$, 샘플링 비율 p ($0 < p \leq 1$)

출력: 분류기 앙상블 $C = \{c_t, 1 \leq t \leq T\}$

알고리즘:

1. $t = 0$;
2. $C = \emptyset$;
3. **repeat** {
4. $t++$;
5. X 에서 임의로 pN 개의 샘플을 뽑아 X_t 라 한다. 이때 대치를 허용한다.
6. X_t 로 분류기 c_t 를 학습시킨다.
7. $C = C \cup c_t$;
8. } **until** (멈춤 조건);

12.4.2 부스팅

부스팅은 **boosting** 배경에 비해 보다 정교한 재 샘플링 연산을 사용한다. 부스팅에서는 t 번째 분류기 c_t 와 그 다음의 c_{t+1} 이 서로 연관성을 가지고 만들어진다. (배경에서는 이들이 독립적으로 만들어졌다.) X 의 샘플은 분류기 c_t 가 맞추는 것과 틀리는 것으로 나눌 수 있다. 맞춘 샘플은 이제 인식이 가능해졌으므로 가중치를 낮추어도 된다. 하지만 틀린 샘플은 여전히 ‘까다로운’ 상대이므로 가중치를 높여 준다. 이 가중치는 $t+1$ 번째 샘플 집합을 뽑는데 중요한 역할을 한다. 즉 가중치가 높은 샘플이 뽑힐 가능성을 높게 조절하는 정책을 사용한다. 부스팅에서는 대치를 허용하지 않는다. 즉 한번 뽑힌 샘플은 다시 뽑히지 않는다.

알고리즘 [12.4] AdaBoost

입력: 훈련 집합 $X = \{(\mathbf{x}_1, t_1), (\mathbf{x}_2, t_2), \dots, (\mathbf{x}_N, t_N)\}$

출력: 분류기 앙상블 $C = \{c_t, 1 \leq t \leq T\}$, 분류기 신뢰도 $\alpha_t, 1 \leq t \leq T$

알고리즘:

1. $t = 1$;
2. $C = \emptyset$;
3. **for** ($j = 1$ **to** N) $w_t(j) = 1/N$; // 처음에는 같은 가중치를 줌

```

4. repeat {
5.      $w_t$ 를 감안하여 분류기  $c_t$ 를 학습시킨다. // 가중치가 큰 샘플을 보다 중요
        시험
6.      $\varepsilon_t = 0$ ;
7.     for ( $j = 1$  to  $N$ ) if ( $c_t(\mathbf{x}_j) \neq t_j$ )  $\varepsilon_t = \varepsilon_t + w_t(j)$ ; // 오류 (틀린 샘플의 가중치
        합) 계산
8.     if ( $\varepsilon_t < 0.5$ ) { // 0.5보다 작은 경우만 분류기  $c_t$ 를 취함
9.          $\alpha_t = \frac{1}{2} \ln \left( \frac{1 - \varepsilon_t}{\varepsilon_t} \right)$ ; // 분류기  $c_t$ 의 신뢰도
10.        for ( $j = 1$  to  $N$ )
11.            if ( $c_t(\mathbf{x}_j) \neq t_j$ )  $w_{t+1}(j) = w_t(j) * e^{\alpha_t}$ ; // 틀린 샘플의 가중치 높임
12.            else  $w_{t+1}(j) = w_t(j) * e^{-\alpha_t}$ ; // 맞춘 샘플의 가중치 낮춤
13.         $sum = \sum_{j=1}^N w_{t+1}(j)$ ;
14.        for ( $j = 1$  to  $N$ )  $w_{t+1}(j) = w_{t+1}(j)/sum$ ; // 가중치 정규화
15.         $C = C \cup c_t$ ;
16.         $t++$ ;
17.    }
18. } until (멈춤 조건);
19.  $T = t - 1$ ;

```

12.4.2 부스팅

- 라인 5와 라인 10-12의 가중치 관리가 부스팅의 핵심

- 라인 5

- 가중치가 큰 샘플을 보다 잘 맞추도록 분류기 c_t 를 학습 시킴
 - 여러 구현 방식이 가능

5. // w_t 를 감안하여 분류기 c_t 를 학습시킨다. (가중치가 큰 샘플을 보다 중요시함)

5.1 X 에서 pN 개의 샘플을 뽑아 X_t 라 한다. 이때 \mathbf{x}_j 가 뽑힐 확률이 $w_t(j)$ 가 되도록 한다. 대치는 허용하지 않는다.

5.2 X_t 로 c_t 를 학습시킨다.

또는

5. $\varepsilon_t = \sum_{j=1, c_t(\mathbf{x}_j) \neq t_j}^N w_t(j)$ 가 최소가 되도록 c_t 를 학습시킨다.

- 라인 10-12 (가중치를 조절해 줌)

- 맞춘 샘플은 가중치 낮추어 주고, 틀린 샘플은 높여 준다.
 - 왜?

12.4.2 부스팅

AdaBoost는 분류기 앙상블 C 와 그것을 구성하는 분류기 각각의 신뢰도 α_t , $1 \leq t \leq T$ 를 출력한다. 따라서 앙상블 결합은 단순한 다수 투표 방식이 아니라 신뢰도가 높은 분류기에 비중을 더 주는 가중 다수 투표 **weighted majority voting** 방식을 사용한다. 앙상블 결합 방식은 12.5절을 참조하라. 가중 다수 투표 방식을 사용한다는 점이 AdaBoost가 배깅에 비해 우수한 요인 중의 하나이다. AdaBoost의 또 다른 강점은 분류기들이 상호 보완적이라는 데서 찾을 수 있다. c_t 의 약점을 c_{t+1} 이 보완해주는 것이다.

12.4.2 부스팅

지금까지 재 샘플링 기법으로 교차 검증, 붓스트랩, 배깅, 그리고 부스팅의 네 가지를 소개하였다. 이들은 대치를 허용하는 지 여부와 어디에 쓰이는 지에 따라 표 12.1과 같이 구분할 수 있다.

표 12.1 재 샘플링 기법의 구분

	대치 허용 안 함	대치 허용함
성능 평가에 쓰임	교차 검증	붓스트랩
앙상블 생성에 쓰임	부스팅	배깅

12.5 앙상블 결합

앙상블 결합은 다중 분류기의 출력을 결합하여 하나의 분류 결과를 만드는 과정을 일컫는다. 따라서 요소 분류기의 출력 특성에 대해 먼저 살펴보아야 한다. 요소 분류기의 출력은 아래 세 가지 형태 중의 하나를 취한다. 부류의 개수는 M 이라 한다. i 번째 부류는 ω_i 로 표기하며 혼돈 우려가 없는 경우 편의상 i 로 표기하기도 한다.

1. 부류 표지 **class label**: 패턴 \mathbf{x} 는 ω_i 에 속한다. 이것을 표지 벡터 $\mathbf{L} = (l_1, l_2, \dots, l_M)^T$ 으로 표현할 수 있다. l_i 는 1이고 나머지 $l_j, j \neq i$ 는 0이다.
2. 부류 순위 **class ranking**: 패턴 \mathbf{x} 가 ω_i 에 속할 가능성의 순위는 r_i 이다. 이것을 모든 부류에 대해 쓰면 순위 벡터 $\mathbf{R} = (r_1, r_2, \dots, r_M)^T$ 이 된다. r_i 는 $[1, M]$ 사이의 정수이며 $i \neq j$ 에 대해 $r_i \neq r_j$ 이어야 한다.
3. 부류 확률 **class probability**: 패턴 \mathbf{x} 가 ω_i 에 속할 확률은 p_i 이다. 이것을 모든 부류에 대해 쓰면 확률 벡터 $\mathbf{P} = (p_1, p_2, \dots, p_M)^T$ 이 된다. p_i 는 $[0, 1]$ 사이의 실수이며 $\sum_{i=1, M} p_i = 1$ 이어야 한다.

12.5 앙상블 결합

■ 분류기들의 출력 특성

- 베이시언 분류기와 HMM은 엄밀한 부류 확률을 출력한다.
- 신경망과 SVM은 실수를 출력하지만 확률이 아님. 확률이 필요한 경우 softmax 함수로 변환한 값을 확률로 간주

$$p_i = \text{softmax}(o_i) = \frac{e^{o_i}}{\sum_{j=1}^M e^{o_j}} \quad (12.2)$$

예제 12.1 부류 표지, 부류 순위, 그리고 부류 확률

부류의 개수가 $M=5$ 라 하자. MLP가 \mathbf{x} 라는 미지의 패턴에 대해 아래와 같은 벡터를 출력했다고 하자.

$$\mathbf{O} = (0.001, 0.0, 0.2, 0.75, 0.1)^T$$

이것을 순위 벡터로 변환하면 $\mathbf{R} = (4, 5, 2, 1, 3)^T$ 이 된다. 또한 네 번째 부류의 출력이 최대이므로 부류 표지 형태의 출력은 ω_4 가 되고 벡터 형태로 표현하면 $\mathbf{L} = (0, 0, 0, 1, 0)^T$ 이 된다.

MLP 출력 \mathbf{O} 를 softmax 함수로 확률 형태로 변환해 보자. 첫 번째 부류의 확률 p_1 은 아래와 같이 계산할 수 있다.

$$p_1 = \frac{e^{0.001}}{e^{0.001} + e^{0.0} + e^{0.2} + e^{0.75} + e^{0.1}} = 0.15532$$

나머지에 대해서도 계산해 보면 아래와 같은 확률 벡터를 얻게 된다.

$$\mathbf{P} = (0.15532, 0.15517, 0.18952, 0.32849, 0.17149)^T$$



12.5.1 부류 표지

이 방법은 부류 표지 벡터 $\mathbf{L} = (l_1, l_2, \dots, l_M)^T$ 을 사용하여 다중 분류기를 결합한다. 분류기가 T 개 있으므로 표지 벡터가 T 개가 있다. 이들을 $\mathbf{L}_1, \mathbf{L}_2, \dots, \mathbf{L}_T$ 로 구분하기로 하자. t 번째 분류기의 출력은 $\mathbf{L}_t = (l_{t1}, l_{t2}, \dots, l_{tM})^T$ 과 같이 표기한다.

■ 다수 투표 majority voting

□ 최다 득표자 선택

$$q = \arg \max_{j=1, M} \sum_{t=1}^T l_{tj} \quad (12.3)$$

□ 여러 변형: 과반 득표 또는 만장 일치

■ 가중 다수 투표 weighted majority voting

□ 분류기의 신뢰도 α_t 고려 (Adaboost로 생성한 앙상블의 경우에 적용 가능)

$$q = \arg \max_{j=1, M} \sum_{t=1}^T \alpha_t l_{tj} \quad (12.4)$$

부류의 개수가 $M=3$ 이고 분류기의 개수는 $T=5$ 라 하자. 미지의 패턴 \mathbf{x} 에 대해 다섯 개의 분류기가 그림 12.5와 같은 표지 벡터를 출력했다고 하자.

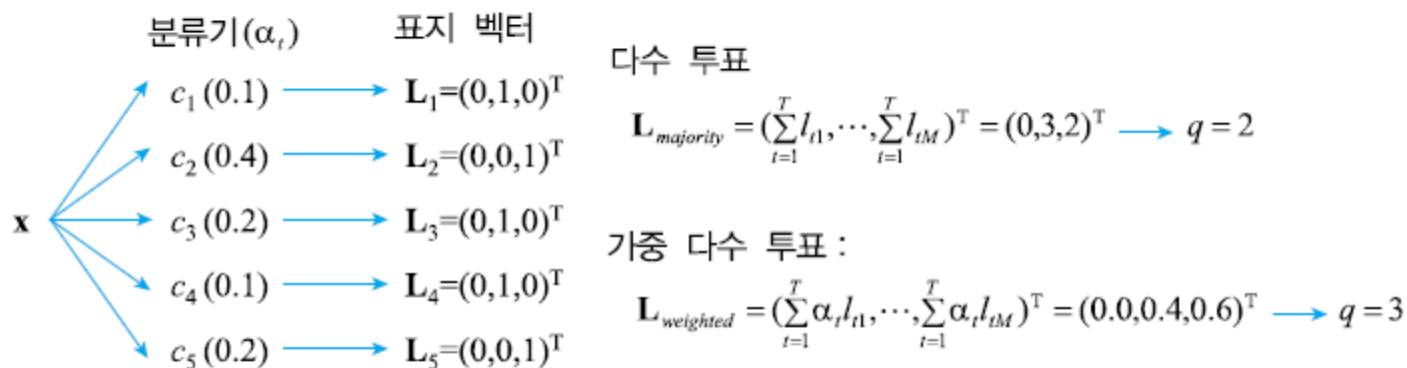


그림 12.5 다수 투표와 가중 다수 투표에서의 의사 결정

다수 투표 알고리즘에서는 두 번째 부류가 최다 득표했으므로 ω_2 로 분류한다. 하지만 가중치 α_t 를 고려한 경우에는 세 번째 부류가 가장 높은 값을 획득했으므로 ω_3 으로 분류한다. 이 결과는 신뢰도가 0.4로 다른 것에 비해 높은 두 번째 분류기 c_2 가 ω_3 을 선택하였기 때문에 발생한 것으로 볼 수 있다. ■■■

12.5.2 부류 순위

- 먼저 순위를 점수로 환산한다.

- 여러 방식

- r 순위에 $M-r$ 점 부여

- r 순위에 $1/r$ 점 부여

- 예) c_t 가 $R_t=(4, 5, 2, 1, 3)^T$ 를 출력했다면, 첫번째 방법은 $S_t=(1, 0, 3, 4, 2)^T$ 가 되고, 두번째 방법은 $S_t=(0.25, 0.2, 0.5, 1, 0.333)^T$ 이 됨

- 보다 계수에 Borda count 의한 결합

$$q = \arg \max_{j=1,M} \sum_{t=1}^T s_{tj} \quad (12.5)$$

12.5.3 부류 확률

- 합, 가중 합, 최소, 최대, 메디언, 그리고 곱 규칙
(c_t 의 출력을 $\mathbf{P}_t=(p_{t1}, p_{t2}, \dots, p_{tM})^T$ 으로 표기)

$$\text{합} \quad \beta_j = \sum_{t=1}^T p_{tj} \quad (12.6)$$

$$\text{가중 합} \quad \beta_j = \sum_{t=1}^T \alpha_t p_{tj} \quad (12.7)$$

$$\text{최대} \quad \beta_j = \max_{t=1, T} p_{tj} \quad (12.8)$$

$$\text{최소} \quad \beta_j = \min_{t=1, T} p_{tj} \quad (12.9)$$

$$\text{메디언} \quad \beta_j = \text{median}_{t=1, T} p_{tj} \quad (12.10)$$

$$\text{곱} \quad \beta_j = \prod_{t=1}^T p_{tj} \quad (12.11)$$

- 위의 규칙으로 구한 값을 사용하여, (12.12) 식으로 ω_q 로 분류

$$q = \arg \max_{j=1, M} \beta_j \quad (12.12)$$

예제 12.3

연산 규칙에 따른 분류기 결합

그림 12.8은 미지의 패턴 \mathbf{x} 에 대해 다섯 개 분류기의 분류 결과를 확률 벡터로 보이고 있다. 이들 벡터에 대해 여섯 가지 연산에 따라 구한 벡터 \mathbf{B} 와 최종 분류 결과를 보여 주고 있다. 최대 규칙만 제외하고 모두 ω_1 로 분류하였다. 최대 규칙은 ω_2 로 분류하였다. 곱 규칙은 합 규칙에 비해 부류간 값 차이가 크게 나는 것을 볼 수 있다.

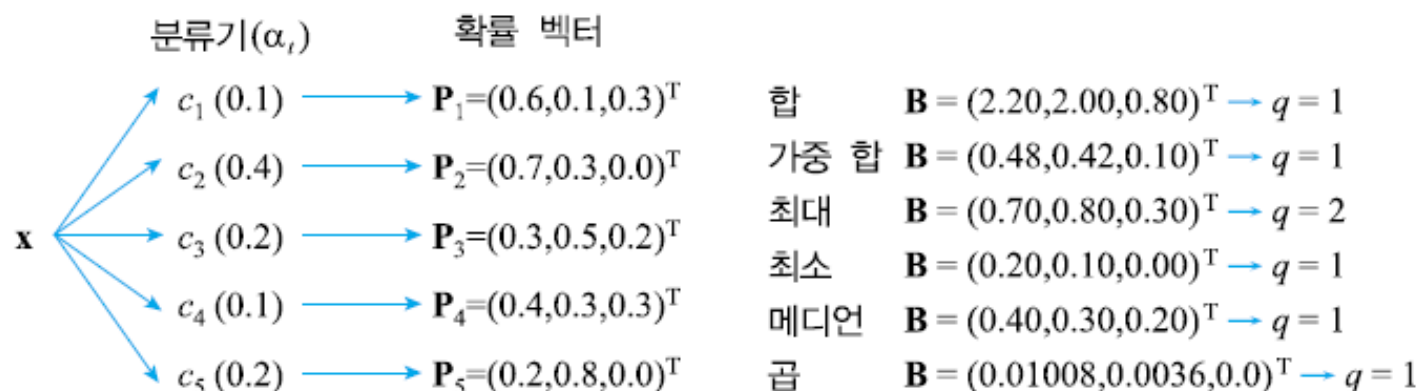


그림 12.8 연산 규칙에 따른 분류기 결합



12.6 앙상블 선택

앙상블 선택은 앙상블 생성이나 앙상블 결합에 비해 소홀히 취급되어 온 것이 사실이다. 하지만 전체 앙상블 시스템의 성능을 향상시킨다는 목표에 따른다면 중요성을 간과할 수 없다. 앙상블 선택은 앙상블이 어떤 알고리즘으로 생성되었느냐에 따라 그 중요성이 다를 수 있다. 예를 들어 AdaBoost로 생성된 앙상블 분류기는 상호 보완성이 뛰어나다. 왜냐하면 어떤 분류기는 그보다 먼저 생성된 분류기가 틀린 샘플에 대해 튜닝이 되어 있기 때문이다. 따라서 이런 경우에는 굳이 앙상블 선택을 적용할 필요가 없다. 하지만 배깅이나 특징 부분 집합으로 만든 앙상블의 경우는 앙상블 선택의 필요성이 상대적으로 높다.

12.6.1 다양성 척도

먼저 표기에 대해 살펴 보자. 앞에서와 마찬가지로 분류기 앙상블은 $C = \{c_1, c_2, \dots, c_T\}$ 라 하고 부류는 $\omega_1, \omega_2, \dots, \omega_M$ 으로 표기한다. 다양성 측정에 사용할 훈련 집합은 $X = \{(\mathbf{x}_1, t_1), (\mathbf{x}_2, t_2), \dots, (\mathbf{x}_N, t_N)\}$ 이라 한다. 여기에서 소개하는 척도는 샘플 \mathbf{x}_j 를 분류기 c_i 가 맞추었는지 여부만 가지고 측정한다. c_i 가 \mathbf{x}_j 를 맞추면 (즉 $c_i(\mathbf{x}_j) = t_j$) $y_{ji} = 1$ 로 하고 틀리면 $y_{ji} = 0$ 으로 한다. 분류기 c_i 의 인식 결과를 벡터로 쓰면 $\mathbf{y}_i = (y_{1i}, y_{2i}, \dots, y_{Ni})^T$ 가 된다.

표 12.2는 두 개의 분류기 c_i 와 c_k 의 관계를 기술한다. 여기에서 n^{11} 은 두 분류기가 모두 맞춘 샘플의 개수이고 n^{00} 은 둘 다 틀린 것의 개수이다. n^{10} 은 c_i 는 맞추고 c_k 는 틀린 샘플의 개수이고 n^{01} 은 c_i 는 틀리고 c_k 는 맞춘 것의 개수이다. 물론 $N = n^{11} + n^{00} + n^{10} + n^{01}$ 이다.

표 12.2 두 개의 분류기 c_i 와 c_k 의 관계

	c_k 맞춤	c_k 틀림
c_i 맞춤	n^{11}	n^{10}
c_i 틀림	n^{01}	n^{00}

12.6.1 다양성 척도

■ Q-통계, 상관 계수, 불일치, 이중 과실

- 두 개 분류기 간의 다양성을 정의하고 그것으로부터 전체 분류기 다양성 측정
- 분류기 i 와 k 간의 다양성

$$\text{Q-통계} \quad q_{ik} = \frac{n^{11}n^{00} - n^{01}n^{10}}{n^{11}n^{00} + n^{01}n^{10}} \quad (12.13)$$

$$\text{상관 계수} \quad r_{ik} = \frac{n^{11}n^{00} - n^{01}n^{10}}{\sqrt{(n^{11} + n^{10})(n^{01} + n^{00})(n^{11} + n^{01})(n^{10} + n^{00})}} \quad (12.14)$$

$$\text{불일치} \quad d_{ik} = \frac{n^{01} + n^{10}}{N} \quad (12.15)$$

$$\text{이중 과실} \quad f_{ik} = \frac{n^{00}}{N} \quad (12.16)$$

12.6.1 다양성 척도

■ Q-통계, 상관 계수, 불일치, 이중 과실 (계속)

□ T 개 분류기 간의 다양성

$$Q = \frac{1}{T(T-1)/2} \sum_{i=1}^{T-1} \sum_{k=i+1}^T q_{ik} \quad (12.17)$$

$$R = \frac{1}{T(T-1)/2} \sum_{i=1}^{T-1} \sum_{k=i+1}^T r_{ik} \quad (12.18)$$

$$D = \frac{1}{T(T-1)/2} \sum_{i=1}^{T-1} \sum_{k=i+1}^T d_{ik} \quad (12.19)$$

$$F = \frac{1}{T(T-1)/2} \sum_{i=1}^{T-1} \sum_{k=i+1}^T f_{ik} \quad (12.20)$$

□ 모두 $[0,1]$ 사이의 값을 가짐

□ 불일치를 빼고 값이 클수록 다양성 떨어짐 (불일치는 반대)

예제 12.4 Q-통계, 상관 계수, 불일치, 그리고 이중 과실

- 세 개 분류기의 10 개 샘플에 대한 인식 결과
 - $n^{11}=6, n^{00}=1, n^{10}=2, n^{01}=1$
- $$\mathbf{y}_1 = (1, 1, 0, 1, 1, 1, 0, 1, 1, 1)^T$$
- $$\mathbf{y}_2 = (1, 0, 1, 1, 1, 1, 0, 1, 0, 1)^T$$
- $$\mathbf{y}_3 = (0, 0, 1, 1, 1, 1, 0, 1, 1, 0)^T$$

- c_1 과 c_2 에 대해 계산해 보면,

$$q_{12} = \frac{6*1 - 1*2}{6*1 + 1*2} = 0.5$$

$$r_{12} = \frac{6*1 - 1*2}{\sqrt{(6+2)(1+1)(6+1)(2+1)}} = 0.21822$$

$$d_{12} = \frac{1+2}{10} = 0.3$$

$$f_{12} = \frac{1}{10} = 0.1$$

- Q-통계를 계산해 보면, (다른 척도도 계산해 보길...)

$$Q = \frac{1}{3(3-1)/2} \sum_{i=1}^{3-1} \sum_{k=i+1}^3 q_{ik} = \frac{q_{12} + q_{13} + q_{23}}{3} = \frac{0.5 + 0.25 + 0.66667}{3} = 0.47222$$

12.6.1 다양성 척도

- 엔트로피, Kohavi-Wolpert 분산, 그리고 평가자 동의

$$\left. \begin{array}{l} \text{엔트로피} \quad E = \frac{1}{N(T - \lceil T/2 \rceil)} \sum_{j=1}^N \min(l_j, T - l_j) \\ \circ \text{이 때 } l_j = \sum_{i=1}^T y_{ji} \end{array} \right\} \quad (12.21)$$

$$\text{Kohavi - Wolpert 분산} \quad V = \frac{1}{NT^2} \sum_{j=1}^N l_j (T - l_j) \quad (12.22)$$

$$\left. \begin{array}{l} \text{평가자 동의} \quad A = 1 - \frac{(1/T) \sum_{j=1}^N l_j (T - l_j)}{N(T-1)p(1-p)} \\ \circ \text{이 때 } p = \frac{1}{NT} \sum_{j=1}^N \sum_{i=1}^T y_{ji} \end{array} \right\} \quad (12.23)$$

예제 12.4의 데이터 $\mathbf{y}_1 = (1, 1, 0, 1, 1, 1, 0, 1, 1, 1)^T$, $\mathbf{y}_2 = (1, 0, 1, 1, 1, 1, 0, 1, 0, 1)^T$, $\mathbf{y}_3 = (0, 0, 1, 1, 1, 1, 0, 1, 1, 0)^T$ 을 재활용하자. 식이 사용하는 l_j , $1 \leq j \leq 10$ 을 먼저 구해보면 아래와 같다.

$$l_1 = 2, l_2 = 1, l_3 = 2, l_4 = 3, l_5 = 3, l_6 = 3, l_7 = 0, l_8 = 3, l_9 = 2, l_{10} = 2$$

$$E = \frac{1}{10(3 - \lceil 3/2 \rceil)} \sum_{j=1}^{10} \min(l_j, 3 - l_j) = \frac{1}{10} (1 + 1 + 1 + 0 + 0 + 0 + 0 + 0 + 1 + 1) = 0.5$$

$$\begin{aligned} V &= \frac{1}{10 * 3^2} \sum_{j=1}^{10} l_j (3 - l_j) \\ &= \frac{1}{90} (2 * 1 + 1 * 2 + 2 * 1 + 3 * 0 + 3 * 0 + 3 * 0 + 0 * 3 + 3 * 0 + 2 * 1 + 2 * 1) = 0.11111 \end{aligned}$$

$$A = 1 - \frac{1/3(2 * 1 + 1 * 2 + 2 * 1 + 3 * 0 + 3 * 0 + 3 * 0 + 0 * 3 + 3 * 0 + 2 * 1 + 2 * 1)}{10(3 - 1)(0.7)(0.3)} = 0.20635$$

12.6.2 선택 알고리즘

- 특징 선택 (9 장)과 유사
 - 따라서 9 장의 알고리즘을 그대로 활용할 수 있음

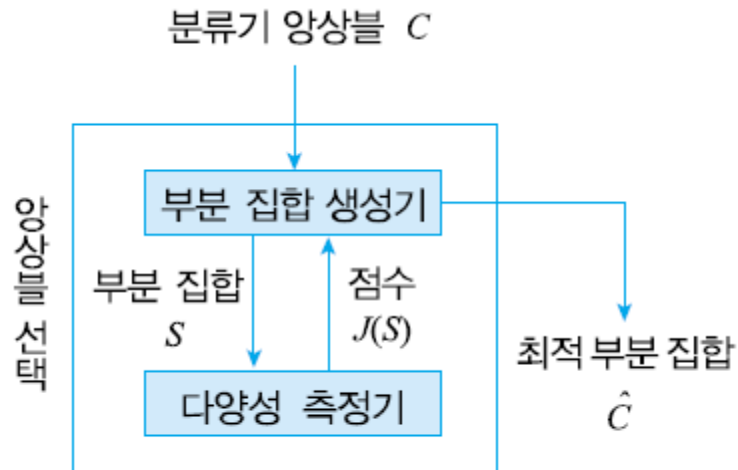


그림 12.9 앙상블 선택 알고리즘의 골격

12.6.2 선택 알고리즘

■ 혼성 유전 알고리즘

- 자식해를 해집단에 넣기 전에 약간 개선함
- 미세 조정력 향상

알고리즘 [12.5]

혼성 유전 알고리즘

1. 초기 해 집단 P 를 설정한다. // P 는 여러 개의 해를 가짐
2. P 의 해들을 평가하여 적합도를 부여한다.
3. **repeat** {
4. P 에서 두 개의 해 $offspring_1$ 과 $offspring_2$ 를 선택한다.
5. $offspring_1$ 과 $offspring_2$ 를 교차시켜 $offspring$ 을 얻는다.
6. $offspring$ 에 변이 연산을 가한다.
7. **improve_by_local_search**($offspring$); // 지역 탐색 연산으로 $offspring$ 을 개선함
8. $offspring$ 을 P 에 대치한다.
9. } **until** (멈춤 조건);

12.7 알고리즘을 바라보는 관점

- 아주 다양한 알고리즘이 있다.
 - 이들은 근본적으로 다른가?
 - 어느 것을 선택하여 사용해야 하나?
- 개성과 통일
 - 알고리즘도 개성이 있다.
 - 특성을 충분히 이해해야 적재적소에 활용할 수 있다.
 - 개성이 다르지만 통일성도 지닌다. 이들에 대한 깊은 이해와 높은 통찰력을 지녀야 새로운 아이디어를 창출할 수 있다.
- 해 보라.
 - 알고리즘 선택에 마술은 없다.
 - 직접 해 보는 것이 최선이다.

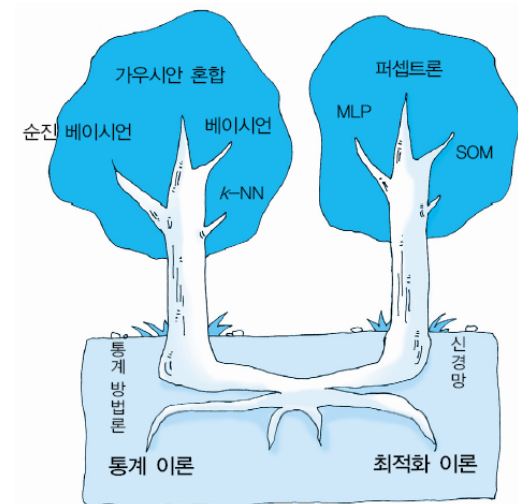


그림 12.10 패턴 인식 알고리즘의 개성과 통일성

Santa Fe Institute

(1984 ~) 미국

지난 수 세기 동안 과학 기술의 경향은 한 마디로 ‘세분화’라고 말할 수 있다. 연구자들의 전공에도 그러한 경향이 나타나 ‘컴퓨터 과학 → 패턴 인식 → HMM



을 이용한 동적 패턴의 인식’ 전공을 가졌다고 말하는 것이 결코 이상해 보이지 않는다. 학문 분야간의 벽도 점점 견고해져 왔다. 하지만 20세기 후반에 이러한 방식의 한계를 깨닫고 그걸 극복하려는 새로운 조류가 나타난다. 이런 조류는 보통 학제간 interdisciplinary, 융합 fusion, 컨버전스 convergence 라는 키워드로 표현된다. 산타페 연구소는 Santa Fe Institute 이러한 학문 분위기를 감지하고 선도적인 연구 그룹으로 1984년 탄생하였다. 그들은 ‘혼성 마인드’를 hybrid mind 강조한다. 그들의 연구 주제는 복잡계 complex systems 이다. 시스템을 구성하는 요소와 그들간의 상호 작용, 그리고 그들이 어떤 메커니즘을 통해 예측하지 못한 현상을 창발 emergence 시키는 지 규명하려는 학문 영역이다. 기존의 뉴턴적 사고는 모든 시스템을 질서라는 눈으로 바라본다. 반면 복잡계 연구에서는 시스템을 무작위와 질서의 긴 스펙트럼 위에 두고 적절한 시점에서 그걸 바라본다. 인터넷이 대표적인 복잡계이다. 네트워크에 연결된 수많은 개인이 일정한 질서 속에서 무작위적으로 보이는 자유로운 행위를 한다. 그들의 행위는 상호 작용을 통해 정치, 경제, 예술, 학문 등에 큰 영향을 미치고 결국 우리 삶을 어떤 방향으로 이끌어 간다. 언뜻 학제간이라는 단어의 표피적인 의미만 보고 산타페 연구소는 여러 분야에서 온 거대한 연구자 집단으로 생각할 수 있다. 하지만 약 35명의 상근 연구원과 60여 명의 외부 교수로 구성된 아주 ‘아담하지만 강한’ 연구 집단이다. 학생들을 위해 다양한 주제의 인턴십과 여름 학교를 개설한다.

패턴 인식과 복잡계는 어떤 관련이 있을까? 복잡계 연구가 패턴 인식 시스템 성능에 기술 돌파를 breakthrough 가져다 줄 수 있을까?