모델 평가 및 선택 (chapter2)

파이널 프로젝트 3주차 스터디

2023.11.24(금) 9:30

김설진

목차

2.1 경험 오차 및 과적합

용어 정리

- 오차율
  - 전체 샘플 수와 잘못 분류한 샘플수의 비율
  - M개의 샘플 중 a개의 잘못 분류된 샘플
  - $\mathbf{E} = {}^{a}/_{m}$
- 정밀도
  - 훈련오차(경험오차) 학습기가 훈련 세트상에서 만들어낸 실제 예측값과 샘플의 실제 값 사이 차이
  - 일반화 오차학습기가 새로운 샘플 위에서 만들어낸 오차
  - '$\mathbf{1}$ – 오차율' => $\mathbf{1} - {}^{a}/_{m}$

* 분류 오차율이 0인, 정밀도가 100%에 달하는 학습기는 좋은 결과를 내지 못함!!

- 과적합
  학습기가 훈련데이터에서 학습을 '과도하게 잘하면', 훈련 데이터 중의 일정한 특성을 모든 데이터에서 내재된 일반 성질이라 오해하게 만드는 것
- 과소적합
  학습기가 훈련 데이터의 일반 성질을 제대로 배우지 못했다는 뜻

예시

나뭇잎을 훈련했을 때,

과적합 모델 분류 결과
=〉 나뭇잎은 끝이 톱니 모양이어야 한다고 잘못 학습
과소적합 모델 분류 결과
=〉 초록색을 모두 나뭇잎으로 인식

2.1 경험 오차 및 과적합

과적합 – 모델 선택 문제

특징 / 성능	과적합	과소적합
원인	학습 능력이 너무 뛰어나 관련 데이터들이 가진 일반적이지 않은 특성까지 학습	좋지 못한 학습 능력
특성	- 까다로우며 피할 수 없음 - 완화하고 위험을 최소화하는 것에 만족해야함	- 의사결정 트리의 경우는 가지치기를 더 진행 - 신경망 학습의 경우에는 epoch수를 늘리면 극복 가능
→ 학습 알고리즘에 따라 과적합을 피할 수 있는 학습 알고리즘과 파라미터를 선택해야함		

모델 선택 문제(model selection 문제)
: 어떤 학습모델과 파라미터를 선택해야하는지 고민하는 문제

2.2 평가 방법

테스트 세트와 오차

- 테스트라는 과정을 통해 학습기의 일반화 오차에 대해 평가를 진행하고 모델을 선택
- **테스트 세트**를 활용해 새로운 샘플에서 어떻게 작동할 지 예측
- 테스트 세트에서 나온 **테스트 오차**를 실제 일반화 오차의 근삿값으로 생각
- 주의점 : 테스트 세트와 훈련 세트의 중복을 최대한 피해야함 / 낙관적일 수 있음

데이터 세트 D를 적절히 처리하여 훈련 세트 S와 테스트 세트 T로 나눠서 훈련해야함

2.2 평가 방법

테스트 세트와 오차

1. 홀드아웃
2. 교차검증
3. 부트스트래핑
4. 파라미터 튜닝과 최종 모델

2.2 평가 방법

홀드아웃

- S : 훈련 세트 집합
- T : 테스트 세트
- $D = S \cup T, S \cap T = x$
- 데이터 세트 D를 겹치지 않는 임의의 두 집합의 합으로 나눔
- S를 통해 훈련된 모델로 T를 이용해 오차측정하고, 일반화 오차에 대한 추정치 제공

주의점

훈련/테스트 세트를 나눌 때 되도록이면 데이터 분포가 같게 나눠야함
그렇지 않으면 데이터 분포의 편향으로 인해 원치 않은 결과를 얻을 수 있음

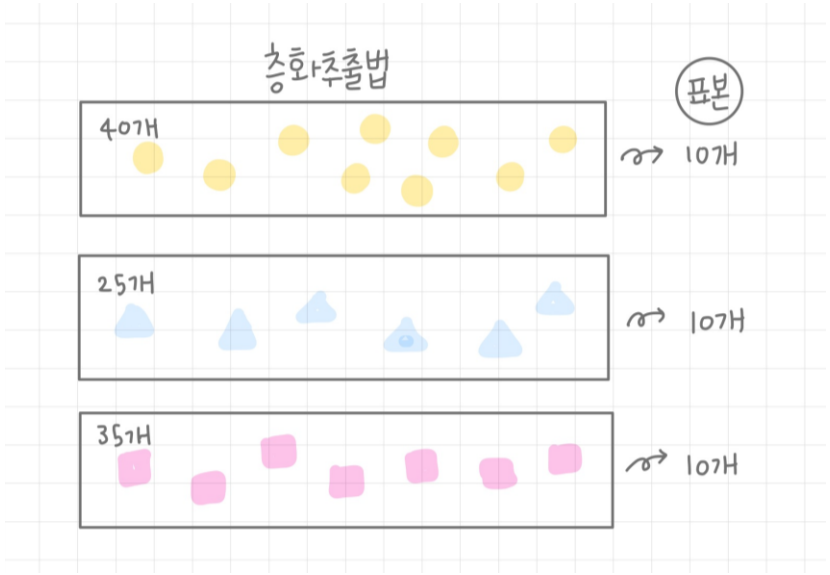
→ 층화추출법 사용

2.2 평가 방법

홀드아웃

층화 추출법

- 모집단을 몇 개의 집단(층)으로 나누고 각 층을 모집단으로 생각하여 어떤 방법으로 미리 할당된 수에 따라 각 층에서 표본을 추출
- 각 샘플의 특징이 같은 것들로 묶어서 층을 나눈 다음 각 층에서 샘플을 추출
- 같은 것으로 묶는 절차가 있기 때문에 집단 내에서 샘플들은 성질이 동일
- 각 집단 간에는 서로 다른 특징을 갖음



출처:data berry

- 만약 T가 적다면, 모델이 D 전체를 훈련하지만, T가 적어 안정적인 평가 결과 불가
- T가 많다면, S와 D의 차이가 높아져, D를 가지고 훈련한 모델과 차이가 증가

=〉 $\frac{2}{3} \sim \frac{4}{5}$ 정도를 훈련 세트로 사용

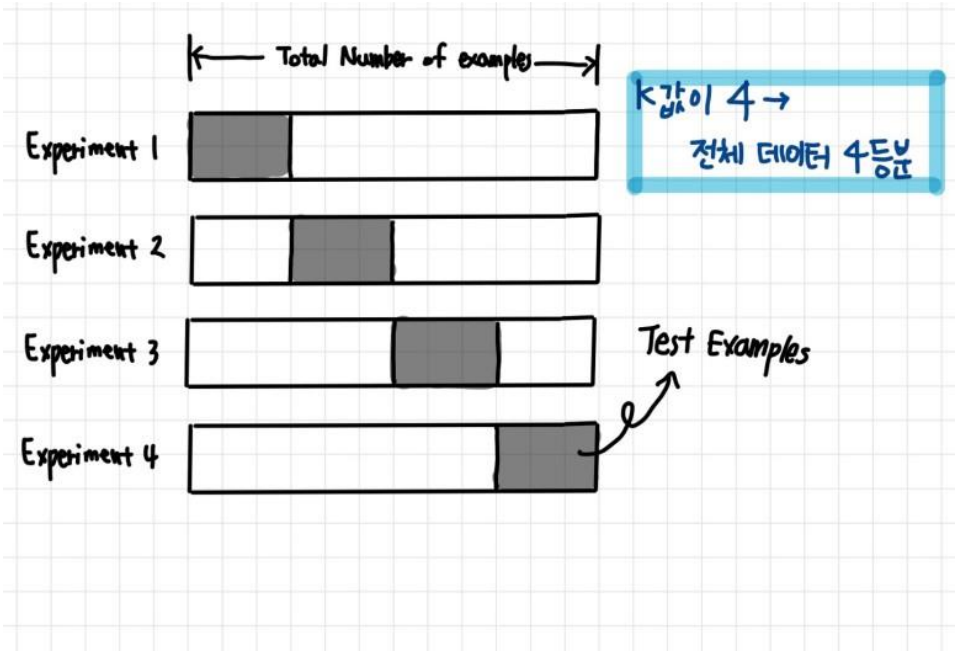
2.2 평가 방법

교차검증

K-fold 교차검증

- 데이터 세트 D를 K개의 서로소 집합으로 만든 것
- $D = D_1 \cup D_2 \cup \cdots \cup D_k$
- $D_i \cap D_j = \emptyset \ (\mathrm{i} \neq \mathrm{j})$
- K-1개의 부분 집합들 : 훈련세트
- 나머지 하나 : 테스트 세트
- K값에 따라 안정성, 정확도가 달라짐

  => K겹 교차검증(k-fold cross validation)
- 일반적으로 10 설정



출처: https://blog.naver.com/nmj936/223154884633

샘플을 나누는 과정에서 생길 수 있는 차별을 최소화하기 위해 k겹 교차 검증은 p번 랜덤 반복하여 나누어 진행 (= 10차 10겹 교차 검증)

2.2 평가 방법

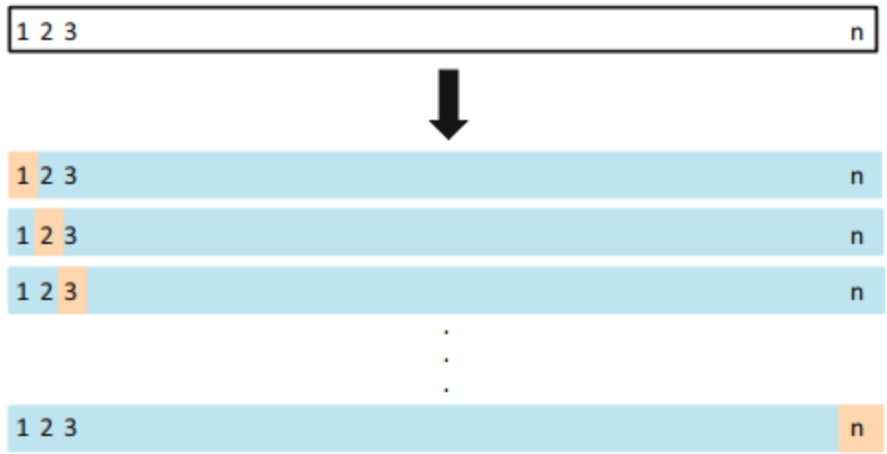
교차검증

LOOCV

- 정의 : m개의 샘플이 있는 데이터 세트 D를 $k = m$으로 설정하고 교차 검증을 실행
- 특징

  - m개의 샘플 분류는 m개의 부분집합을 만들기 때문에 샘플 분류 방법에 대한 영향받지 않음

  - 모든 데이터세트D를 활용해 훈련한 모델과 비슷한 성능
- 장점 : 편향이 작음
- 단점 : D가 매우 클 때, 모델을 m번 적합해서 계산량이 많아짐



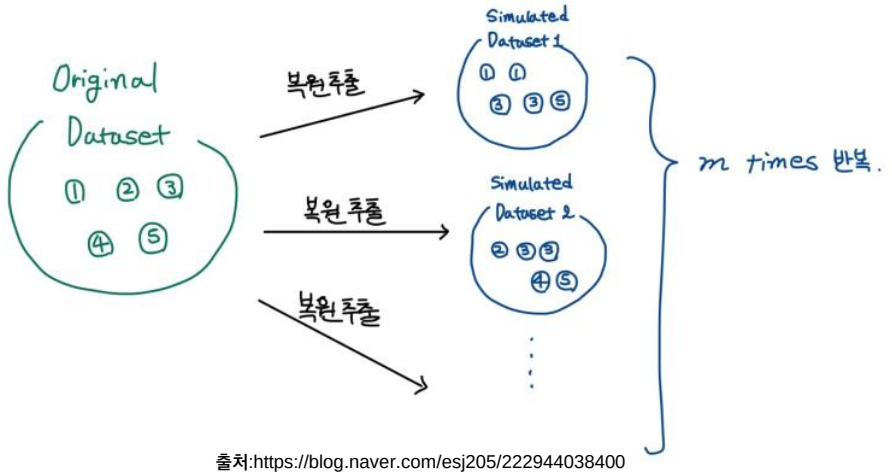
출처:https://deep-learning-study.tistory.com/623

2.2 평가 방법

부트스트래핑

부트스트래핑

- 데이터세트D에서 복원 추출로 m개의 데이터를 m번 반복하여 추출 => D'
- 어떤 샘플은 아예 뽑히지 않을 수도 있음 => 확률은 ((1-1/m)**2)
- $\lim_{n\to\infty}\left(1-\frac{1}{m}\right)^{m}=\frac{1}{e}\approx 0.368$
- 데이터 중 36.8%가 훈련 데이터 세트에 들어가지 못함
- 들어가지 못한 1/3의 샘플들은 테스트 샘플로 활용 => Out-of-Bag 예측



출처:https://blog.naver.com/esj205/222944038400

- 정의 : 부트스트랩 샘플링에 기반을 둔 샘플 추출 기법
- 특징
  - 데이터 세트가 비교적 적거나, 훈련/테스트 세트로 분류하기 힘들 때 사용
  - 앙상블 기법에 적용하기 좋음
  - 생성된 데이터 세트들은 초기 데이터의 분포와 다를 수 있으므로 편향을 크게 만들 수 있음

2.2 평가 방법

파라미터 튜닝과 최종 모델

파라미터 튜닝

모델 평가 및 선택시 학습 알고리즘의 선택뿐 아니라 알고리즘 파라미터에 대한 설정도 고려

알고리즘

- 하이퍼 파라미터라고 함
- 일반적으로 10개 이내

모델

- 개수가 많을 수 있음

⇨

- 모든 파라미터를 모델에 적합시키기 어려움
  => 파라미터의 범위와 변화간격을 설정
- 파라미터 개수를 고려하고 절충을 취해야 좋은 모델을 얻을 수 있음

2.3 모델 성능 측정

성능 측정

성능 측정이란?

- 학습기의 일반화 성능에 대해 평가할 때 유효하고 실험 가능한 테스트 방법뿐만 아니라 모델의 일반화 성능을 평가할 기준

평균제곱오차

- 회귀분석에서 가장 자주 사용하는 성능 측정 방법
- $E(f;D) = \frac{1}{m}\sum_{i=1}^{m}(f(x_i) - y_i)^2$

2.3 모델 성능 측정

오차율과 정확도

오차율

- 모든 샘플 수에서 잘못 분류한 샘플 수가 차지하는 비율
- $E(f;D) = \frac{1}{m}\sum_{i=1}^{m} \Pi(f(x_i) \neq y_i)$

정확도

- 전체 샘플 수에서 정확히 분류한 샘플 수가 차지 하는 비율
- $\mathrm{arr}(f;D) = \frac{1}{m}\sum_{i=1}^{m} \Pi(f(x_i) = y_i)$
  $= 1 - E(f;D)$

혼동행렬

실제 값	예측값	
	양성	음성
양성	TP(실제 양성)	FN(거짓 음성)
음성	FP(거짓 양성)	TN(실제 음성)

2.3 모델 성능 측정

재현율, 정밀도 그리고 F1 스코어

정밀도

- **실제 Negative를 Positive로 잘못 판단하면 큰 문제인 경우**

-> 모델이 True라고 분류한 것 중에서 실제 True인 것의 비율

$$Precision = \frac{TP}{TP + FP}$$

ex) 스팸 음성(일반 메일)을 양성(스팸 메일)으로 판단하여 못읽는 경우

재현율

- **실제 Positive를 Negative로 잘못 판단하면 큰 문제인 경우**

-> 실제 True인 것 중에서 모델이 True라고 예측한 것의 비율

$$Recall = \frac{TP}{TP + FN}$$

ex) 암 양성을 음성으로 판단하면 위험한 경우

⇨

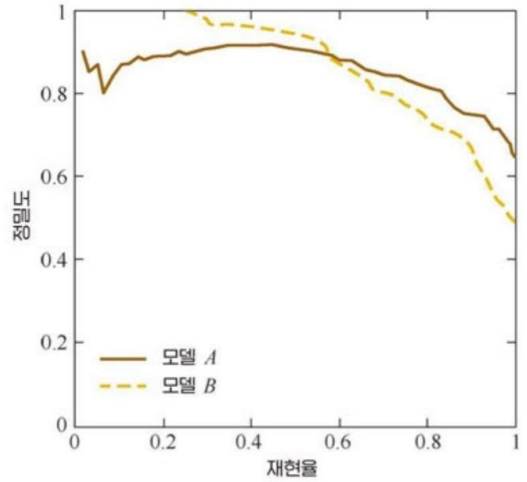
- 정밀도와 재현율 사이에는 트레이드 오프(trade off)가 존재
- 정밀도가 높으면 재현율이 낮음
- 재현율이 높으면 정밀도가 낮음

2.3 모델 성능 측정

오차율과 정확도

P-R 곡선

- 정밀도 precision와 재현율 recall의 관계를 나타내는 그래프(Y축 정밀도, X축 재현율)
- 샘플을 내림차순으로 정렬했을 때 앞쪽부터 양성샘플 가능성이 큼
- 이 순서대로 모든 샘플이 양성값이라 가정하고 예측 진행
- 재현율이 증가함에 따라 정밀도는 전체적으로 감소



1) 재현율이 0에 가까울 때 **모델 A의 정밀도는 0.9이고 모델 B의 정밀도는 1**

2) 재현율이 1일 때 모델 A의 정밀도는 모델 B의 정밀도보다 큼

2.3 모델 성능 측정

오차율과 정확도

손익분기점

- 정밀도 = 재현율
- 간소화한 면이 있어서 F1스코어를 더 많이 사용

F1 스코어

- 재현율과 정밀도의 조화 평균
- $F_\beta = \frac{(1+\beta^2)\times P\times R}{(\beta^2\times P)+R}$
- $\beta = 1$일 때는 일반적인 F1스코어가 됨
- $\beta > 1$ 일 때는 재현율의 영향력이 더 크고, $\beta < 1$일 때는 정밀도의 영향력이 커짐
- $\frac{1}{F1} = \frac{1}{2}(\frac{1}{P} + \frac{1}{R})$

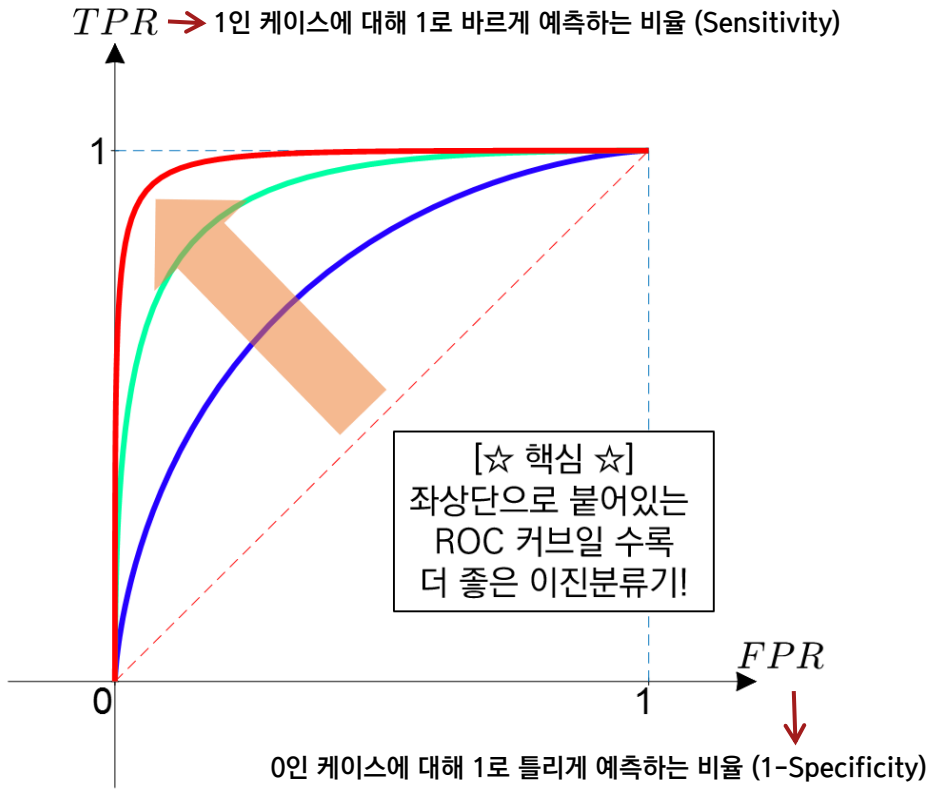
2.3 모델 성능 측정

ROC와 AUC

ROC 곡선

- 배열 순서 자체의 품질에 따라 다른 문제에서 각 학습기의 일반화 성능이 결정하는 도구
- 수신기 조작 특성(Receiver Operating Characteristic)의 약자
- 학습기의 예측 결과를 기반으로 샘플에 대해 순서를 매기고, 해당 순서에 따라 샘플이 양성값이 될 확률을 계산
- TPR과 FPR값을 계산하여 x축(거짓 양성률)과 y축(참 양성률)에 그려 넣으면 ROC 곡선이 완성

$$TPR = \frac{TP}{TP + FN} \qquad FPR = 1 - TNR = \frac{FP}{TN + FP}$$



출처: https://angeloyeo.github.io/2020/08/05/ROC.html#tpr%EA%B3%BC-fpr%EC%9D%98-%EA%B4%80%EA%B3%84

2.3 모델 성능 측정

ROC와 AUC

AUC

- AUC (Area Under the ROC Curve)는 ROC curve의 밑면적
- 1에 가까울수록 그래프가 좌상단에 근접하게 되므로 좋은 모델
- AUC가 고려하는 것은 샘플 예측의 배열 순서 품질 => 순서오차와 긴밀한 관계

$$AUC = \frac{1}{2}\sum_{i=1}^{m-1}(x_{i+1} - x_i)(y_i + y_{i+1}) \ = 1 - l_{rank}$$

l_{rank} 은 ROC곡선 위의 면적

2.3 모델 성능 측정

비용민감 오차율과 비용곡선

비균등비용

- 서로 다른 종류의 오차가 일으키는 서로 다른 종류의 비용에 대한 균형을 맞추기위해 적용하는 개념

비용행렬

- $cost_{ij}$는 I 클래스 샘플이 j 클래스 샘플로 분류될 경우의 비용으로 정의
- $cost_{ii} = 0$
- 만약 0클래스가 1클래스로 분류될 경우 발생하는 손실이 크면 $cost_{01} > cost_{10}$
- 손실이 커질수록 두 값의 차이가 커짐

실제 클래스	예측클래스	
	0 type	1 type
0 type	0	$cost_{01}$
1 type	$cost_{10}$	0

2.4 비교검증

성능비교 및 가설검증이란?

성능비교

1. 우리가 비교하고자 하는 것은 학습기의 일반화 성능
2. 테스트 세트상에서 성능은 테스트 세트 그자체와 큰 상관 관계
3. 모든 학습기는 자체적으로 일종의 무작위성을 포함

통계가설검증

- 학습기 성능을 비교하는데 중요한 근거 제공
- 학습기 A가 B보다 테스트 세트 상에서 성능이 좋다면 A의 일반화 성능이 통계적으로도 B보다 좋은 것인지, 판단이 어느정도로 정확한지 도와줌

2.4 비교검증

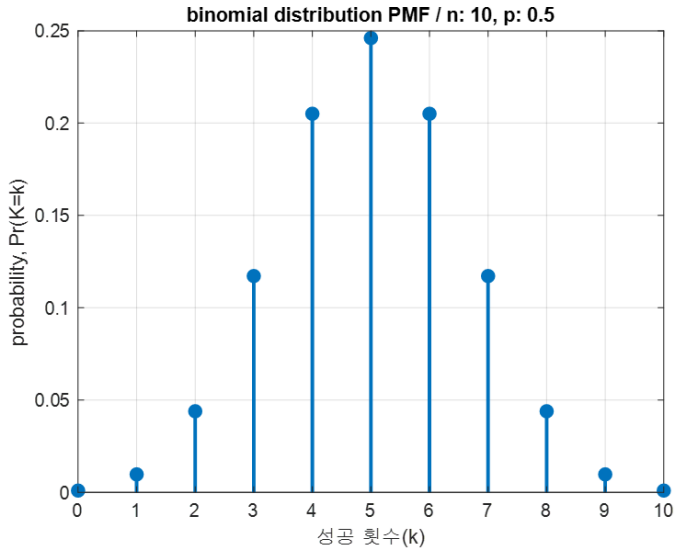
가설검정 – 이항검정

가설
- 학습기 일반화 오차율 분포에 대한 모종의 판단이나 가정
- $\epsilon = \epsilon_0$ (귀무가설)

이항검정

- **결과가 두 가지 값**을 갖는 확률변수의 분포(**이항분포**)를 판단하는데 효과적
- 정규분포는 연속변량인데 반해 이항분포는 이산변량을 따름
- N개의 샘플을 가진 테스트 세트에서 일반 오차율이 p인 학습기를 테스트 했을때 테스트 오차율을 얻을 확률
- $1-\alpha$ 는 결과의 신뢰도를 반영

$$\Pr(K=k) = \binom{n}{k} p^k (1-p)^{n-k}$$



p가 너무 작거나 크면 경우 → 포아송 근사
p가 0.5에서 많이 벗어나지 않는 경우 → 정규근사

2.4 비교검증

가설검정 – 이항검정

이항검정 결론

1. 테스트 오차율이 임계치보다 작다면 유의성 α 하에서 가설 $\epsilon \leq \epsilon_0$은 기각할 수 없고, $1-\alpha$ 의 신뢰도로 학습기의 일반 오차율이 귀무가설보다 작다고 볼 수 있음

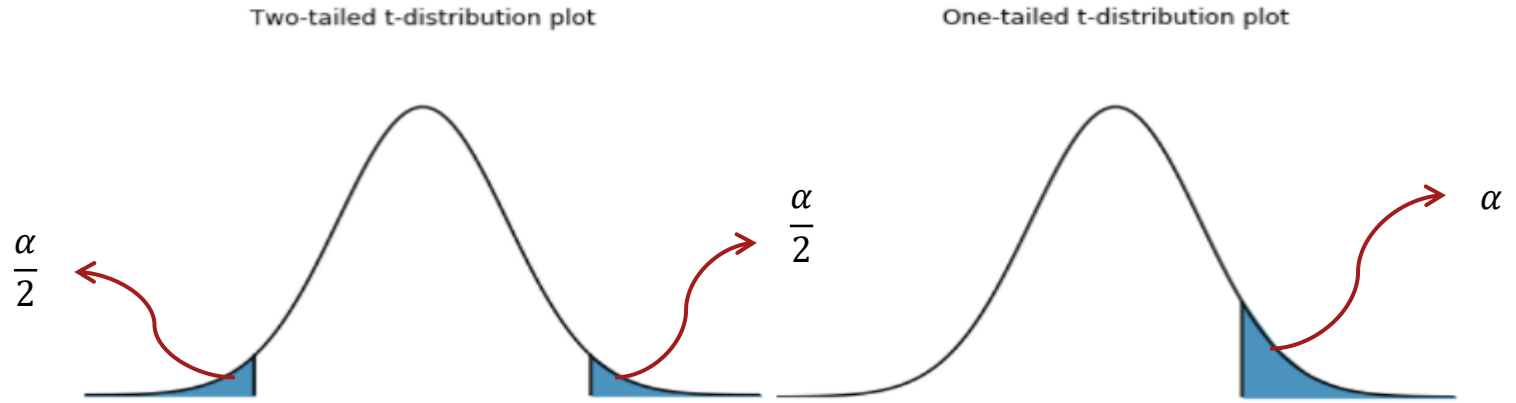
2. 가설이 기각된다면, 유의성 α 하에서 학습기의 일반화 오차보다 크다는 것을 알 수 있음

2.4 비교검증

가설검정 – T검정

T검정(T-test)

- 다양한 검정 세트 기법, 교차 검증법 등 여러 번의 훈련/테스트 세트에 대해 테스트 오차율을 구할 때 사용
- K개의 테스트 오차율 여러 개를 얻는다면 이들의 평균 오차율인 μ와 분산인 σ^2를 구할 수 있음
- K개의 테스트 오차율이 일반화 오차율 $\in_0$의 독립 표본일 때, $T_t = \frac{\sqrt{k}(\mu - \in_0)}{\sigma}$

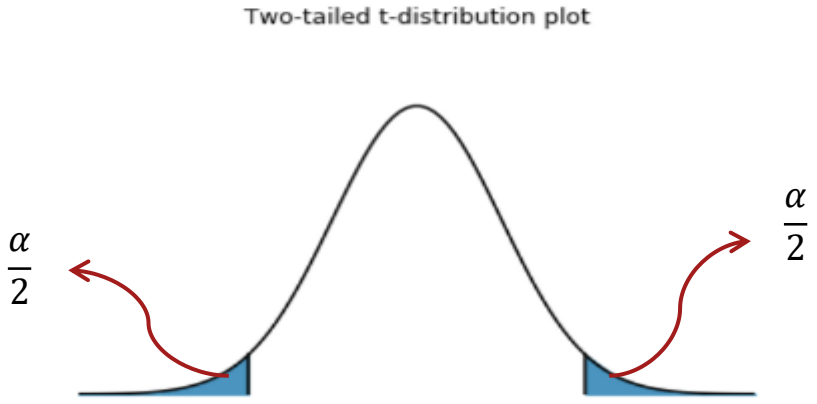


2.4 비교검증

가설검정 – T검정

T검정(T-test)

- 자유도 $k-1$개의 t분포를 따르는 그래프
- 평균값이 $\in_0$일 때, $1-\alpha$ 확률 내에서 관측할 수 있는 최대 오차율
- 임계값을 계산할 수 있음
- 양변에서 음영으로 표시한 영역의 면적은 $\frac{\alpha}{2}$
- 범위는 $(-\infty, t_{-\frac{\alpha}{2}}]$와 $[t_{\frac{\alpha}{2}}, \infty)$



α의 값은 주로 0.05와 0.1을 사용

결론: 평균오차율 μ와 ϵ_0 의 차이 $|\mu-\epsilon_0|$가 임계값 범위 내에 있다면 가설 '$\mu=\epsilon_0$'을 기각할 수 없음

2.4 비교검증

가설검정 – 맥니마 검정

분할표

- 두 학습기 각각의 분류 결과에 대한 차이
- 모두 맞혔을 때, 모두 틀렸을 때, 하나만 맞혔을 때의 수를 표로 나타냄

알고리즘 A	알고리즘 B	
	정답	오답
정답	e_{00}	e_{01}
오답	e_{10}	e_{11}

맥니마 검정

- 자유도가 1인 카이제곱 분포를 따르는 표준정규분포의 제곱

$$T_{x^2} = \frac{(|e_{01} - e_{10}| - 1)^2}{e_{01} + e_{10}}$$

1) 가설 채택: 유의성α에서 위 값이 임계값$(x_\alpha)^2$ 보다 작을 때, 두 학습의 성능에 큰 차이가 없다고 간주
2) 가설 기각: 두 학습기의 성능에는 큰 차이가 있다고 간주하고, 평균 오차율이 비교적 작은 학습기의 성능이 뛰어나다고 판단

2.4 비교검증

가설검정 – 프리드먼 검정/네메니 사후 검정

프리드먼 검정

- 알고리즘을 비교할 때 알고리즘에 등수를 매기는 검정

1) 알고리즘의 테스트 결과 얻기
2) 데이터 세트별로 순위 매기기
3) 좋은 순서대로 값 부여(성능이 같다면 평균값 부여)
4) 마지막엔 알고리즘별 평균값 구하기
5) 프리드먼 검정을 사용하여 알고리즘들의 성능이 같은지 판단

=> 성능이 같다면 알고리즘들의 평균값이 같을 것

네메니 사후 검정

- 모든 알고리즘의 성능이 같다는 가설이 기각되고,

=> 알고리즘들의 성능이 다르다는 것을 설명하기 위해 추가 검정시 사용

- 평균값 차이의 임계값 영역을 계산

2.5 편향과 분산

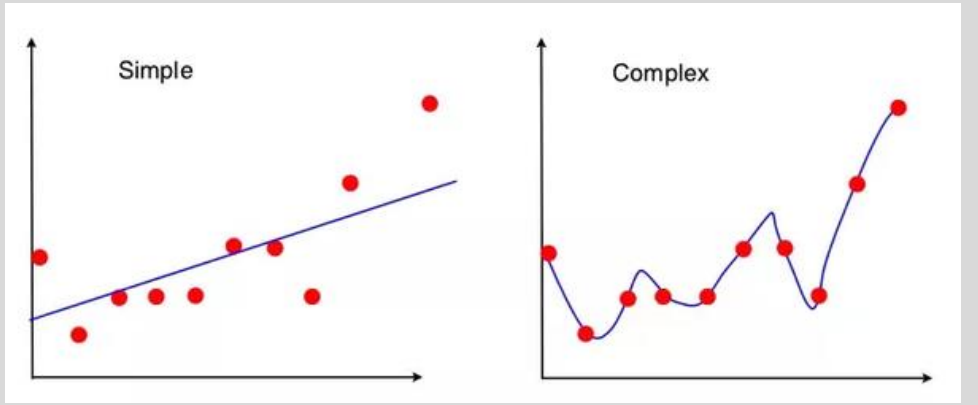
편향-분산 분해

편향-분산 분해

- 알고리즘의 일반화 성능을 해석할 수 있는 중요한 도구
- 학습 알고리즘의 기대 일반화 오차를 분해

$$E(f;D) = bias^2(x) + var(x) + \varepsilon^2$$

- 일반오차는 편향, 분산, 노이즈의 합으로 분해 가능

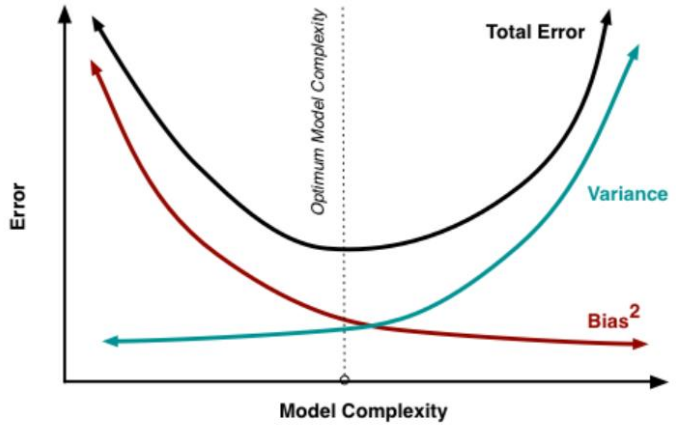


큰 편향, 작은 분산

편향이 크면 과소 적합

작은 편향, 큰 분산

분산이 크면 과대 적합



편향-분산 Trade-off

- 모델이 복잡해질 수록 편향은 작아지고, 분산은 커지며 over-fitting 됨
- 모델이 단순해질수록 편향은 커지고, 분산은 작아지며 under-fitting 됨
- 오류를 최소화하려면 편향과 분산의 합이 최소가 되는 적당한 지점을 찾아야 함

감사합니다.
피드백부탁드려요!