

Motivation

1

동기

- 배경지식
 - 기존 통계학적 모델의 경우 MVUE를 목적으로 함
 - MVUE: Minimum Variance under unbiased estimator(최소분산 비편향 추정량)
 - 추정값이 비편향이지만, 성능이 떨어진다는 단점
 - 머신러닝의 경우 Minimize overall Error
 - 전반적인 error가 감소하면서 성능은 뛰어나지만 편향된 추정을 한다는 단점
 - 따라서 머신러닝에서는 단순히 성능을 높이는 것뿐만 아니라 정규화(Regularization)을 진행하는 것이 필요함
 - 단순 훈련 데이터뿐만 아니라 전반적인 상황에서도 잘 작동하는 일반 구조를 알아내기 위함

The Goal of Cross-Validation

교차-검증의 목표

• 모델을 만드는 목적

- 일반 구조를 알아내기 위함
- 머신러닝의 경우 훈련 데이터셋은 충분히 잘 모사할 수 있음
 - 하지만 이것이 일반구조를 표상한다고 주장할수는 없음
 - 현데이터에만 적합했을 위험이 있기 때문
- CV는 IID 프로세스를 통해 추출된 관측 자료를 훈련 집합과 테스트 집합으로 나눔
 - 훈련 집합에 있는 정보가 테스트 집합에 들어가는 것을 방지해야 함

K-Fold CV

- 알고리즘
 - 1. 데이터 세트를 k개의 부분 집합으로 분할
 - 2. I = 1,···,k에 대해
 - (a) 머신러닝 알고리즘이 i를 제외한 모든 부분집합에 대해 훈련한다
 - (b) 적합화된 머신 러닝 알고리즘은 i에 테스트함

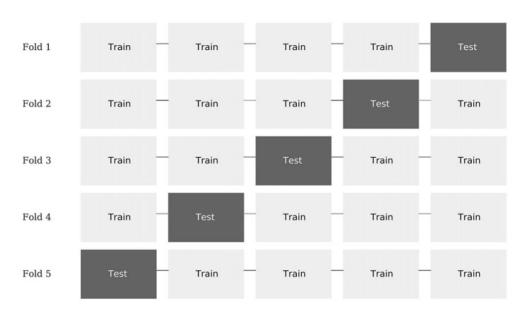


FIGURE 7.1 Train/test splits in a 5-fold CV scheme

교차-검증의 목표

- 금융에서 CV의 활용 현황
 - 1. 모델의 개발
 - Hyper Parameter Tuning
 - 본 장의 목적
 - 2. 백테스팅
 - 10장부터 16장에서 다룰 내용

Why K-Fold CV Fails in Finance

금융에서 K-폴드 교차 검증이 실패하는 이유

• 문제 사항

- 1. 관측값이 IID 프로세스에서 추출되었다는 가정 할 수 없음
 - 계열 상관된 특징 X가 중첩된 데이터에서 형성된 레이블 Y와 상관되어 있음
 - 계열 상관관계에 의해 $X_t \approx X_{t+1}$, 레이블이 중첩된 데이터 포인터에서 유도됐으므로 $Y_t \approx Y_{t+1}$
 - 이 때 t와 t + 1을 다른 집합에 두면 정보가 누출됨
 - 정보 누수 해결 문제
 - 1. Y_i 가 Y_i 를 결정하기 위해 사용된 정보 함수고, j가 테스트 집합에 속할 때 모든 관측값 i를 훈련 집합에서 퍼지함
 - Y_i 와 Y_i 는 중접된 기간이 없어야 함
 - 2. 분류기의 과적합을 피해야 함
 - 기저 추정기의 조기 종료(6장)
 - 중복된 예제에 대한 과표본을 통제하면서 분류기를 배깅해 개별 분류기가 최대한 다양해질 수 있도록 함
 - → Max_samples를 평균 고유성으로 설정
 - → 순차적 부트스트랩 적용(4장)
- 2. 테스트 집합이 모델 개발 과정 프로세스에서 여러 번 사용되었기 때문
 - 다중 테스트, 선택의 편향 초래(11~13장)

A Solution:Purged K-Fold CV

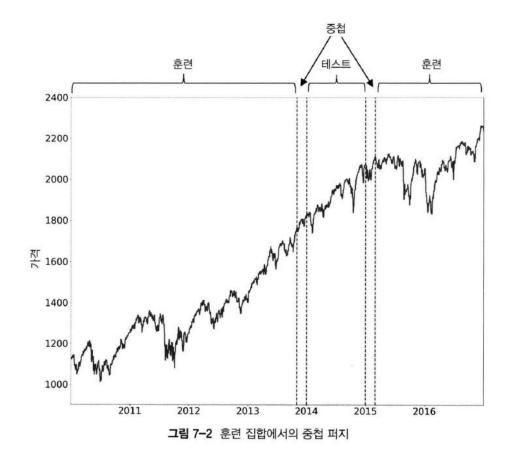
• 엠바고 프로세스

- 훈련 집합에서 테스트 집합에 있는 관측값을 그대로 따르는 관측값을 퍼지해야 함
- 1. 훈련 집합에서의 퍼지
 - **가정** : Y_i인 테스트 관측값이 정보 집합 Φ_i에 의해 결정
 - 목적
 - 훈련 집합에서 정보 집합 Φ_i 에 근거한 Y_i 의 모든 값을 퍼지하여 $\Phi_i \cap \Phi_i = \emptyset$ 를 만듦
 - 중첩 여부 판단
 - $Y_j = f\left[\left[t_{j,0},t_{j,1}\right]\right]$ 에 대하여 $Y_i = f\left[\left[t_{i,0},t_{i,1}\right]\right]$ 은
 - 1. $t_{j,0} \le t_{i,0} \le t_{j,1}$
 - 2. $t_{j,0} \le t_{i,1} \le t_{j,1}$
 - 3. $t_{i,0} \le t_{i,0} \le t_{i,1} \le t_{i,1}$ (포함)
 - 일 경우 중첩된다

```
def getTrainTimse(t1, testTimes):
   주머진 TestTimes에 대해 훈련 관측값의 시간을 찾음
   input :
       - t1
          - t1.index : 관측 시작 시간
          - t1.values : 관측 종료 시간
       - testTime
          테스트 관측 시간
   trn = t1.copy(deep = True)
   for i,j in testTimes.iteritems():
       # 테스트 내 훈련 시작
       df0 = trn[(i<=trn.index)&(trn.index<=j)].index</pre>
       # 테스트 내 훈련 종료
       df1 = trn[(i <= trn) & (trn <= j)].index
       # 훈련이 테스트 포함
       df2 = trn[(trn.index<=i)&(j<=trn)].index</pre>
       trn=trn.drop(df0.union(df1).union(df2))
   return trn
```

• 엠바고 프로세스

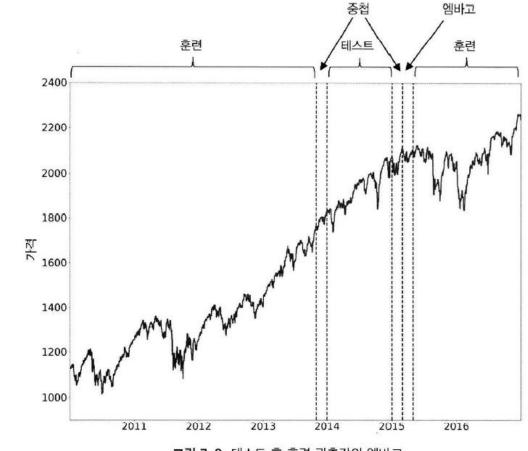
- 훈련 집합에서 테스트 집합에 있는 관측값을 그대로 따르는 관측값을 퍼지해야 함
- 1. 훈련 집합에서의 퍼지
 - 가정 : Y_i 인 테스트 관측값이 정보 집합 Φ_i 에 의해 결정
 - 목적
 - 훈련 집합에서 정보 집합 Φ_i 에 근거한 Y_i 의 모든 값을 퍼지하여 $\Phi_i \cap \Phi_i = \emptyset$ 를 만듦
 - 중첩 여부 판단
 - $Y_j = f\left[\left[t_{j,0},t_{j,1}\right]\right]$ 에 대하여 $Y_i = f\left[\left[t_{i,0},t_{i,1}\right]\right]$ 은
 - 1. $t_{j,0} \le t_{i,0} \le t_{j,1}$
 - 2. $t_{j,0} \le t_{i,1} \le t_{j,1}$
 - 3. $t_{i,0} \le t_{j,0} \le t_{j,1} \le t_{i,1}$ (포함)
 - 일 경우 중첩된다



- 엠바고 프로세스
 - 2. 엠바고
 - 목적
 - 퍼지로도 누수를 방지하지 못할 때 설정
 - 훈련 레이블 $Y_i = f\left[\left[t_{i,0},t_{i,1}\right]\right]$ 은 $t_{i,1} < t_{j,0}$ 테스트 시간 $t_{j,0}$ 에 있었던 정보 포함
 - $t_{j,1} \leq t_{i,0} \leq t_{j,1} + h$ 에 발생하는 훈련 레이블 $Y_i = f\left[\left[t_{i,0,},t_{i,1}\right]\right]$ 에만 관련

```
def getEmbargoTimes(times, pctEmbargo):
## 学 好例 대意 塑料고 시간 劃与
step=int(times.shape[0]*pctEmbargo)
if step==0:
    mbrg=pd.Series(times, index =times)
else:
    mbrg=pd.Sereis(times[step:], index=times[:-step])
    mbrg=mbrg.append(pd.Sereis(times[-1],index=times[-step:]))
return mbrg
```

- 엠바고 프로세스
 - 2. 엠바고
 - 목적
 - 퍼지로도 누수를 방지하지 못할 때 설정
 - 훈련 레이블 $Y_i = f\left[\left[t_{i,0},t_{i,1}\right]\right]$ 은 $t_{i,1} < t_{j,0}$ 테스트 시간 $t_{j,0}$ 에 있었던 정보 포함
 - $t_{j,1} \leq t_{i,0} \leq t_{j,1} + h$ 에 발생하는 훈련 레이블 $Y_i = f\left[\left[t_{i,0,},t_{i,1}\right]\right]$ 에만 관련



Bugs in Sklearn's Cross-Validation

Implementation

- 문제점 1.
 - Score 함수가 pandas series가 아니라 numpy 배열로 구현되어 있기 때문에 classes_를 알지 홋함
- 문제점2
 - cross_val_score는 가중값을 fit 메서드렝 전달하고, log_lss에 전달하지 않으므로 다른 결과 산출

