Chapter 5

Resampling Methods (재표본추출)

염정운

5장은 재표본추출 방법에 대해 다루고 있습니다. 재표본추출이란 이미 가지고 있는 데이터에 반복해서 표본을 추출함으로써 추가적인 정보를 획득하는 과정이라고 할 수 있겠습니다.

ISL에서는 크게 **Cross Validation**(교차검증)과 **Bootstrap**를 다룹니다.

1. Cross-Validation (교차검증)

모델을 적합한 Training error rate과 적합된 모델에 새로운 데이터셋을 적용했을 때 관측되는 Test error rate는 차이가 존재합니다(2.2.1). Training error rate을 최소화하는 과정(*least square method etc...*)에서 실제 Test error rate을 크게 과소추정할 수도 있습니다. Cross-Validation은 이러한 문제를 해결하기 위해 사용합니다.

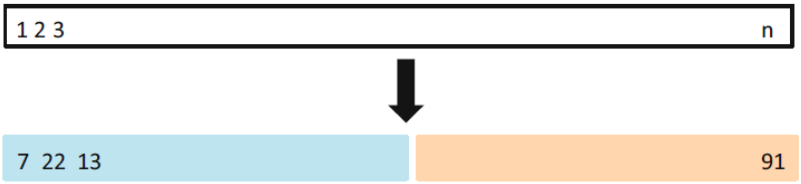
근본적으로 Cross-Validation은 **Test error rate을 적절히 추정**하는 것에 목적이 있다고 이해할 수 있겠습니다.

5.1 절에서는 세 가지 교차검증 기법(Validation Set Approach, LOOCV, K-Fold)에 대해 알아보고 가장 널리 사용되는 K-Fold 방식의 Variance-Bias Trade off 특징과 Classification 문제에서의 Cross-Validation 적용을 알아봅니다.

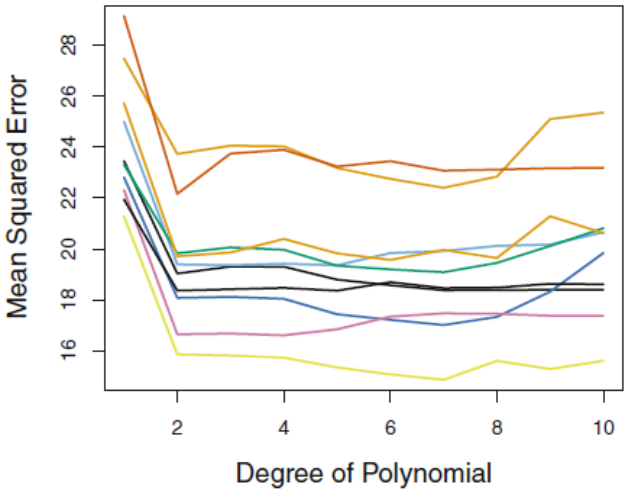
*5.1.1~5.1.4절에서 Regression 상황을 가정하나, Cross-Validation은 Classification에서도 동일한 개념으로 적용된다고 밝혀져 있습니다.

1.1 Validation Set Approach (검증 셋 기법)

총 n개의 관측치(obs)를 가진 Data Set을 생각해 봅시다. Validation Set Approach는 다음과 같이 표현할 수 있습니다.



전체 Data Set을 임의로 Training Set과 Validation Set으로 나눕니다. Training Set으로 모델을 적합하고, Validation Set으로 Test MSE를 추정합니다. 책에서는 분할을 10번 시도해보았고, 각각에 대해 MSE 값을 도식해본 결과는 아래와 같습니다.



Validation Set Approach는 쉽고 간단하지만 두가지 결점을 가집니다.

1. Data Set을 임의로 분할하기에 관측이가 나뉘어지는 것에 따라 추정치의 변동이 심하다

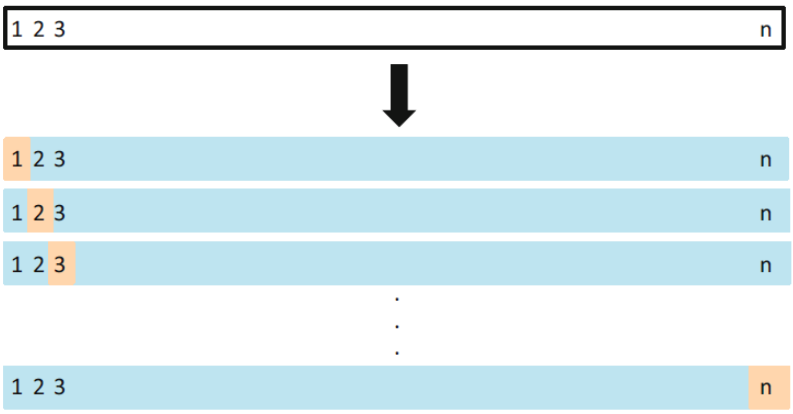
2. 모델 적합에 Training Set만 사용됨에 따라 실제 학습되는 관측치 수가 적어 Test error rate이 과대 추정 될 수 있다.

1.2 LOOCV (Leave-One-Out Cross-Validation)

이 방식은 위 Validation Set Approach의 결점을 극복하려고 합니다. 우선 Data Set에서 임의로 추출한 단 하나의 관측치만 Validation Set으로 활용하고, 나머지 n-1개 관측치를 Training Set으로 모델을 학습시킵니다. 이때 Validation Set을 적용해 임의의 관측치 i 에 대한 $MSE_i = (y_i - \hat{y}_i)^2$가 구해지고, 이를 n개의 관측치에 대해 전부 시행한 후 이를 평균을 내어 추정치를 구합니다. LOOCV의 Test error rate 추정치는

$$\mathrm{CV}_{(n)} = \frac{1}{n}\sum_{i=1}^{n}\mathrm{MSE}_i.$$

와 같이 작성될 수 있습니다. 이를 도식화 하면 아래와 같습니다.



이 때 모델을 관측치 수만큼 적합 시키기 때문에 이 방법을 여러 번 시도해도 항상 같은 추정치 결과를 얻게 됩니다. 그러나 관측치 n이 아주 커질 때에는 물리적인 제약(시간과 기계의 성능 등)이 제약이 발생할 수 있습니다.

OLS를 통한 단순/다중선형회귀를 진행하는 경우에 한해 시간적 제약을 쉽게 하는 방법은 Leverage(3.37 참고)를 이용하는 방법이고, 그 방식은 아래와 같습니다.

$$\mathrm{CV}_{(n)} = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{y_i - \hat{y}_i}{1 - h_i}\right)^2,$$

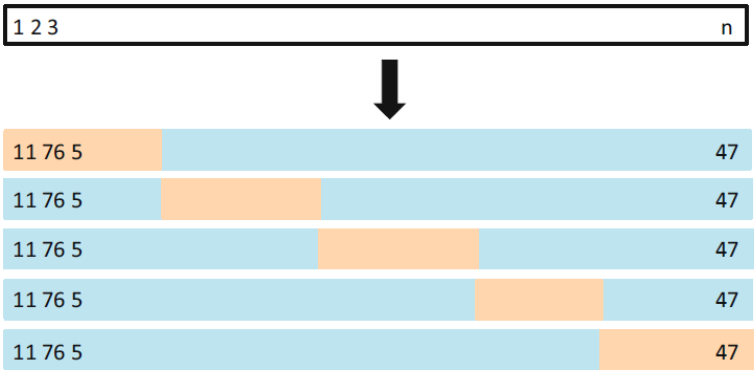
1.3 K-Fold Cross Validation (K-Fold 교차검증)

LOOCV의 한계에 대한 대안 방식입니다. n개의 관측치를 가진 전체 Data Set을 크기가 거의 동일한 k개의 Data Set으로 분할합니다. K개의 Set 중 임의의 하나를 Validation Set으로 취급하고, k-1개 Set에 대해 Training을 진행합니다. 임의의 i번째 Validation Set에 대해 $MSE_i = (y_i - \hat{y}_i)^2$가 구해지고, 이 작업을 k번 반복할 수 있습니다.

K-Fold를 이용한 Test error 추정치는 이 k개의 MSE 값들을 평균 내어 계산합니다.

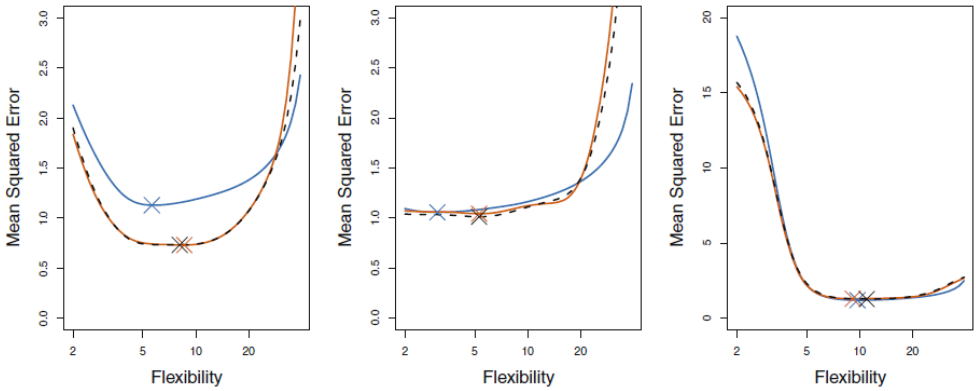
$$\mathrm{CV}_{(k)} = \frac{1}{k}\sum_{i=1}^{k}\mathrm{MSE}_i.$$

단순하게 생각하면 개별 관측치에 대해 MSE를 일일이 계산하는 LOOCV를 조금 더 뭉텅뭉텅하게 만든 방법이라고 이해하면 편할 것 같습니다.



아래 세 그래프는 특정 모의 Data Set에 실제 Test MSE(파란색), LOOCV 추정 MSE(검정색 파선), 10-Fold 추정 MSE(오렌지색)을 도식한 결과입니다.

10-Fold 추정 MSE가 LOOCV 추정과 크게 다르지 않으며, 중앙과 오른쪽 그림의 경우 실제 Test MSE 추세와 상당히 유사한 모습을 보여주는 것을 알 수 있습니다.



▣ Cross-Validation 수행하는 이유 ▣

1. 실질적인 Test MSE의 추정
2. 최소의 Test MSE가 도출되는 위치 추정(모델 유연성 수준 식별)

1.4 Bias-Variance Trade-off for K-Fold Cross Validation

K-Fold Cross Validation은 LOOCV에 비해 가지는 우위는 다음과 같습니다.

1. 연산 수준
2. Bias-Variance Trade-Off로 인한 더 정확한 Test error rate 추정

n-1개의 관측치에 대해 Training을 진행하는 LOOCV는 Low Bias-High Variance를 가질 것이고, 반대로 Validation Set Approach의 경우 반대로 High Bias-Low Variance 특성을 보일 것입니다. 그리고 K-Fold CV는 그 사이 약 $\frac{(k-1)n}{K}$ 개의 관측치를 Training 하므로 **Bias와 Variance가 절충**되어 나타나게 됩니다.

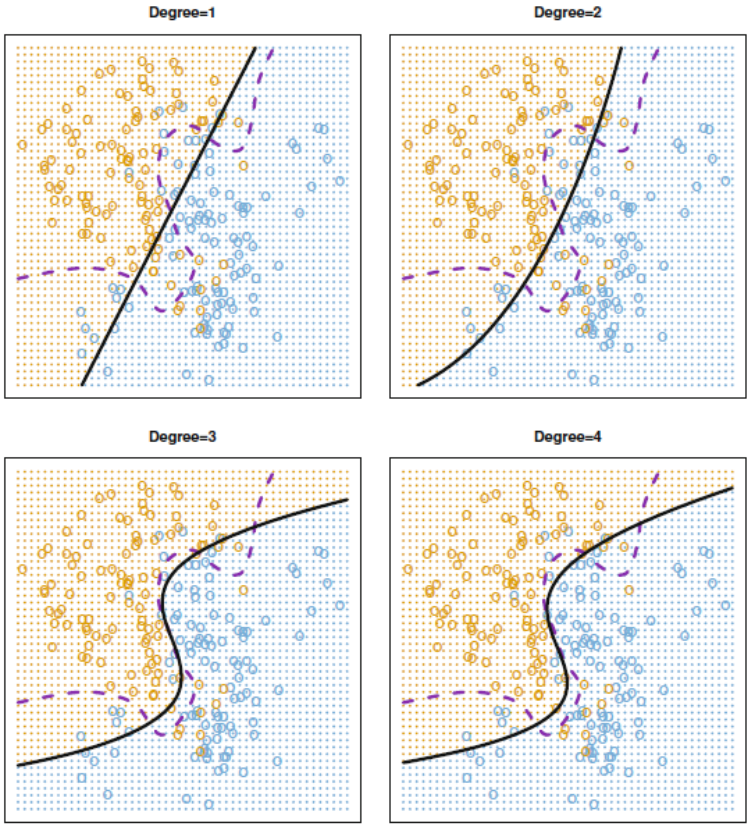
경험적으로, k=5 혹은 k=10에서 적당히 절충되어 합리적인 Test error rate 추정치를 얻을 수 있다는 것이 알려져 있다고 합니다.

1.5 Cross-Validation on Classification Problem

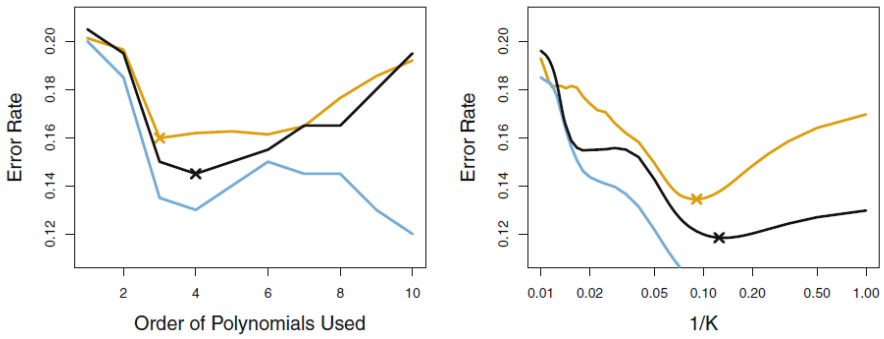
Classification 상황에서 Cross-Validation은 MSE 등 Test Error rate이 아닌 잘못 분류 된 관측치(오분류율)을 추정하려 합니다. LOOCV에서의 오차율은 이하와 같이 표현됩니다.

$$\mathrm{CV}_{(n)} = \frac{1}{n}\sum_{i=1}^{n}\mathrm{Err}_i, \qquad \text{where } \mathrm{Err}_i = I(y_i \neq \hat{y}_i)$$

여러가지 분류 모델을 사용할 때, 어떤 모델을 분류기로 사용해야 하는가 판단해야 할 때, Cross-Validation을 통해 결정할 수 있습니다. 로지스틱 회귀 모델을 차수별로 만든다고 가정해 봅시다.



모의 Data Set의 경우 위와 같이 베이즈 결정 경계(점선)와 각 모델 별 분류 경계(실선)를 확인할 수 있습니다만 실제 데이터의 경우는 그렇지 않습니다. 이 때 차수별로 오분류율을 도식해보면 아래와 같습니다.



좌측은 로지스틱 회귀에서 차수별 오분류율을, 우측은 KNN 분류에서 K값 변화에 따른 오분류율을 나타냅니다. Test 오분류율은 갈색(실제 상황에서는 알 수 없는 값), Training 오분류율은 파란색, 10-Fold CV 오분류율은 검은색으로, 10-Fold 오분류율이 Test 오분류율에 잘 근사됨을 알 수 있습니다.

2. Bootstrap

Bootstrap은 실제로는 알기 어려운 추정량(모집단의 평균, 분산 etc)들의 불확실성을 계산하는데 널리 쓰이는 강력한 기법입니다. Cross Validation과 비교하여 다음과 같은 특징을 가집니다.

- Cross Validation uses sampling *without replacement*
  - The same instance, once selected, can not be selected again for a particular training/test set
- The *bootstrap* uses sampling *with replacement* to form the training set
  - Sample a dataset of *n* instances *n* times *with replacement* to form a new dataset of *n* instances
  - Use this data as the training set
  - Use the instances from the original dataset that don't occur in the new training set for testing

책에서 나온 예시를 통해 확인해봅시다. 임의의 확률변수 쌍 (X,Y)을 생각해봅시다. X, Y에 자본을 나누어 투자하는데, 투자 이익에 관한 변동을 최소화하는(위험을 최소화하는) 비율로 X에 α만큼, Y에 $1-\alpha$만큼 투자를 하고, $\min(\ Var(\alpha X + (1-\alpha)Y)\ )$ 를 만족하는 α를 찾고자 합니다. 이 때 수식을 정리하여 미분하면 조건을 만족시키는 α는 다음과 같습니다*.

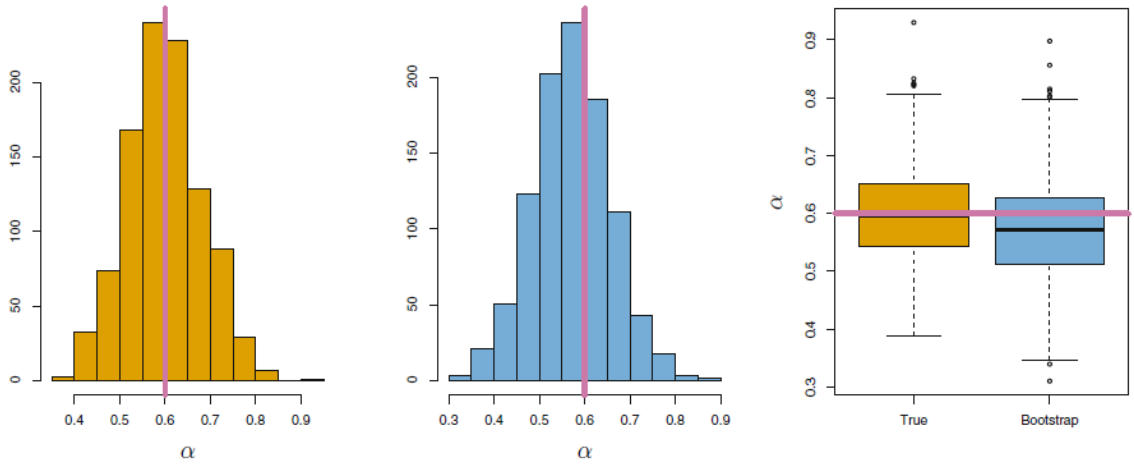
$$\alpha = \frac{\sigma_Y^2 - \sigma_{XY}}{\sigma_X^2 + \sigma_Y^2 - 2\sigma_{XY}},$$

이 때 모집단 X, Y의 분산과 공분산은 알 수 없는 값입니다. 때문에 모집단의 표본으로부터

추정값인 $\hat{\alpha}$를 구합니다.

$$\hat{\alpha} = \frac{\hat{\sigma}_Y^2 - \hat{\sigma}_{XY}}{\hat{\sigma}_X^2 + \hat{\sigma}_Y^2 - 2\hat{\sigma}_{XY}}.$$

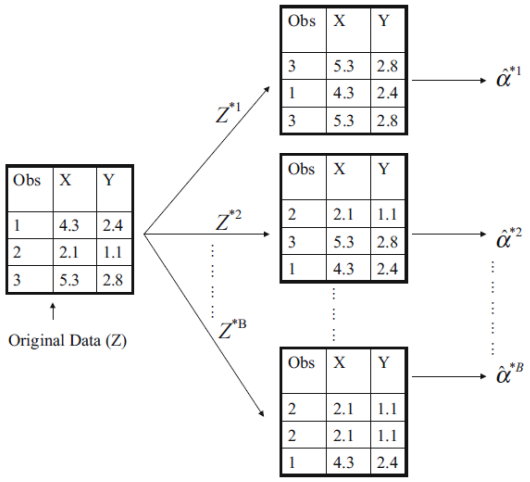
1000개의 (X,Y) 표본을 가지고, 이 중에서 임의로 **복원추출 을 허용하여** 100개의 (X,Y) 표본을 뽑아 $\hat{\alpha}$를 구하는 작업을 1000번 반복합니다. 모집단으로부터 1000개 모의 Data Set을 생성하여 얻은 $\hat{\alpha}$의 히스토그램과(좌측) 앞선 Bootstrap 방식을 통해 총 1000개의 $\hat{\alpha}$에 대한 히스토그램(중앙)은 다음과 같습니다.



모집단으로부터 1000개의 Data Set을 생성하여 얻은 왼쪽 히스토그램과, 하나의 Data Set을 Bootstrap을 이용하여 얻은 히스토그램이 상당히 유사한 모양을 보여주고 있습니다.

실제 최적 α 값이 0.6임을 고려했을 때 추정치의 평균이 실제 최적 값에 상당히 근사하는 정규분포 형태의 히스토그램을 보여주고 있습니다. 또한 $\hat{\alpha}$의 표준편차는 0.083으로 이는 α와 $\hat{\alpha}$가 평균적으로 0.08만큼 차이를 보인다고 판단할 수 있습니다.

n=3의 작은 표본에서 Bootstrap 과정을 간단히 도식하면 아래와 같습니다.



하나의 데이터셋 Z로부터 B번의 반복추출을 통해 B개의 $\hat{\alpha}$를 형성하며, 이에 대한 표준 오차는 다음과 같이 계산됩니다.

$$\mathrm{SE}_B(\hat{\alpha}) = \sqrt{\frac{1}{B-1}\sum_{r=1}^{B}\left(\hat{\alpha}^{*r} - \frac{1}{B}\sum_{r'=1}^{B}\hat{\alpha}^{*r'}\right)^2}.$$

Bootstrap 방식은 마치 무에서 유를 만들어내는 것과 같은 느낌을 주지만, 검증된 강력한 통계분석 방법론이라고 합니다.

참고로 ML 모델 앙상블 기법 중 하나인 bagging은 이 bootstrap을 활용한 방식으로, bootstrap aggregating의 준말입니다.

Bagging: (Full name: bootstrap aggregating)동일한 모델을 random한 training data subset에 fitting하여 여러개의 모델을 얻는것 (training data sampling시 **복원추출 허용**)

---

* Proof)

$$\mathrm{Var}(\alpha X + (1-\alpha)Y) = \alpha^2 \times \sigma_X^2 + (1-\alpha)^2 \times \sigma_Y^2 + 2\,\alpha(1-\alpha)\sigma_{XY}$$

$$\frac{\partial}{\partial \alpha}\left(\mathrm{Var}(\alpha X + (1-\alpha)Y)\right) = 2\alpha\,\sigma_X^2 - 2\sigma_Y^2 + 2\alpha\,\sigma_Y^2 + 2\sigma_{XY} - 4\alpha\sigma_{XY}$$

$$2\alpha\,\sigma_X^2 - 2\sigma_Y^2 + 2\alpha\,\sigma_Y^2 + 2\sigma_{XY} - 4\alpha\sigma_{XY} = 0$$

$$\alpha = \frac{\sigma_Y^2 - \sigma_{XY}}{\sigma_X^2 + \sigma_Y^2 - 2\sigma_{XY}}$$