Seoul National University
College of Engineering
Department of Naval Architecture and Ocean Engineering
1, Gwanak-ro, Gwanak-gu, Seoul 151-744, Korea

Fall 2020

머신러닝

PA # 5

Instructor name	김태완 교수님
Student name	이용준
Department	조선해양공학과
Student ID	2015-19595
Submission date	2020.12.04

Contents

1. Problem Definition ---------- 3

2. Problem Analysis ---------- 3

3. Code Explanation ---------- 4

4. Result & Conclusion ---------- 4

4.1 Result ---------- 4

4.2 Conclusion ---------- 7

1. Problem Definition

Drag.csv :

Types	Column	Explanation
INPUT	Age	환자의 나이
	Sex	환자의 성별
	BP	환자의 혈압(blood pressure)
	Cholesterol	환자의 콜레스테롤 수치
	Na_to_K	혈액 내 나트륨 대 칼륨 비율
OUTPUT	Drug	처방 약의 종류

Mnist_400.csv : MNIST data 중 일부 400개를 추출하여 구성한 데이터

1. Random forest 알고리즘을 drag.csv 데이터에 적용시켜 분류하는 프로그램 작성, 그 결과를 Decision Tree와 비교하여 Bagging 측면에서 분석

2. VotingClassifier를 활용, drag.csv 데이터로 ensemble 학습을 수행하는 프로그램 작성. Hard voting과 Soft voting을 통해 각각 학습하고 그 결과를 분석. 모델은 KNN, SVM, Decision Tree, Random Forest를 활용하여 여러가지 조합 가능. 가중치 또한 여러 조합을 시도해보면서 결과 비교.

3. VotingClassifier를 활용, mnist_400.csv 데이터로 ensemble 학습을 수행하는 프로그램 작성. Hard voting과 Soft voting을 통해 각각 학습하고 그 결과를 분석. 모델은 KNN, SVM, Decision Tree, Random Forest를 활용하여 여러가지 조합 가능. 가중치 또한 여러 조합을 시도해보면서 결과 비교.

2. Problem Analysis

1. sklearn 라이브러리를 활용하여 KNN, SVM, Decision Tree, Random Forest 모델을 만들고 학습하여 테스트를 진행한다. 테스트 정확도를 출력한다.

2. 배깅

- 배깅은 과대적합이 쉬운 모델에 상당히 적합한 앙상블
- 배깅은 한 가지 분류 모델을 여러 개 만들어서 서로 다른 학습 데이터로 학습시킨 후(부트스트랩), 동일한 테스트 데이터에 대한 서로 다른 예측값들을 투표를 통해 (어그리게이팅) 가장 높은 예측값으로 최종 결론을 내리는 앙상블 기법
- 배깅의 어원은 부트스트랩(bootstrap)과 어그리게이팅(aggregating)에서 옴.

3. Ensemble 알고리즘은 sklearn의 VotingClassifier를 활용한다. 이를 활용하면 쉽게 hard voting, soft voting을 학습시킬 수 있다.

3. Code Explanation

프로젝트에 사용된 함수에 관한 설명이다. 문제 별 사용한 함수는 동일하다.

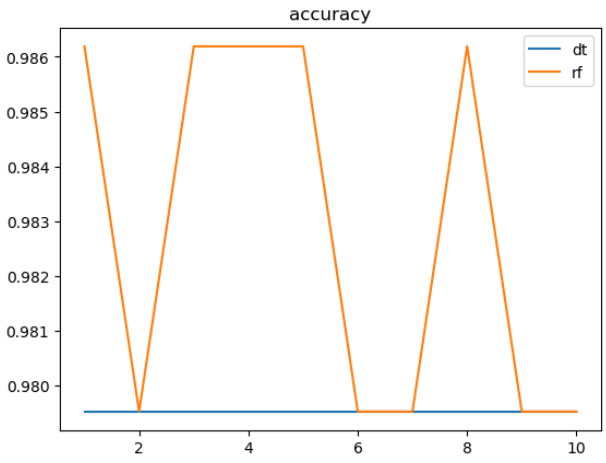
normalized(x)		
Parameter	Type	Explanation
x	array	정규화 할 input array
data	dataframe	csv파일의 내용을 담고 있는 dataframe

cross_validation(classifier,features, labels)		
Parameter	Type	Explanation
classifier	분류기	학습 모델
features	Dataframe	학습 및 검증을 위한 Input data
lables	Dataframe	학습 및 검증을 위한 Output data

4. Result & Conclusion

4.1 Result

1번 문제 : drag 데이터를 random forest 알고리즘을 이용하여 분류기를 학습시켜 보았다. 그리고 이를 decision tree와 결과를 비교해보았다. 그 결과를 아래와 같이 시각화해서 비교하였다.

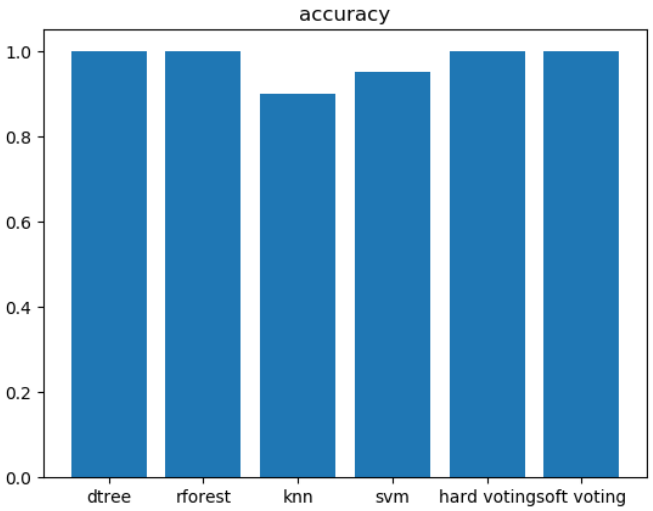


```
validation
decision tree accuracy :  0.9795238095238096
random forest accuracy :  0.9828571428571429

test
decision tree accuracy : 1.0
random forest accuracy : 1.0
Press any key to continue . . .
```

분석 결과 10번의 교차검증을 통해 얻어낸 정확도는 두 경우 모두 높았지만 Random Forest가 살짝 더 높았다. 테스트 데이터에 대한 정확도는 모두 1.0이었다.

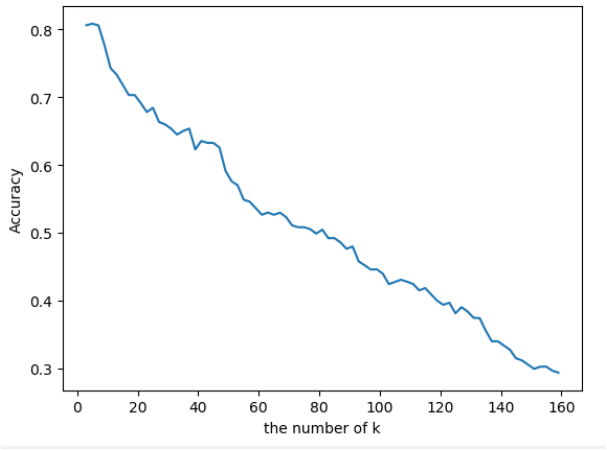
Random Forest의 정확도가 더 높을 수 있었던 이유는 Decision Tree를 배깅해서 모델을 만들고 예측하기 때문이다. 이와 함께 Random Forest는 모델의 편향을 증가시켜 과대적합의 위험성을 줄이기 때문이다.

2번 문제 : 2번 문제의 경우 ensemble을 이용하여 drac.csv 데이터에 대한 분류기를 학습시켰다. 그런데 개별 분류기(knn, svm, dtree, rforest)의 정확도가 이미 매우 높은 수준에 있었기 때문에 ensemble의 중요도는 사실 그리 높지 않았다. 그럼에도 dtree, knn, svm 모델을 조합하여 ensemble을 구성하여 학습시켜 보고 그 결과를 살펴보았다. 그 결과, hard voting과 soft voting에서 모두 1.0의 정확도를 얻었다. 아래의 그림은 각 분류기와 앙상블 분류기에 대한 정확도를 시각화한 것이다.



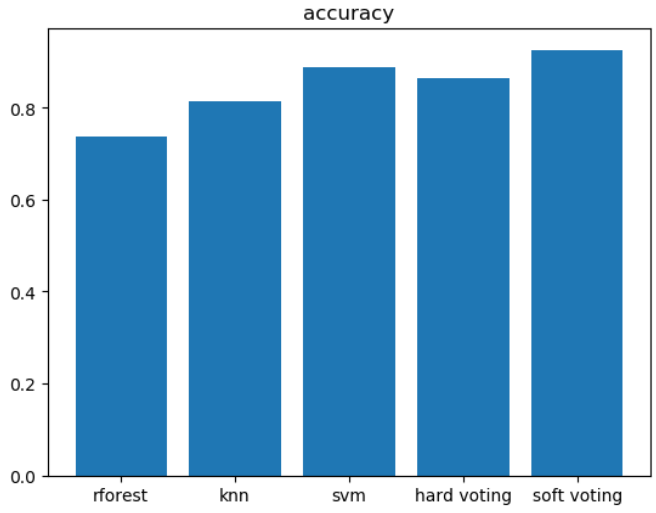

```
 [accuarcy]
tree          :  1.0
random forest :  1.0
knn           :  0.9
svm           :  0.95

 [ensemble]
 hard voting accuracy :  1.0
 soft voting accuracy :  1.0
```

3번 문제 : 3번 문제의 경우 ensemble을 이용하여 mnist_400.csv 데이터에 대한 분류기를 학습시켰다. 우선 개별 분류기(knn, svm, dtree, rforest)에 대해 정확도를 구하였다. 이때 파라미터 최적화가 필요할 시 교차검증을 통해 최적의 파라미터를 찾았다. 밑의 그래프는 knn의 예로, 가장 적합한 k를 교차검증을 통해 찾은 것을 가시화한 것이다.



개별 분류기에 대한 정확도를 분석한 결과, Decision tree의 경우 정확도가 0.5로 매우 낮았다. 그리고 random forest는 0.74, knn은 0.81, svm은 0.89의 정확도를 보였다. 따라서 정확도가 가장 낮은 dtree를 제외하고 3개로 앙상블을 구성하여 학습시켜보았다.

그리고 hard voting, soft voting으로 나누어 결과를 가시화해보았다. 이때 가중치를 자유롭게 조정할 수 있는데 여러 조합을 시도해본 결과, random forest, knn, svm의 가중치를 각각 1, 1, 2로 주었을 때 테스트 정확도가 가장 좋게 나왔다. 아래의 그림은 그 결과이다.




```
 [accuarcy]
tree          :  0.5
random forest :  0.7375
knn           :  0.8125
svm           :  0.8875

 [ensemble]
 hard voting accuracy :  0.8625
 soft voting accuracy :  0.925
Press any key to continue . . .
```

Hard voting의 경우 0.8625의 정확도를, soft voting의 경우 0.925를 얻었다. Soft voting은 확률로써 계산하기 때문에 hard voting의 경우보다 더 좋은 성능을 낸 것이라고 생각할 수 있다.

4.2 Conclusion

1. 1번 문제에서 Decision Tree보다 Random Forest에서 더 높은 정확도를 보였다. Random Forest의 정확도가 더 높을 수 있었던 이유는 Decision Tree를 배깅(bagging)해서 모델을 만들고 예측하기 때문이다. 이와 함께 Random Forest는 모델의 편향을 증가시켜 과대적합의 위험성을 줄이기 때문이다.

2. 2번 문제에서는 앙상블 기법으로 학습을 시켰는데, hard voting과 soft voting 모두에서 높은 학습 성과를 보였다. 그 이유는 개별 분류기의 정확도가 이미 충분히 높았기 때문이라고 생각할 수 있다. 개별 분류기가 이미 drug의 종류를 잘 구별해주기 때문에 투표로 결정되는 앙상블에서는 당연히 매우 높은 정확도를 얻을 수 있다. 가중치를 자유롭게 설정할 수 있었는데 2번 문제는 가중치 설정에 거의 영향을 받지 않고 정확도가 1.0이 나왔다.

3. 3번 문제에서는 앙상블 기법을 사용한 결과, 개별 분류기를 사용했을 때보다 더 높은 정확도를 얻을 수 있었다. 그리고 Hard voting의 경우보다 soft voting에서 더 높은 정확도를 얻었다. Soft voting은 개별 분류기의 결과를 확률로써 계산하기 때문에 hard voting의 경우보다 더 좋은 성능을 낸 것이라고 생각할 수 있다. 또한 가중치는 개별 정확도가 가장 높았던 svm에 더 높은 가중치를 주었을 때 더 좋은 결과를 냈다. 또한 개별 분류기 정확도가 낮았던 decision tree는 앙상블 조합에서 제외시키는 것이 정확도 측면에서 더 좋은 결과를 가져왔다.

4. 이번 과제를 통하여 앙상블 기법을 처음 사용하여 여러 분류기의 조합들을 이용하여 더 좋은 성능을 낼 수 있었다.