**2018-1 IT 집중교육 1**

**homework 1**

1조

민태홍

201320991

제출일자 :03/11

1. 데이터 설명(EDA 포함)

Au4\_2500.train.csv는 총 2000개의 data가 있고 101개의 feature가 있다.

여기서 6개의 feature를 뽑았는데 'V3','V8','V45','V51','V88','Class' 이 중 5개 v3, v8, v45, v51, v88을 뽑아서 class를 예측하려고 하였다.

뽑은 feature로 했을 때 nan 값을 처리하기위해 fillna(‘ffill’)을 사용하여 앞의 값을 nan값에 넣었다. 총 2000개의 data가 각 feature별로 있고 v3,v51는 object형식이고 v8,v45,v88는 float64형식으로 나왔다.

Float64형식인 V8,V45,V88는 mean,std,min,25%,50%,75%,max로 볼 수 있었다.

test데이터도 똑 같은 feature로 뽑았을 때 EDA를 보면 500개의 data가 들어있었다. 마찬가지로 있고 v3,v51는 object형식이고 v8,v45,v88는 float64형식으로 나왔다.

1. 데이터 전처리 (결측치 처리 방법 포함)

Object를 float으로 바꾸기 위해 le.fit\_Transform 형식을 이용하였다.

Le.fit\_Transform()함수는 object형식을 float형식으로 바꿔주는 함수이다 이것을 사용해서 모든 데이터를 float형식으로 바꾸어서 k-nn계산을 할 수 있었다.

그리고 데이터에서 v88은 평균이 538로 다른 데이터보다 훨씬 큰 값을 가지므로 정규화를 할 필요성이 있었다. 정규화하기위해 (X\_train-X\_train.mean())/X\_train.std()을 이용하여서 모든 데이터를 정규화 하였다.

결측치는 fillna(‘ffill’)로 앞에 있는 값으로 넣었다.

Test data를 뽑았을 때 똑같인 object형식을 float64형식으로 바꾸고 k-NN의 정확성을 더 높이기 위해 모든 데이터를 정규화를 시켰다.

1. Classification 알고리즘 수행 결과 및 분석

5-NN으로 했을 때 train데이터의 accuracy는 0.7185로 나오고 test데이터는 0.652로 나왔다.

Decision tree 분류기를 사용해서 했을 때는 train data의 accuracy는 0.9995가 나왔고 test data의 accuracy 는 0.604로 나왔다

SVM 분류기를 사용해서 했을 땐 train data의 accuracy는 0.79 test data 는 0.498로 나오게 되었다.

모든 분류기에서 train data의 accuracy가 더 높게 나왔는데 그 이유는 train data로 learning을 하여서 accuracy가 더 높게 나온 것 같다.

1. Evaluation 및 Optimization 수행 결과 및 분석

우선 knn의 optimization을 하기 위해 k의 값을 1~30중 홀수의 값을 구해 k의 값을 변화 시키며 accuracy를 구하였다. K가 13일 때 accuracy는 0.611로 최대값으로 나왔다. Svm을 최적화 하기위해 gamma를 1~5까지 해보았을 때 gamma가 2 일때 0.504로 제일 높게 나왔고 C의 값을 1~30중 홀수로 해보았을 때 C=29일 때 accuracy 0.52564로 제일 높게 나왔다.

Confusion matrix도 구해 보았을 때

Precision(1이라고 예측한 것 중 실제 값이 1인 것 ) 즉 내가 푼 문제 중에 맞춘 정답 개수가 평균 0.60

Recall(실제값이 1 인 것 중 1이라고 예측한 것 )전체 1 중에서 내가 몇 개를 맞췄는지가 0.65로 나왔다.

F1-score = 조화평균 은 평균 0.65로 나오게 되었다.

1. 토의 및 해석

k-nn 분류기를 사용했을 때가 0.652로 decision tree,svm보다 조금 더 높게 나왔다.

Train 데이터를 넣었을 때가 test데이터를 넣엇을 때 보다 훨씬 더 높게 나왔는데 생각해보면 train로 학습을 시켰으므로 train데이터의 결과값을 더 잘 예측할 수 있는 것 같다. 분류기를 학습 시킬 때 최적화를 하기위해 feature의 개수 k의 개수, svm 분류기의 gamma, hyper-parameter설정 등등 으로 accuracy를 높일 수 있다는 것을 확인할 수 있었다. ㄴㄴ