

[ML/DL] python 으로 구현하는 ROC곡선과 AUC — 나무늘보의 개발 블로그

노트북: 첫 번째 노트북

만든 날짜: 2021-01-04 오전 6:00

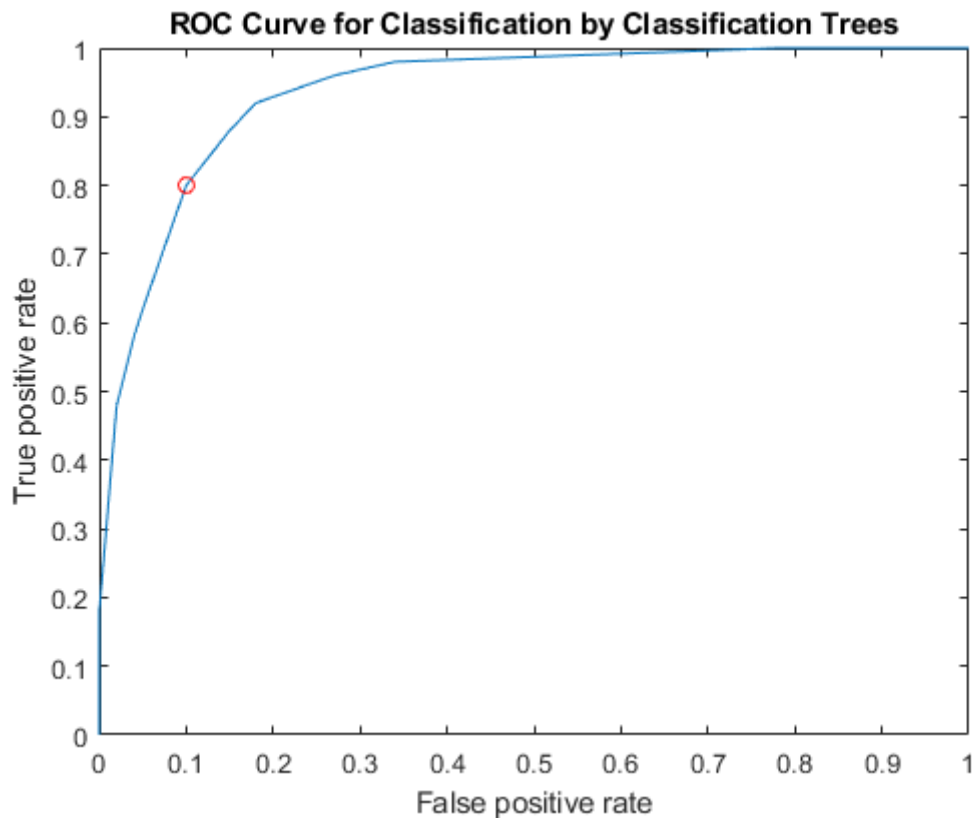
URL: <https://continuous-development.tistory.com/172?category=736685>

ML,DL

## [ML/DL] python 으로 구현하는 ROC곡선과 AUC

2020. 11. 2. 08:55 수정 삭제 공개

# ROC곡선과 AUC



ROC 곡선은 FPR이 변할 때 TPR이 어떻게 변하는지 나타내는 곡선이다.

ROC 곡선은 다양한 threshold에 대한 이진 분류기의 성능을 한 번에 표시한 것이다.

이진 분류의 성능은 True Positive Rate를 y축으로 False Positive Rate를 x 축으로 이용해서 표현하게 된다.

ROC 커브는 좌상단에 붙어있는 커브가 더 좋은 분류기를 의미한다고 생각할 수 있다.

즉 곡선이 100에 가까워질수록 positive에 가까워진다. 곡선이 커진다는 것은 positive라고 예측할 확률이 높아진다.

이 곡선의 면적은 AUC라고 한다. 이것은 ROC 곡선 밑의 면적을 구한것으로서 일반적으로 1에 가까울수록 좋은 것이다.

또한, ROC가 0.5에 가깝다는 건 분류가 제대로 안됐다는 것이다. 이진 분류로서 의미가 없다. 이 곡선이 직선에 가까울수록 성능이 안 좋다.

임계값이 1이면 positive를 예측을 할 수 없게 되고 0에 가까워질수록 positive라고 예측할 확률이 높아진다.

## # roc\_curve(실제값, 예측 확률 값): FPR, TPR, 임계값

(타이타닉을 이용한 생존자 예측 FPR, TPR, 임계값)

앞단에 타이타닉 데이터를 X\_train, y\_train, X\_test, y\_test를 만드는 작업이 있습니다.

```

from sklearn.metrics import roc_curve

# 레이블 값이 1 일때의 예측확률을 추출

pred_positive_label = lr_model.predict_proba(X_test)[:,1]

fprs, tprs, thresholds = roc_curve(y_test, pred_positive_label)

print("샘플추출")

print()

thr_idx = np.arange(1,thresholds.shape[0],6)

print('thr idx:',thr_idx)

print('thr thresholds value:',thresholds[thr_idx])

print('thr thresholds value:',fprs[thr_idx])

print('thr thresholds value:',tprs[thr_idx])

```

## 샘플추출

```

thr idx: [ 1  7 13 19 25 31 37 43 49]
thr thresholds value: [0.94764846 0.69009268 0.63604005 0.41757306 0.29194688 0.22945312
 0.14669529 0.12959111 0.05506945]
thr thresholds value: [0.          0.02564103 0.08547009 0.15384615 0.24786325 0.31623932
 0.5982906  0.69230769 0.98290598]
thr thresholds value: [0.01612903 0.51612903 0.58064516 0.75806452 0.80645161 0.88709677
 0.90322581 0.9516129  0.98387097]

```

샘플을 위와 같이 추출한 다음 그 값을 인덱스로 사용함에 따른 FPRS, TP RS를 볼 수 있다.

이 값을 가지고 그래프를 그리면 아래와 같이 생긴다.

```

pred_positive_label = lr_model.predict_proba(X_test)[:,1]

fprs, tprs, thresholds = roc_curve(y_test, pred_positive_label)

```

```
precisions, recalls, thresholds = roc_curve(y_test, pred_positive_label)
```

```
plt.figure(figsize=(15,5))
```

```
# 대각선
```

```
plt.plot([0,1],[0,1],label='STR')
```

```
# ROC
```

```
plt.plot(fprs,tprs,label='ROC')
```

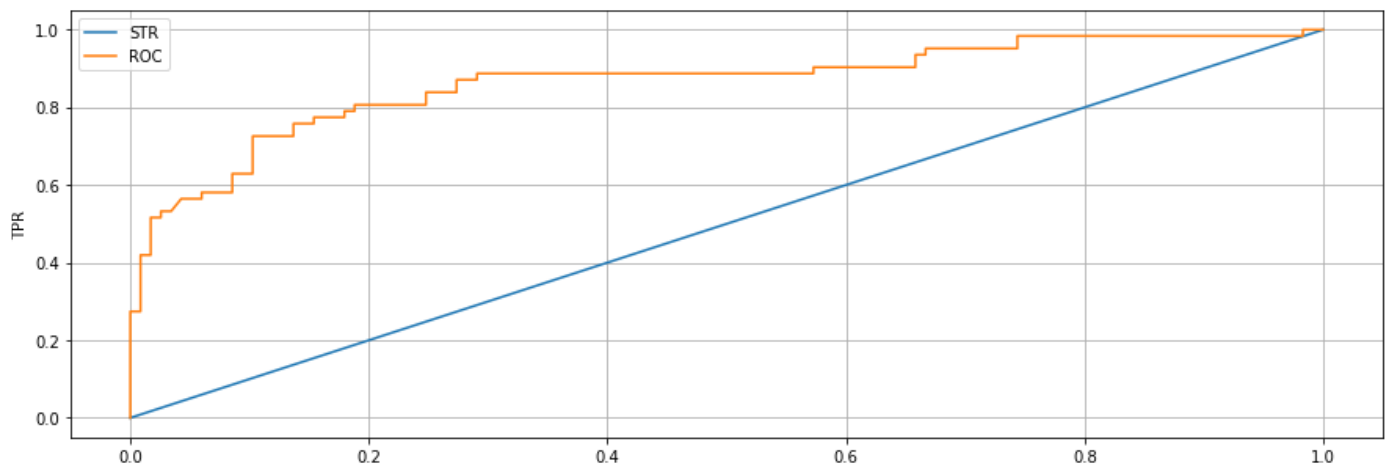
```
plt.xlabel('FPR')
```

```
plt.ylabel('TPR')
```

```
plt.legend()
```

```
plt.grid()
```

```
plt.show()
```



## AUC 면적 구하기

```

from sklearn.metrics import roc_auc_score

from sklearn.linear_model import LogisticRegression


# 위 코드 확인 후

# 로지스틱 회귀

lr_model = LogisticRegression()

lr_model.fit(X_train,y_train)

prediction = lr_model.predict(X_test)

print('roc auc value {}'.format(roc_auc_score(y_test,prediction))) # 이 value는 auc에 대한 면적을 나타내

# display_eval(y_test,prediction)

```

roc auc value 0.807347670250896

## 'ML,DL' 카테고리의 다른 글

[ML/DL] 앙상블 학습 (Ensemble Learning): bagging,voting,boosting

[ML/DL] DecisionTree 구현 및 hyper parameter 설정

**[ML/DL] python 으로 구현하는 ROC곡선과 AUC**

[ML/DL] 정밀도와 재현율의 트레이드 오프 정의와 구현

[ML/DL] python 을 통한 분류(classification) 성능평가지표 사용법(Accuracy,Pre...

[ML/DL] python 을 통한 교차검증 ( k -Fold , stratifiedkFold)

---

AUC

Python AUC

python ROC

Roc

---



나아무늘보

혼자 끄적끄적하는 블로그 입니다.