ada\_boost Open Source code

# 라이브러리 링크

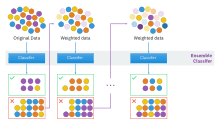
**install :**

[**https://scikit-learn.org/stable/install.html**](https://scikit-learn.org/stable/install.html)

**github :**

[**https://github.com/scikit-learn/scikit-learn/blob/7e1e6d09b/sklearn/ensemble/\_weight\_boosting.py#L313**](https://github.com/scikit-learn/scikit-learn/blob/82df48934eba1df9a1ed3be98aaace8eada59e6e/sklearn/tree/_classes.py)

# 기초 설명

****

AdaBoost 는 yoav freund와 robert schapire가 개발한 기계 학습 메타 알고리즘으로 AdaBoost는 성능을 향상 시키기 위하여 다른 많은 형태의 학습 알고리즘과 결합하여 사용할수 있으며. 다른 학습 알고리즘의 결과물들을 가중치를 두어 더하는 방법으로 가속화 분류기의 최종 결과물을 표현할수 있다.

# 버전 정보

* Python ( >=3.7 )
* NumPy( >= 1.14.6 )
* Scipy( >= 1.1.0 )
* joblib( >= 0.11 )
* threadpoolctl( >= 2.0.0 )
* For example code :
* pandas (pip install pandas)

# 데이터셋 설명 및 출처

* train.scv: The Otto Group is one of the world’s biggest e-commerce companies, with subsidiaries in more than 20 countries, including Crate & Barrel (USA), Otto.de (Germany) and 3 Suisses (France). We are selling millions of products worldwide every day, with several thousand products being added to our product line.
* A consistent analysis of the performance of our products is crucial. However, due to our diverse global infrastructure, many identical products get classified differently. Therefore, the quality of our product analysis depends heavily on the ability to accurately cluster similar products. The better the classification, the more insights we can generate about our product range.
* For this competition, we have provided a dataset with 93 features for more than 200,000 products. The objective is to build a predictive model which is able to distinguish between our main product categories. The winning models will be open sourced.
* Sources : https://www.kaggle.com/c/otto-group-product-classification-challenge

# 코드 설명

* 데이터셋을 학습용 데이터셋과 검증용 데이터셋으로 나눈뒤, 이를 각각의 decision tree, adaboost, 그리고 random forest 모델에 학습시킨후 검증용 데이터셋을 입력하여 각 모델의 예측결과를 실제 검증 데이터셋과 비교하여 이에대한 각 모델의 정확도를 출력한다.

# 검증 방법

* 코드 내부에서 데이터셋을 학습용 데이터셋과 검증용 데이터셋으로 나누어 이를 검증함.

(test\_size = 0.2, random\_state = 0)

* 추가적으로 sklearn의 accuracy\_score 함수를 사용하여 실제값과 예측값 사이의 적합도를 평가.