ElasticNet

# 라이브러리 링크

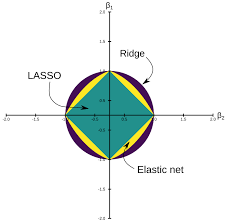
**install :**

<https://scikit-learn.org/stable/install.html>

**github :**

https://github.com/scikit-learn/scikit-learn/blob/7e1e6d09b/sklearn/linear\_model/\_coordinate\_descent.py#L679

# 기초 설명

****

ElasticNet regression 은 여러 정규화 방법들중 하나로 선형회수의 계수값이 매우 크게 나타나게 되는 상황을 방지하기 위해 사용된 RIDGE, LASSO 방식의 L1-norm,L2-norm 값을 모두 사용하는 정규화 방식이다.

# 버전 정보

* NumPy >= 1.14.6 (pip install numpy)
* Scipy >= 1.1.0 (pip install scipy)
* Joblib >= 0.11 (pip install joblib
* Threadpoolctl >= 2.0.0 (pip install threadpoolctl)
* pandas >= 1.2.4 (pip install pandas)
* matplotlib == 3.22 (pip install matplotlib)

# 데이터셋 설명 및 출처

* winequality-red.scv:
* The two datasets are related to red and white variants of the Portuguese "Vinho Verde" wine. For more details, consult: [Web Link] or the reference [Cortez et al., 2009]. Due to privacy and logistic issues, only physicochemical (inputs) and sensory (the output) variables are available (e.g. there is no data about grape types, wine brand, wine selling price, etc.).
* These datasets can be viewed as classification or regression tasks. The classes are ordered and not balanced (e.g. there are many more normal wines than excellent or poor ones). Outlier detection algorithms could be used to detect the few excellent or poor wines. Also, we are not sure if all input variables are relevant. So it could be interesting to test feature selection methods.
* P. Cortez, A. Cerdeira, F. Almeida, T. Matos and J. Reis.  
  Modeling wine preferences by data mining from physicochemical properties. In Decision Support Systems, Elsevier, 47(4):547-553, 2009.
* Sources : https://archive.ics.uci.edu/ml/datasets/wine+quality
* or : https://www.kaggle.com/uciml/red-wine-quality-cortez-et-al-2009

# 코드 설명

* Sklearn의 ElasticNet 을 사용하여 레드와인에 대한 데이터셋 내부의 데이터를 학습하여 어떤 와인의 total sulfur dioxide 를 예측하는 ElasticNet regression 모델을 생성하고 이때 파라미터로 사용되는 alpha, l1-ratio 값의 변동에 따라 변화하는 모델의 예측값과 실제값을 그래프를통해 나타내고 각 그래프에 대한 Mean\_Squared\_error 를 계산하여 별도의 그래프에 나타낸다.

# 검증 방법

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

(random\_state = 30)

* 추가적으로 sklearn의 Mean\_Squared\_error 함수를 사용하여 실제값과 예측값 사이의 적합도를 평가.