[5주차 과제1] Logistic Regression의 하이퍼파라미터 변경에 따른 성능 결과 분석

은하계 종류 예측을 통한 분석

18011480 이진형 I 컴퓨터공학과

문제 분석

문제에서 주어진 데이터 셋(trainX.csv, testX.csv)는 2D 은하계 사진을 vgg16과 pca를 통해 1차원으로 기술한 데이터이다. 각 컬럼은 기술된 벡터의 각 차원을 나타낸다. 해당 데이터 셋으로 Logistic Regression 모델을 학습시킨 뒤, Spiral, Edge, Smooth 세 가지의 은하 종류로 분류하는 문제이다.

실험 조건

scikit-learn 문서[1]에 따르면 Logistic Regression의 파라미터는 penalty, dual, tol, C, fit_intercept, intercept_scaling, class_weight, random_state, solver, max_iter, multi_class, verbose, warm_start, n_jobs, l1_ratio 가 존재한다. 이번 실험에선 이들 중 penalty, C, solver 세 가지 파라미터의 변화를 통해 실험을 진행했다.
실험에 사용할 파라미터의 종류를 나타내면 다음과 같다.

Solver

liblinear	Library for large linear classification. 공식 문서에 따르면 작은 크기의 데이터셋에 유리하다.	l1, l2
saga	Stochastic average gradient descent인 sag에 L1 정규화가 가능하도록 한 solver.	elasticnet, l1, l2, none
lbfgs	Limited-memory Broyden-Fletcher-Goldfarb-Shanno. 0.22.0 버전부터 기본값이다.	l2, none

Penalty

l1	Lasso. 영향을 크게 미치는 핵심적인 피처(x_i)들만 반영하도록 한다.
l2	Ridge. 전체적인 W값을 감소시켜 그래프의 곡률을 줄인다.

C

C는 규제 강도를 나타내는 alpha 값의 역수이다. C값이 작을 수록 규제 강도가 크다.

[1] https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html

실험

진행한 실험의 조건과 결과는 다음과 같다.

Solver	Penalty	C	Score
liblinear	l1	0.1	0.79400
liblinear	l1	1.0	0.79400
liblinear	l1	10.0	0.79466
liblinear	l2	0.1	0.79533
liblinear	l2	1.0	0.79466
liblinear	l2	10.0	0.79466
saga	**none**	**0.1**	**0.79666**
saga	**none**	**1.0**	**0.79666**
saga	**none**	**10.0**	**0.79666**
saga	l1	0.1	0.79600
saga	**l1**	**1.0**	**0.79666**
saga	l1	10.0	0.79600
saga	l2	0.1	0.79533
saga	l2	1.0	0.79533
saga	l2	10.0	0.79600
lbfgs	**none**	**0.1**	**0.79666**
lbfgs	**none**	**1.0**	**0.79666**
lbfgs	**none**	**10.0**	**0.79666**
lbfgs	l2	0.1	0.79533
lbfgs	l2	1.0	0.79533
lbfgs	**l2**	**10.0**	**0.79666**

실험 결과

위와 같은 조건으로 실험을 진행한 결과, 최고 점수인 0.79666을 기록한 파라미터는 다음과 같았다.
saga/none/(0.1, 1.0, 10.0)
saga/l1/1.0
lbfgs/none/(0.1, 1.0, 10.0)
lbfgs/l2/10.0
이번 문제에서는, 전반적으로 낮은 점수를 기록한 liblinear를 제외하고는 오히려 정규화를 진행하지 않거나(none) 보다 약한 규제 강도(C값 1.0, 10.0)에서 높은 점수가 나온 것을 볼 수 있다. 또한, penalty가 none으로 지정됐을때는, 강도를 적용할 규제가 없기 때문에 C값에 구애받지 않고 같은 모델이 생성된 것으로 보인다.

개선사항

규제정도를 나타내는 C값에 따른 개선사항을 더 효과적으로 확인하기 위해서는 학습 데이터와 테스트 데이터의 점수를 확인하여 생성된 모델이 fit된 정도를 확인해야 하는데, 해당 추이를 확인하기엔 무리가 있는 결과가 나와 아쉬웠다.