**Machine Learning Assignment\_2**

조 병 웅

2019312570

원전공 사학과/복수전공 인공지능 융합전공

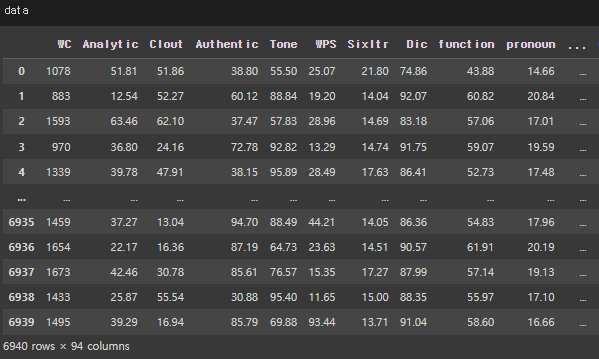
1. **Problem Definition**

이번 프로젝트는 MBTI를 공개한 사람들이 올린 텍스트의 LIWC분석값은 데이터로 사용해 텍스트를 쓴 사람이 T(사고형)인지, F(감정형)인지를 분류(classification)하는 문제이다. 프로젝트에서 feature는 사람들이 쓴 텍스트의 LIWC 분석값들이며, 정답은 해당 데이터에 포함된 label 데이터이다. 필자는 이진분류가 가능한 이번 모델의 특성상, 이진분류 로지스틱 회귀 분류 모델을 분류를 수행할 모델로 채택하였다.

1. **Data**
   1. **Data Description**

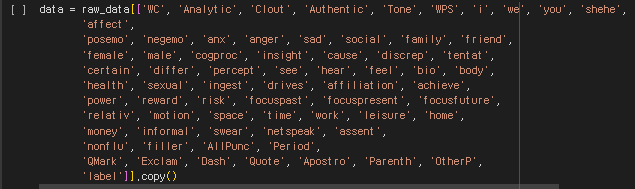
input data:

해당 데이터는 6940개의 각 인물에 대한 텍스트 분석과 label로 이루어져 있다. 텍스트 분석은 LIWC 분석을 거친 데이터들로 약 90여가지의 열들로 이루어져 있고, wc열 빼고는 모두 퍼센테이지로 분석된다. wc, analytic, clout, authentic, tone, wps는 각 텍스트의 단어 수와 같은 텍스트 분석 값이고, dic, function과 같은 구문 분석값, 그리고 feel과 같은 감정 표현을 나타내는 분석값 또한 존재한다. wc를 제외한 클래스는 퍼센테이지 형태 내에서 크기가 다양하지만, wc는 단순히 숫자값이기 때문에 이러한 피처 데이터를 그대로 활용하는 건 무리가 있어보인다. 후술하겠지만, 이러한 데이터 값을 standardscaler를 통해 스케일해서 사용한다.

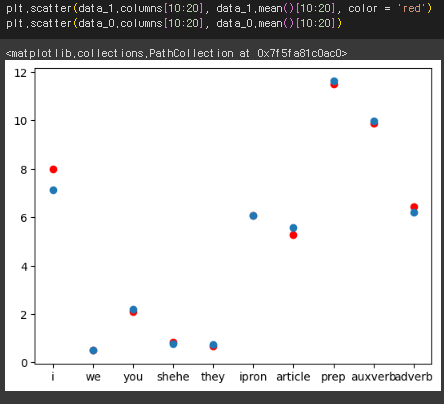


* 1. **Data Pre-processing**

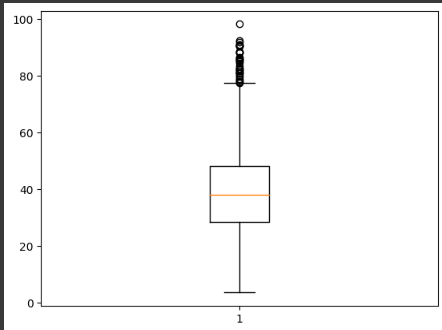
가설 단계에서, 모든 피처 데이터를 사용한 모델보다 특정 감정과 같은 피처를 사전에 추출해서 피처의 수를 줄이고자 생각하였다. 사고형과 감정형이 인간의 감정표현과 관련되어있기 때문이다. (사용한 피처들)



그러나 인위적으로 조작한 피처의 모델은 생각보다 좋은 결과를 내지 못했다. 모든 피처 데이터를 사용한 모델을 생성해서 서브미션한 결과는 0.71로 나왔는데, 이후 조작한 피처를 사용한 모델들이 모두 0.4~0.5와 비슷한 결과를 보였다. 때문에 사전에 피처를 고려하여 조작하기보다 피처가 가지고 있는 값들을 비교하여 큰 차이가 나는 피처를 사용해보고자도 노력하였다.

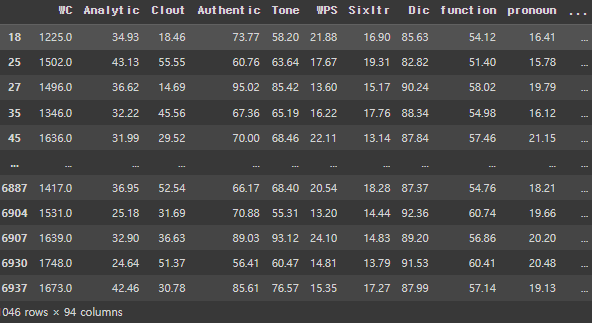
  
하지만 이러한 피처들의 비교 또한 그다지 유의미한 결과를 얻지 못하였다. 때문에 이러한 피처 조작보다는, 데이터의 이상치들을 제거하는 방법만을 고려하고자 하였다.

데이터 이상치를 검출하기위해서는 IQR(사분범위)를 이용하였다. boxplot에서 나온 이상치들을 제거하고자 떠오른 아이디어였다. 먼저 boxplot을 사용해 이상치를 검출해보니 꽤 많은 양의 이상치들이 검출되는 것을 확인할 수 있었다.



이를 한계치를 구하고, 해당 한계치보다 높고 낮은 값들을 drop()하는 방식의 전처리 과정을 통해 이상치를 줄일 수 있었다.

(전처리 후의 데이터)



1. **Analysis**

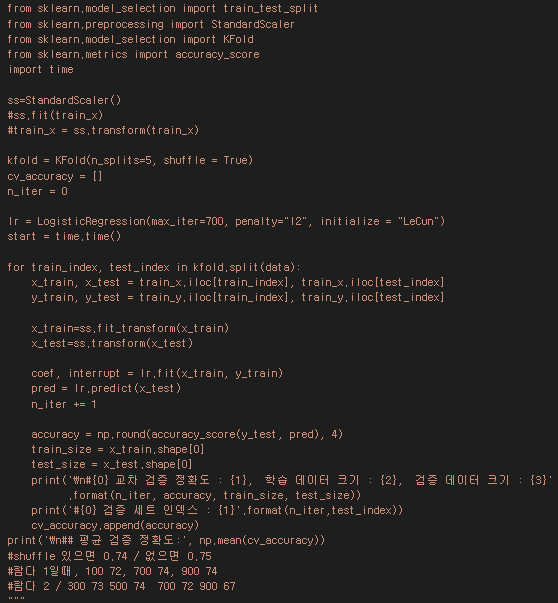
이러한 과정을 통해 이상치들을 제거하여 유의미하다고 생각되는 수치들만을 남길 수 있었지만, 약 6000개 정도의 데이터 수들이 1000개 정도로 줄어들었다. 데이터들의 수가 많이 줄어들어 서브미션의 정확도가 높지는 않았지만, 데이터 증식에 대한 방법을 찾지 못해 데이터를 늘리는 것에는 아쉽지만 실패했다. 데이터 하나하나가 유의미한 값들을 남기는데 성공했으나, 많은 피처수에 비해 데이터의 확보가 부족했다고 평가된다.

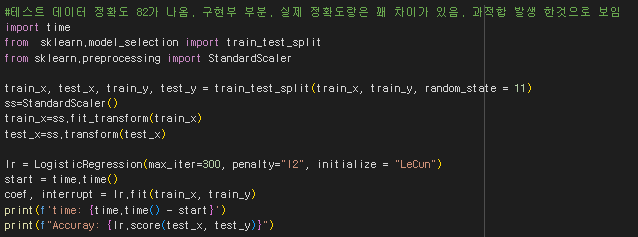
1. **Model**
   1. **Classifier**

분류기는 수업 때 scratch로 구현한 이진분류 로지스틱 회귀를 사용하였다. 데이터의 각 피처들이 입력되면, 내부에서 fwpass 함수가 수행되면서 가중치와의 내적이 수행된다. 이러한 과정 중에서 오차가 발생한다면, 그 오차를 가중치에 업데이트하는 방식으로 가중치들이 정해진다. 이러한 과정 중에서 가중치들이 오버피팅되는 것을 방지하기 위해 L2 규제를 적용하였다.

* 1. **Design Consideration**

이렇게 가공된 데이터들을 그냥 사용할 수는 없었다. 데이터의 범위가 다르고,(특히 WC의 값이 크다) 데이터를 분류해야하기 때문이다. 필자는 이 부분에서 먼저 데이터의 스케일을 조정하기 위해 standardscaler를 사용하였다. 그리고 데이터 분류 측면에서느 2가지 방법을 시도하였는데, 먼저는 k폴드 교차검증을 사용하였고, 그 다음은 일반적은 test\_and\_split 라이브러리를 사용하였다.





k폴드 교차검증을 활용해 모델의 파라미터를 적절한 값으로 구할 수 있었으나, 아마도 scratch 코드의 구현상의 문제로 인해 결과값이 좋지 않게 작동하였다. 그래서 실제 머신을 작동시킬 때에는 파라미터 값만을 반영하여 일반적은 test\_and\_split을 적용하였다.

1. **Experiments**
   1. **Setting**

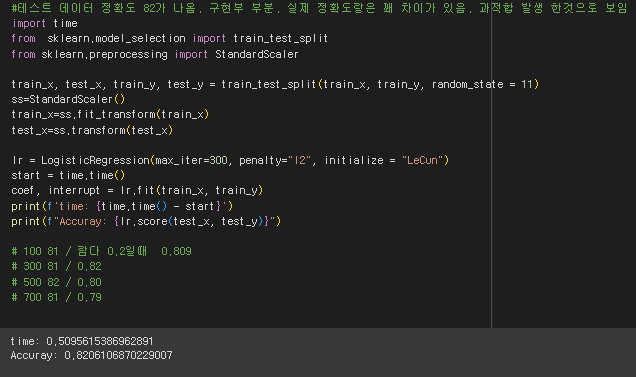
k폴드 교차 검증으로 활용해 구한 모델의 파라미터는 lr : 0.001, lambda : 0.01, iter : 300이었다. 이후에도 많은 값들을 서브미션 시도해보았지만, 모두가 0.5 정도에 그쳐 큰 변화를 만들어 내지는 못했다. 아마 피처수가 많아 유의미한 정답률의 변화를 파라미터만으로는 가져오기 힘들었던 것으로 보인다.

* 1. **Performance Metrics**

평가 지표는 기존의 로지스틱 회귀 결과값 평가 방식과 동일하게 prediction된 데이터와 정답 데이터를 ==연산하고, True값의 mean()을 측정하는 방식으로 구하였다.

* 1. **Results**

먼저 테스트 데이터로 구한 정답률은 다음과 같다.



그러나 실질적인 서브미션에서는 이보다 10퍼센트가 낮은 71퍼센트로 나왔다. 아마 많은 피쳐 수에 비해 데이터 수가 부족하여 모델이 다소 오버피팅된 것으로 추정된다.

1. **Discussion and Limitation**

먼저 이상치 제거를 통해 유의미한 데이터 값들만 남길 수 있었던 것은 정답률 상승에 큰 도움이 된 것으로 보인다. 허나, 이를 통해 데이터 수들이 부족해지면서 모델이 다소 과적합된 아쉬움이 있다. 이를 해결하기 위해서는 파라미터의 조작보다는 보다 많은 데이터의 확보가 중요하다고 생각되었다. 여러가지로 데이터 증식을 시도해 보았으나, 아직 견식의 부족으로 인해 구현하지 못한 한계점이 있다. 만약 데이터 증식이나 새로운 데이터의 제공으로 데이터 수를 충분히 유의미할 만큼 모을 수 있다면, 해당 모델이 충분이 높은 결과를 제공할 것이라고 기대한다.

1. **References**

-코랩에서 작업하였고, api코드들은 캐글 문서에서 가져왔다.

-로지스틱 회귀 모델은 기존 수업을 통해 배운 토대로 구현한 스크래치 모델을 사용하였다.

- iqr을 통한 이상치 제거는 해당 블로그를 레퍼런스했다.

https://hwi-doc.tistory.com/entry/IQR-%EB%B0%A9%EC%8B%9D%EC%9D%84-%EC%9D%B4%EC%9A%A9%ED%95%9C-%EC%9D%B4%EC%83%81%EC%B9%98-%EB%8D%B0%EC%9D%B4%ED%84%B0Outlier-%EC%A0%9C%EA%B1%B0