21. 11. 24

* 데이터 내의 결측치 검사 : 없음
* 데이터 내 기록되지 않은 열 제거
* 데이터를 받아 오류 발생 여부를 예측 모델 1 [ 열 : PassOrFail ]
* 데이터를 받아 오류의 원인을 예측하는 모델2 [ 열 : Reason ]

1. 모델\_1

* PassOrFail 의 Y/N 을 이진 표기 : 0 or 1 으로 변환 >> 결과값으로 사용

1. 모델\_2

* 원인의 세가지 종류를 클래스 코드로 지정 { 가스 : 1 , 미성형 : 2, 초기허용불량 : 3 }
* 기기의 요구된 결과값의 추정 과정에 불필요한 열 삭제 : 총 9개 열
* 데이터 정규 분포 스케일링 훈련 셋 70 % / 검증 셋 30%

**# 구현법**

* **Random Forest :**

1. 상관관계 따른 특징 추출 O : F1 73%, Recall 65%, Precision 96%, 정확도 99%
2. Train / Valid 50% 씩 : F1 70, Recall 63점, Precision 96점, 정확도 99%

데이터 셋 자료 특성상 오류의 가짓수가 지나치게 적어 생긴일 // 비균형

* **MLPClassifier**

1. 상관관계 따른 특징 추출 O: F1 64점, Recall 58점, Precision 100점, 정확도 99%

* **ANN**

1. 상관관계 따른 특징 추출 O : Acc : 99.1, Recall: 100, Precision : 99.1, F1 : 99.5

* POF : PassOrFail 로서 통과 이상수치를 판단하는 것 자체는 쉬움, 하지만 판단하고 원인을 규명하지 못하기 때문에 수정 → Reason 모델 자체가 POF 를 내포하기 때문
* 다중 클래스 분류로 목표를 변경.
* Selu 레이어보다 LeakyRelu 가 효과가 좋음
* Valid set 비율을 낮춰 학습량 증가시킴
* 상관도 분석 임계를 낮춰 feature를 늘림.
* Loss 의 상태를 보아 Underfitting 이었던 것으로 추정
* 확실히 Underfitting 중이다
* 애초에 데이터 분류가 너무 적다.
* Boosting 알고리즘으로 전환 예정.