Anomaly Detection with Machine Learning

중앙정보처리학원

『빅데이터 기반 AI 응용 솔루션 개발자 전문과정』

기간

2021/7/26 - 2021/8/26

팀원

구혜진 김민기 서영흔
이세희 이호영 황정석

Contents

1 프로젝트 개요

2 데이터 탐색

3 데이터 변환

4 모델링

5 결론

6 느낀 점

프로젝트 개요

1

1) 프로젝트 배경 및 목적

2) 일정 및 환경 도구

3) 데이터 분석 방법론

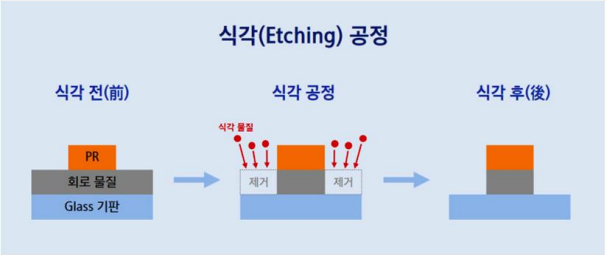
프로젝트 목적

- 웨이퍼 식각 후 테스트 단계에서 **센서 측정값의 1시간 평균 데이터를 분석하여 저수율 요인을 찾고자 함**
- 테스트 과정에서 웨이퍼 상태의 반도체 칩의 불량여부를 선별 가능함
- 저수율 요인을 찾아 설계상의 문제점이나 제조상의 문제점을 발견해 수정 가능함

프로젝트 배경

웨이퍼 식각 공정

- 반도체 공정 중 하나
- TFT(박막트랜지스터)의 회로 패턴을 만들기 위해 웨이퍼의 필요한 부분만 남기고 불필요한 부분은 깎아내는 공정



저수율 웨이퍼가 발생하는 원인

- 반도체 공정의 각 프로세서에서 레시피(온도,압력, 가공시간 등)대로 작업이 이루어지지 않았기 때문

저수율 요인 파악의 필요성

- 공정 중 저수율 요인을 찾아내면 해당 프로세서의 집중적인 관리를 통해 고수율 웨이퍼의 생산효율을 극대화 할 수 있음
- 최적의 Etching 공정 레시피를 제공하고자 함

생산공정 중 양품/불량품에 큰 영향을 미치는 저수율 요인 5개를 찾아내는 것을 목표로 함

프로젝트 일정표

	7월							8월																
	1주차							2주차							3주차							4주차		
멘토링	월	화	수	목	금	토	일	월	화	수	목	금	토	일	월	화	수	목	금	토	일	월	화	수	목
	26	27	28	29	30	31	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
데이터 탐색	■	■	■	■	■	■																			
데이터 변환							■	■	■	■	■	■	■	■	■	■	■								
모델링															■	■	■	■	■	■	■	■	■	■	■
검증 및 평가																		■	■	■	■	■	■	■	■

분석 환경 및 도구

Hardware/Server

- ubuntu — Ver 18.04
- Windows 10 — CPU i7, RAM 16

Language

- python 3 — Ver 3.7.11

Analysis Library

- pandas — Ver 1.1.5
- NumPy — Ver 1.19.5
- scikit learn — Ver 0.22.2
- SciPy — Ver 1.4.1
- matplotlib — Ver 3.2.2
- seaborn — Ver 0.11.1

Tools

- github
- Google Drive

Development Tools

- Colab
- Jupyter lab — Ver 2.2.6

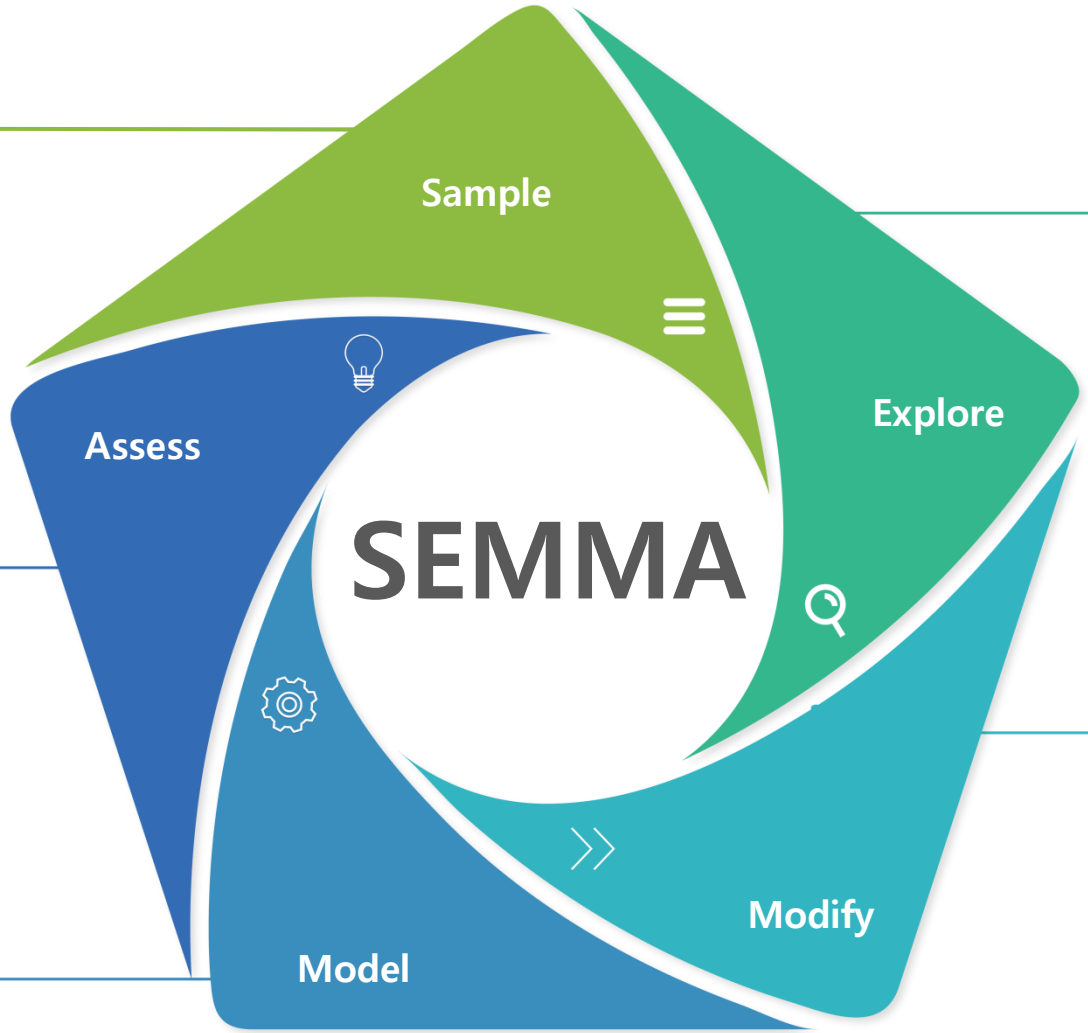
SEMMA 방법론

Sample
분석 데이터 수령
생산 설비에서 나오는
센서 정보

Assess
모델 평가 및 검증
분류 모델 평가 지표 중
재현율(Recall) 선택

모델간 성능 비교
각 모델 중요 피처 추출 후
레이블 분류 성능 시각화

Model
모델 구축 및
알고리즘 적용
XGBoost
Random Forest
SVM
Decision Tree



Explore
데이터 탐색 및
기술 통계 분석
피처 이름으로 상관분석
(Left, Right)

Modify
데이터 전처리
결측치 제거
분산도 0 제거
상관계수 0.9이상 제거

데이터 변환
PCA
Feature Importance
RFE

데이터 탐색

2

초기 데이터 shape 확인

8145 rows x 841 columns

※ dat...date.name.피처를 index로 지정함

· 초기 데이터는 8145 rows x 840 columns의 int, float 타입 데이터로 구성됨

2016년 1월1일 07시~
2016년 12월31일 23시까지의
1시간 단위의 평균 센서값

센서 피처

RollDown, 진공폐기
x Left, Right의 폐기율 피처

dat...date.name.	LIFT.OUT.ROL.MTR.M1.SPD.1WSI40101	...	STD_GLS_R	투입.LEFT.	...	R.Vac
2016-01-01 07						
...						
2016-12-31 23						

841 columns

8145 rows

데이터 전처리 및 변환

3

1) 데이터 전처리 요약

2) 데이터 전처리 상세 내용

3) 피처 구분

4) 변환 – PCA 분석

5) 레이블 분류 기준

데이터 전처리 및 변환

1) 데이터 전처리 요약

데이터 전처리 요약

초기 데이터 shape

8145 rows x 840 columns

전처리 후 데이터 shape

8145 rows x 324 columns

STEP 01 결측치 처리

모든 행(8145rows)이 결측치인 피처 제거
나머지 결측치가 3개의 행에 존재
=> 시계열 데이터이므로 ffill로 결측치 처리

7개 피처 제거

STEP 02 분산을 이용한 피처 제거

분산을 출력하여 분산 값이
0에 가까운 피처 제거

7개 피처 제거

STEP 03 피처 분류를 통한 피처 제거

피처 이름에 Left, Right가 반복 등장함
➔ 동일한 공정을 수행하는
별개 라인으로 판단

Right 피처 87개 제거

STEP 04 상관계수를 이용한 피처 제거

상관계수 0.9 이상인
피처 제거

401개 피처 제거

~~STEP 05~~ VIF를 이용한 피처 제거

VIF 계수가 30이하가 될 때까지
VIF 계수가 가장 큰 피처 제거

159개 피처 제거

모델 훈련 시 생략함

VIF를 이용하여 피처를 제거한 결과,
남은 피처가 모델을
충분히 설명하지 못하는 것으로 판단

데이터 전처리 및 변환

2) 데이터 전처리 상세 내용

결측치 처리

7개 피처 제거

- 열 : 모든 행(8145rows)이 결측치인 피처 제거
- 행 : 나머지 결측치가 3개의 행에 존재

=> 시계열 데이터이므로 ffill로 결측치를 채움

분산을 이용한 피처 제거

7개 피처 제거

- 분산을 출력하여 0에 가까운 분산값을 가진 피처 제거함
- 피처의 분산이 0이라는 것은 그 피처의 데이터가 모든 행에 대해 거의 변하지 않은 것을 의미
- 어떤 경우에도 같은 값을 내는 피처가 불량률에 영향을 주고 있다고 보기 어려움

피처 분류하여 Right피처 제거

87개 피처 제거

상세 내용은 다음페이지에서 설명

상관계수를 이용한 피처 제거

401개 피처 제거

- 상관계수 0.9 이상인 피처 제거
- 두 피처 간의 상관관계가 높다는 것은,
  하나의 피처 값이 다른 피처의 값에 큰 영향을 줌을 의미함
- 모델링에 영향을 미치는 원인들이 모두 비슷한 중요도로 반영되게 하려면
  종속성이 낮은 피처들만을 이용하여 모델을 만드는 것이 타당함

VIF를 이용한 피처 제거

159개 피처 제거

- VIF계수가 30이하가 될 때까지 VIF계수가 높은 피처 제거
- 이 방법으로 피처를 제거한 결과, 남은 피처가 모델을 충분히 설명하지 못하는 것 같음
  => 순차적으로 진행되는 작업으로 수행 시간이 오래 걸리는 작업인 것에 비해
  중요 피처가 삭제되는 것으로 판단하고 모델 훈련 데이터 생성 시 이를 생략함

피처이름으로 Left, Right 공정 구분

- 피처이름에 Left, Right가 반복하여 등장함
- Left, Right 동일한 공정을 수행하는 별개의 라인이라면
  두 라인의 피처를 모두 사용하는 것은 동일한 공정이 모델링에 중복하여 영향력을 주는 것과 같음

분류 방법

LR, L_R 등 판단 불가한 피처를 우선 분류

↓

Left, Right 분류

↓

Left, Right 외의 모든 피처는 공통 생산라인으로 간주

라인별 피처 개수

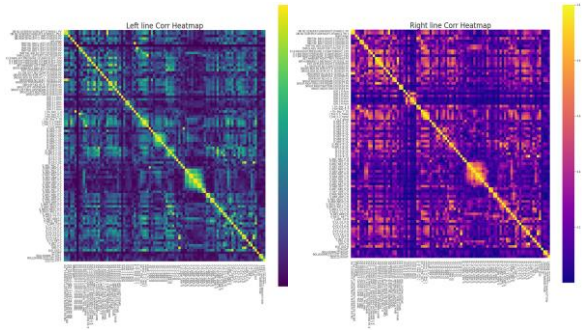
Left 피처 : 95 개
Right 피처 : 91 개

Left, Right 피처이름 비교

```
'Left': [   'D8.N2.SCREEN.FLOW.LEFT.1FI49011_PV',
            'D8.N2.SCREEN.FLOW.LEFT.1FI49011_PV.1',
            'D8.HEAT.BTM.LEFT.1.TMP.1TI40315.PV',
            'D8.HEAT.BTM.LEFT.2.TMP.1TI40316.PV',
            'Left.edge',
            'TMP.TIN..BAY.1.LEFT.1TI30201.PV',
            'TMP.TIN..BAY.4.LEFT.1TI30203.PV',
            'TMP.TIN..BAY.5.LEFT.1TI30205.PV',
            'TMP.TIN..BAY.7.LEFT.1TI30207.PV',
            'TMP.TIN..BAY.10.LEFT.1TI30209.PV',
            'TMP.GLASS..LEFT.EXIT.1TIC30109.PV',
            'X.10.BAY.LEFT.PRESSURE.1CLBAY10LEFT_CP
            'X.1.BAY.LEFT.PRESSURE.1CLBAY1LEFT_CPV',
            'X.7.BAY.LEFT.PRESSURE.1CLBAY7LEFT_CPV',
            'RET.AMBIANT.LEFT.TMP.1TI42602.PV',
            'BATH.AMBIENT.3.BAY.LEFT.1TI31003_PV',
            'EXIT.LIP.PLATE.LEFT.1TI30604.PV',
'Right': [  'D8.N2.SCREEN.FLOW.RIGHT.1FI49012_PV',
            'D8.N2.SCREEN.FLOW.RIGHT.1FI49012_PV.1',
            'Right.edge',
            'TMP.TIN..BAY.1.RIGHT.1TI30202.PV',
            'TMP.TIN..BAY.5.RIGHT.1TI30206.PV',
            'TMP.TIN..BAY.7.RIGHT.1TI30208.PV',
            'TMP.TIN..BAY.10.RIGHT.1TI30210.PV',
            'TMP.GLASS..RIGHT.EXIT.1TIC30111.PV',
            'X.10.BAY.RIGHT.PRESSURE.1CLBAY10RIGHT_CPV',
            'X.1.BAY.RIGHT.PRESSURE.1CLBAY1RIGHT_CPV',
            'X.7.BAY.RIGHT.PRESSURE.1CLBAY7RIGHT_CPV',
            'RET.AMBIANT.RIGHT.TMP.1TI42603.PV',
            'BATH.AMBIENT.3.BAY.RIGHT.1TI31002_PV',
            'EXIT.LIP.PLATE.RIGHT.1TI30605.PV',
            'E.L.DB.RIGHT.N2.FLOW.1FI33502.PV',
            'DROSS.BOX.N2.BTM.HT1.R.1JI39006.PV',
            'DROSS.BOX.N2.BTM.HT..R..1JI39010.PV',
```

- 각 라인에 속한 피처 이름이 매우 유사함
- 동일한 이름 패턴에서 라인 별로 피처 수가 다소 차이남

Left, Right의 상관계수 히트맵



- 각 라인에서 피처간 관계는 비슷함

두 생산라인 중 하나만 선택하여 피처의 영향력을 조사함
피처 수가 적은 Right 라인의 피처를 제외함

PCA 분석 수행

- 전처리 수행한 피처 개수만큼 PCA 분석을 통해 주성분을 추출함
- **각 피처와 1:1 대응되는 누적 기여율**과 **Elbow Point**를 확인함
- MinMax Scaler를 이용하여 스케일링 수행함

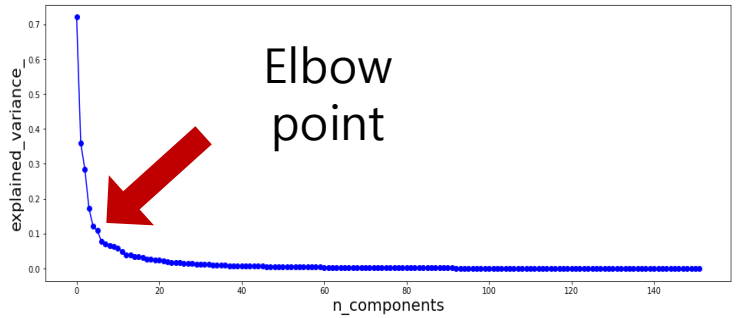
누적기여율 0.9이상의 피처개수 – 24개

	설명가능한 분산 비율(고윳값)	기여율	누적기여율
pca1	23.799554	0.146029	0.146029
pca2	13.579613	0.083322	0.229351
pca3	10.040242	0.061605	0.290956
pca4	7.481348	0.045904	0.336861
pca5	6.179061	0.037914	0.374774
...	...	...	...
pca53	0.717442	0.004402	0.882534
pca54	0.695238	0.004266	0.886800
pca55	0.661487	0.004059	0.890859
pca56	0.647054	0.003970	0.894829
pca57	0.610124	0.003744	0.898572

- **누적 기여율이 0.9 이상이 되는 피처의 개수는 24개**

→ **24개의 피처로 전체 데이터의 90% 설명 가능함**

PCA설명변수 Elbow Point - 7개



- 전체 데이터에 대해 가장 잘 설명하는 주성분의 설명변수는 0.7임
- 감소폭이 완만해지기 전까지 7개의 주성분으로 전체 데이터를 전반적으로 설명할 수 있음

→ **데이터를 전반적으로 설명하기 위해 최소 7개의 피처가 필요함**

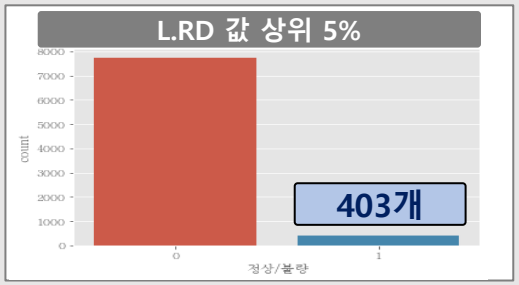
데이터 전처리 및 변환

5) 레이블 분류 기준

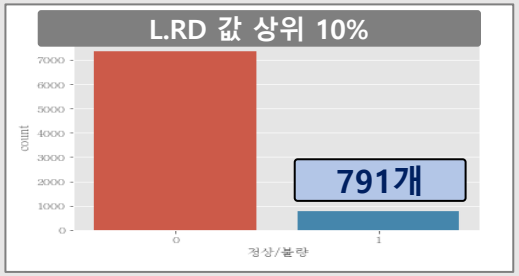
레이블 분류 기준 정하기

불량품 선정 기준

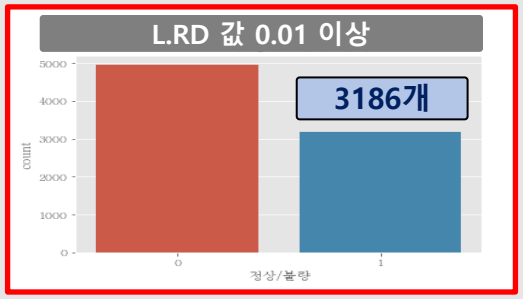
L.RD 값 상위 5%



L.RD 값 상위 10%

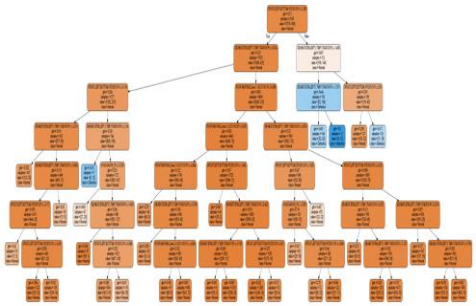


L.RD 값 0.01 이상

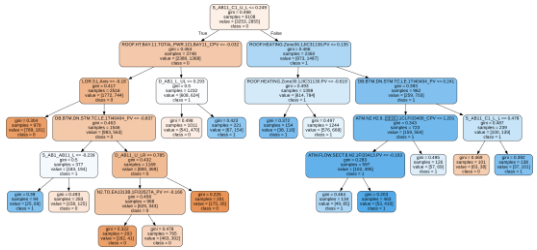


XGBoost Tree

L.RD 값 상위 10%



L.RD 값 0.01 이상



결론

- 상위 10% 이내의 값을 불량품으로 정하는 경우, 불균형한 Tree가 생성됨
- Tree 구조의 개선을 위해 불량품 데이터에 대해 OverSampling을 수행할 수 있으나, 실제 데이터에 대한 Tree 탐색의 의미가 사라짐
- 양품/불량품 개수가 불균형한 레이블 분류 기준은 불량품에 영향을 줄 수 있다고 판단되는 피처 탐색에 타당하지 않음

L.RD 값 0.01 이상을 불량품으로 처리함

불량에 영향을 주는 피처를 찾는 것이 프로젝트의 핵심임

→ 양품/불량품 비율이 비슷한 0.01로 설정하여 분류 모델 재수행

모델링

1) 평가지표 개념정리 및 선정

2) 분류 모델 알고리즘 개념정리

3) Decision Tree의 분류 기준

4) XGBoost 훈련 및 검증

5) Random Forest 훈련 및 검증

6) SVM 훈련 및 검증

7) RFE 수행

4

평가지표 선정

- 이 데이터에서는 극소수의 불량품을 판정하지 못하는 것이 치명적임
- 따라서 정확도는 적절치 않음
- 양품이라고 판정했지만 실제 불량이 발생한 경우를 예방해야 하므로 **재현율(Recall)**이 적절한 평가지표

정확도(Accuracy)	전체데이터에서 양품과 불량품을 올바르게 분류한 것의 비율
정밀도(Precision)	불량이라고 판정된 것 중에 실제 불량률인 것의 비율
재현율(Recall)	실제 불량인 것 중에서 불량이라고 예측한 것의 비율
F1 Score	정밀도와 재현율의 조화평균
ROC 곡선	FP 비율에 대한 TP의 비율을 나타내는 곡선

SVM, RandomForest, XGBoost의 이론 비교

단일모델

SVM

- 범주, 수치 예측 문제에 사용
- 오류 데이터 영향이 적음
- 신경망보다 사용하기 쉬움

Emsemble

Random Forest

- Bagging 방식을 이용함
- 결정트리 기반의 알고리즘 모델을 여러 개 생성하고 각각 다르게 샘플링 된 데이터로 훈련함
- 각 모델 중 분류 성능이 뛰어난 모델을 선택함

XGBoost

- Gradient Boost 방식을 이용함
- 여러 개의 분류 모델을 순차적으로 학습시키는 방식
- 이전 분류에서 잘못 분류한 데이터에 대해 가중치를 부여하여 다음 분류 모델을 훈련함
- 내부적으로 교차 검증을 수행하여 최적화된 반복 수행 횟수를 가질 수 있음

Decision Tree의 분류 기준 지표

엔트로피

$$H = -\sum_{i=1}^{N} p_i \log(p_i)$$

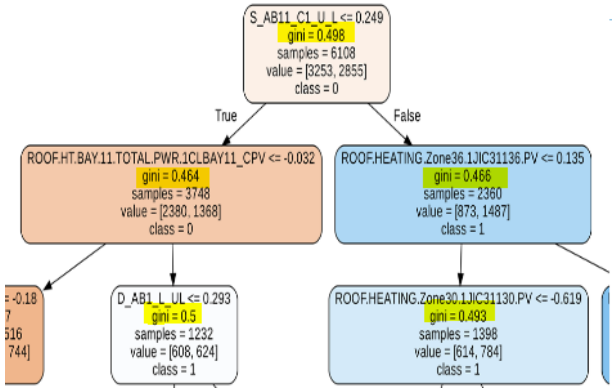
데이터가 불균형하게 분류되면
엔트로피가 감소함

Gini 계수 (Gini 불순도)

$$GINI(t) = 1 - \sum_{k} \left[p(C_k|t)^2\right]$$

클래스가 공평하게 섞여 있을수록
지니 계수가 상승함

➢ 트리구조 예시



Tree 구조에서 중요한 피처 찾는 방법

- **다음에 해당하는 노드를 찾아냄**
  - 지니계수가 낮고
  - 분류할 샘플 데이터 수가 많고
  - 불량(1)으로 분류한 것의 개수가 많은 노드
- 어떤 노드를 기준으로 그 아래 파랑, 빨강 박스가 나뉘면 그 피처 기준으로 잘 분류한 것이므로 중요한 피처임
- 피처 수가 적고, 트리 구조가 깊어서 어떤 피처가 반복하여 등장한다면 많이 등장한 피처가 중요한 피처임
- 가장 먼저 노드를 나누는 피처가 중요한 피처임

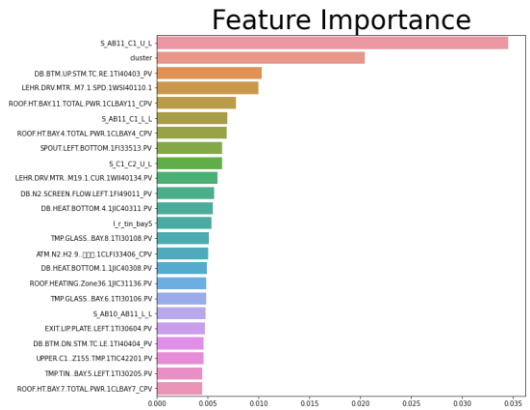
✲ 모델링

하이퍼파라미터 조정

```
테스트 데이터 수 (2037,)
오차행렬:
 [[756 314]
 [372 595]]
정확도: 0.663
정밀도: 0.655
재현율: 0.615
f1 스코어: 0.634
roc_auc_score: 0.723
```

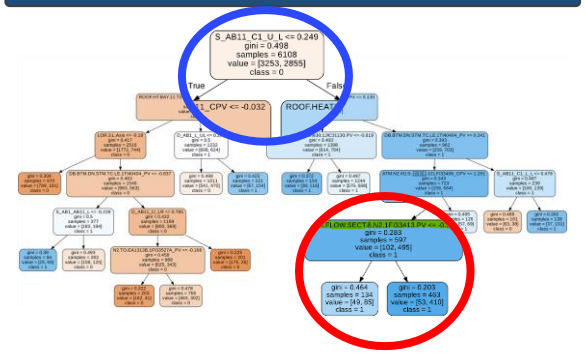
✲ 중요한 피처 탐색

feature_importance 확인



S_AB11_C1_U_L
cluster
DB.BTM.UP.STM.TC.RE.1TI40403_PV
LEHR.DRV.MTR..M7.1.SPD.1WSI40110.1
ROOF.HT.BAY.11.TOTAL.PWR.1CLBAY11_CPV

트리구조 탐색



S_AB11_C1_U_L
ROOF.HEATING.Zone36.1JIC31136.PV
DB.BTM.DN.STM.TC.LE.1TI40404_PV
ATM.N2.H2.9..함유율.1CLFI33406_CPV
ATM.FLOW.SECT.8.N2.1FI33413.PV

* feature_importance_의 상위 24개에 속하지 않은 피처를 파란색으로 표시함

XGBoost 중요 피처

- ➢ feature importance와 트리구조 모두에서 중요한 피처로 선정됨
  - S_AB11_C1_U_L
- ➢ 아래 피처는 두 기준 중 하나에서 중요한 피처로 추정됨
  - DB.BTM.DN.STM.TC.LE.1TI40404_PV
  - DB.BTM.UP.STM.TC.RE.1TI40403_PV
  - ROOF.HEATING.Zone36.1JIC31136.PV
  - ROOF.HT.BAY.11.TOTAL.PWR.1CLBAY11_CPV
  - ATM.N2.H2.9..함유율.1CLFI33406_CPV
  - ATM.FLOW.SECT.8.N2.1FI33413.PV

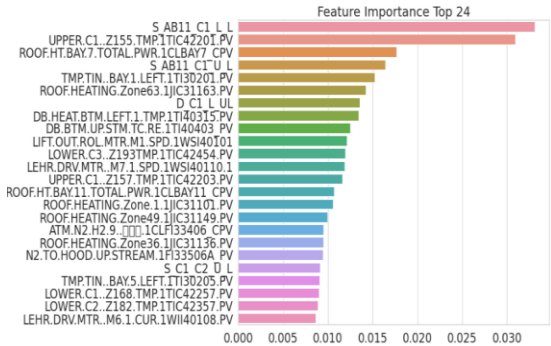
✲ 모델링

하이퍼파라미터 조정

```
오차행렬
[[620 233]
 [291 485]]
정확도 : 0.6783
정밀도 : 0.6755
재현율 : 0.6250
F1 스코어 : 0.6493
roc_auc_score : 0.7378
```

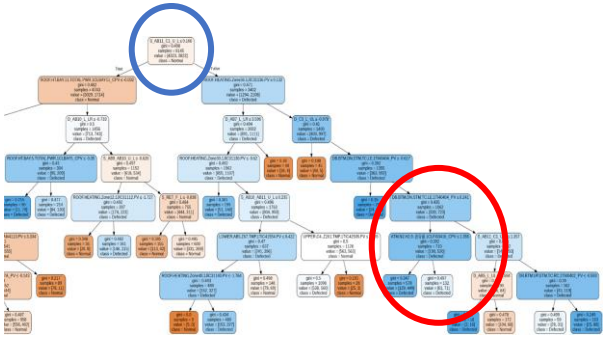
✲ 중요한 피처 탐색

feature_importance 확인



S_AB11_C1_L_L
UPPER.C1..Z155.TMP.1TIC42201.PV
ROOF.HT.BAY.7.TOTAL.PWR.1CLBAY7_CPV
S_AB11_C1_U_L
TMP.TIN..BAY.1.LEFT.1TI30201.PV

트리구조 탐색



ROOF.HT.BAY.5.TOTAL.PWR.1CLBAY5_CPV
ATM.N2.H2.9..함유율.1CLFI33416_CPV
DB.BTM.UP.STM.TC.RC.1TI40402_PV
DB.BTM.UP.STM.TC.LE.1TI40404_PV
D_C3_L_UL

RandomForest 중요 피처

➢ feature importance와 트리구조 모두에서 중요한 피처로 선정됨

- S_AB11_C1_U_L
- ROOF.HT.BAY.11.TOTAL.PWR.1CLBAY11_CPV
- ROOF.HEATING.Zone36.1JIC31136.PV

모델링

SVC 수행

```
테스트 데이터 수 (2037, 1)
오차행렬:
 [[703 399]
 [315 620]]
정확도: 0.649
정밀도: 0.608
재현율: 0.663
f1 스코어: 0.635
```

중요한 피처 탐색

SVR 수행

```
MAE: 8.968 MSE: 80.434 RMSE: 8.969
```

SVR의 Coef_ 계수 확인

bay1_2	0.047711
D_AB1_L_LR	0.046373
ROOF.HEATING.Zone31.1JIC31131.PV	0.039751
ROOF.HEATING.Zone86.1JIC31186.PV	0.034475
TMP.GLASS..BAY.6.1TI30106.PV	0.032489
Gross.width	0.032393
ROOF.HEATING.Zone45.1JIC31145.PV	0.026526
D_GLS_AB11_LR	0.023391
ATM.FLOW.SECT.4.N2.1FI33405.PV	0.022054
D_AB11_L_LR	0.021101

SVM 중요 피처

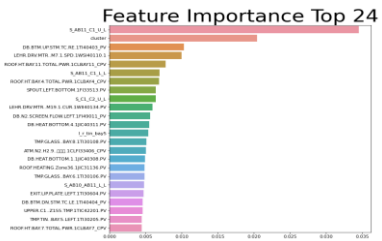
➢ feature importance와 트리구조 모두에서 중요한 피처로 선정됨

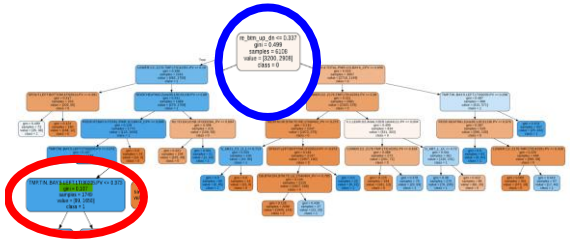
- S_AB11_C1_U_L
- ROOF.HT.BAY.11.TOTAL.PWR.1CLBAY11_CPV
- ROOF.HEATING.Zone36.1JIC31136.PV
- TMP.GLASS..BAY.61TI30106.PV
- ROOF.HEATING.Zone31.1JIC31131.PV

RFE수행

- 앞서 수행한 PCA주성분 분석 결과에 따르면 24개의 피처로 전체 데이터의 90%설명이 가능함
- 따라서 RFE를 이용하여 24개 피처를 추출함
- **XGBoost**와 **RandomForest** 알고리즘을 이용하여 비교함

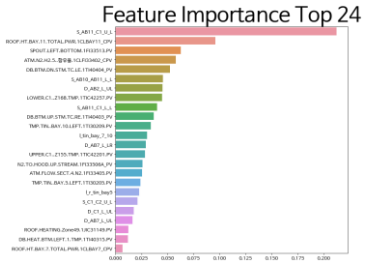
XGboost

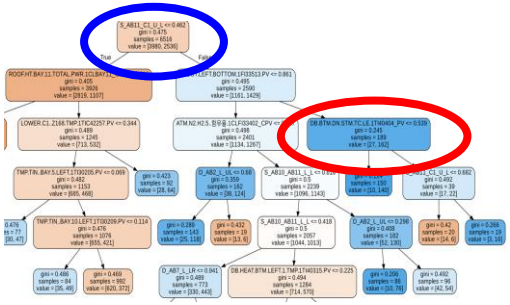




TMP.TIN..BAY.5.LEFT.1TI30205.PV
re_btm_up_dn
LOWER.C2..Z176.TMP.1TIC42351.PV
ROOF.HEATING.Zone36.1JIC31136.PV
ROOF.HT.BAY.4.TOTAL.PWR.1CLBAY4_CPV

RandomForest





S_AB11_C1_U_L
SPOUT.LEFT.BOTTOM.1FI33513.PV
DB.BTM.DN.STM.TC.LE.1TI40404_PV
D_AB2_L_UL

결론

1) 중요 피처 선택

2) 프로젝트 결론

5

최종 중요 피처 선택

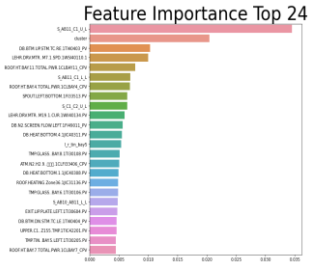
XGBoost 모델

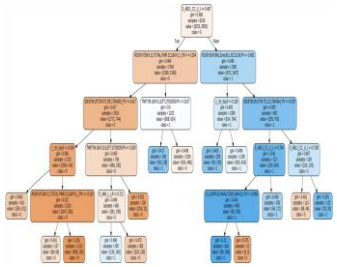
- 모델링 방식 중 재현율이 가장 우수한 XGBoost 모델을 이용함

```
테스트 데이터 수 (2037,)
오차행렬:
 [[756 314]
 [372 595]]
정확도: 0.663
정밀도: 0.655
재현율: 0.615
f1 스코어: 0.634
roc_auc_score: 0.723
```

중요 피처 확인

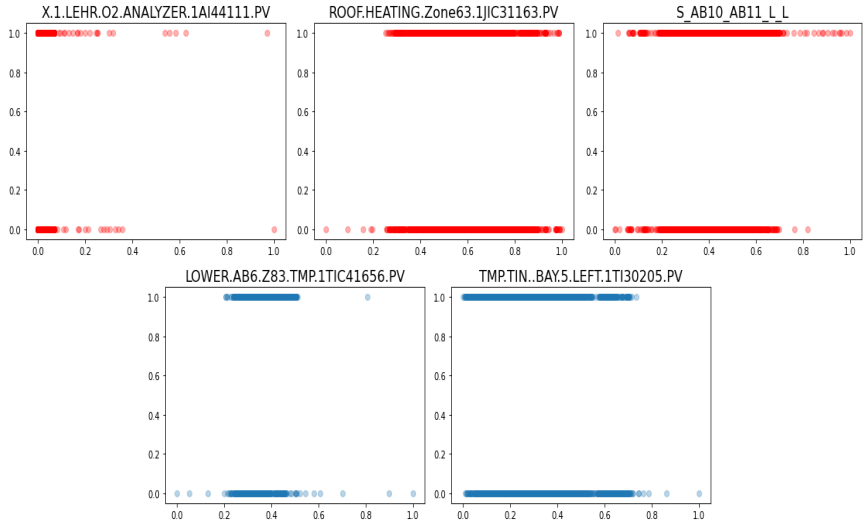
- feature importance 상위 24개의 피처에 대해 Tree 구조 확인 및 Gini 계수 탐색





중요 피처 선택

- 중요한 피처란 불량률에 큰 영향을 미치는 피처임
- 중요 피처를 선정하여 레이블과의 관계를 확인함



중요 피처

LOWER.AB6.Z83.TMP.1TIC41656.PV
TMP.TIN..BAY.5.LEFT.1TI30205.PV
X.1.LEHR.O2.ANALYZER.1AI44111.PV
ROOF.HEATING.Zone63.1JIC31163.PV
S_AB10_AB11_L_L

생산공정 수율 안정화를 위한 중요 피처 찾기 프로젝트 결론

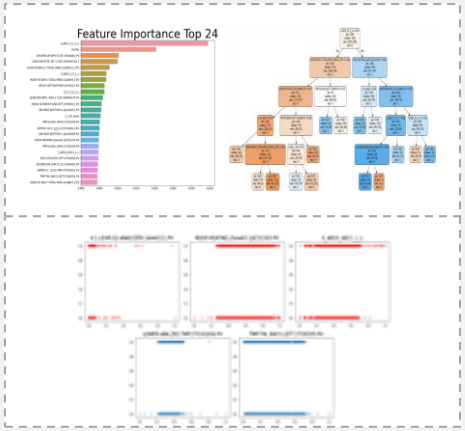
목적

- 피처 값의 범위에 따른 양품/불량품 확률로부터 불량품에 가장 큰 영향을 미치는 피처 5개를 도출함

수행

- 불량품을 검출하는 분류 모델을 수립함
- 불량품에 영향을 주는 피처 5개 검출함
- 피처 조건에 따른 불량률 판단기준 확립함

5개 피처 도출



활용방안

데이터 분석 결과를 생산시스템에 적용하면
다음과 같은 측면에서 현장운영이 수월해질 것으로 기대됨

1) 수율안정화
2) 현장관리 수준향상
3) 지속적으로 불량요인을 발생시키는 피처 데이터를 축척함으로써 지도학습 모델을 이용한 불량탐지 자동 검출 시스템 구축가능

6

느낀 점

06 느낀 점

구혜진

처음부터 끝까지 스스로 코드를 짜고 정리해 본 첫 프로젝트였습니다. 전체 흐름을 정리하면서 머신러닝에 대해 좀더 잘 알게 되었습니다. 팀원들과 효과적으로 의사소통하는 방법을 찾아보는 시간이었습니다.

김민기

4개월이라는 짧고도 긴 시간동안 많은 것을 배우고 따라해 보면서 처음에는 프로젝트를 큰 어려움없이 진행 할 수 있다고 생각했습니다. 막상 프로젝트 데이터를 받고 나니 전처리부터 시작하여 프로젝트 모든 부분에 어려움을 느꼈습니다.
하지만 선생님과 팀원들의 도움으로 프로젝트 방향성을 잡고 수업시간에 배운 것들을 복습, 정리해가면서 프로젝트를 진행해 나갈 수 있었습니다. 비록 팀원들에게는 큰 도움이 되지 못했지만 팀원들과 코드를 공유하고 머신 러닝과 통계,프로그래밍언어 등 다양한 의견을 나눌 수 있어 좋았고 그로 인하여 한단계 더 발전할 수 있었다고 생각합니다.

이세희

아쉬웠던 점은 처음에 레이블이나 센서 정보에 대한 이해가 부족해 프로젝트 방향성을 납득하는데 시간이 걸렸습니다.
비대면으로 진행한 팀원들과 zoom이나 slack을 이용해 소통을 하려고 노력했지만 직접 대면하는 팀원들보다는 소통이 원활하지 않았던 것 같아 아쉽습니다.
반면 좋았던 점은 짧은 훈련 기간동안 한꺼번에 새롭고 많은 지식을 습득하여 이를 활용할 기회가 부족했는데 프로젝트를 통해 파이썬 언어에 대한 실력향상 뿐 아니라 머신 러닝과 기술통계에 대해 다잡을 수 있어 행운이었다고 생각합니다. 또한 성능을 개선시키기 위해 수정반복한 작업들이 지치긴 했지만 다음 프로젝트 때는 모델링 과정에 대해 작업내용을 조건 별로 정리하고 효율적인 방법을 선택해 나간다면 좋을 것 같습니다

이호영

프로젝트를 진행하면서 수업시간에 배운 내용들을 더 적확하게 이해 할 수 있었습니다. 팀원들과 활용 데이터가 조금씩 달라서 같은 작업을 또 수행하는 어려움이 있었습니다. 프로젝트 후반기에는 반복적인 작업만 했기 때문에 지쳤습니다

서영흔

다른 팀원들에 비해 data처리를 소홀히 한 것 같습니다 data mining 생각을 왜 못했을까?너무 코딩 짜는 거에 치우쳐 효율적으로 프로젝트진행을 못한 것 같습니다. 포인트를 잘 파악하지 못했던 것 같아 아쉬움이 남는 프로젝트였습니다.
데이터프레임에 대해 좀더 알게 되고 직접 사용하는 계기가 되어 매우 좋았습니다. 그리고 프로젝트를 하며 Pandas, Numpy 그리고 여러 함수들을 사용하며 그동안 배웠던 것을 응용한 것 같아 보람찼습니다

황정석

프로젝트를 수행하면서 다양한 기술들에 활용됨에 따라 그 기술에 대한 학습의 필요성을 깨우치게 되었습니다.
프로젝트 방향에 대해 어려움이 있었는데 실력 있는 팀원들 덕분에 큰 도움을 얻었습니다. 이번 경험을 통해 기술적인 부분과 data mining에 대한 부분을 더 학습하겠습니다.

프로젝트 소스코드

https://github.com/ammobam/Display_SensorData