상수관로 누수감지 및 분류

2022 인공지능 온라인 경진대회

참여인원: 1명
2022.06 ~ 2022.08

CONTENTS

1. 프로젝트 소개

- 수치해석 분야 - 상수관로 누수감지 및 분류 문제
  - 프로젝트 설명
    - 상수관로 진동 센서 데이터로 누수 유형을 분류하는 문제
  - 추진 배경
    - 전국 수도관의 13%가 30년 이상 된 노후관로이며, 이는 상수도 품질 저하의 주요 원인
    - 상수관로 누수 감지 및 분류를 자동화하여 시간과 비용을 절감할 수 있음
  - 활용 서비스
    - 센서 기반 누수 자동 탐지 솔루션
  - 평가지표
    - Macro F1 Score

2. 데이터

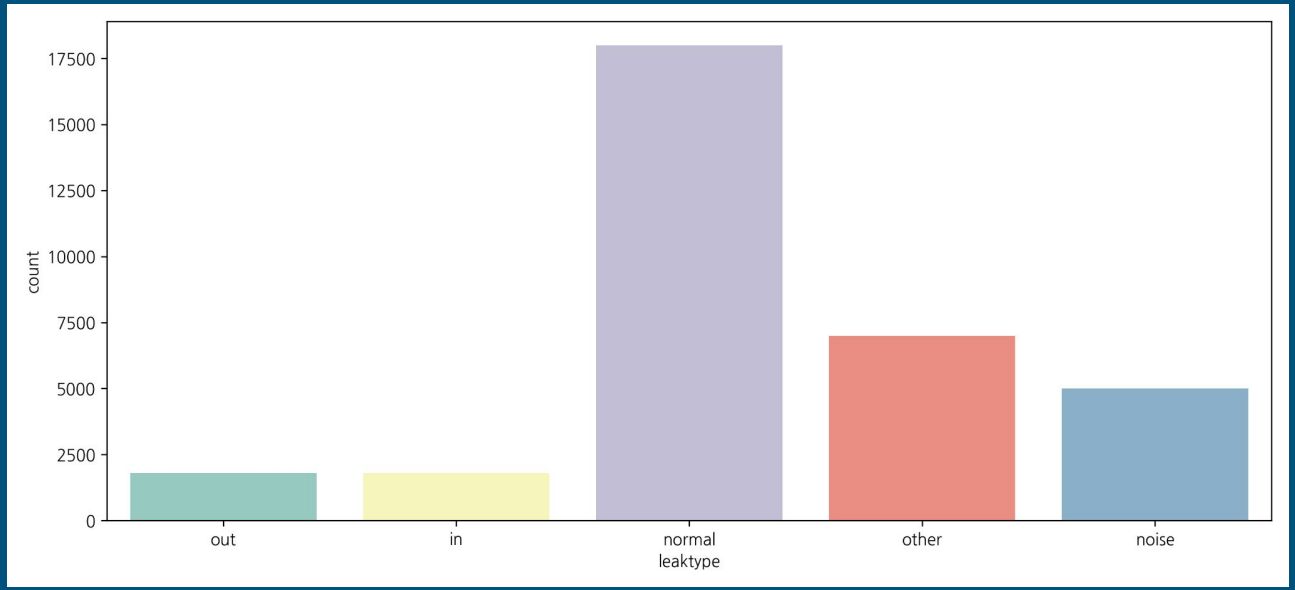
- 독립변수
  - 상수관로 진동 센서 데이터
  - 센서 출력값에 Fourier Transform을 적용하여 계산한 주파수 별 Spectral Density 값
  - 0Hz부터 5120Hz까지 10Hz단위로 수집된 513개 Columns
- 종속변수
  - 누수 구분 클래스(Leaktype)
  - 옥외누수(Out), 옥내누수(In), 정상(Normal), 전기/기계음(Noise), 환경음(Other)

	leaktype	0HZ	10HZ	20HZ	30HZ	40HZ	50HZ	60HZ	70HZ	80HZ	...	5030HZ	5040HZ	5050HZ	5060HZ	5070HZ	5080HZ	5090HZ	5100HZ	5110HZ	5120HZ
0	out	0	2	2	0	2	0	2	2	2	...	2	5	2	2	5	2	2	5	2	5
1	out	0	0	0	3	0	3	0	0	0	...	0	3	3	3	3	3	3	3	3	6
2	out	0	4	4	4	4	5	4	4	5	...	5	5	6	5	6	6	6	5	6	4
3	out	0	6	5	5	6	5	6	6	5	...	6	6	7	7	5	6	5	5	7	7
4	out	0	3	0	0	3	0	0	3	3	...	3	3	3	3	3	3	3	3	3	3

[데이터 프레임 출력 결과]

2. 데이터

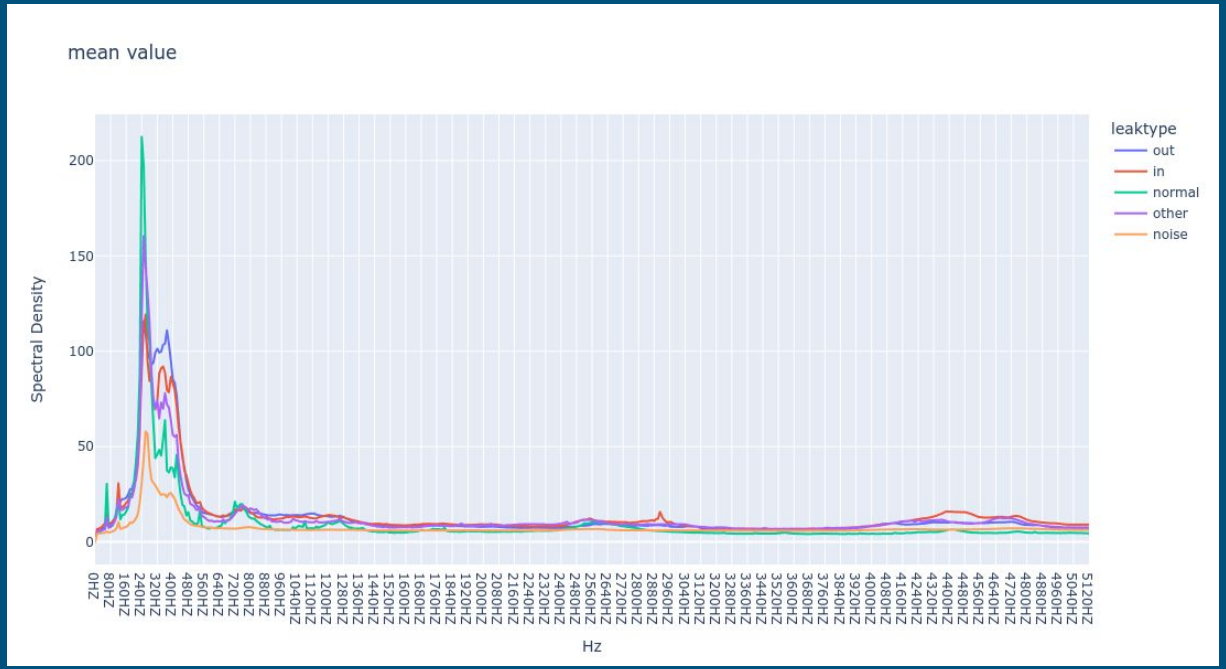
- 종속변수 분포
  - 총 33600개(Normal 18000 / Other 7000 / Noise 5000 / Out 1800 / In 1800)



[종속변수 카테고리별 데이터 수 비교]

2. 데이터

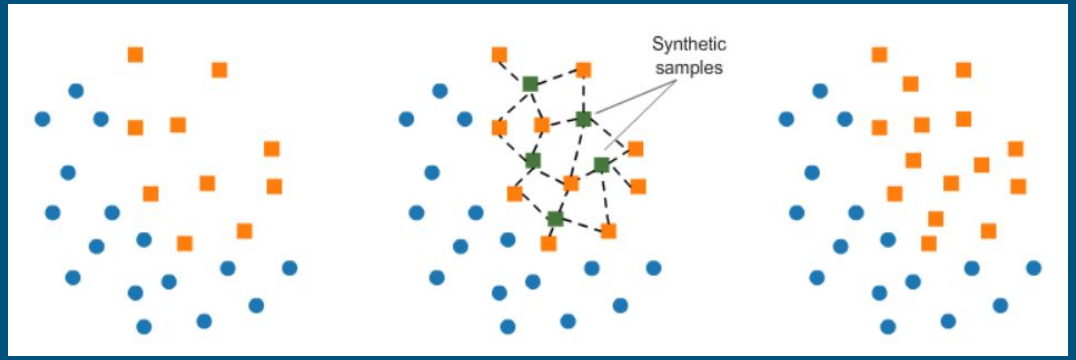
- 카테고리 그룹별 Spectral Density 평균값 시각화



[종속변수의 카테고리 그룹별 Spectral Density 평균값 비교]

3. 전처리

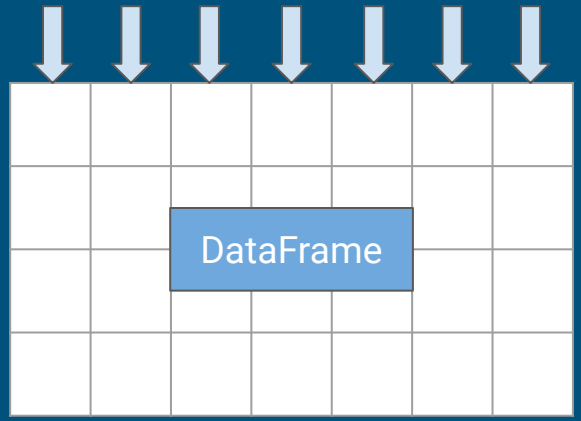
- 클래스 불균형 문제
  - 본 데이터에는 클래스 불균형 문제가 있음('Normal' 카테고리 비율 53.6%)
  - 경진대회의 평가지표가 **F1 Score** 이므로 이 문제를 해결하는 것이 매우 중요
  - 해결 방법으로 SMOTE(Synthetic Minority Over-sampling Technique) 기법을 채택
    - SMOTE: 소수 클래스의 샘플 사이에 가상의 샘플을 생성하여 데이터셋을 균형있게 만드는 오버샘플링 기법



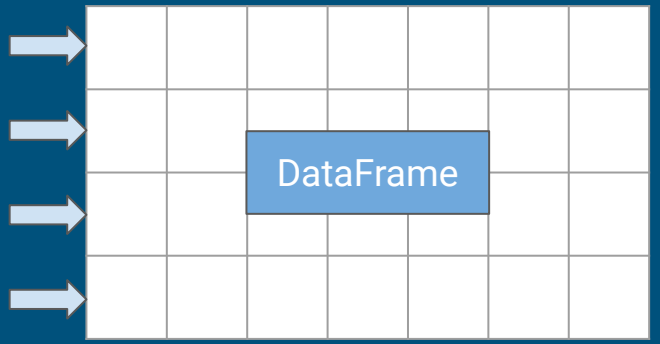
[SMOTE 알고리즘을 표현한 그림]

3. 전처리

- 두 가지 정규화(Normalization) 방법
  - 데이터 내에서 최대한의 정보를 추출하기 위해 두 가지 방법으로 데이터를 해석함
    - Column Based: 일반적인 정규화 방법으로, 열(Column)을 기준으로 Scaler 적용
    - Row Based: **각 데이터의 추세**를 중요하게 생각하여, **행(Row)을 기준으로 Scaler 적용**



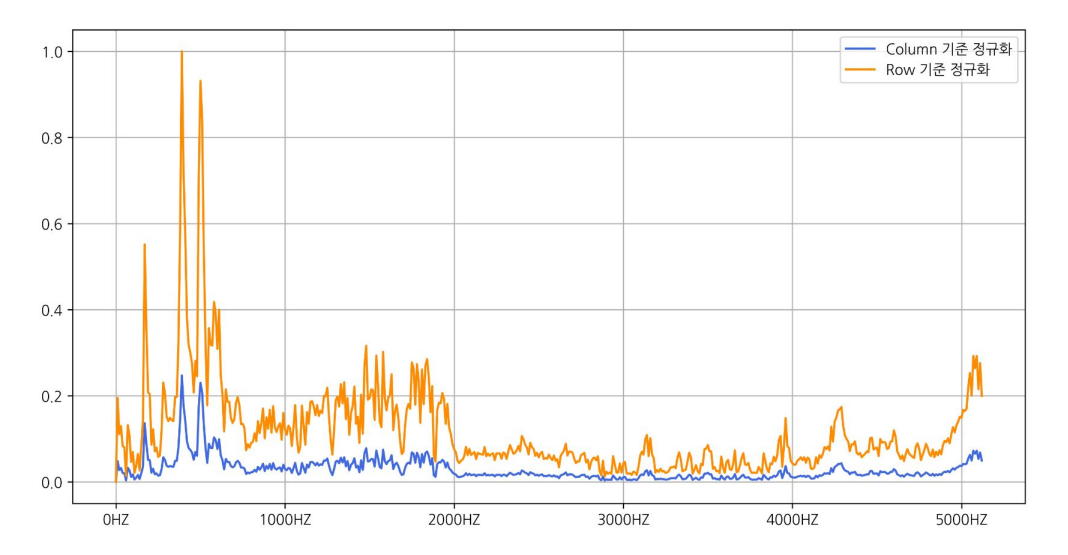
[Column Based Normalization]



[Row Based Normalization]

3. 전처리

- 동일한 데이터에 대하여 두 가지 정규화 방법 적용 결과 비교
  - 파란색 선: Column Based Normalization 적용 결과 (MinMaxScaler)
  - 오렌지색 선: Row Based Normalization 적용 결과 (MinMaxScaler)



[정규화 방법에 따른 데이터 변화 비교]

3. 전처리

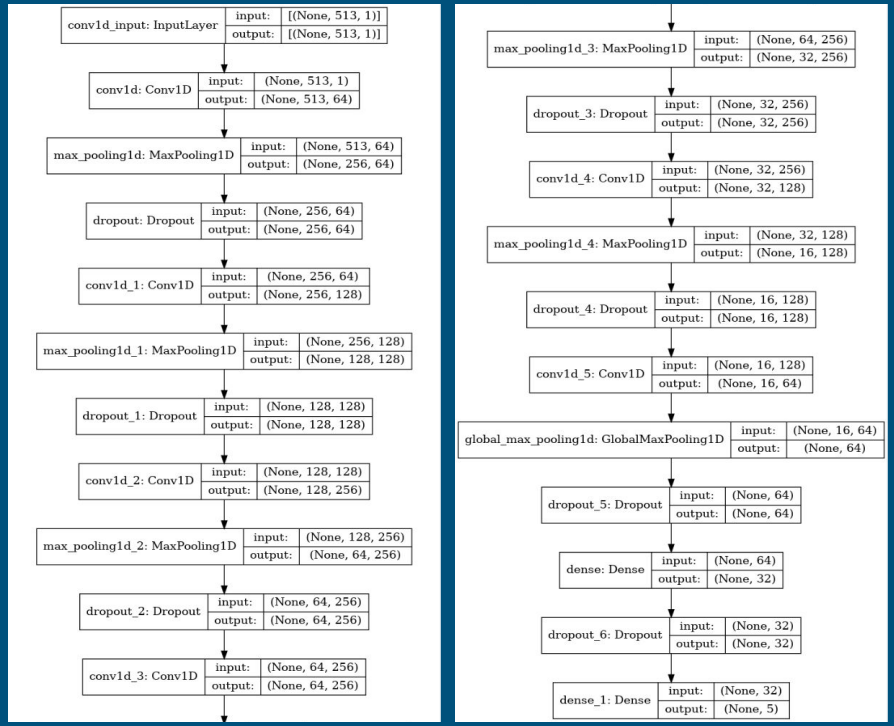
- 실패한 전처리 방법
  - 파생변수 생성
    - 최대 Hz 크기, 범위 정보
  - 군집화
    - Leaktype이 ‘Normal’인 카테고리 군집화, 특이한 특징을 갖는 군집은 학습에서 제외 (Undersampling)
    - ‘Normal’ 데이터를 군집별로 나누어서 각각 학습 후 가중치를 적용하여 모델 앙상블
  - Normal / Abnormal
    - Normal과 Abnormal을 먼저 분류한 뒤, Abnormal을 ‘Other’, ‘Noise’, ‘In’, ‘Out’ 으로 세부 분류

4. 모델링

- Tree-Based Models
  - AutoML 라이브러리 / 플랫폼 활용
    - Pycaret: AutoML 라이브러리
    - AIOTS: 자체 개발 AutoML 플랫폼
  - Ensemble
    - KNeighborsClassifier, XGBClassifier, RandomForestClassifier 모델을 Voting 기법으로 앙상블하여 Voting Classifier 생성
- DL Models
  - Conv1D 모델 1: Column Based Normalization 데이터 학습
  - Conv1D 모델 2: Row Based Normalization 데이터 학습
  - Conv1D 모델 3: Row Based Normalization 데이터 학습 + Class Weight 적용

4. 모델링

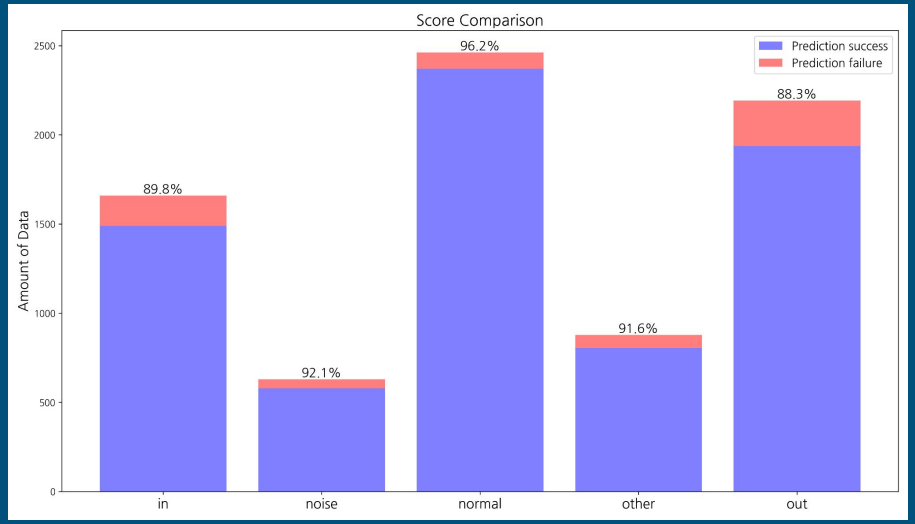
- Conv1D Model Architecture



[Conv1D 모델 구조 시각화]

5. 결과

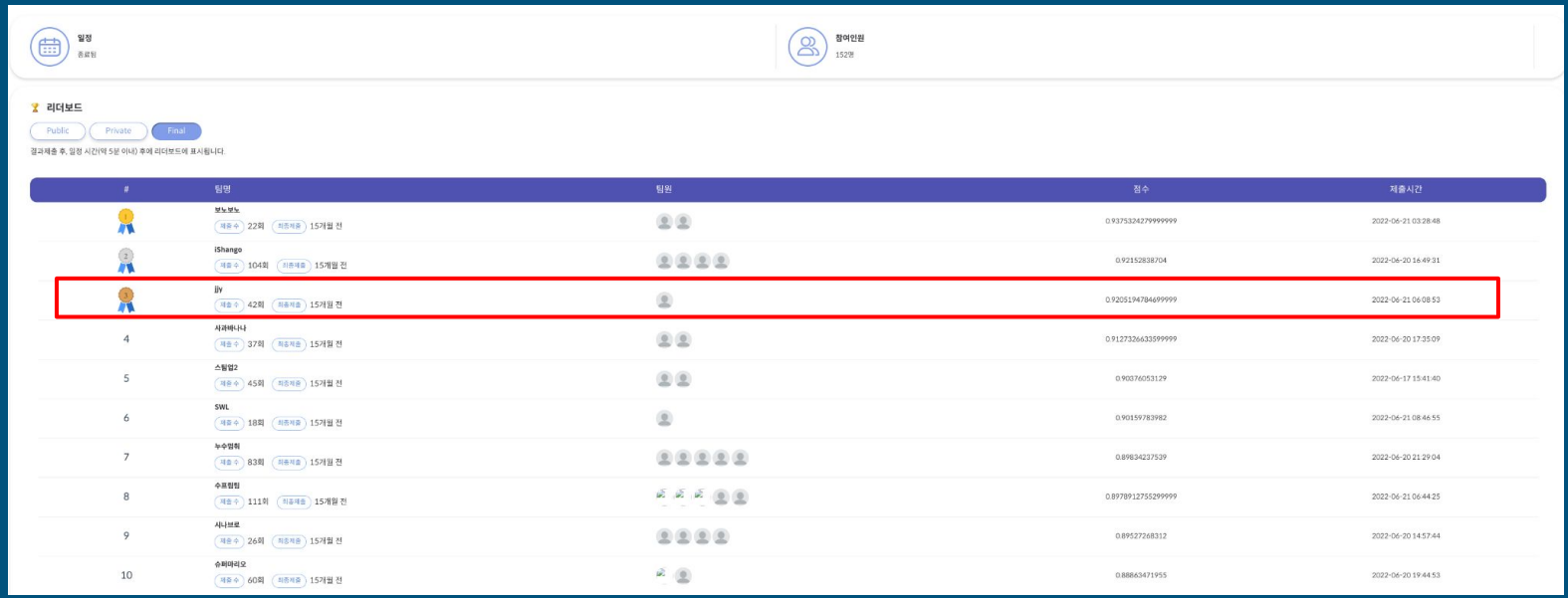
- 최종 모델 및 결과
  - 최종 모델은 Conv1D 모델 1,2,3 + VotingClassifier 총 4개 모델 Ensemble
  - 가장 정확도가 낮은 ‘In’과 ‘Out’에 가중치 적용
  - 각 단일 모델들의 F1 Score: 0.85~0.90 / 최종 모델(Ensemble)의 F1 Score: **0.92**



[카테고리별 예측결과 비교]

5. 결과

- 최종 모델 및 결과
  - Row Based Normalization이 유효하게 작용함(F1 Score: 0.91 -> 0.92로 향상)
  - 최종 3위 입상



일정
종료됨

참여인원
152명

리더보드

Public | Private | Final

결과제출 후, 일정 시간(약 5분 이내) 후에 리더보드에 표시됩니다.

#	팀명	팀원	점수	제출시간
1	보노보노 제출 수 22회 최종제출 15개월 전		0.9375324279999999	2022-06-21 03:28:48
2	iShango 제출 수 104회 최종제출 15개월 전		0.92152838704	2022-06-20 16:49:31
3	jjy 제출 수 42회 최종제출 15개월 전		0.9205194784699999	2022-06-21 06:08:53
4	사과바나나 제출 수 37회 최종제출 15개월 전		0.9127326633599999	2022-06-20 17:35:09
5	스칼업2 제출 수 45회 최종제출 15개월 전		0.90376053129	2022-06-17 15:41:40
6	SWL 제출 수 18회 최종제출 15개월 전		0.90159783982	2022-06-21 08:46:55
7	누수멈춰 제출 수 83회 최종제출 15개월 전		0.89834237539	2022-06-20 21:29:04
8	수표힘힘 제출 수 111회 최종제출 15개월 전		0.8978912755299999	2022-06-21 06:44:25
9	사나브로 제출 수 26회 최종제출 15개월 전		0.89527268312	2022-06-20 14:57:44
10	슈퍼마리오 제출 수 60회 최종제출 15개월 전		0.88863471955	2022-06-20 19:44:53

[대회 최종 리더보드 화면]

6. 보완점

- Tree-Based Model에도 Row Based Normalization 데이터를 학습시켜 볼 필요가 있었음
  - Row Based Normalization 아이디어는 DL Model에 적합하다고 생각하여 Tree-Based Model에 적용하지 않았음. 그러나 테스트는 필요

- ‘In’과 ‘Out’ 카테고리에 더 강한 가중치 적용
  - 예측 Probability를 기준으로 연산하여 소수 카테고리의 절대적인 예측 빈도를 늘려서 Test Set에 대한 F1 Score를 확인했어야 함

- 서버 PC를 충분히 활용하지 못함
  - DL Model의 최적 구조 파악에 더 많은 시간을 투자했어야 함

THANK YOU!