

유선호, 연현중, 정은택

머신러닝 초보가 머신러닝 경진대회에서 의미있는 실패 만들기

2024 제4회 K-water AI 경진대회 : 상수도 관망 이상 감지 AI 알고리즘 개발

“노베이스”였던 우리가 프로젝트를 시작하기도 전에 했던 고민들

“일단 대회는 찾았는데, 어떻게 프로젝트를 시작하지?”

“프로젝트는 어떤 진행 과정으로 진행하지?”

“이 프로젝트에 얼마만큼 투자를 하고, 나에게 얼마나 의미가 있을까?”

1. 프로젝트 소개

2024 제4회 K-water AI 경진대회 : 상수도 관망 이상 감지 AI 알고리즘 개발

알고리즘 | 시계열 | 정형 | 이상 탐지

₩ 상금 : 800만원

🕒 2024.11.22 ~ 2024.12.16 09:59 [+ Google Calendar](#)

👤 1,151명 📅 마감

1	나는야매스티쳐		0.45312
2	PQM		0.45
3	SKKU brAln		0.44444
4	AIME한테 DM해~		0.41538
5	쥬혁이		0.39393
6	일렷		0.13793
7	SeaweedSoup98		0.13725
8	naye971012		0.10644
9	Ryan.Ahn		
10	ahn24		

[배경]

본 경진대회는 상수도 관망의 이상 시점과 누수 발생 구간을 정확하게 탐지할 수 있는 범용 AI 알고리즘 개발을 목표로 하고 있습니다.

대회 참가자들에게 다양한 상수도 관망의 실제 데이터를 제공하여, 이를 기반으로 실시간으로 이상을 감지하고 효율적으로 의사결정을 지원할 수 있는 기술을 개발하도록 지원합니다.

개발된 알고리즘은 향후 상수관망 디지털트윈 및 Water-Net 등 사내 시스템에 내재화하여 상수도 관리의 효율성과 정확성을 높이하고자 합니다.

[주제]

상수도 관망 이상 감지 AI 알고리즘 개발

[설명]

다양한 상수도 관망의 실시간 이상을 감지하는 AI 모델을 개발해야 합니다.

학습 데이터는 A와 B 구조를 가진 상수도 관망 데이터로, 분 단위의 시간 정보가 모두 공개되어 있습니다.

반면, 평가 데이터는 C와 D 구조를 가진 상수도 관망 데이터로 제공되며, 현재 시점 T를 기준으로 시간이 비식별화되어 있습니다.

모델은 평가 데이터에서 최대 1주일 분량의 분 단위 입력 데이터를 바탕으로 T+1분 시점의 이상 여부를 감지해야 하며, 관망 구조 내 존재하는 각각의 압력계(P)에 대해 이상을 감지할 수 있어야 합니다.

[주최 / 운영]

- 주최: K-water
- 운영: 데이콘



대회 정보



대회 정보

2. 프로젝트 접근 계획

2-I. 문제 정의하기



2-II. 문제 쪼개기



2-III. 쪼갬 문제 해결하는 방법론 만들기



2-IV. 실행하기

2-I. 문제 정의하기

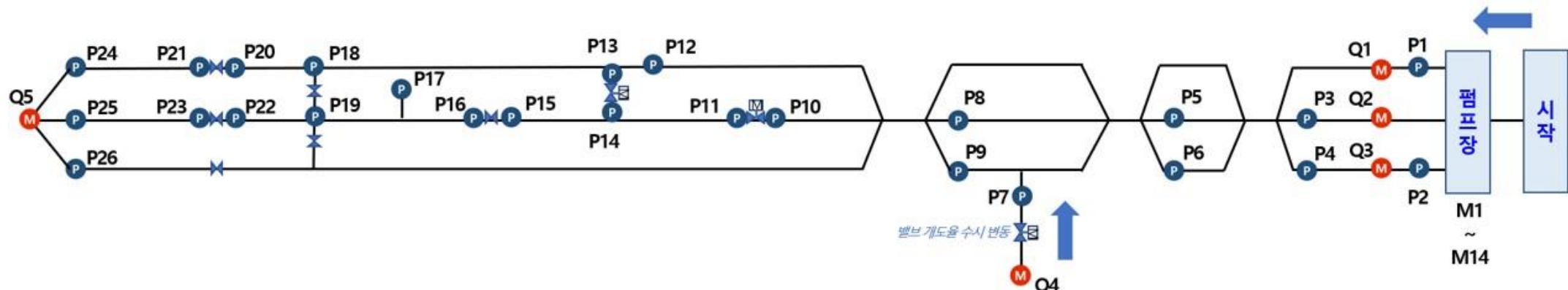
2-I. 문제 정의하기

2-II. 문제 쪼개기

2-III. 쪼갬 문제 해결하는 방법론 만들기

2-IV. 실행하기

관망 구조 A



		Q1	Q2	Q3	Q4	Q5	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16	P17	P18	P19	P20	P21	P22	P23	P24	P25	P26	anomaly	P1 flag	P2 flag	P3													
1	2024-05-27 0.00	17880	37151	24834	6321	85828	0	0	0	1	0	1	1	1	0	0	1	0	0	1	3.206	3.1621	0.7022	0.6956	1.0556	1.0425	3.9288	0.1163	0.1659	5.6972	5.7206	6.2991	6.3019	6.3197	3.2025	3.3588	3.5763	2.51	2.47	3.005	3	2.98	2.9875	2.8981	2.8175	2.7762	0	0	0
2	2024-05-27 0.01	17970	37029	25016	6367	85212	0	0	0	1	0	1	1	1	0	0	1	0	0	1	3.206	3.1566	0.7125	0.7041	1.0688	1.0584	3.9237	0.1144	0.1678	5.7028	5.7319	6.3141	6.2953	6.3019	3.2025	3.3588	3.5763	2.51	2.47	3.005	3	2.98	2.9875	2.8994	2.8163	2.775	0	0	0
3	2024-05-27 0.02	17280	37345	24462	6431	85655	0	0	0	1	0	1	1	1	0	0	1	0	0	1	3.2115	3.1566	0.6797	0.6919	1.0509	1.0191	3.925	0.0984	0.1725	5.6881	5.7319	6.2953	6.3009	6.3197	3.2025	3.3588	3.5763	2.51	2.47	3.005	3	2.98	2.9875	2.8912	2.8144	2.7725	0	0	0
4	2024-05-27 0.03	17280	37345	24462	6431	85619	0	0	0	1	0	1	1	1	0	0	1	0	0	1	3.2115	3.1556	0.6797	0.6919	1.0509	1.0191	3.925	0.0984	0.1725	5.6881	5.7319	6.2953	6.3009	6.3197	3.2025	3.3588	3.5763	2.51	2.47	3.005	3	2.98	2.9875	2.8956	2.8125	2.775	0	0	0
5	2024-05-27 0.04	17920	37075	24896	6206	85619	0	0	0	1	0	1	1	1	0	0	1	0	0	1	3.2005	3.1621	0.6844	0.6909	1.0631	1.0247	3.93	0.0947	0.1762	5.7	5.7431	6.2934	6.2972	6.3206	3.2025	3.3588	3.5763	2.51	2.47	3.005	3	2.98	2.9875	2.8956	2.8125	2.775	0	0	0
6	2024-05-27 0.05	17770	37061	24883	6426	85101	0	0	0	1	0	1	1	1	0	0	1	0	0	1	3.2005	3.1621	0.69	0.7097	1.0538	1.02	3.9225	0.1219	0.1612	5.6831	5.7159	6.3	6.3019	6.3244	3.2188	3.2613	3.6163	2.4362	2.4012	2.9062	3.0363	2.9962	3.0387	2.8881	2.8106	2.7762	0	0	0
7	2024-05-27 0.06	17770	37061	24883	6426	85101	0	0	0	1	0	1	1	1	0	0	1	0	0	1	3.2005	3.1621	0.69	0.7097	1.0538	1.02	3.9225	0.1219	0.1612	5.6831	5.7159	6.3	6.3019	6.3244	3.2188	3.2613	3.6163	2.4362	2.4012	2.9062	3.0363	2.9962	3.0387	2.8881	2.8106	2.7762	0	0	0
8	2024-05-27 0.07	17470	36868	25179	6251	85268	0	0	0	1	0	1	1	1	0	0	1	0	0	1	3.206	3.1621	0.7003	0.6956	1.0688	1.035	3.925	0.1106	0.1753	5.6881	5.745	6.3178	6.2906	6.3019	3.2188	3.2613	3.6163	2.4362	2.4012	2.9062	3.0363	2.9962	3.0387	2.8919	2.8175	2.7769	0	0	0
9	2024-05-27 0.08	17450	37602	25220	6239	85962	0	0	0	1	0	1	1	1	0	0	1	0	0	1	3.206	3.1621	0.69	0.7059	1.0556	1.0256	3.9188	0.09	0.1584	5.6813	5.7159	6.3103	6.2972	6.3047	3.2188	3.2613	3.6163	2.4362	2.4012	2.9062	3.0363	2.9962	3.0387	2.9012	2.8081	2.7706	0	0	0
10	2024-05-27 0.09	17900	37240	24846	6241	85997	0	0	0	1	0	1	1	1	0	0	1	0	0	1	3.2115	3.1456	0.6797	0.6853	1.0669	1.0341	3.9213	0.1013	0.1462	5.6841	5.7103	6.3066	6.3	6.3253	3.2188	3.2613	3.6163	2.4362	2.4012	2.9062	3.0363	2.9962	3.0387	2.8994	2.8087	2.7713	0	0	0
11	2024-05-27 0.10	17900	37240	24846	6241	85997	0	0	0	1	0	1	1	1	0	0	1	0	0	1	3.2115	3.1456	0.6797	0.6853	1.0669	1.0341	3.9213	0.1013	0.1462	5.6841	5.7103	6.3066	6.3	6.3253	3.2188	3.2613	3.6163	2.4362	2.4012	2.9062	3.0363	2.9962	3.0387	2.8994	2.8087	2.7713	0	0	0
12	2024-05-27 0.11	17050	37091	25258	6346	84915	0	0	0	1	0	1	1	1	0	0	1	0	0	1	3.2005	3.1401	0.7097	0.6969	1.0378	1.0175	3.9288	0.1566	0.2756	5.6756	5.7216	6.2925	6.2888	6.3084	3.2188	3.2613	3.6163	2.4362	2.4012	2.9062	3.0363	2.9962	3.0387	2.8906	2.8106	2.7738	0	0	0
13	2024-05-27 0.12	18050	37469	24613	6157	85784	0	0	0	1	0	1	1	1	0	0	1	0	0	1	3.2005	3.1401	0.6891	0.6938	1.0538	1.0181	3.9237	0.0872	0.1462	5.6784	5.7309	6.3141	6.3009	6.3159	3.2188	3.2613	3.6163	2.4362	2.4012	2.9062	3.0363	2.9962	3.0387	2.88	2.7956	2.7569	0	0	0
14	2024-05-27 0.13	18050	37469	24613	6157	85784	0	0	0	1	0	1	1	1	0	0	1	0	0	1	3.2005	3.1401	0.6891	0.6938	1.0538	1.0181	3.9237	0.0872	0.1462	5.6784	5.7309	6.3141	6.3009	6.3159	3.2188	3.2613	3.6163	2.4362	2.4012	2.9062	3.0363	2.9962	3.0387	2.88	2.7956	2.7569	0	0	0
15	2024-05-27 0.14	17460	36988	25076	6141	84996	0	0	0	1	0	1	1	1	0	0	1	0	0	1	3.206	3.1621	0.6834	0.6909	1.0678	1.035	3.923	0.0881	0.1772	5.6916	5.7319	6.2981	6.2888	6.3103	3.2188	3.2613	3.6163	2.4362	2.4012	2.9062	3.0363	2.9962	3.0387	2.8919	2.8031	2.7619	0	0	0
16	2024-05-27 0.15	17380	37211	24724	6291	84996	0	0	0	1	0	1	1	1	0	0	1	0	0	1	3.206	3.1566	0.6891	0.7022	1.0641	1.0181	3.92	0.1031	0.1481	5.6944	5.7141	6.315	6.3066	6.3094	3.2188	3.2613	3.6163	2.4362	2.4012	2.9062	3.0363	2.9962	3.0387	2.8919	2.8031	2.7619	0	0	0
17	2024-05-27 0.16	17400	37379	25147	6108	85554	0	0	0	1	0	1	1	1	0	0	1	0	0	1	3.206	3.1566	0.705	0.7022	1.0669	1.0134	3.9237	0.0919	0.1622	5.685	5.7075	6.3169	6.3019	6.3056	3.2188	3.2613	3.6163	2.4362	2.4012	2.9062	3.0363	2.9962	3.0387	2.8794	2.8194	2.7662	0	0	0
18	2024-05-27 0.17	17400	37379	25147	6108	85554	0	0	0	1	0	1	1	1	0	0	1	0	0	1	3.206	3.1566	0.705	0.7022	1.0669	1.0134	3.9237	0.0919	0.1622	5.685	5.7075	6.3169	6.3019	6.3056	3.2188	3.2613	3.6163	2.4362	2.4012	2.9062	3.0363	2.9962	3.0387	2.8794	2.8194	2.7662	0	0	0
19	2024-05-27 0.18	17530	36896	24558	6301	85844	0	0	0	1	0	1	1	1	0	0	1	0	0	1	3.206	3.1456	0.6816	0.7022	1.0425	1.0228	3.925	0.1078	0.1481	5.7019	5.7337	6.2953	6.285	6.3141	3.2188	3.2613	3.6163	2.4362	2.4012	2.9062	3.0363	2.9962	3.0387	2.8944	2.8131	2.7744	0	0	0
20	2024-05-27 0.19	17270	37466	24336	6173	85017	0	0	0	1	0	1	1	1	0	0	1	0	0	1	3.206	3.1566	0.6731	0.7013	1.0341	1.0041	3.9237	0.0881	0.1678	5.6934	5.7141	6.3187	6.3047	6.3094	3.2188	3.2613	3.6163	2.4362	2.4012	2.9062	3.0363	2.9962	3.0387	2.8881	2.8038	2.7613	0	0	0
21	2024-05-27 0.20	17270	37466	24336	6173	85017	0	0	0	1	0	1	1	1	0	0	1	0	0	1	3.206	3.1566	0.6731	0.7013	1.0341	1.0041	3.9237	0.0881	0.1678	5.6934	5.7141	6.3187	6.3047	6.3094	3.2188	3.2613	3.6163	2.4362	2.4012	2.9062	3.0363	2.9962	3.0387	2.8881	2.8038	2.7613	0	0	0
22	2024-05-27 0.21	16950	37119	24942	6215	84980	0	0	0	1	0	1	1	1	0	0	1	0	0	1	3.206	3.1621	0.6797	0.675	1.0331	0.9947	3.92	0.0956	0.1744	5.7038	5.7253	6.2953	6.2906	6.3225	3.2188	3.2613	3.6163	2.4362	2.4012	2.9062	3.0363	2.9962	3.0387	2.895	2.7975	2.7719	0	0	0
23	2024-05-27 0.22	16950	37119	24942	6215	84980	0	0	0	1	0	1	1	1	0	0	1	0	0	1	3.206	3.1621	0.6797	0.675	1.0331	0.9947	3.92	0.0956	0.1744	5.7038	5.7253	6.2953	6.2906	6.3225	3.2188	3.2613	3.6163	2.4362	2.4012	2.9062	3.0363	2.9962	3.0387	2.895	2.7975	2.7719	0	0	0
24	2024-05-27 0.23	17090	37292	25305	6336	85284	0	0	0																																								

2-1. 문제 정의하기

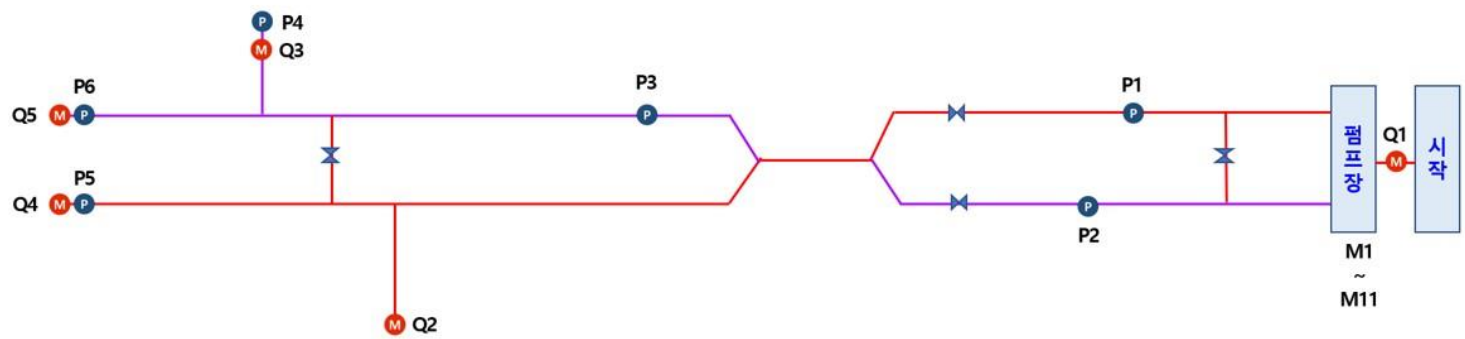
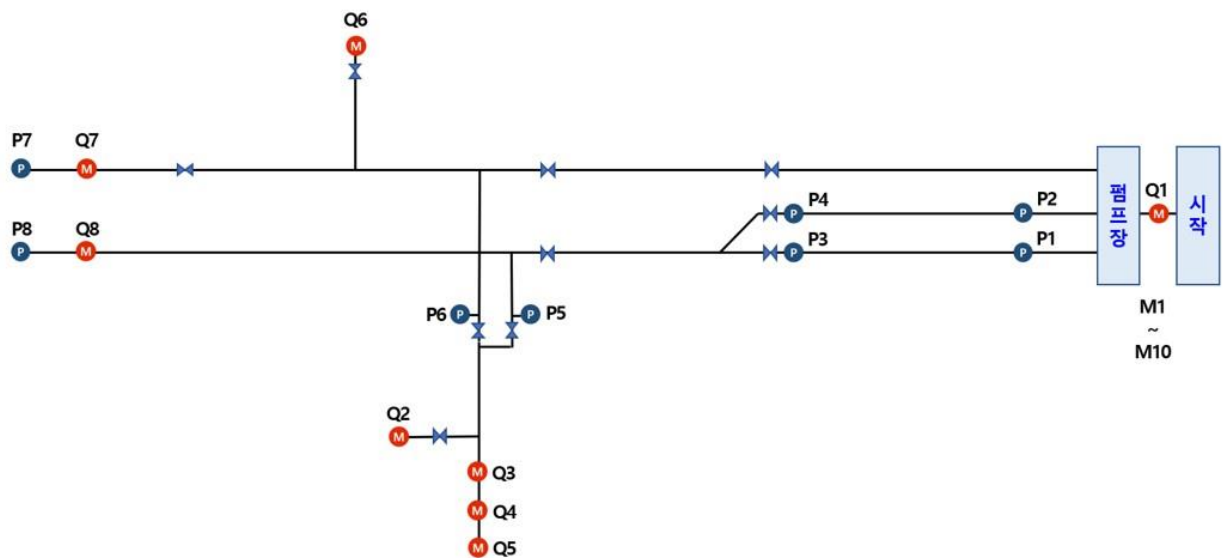
2-1. 문제 정의하기

2-II. 문제 쪼개기

2-III. 쪼갬 문제 해결하는 방법론 만들기

2-IV. 실행하기

관망 구조 C, D



TEST_C_0000	2024-11-20 오전 9:12	Microsoft Excel 실...	1,255KB
TEST_C_0001	2024-11-20 오전 9:12	Microsoft Excel 실...	1,255KB
TEST_C_0002	2024-11-20 오전 9:12	Microsoft Excel 실...	1,255KB
TEST_C_0003	2024-11-20 오전 9:14	Microsoft Excel 실...	1,255KB
TEST_C_0004	2024-11-20 오전 9:11	Microsoft Excel 실...	1,255KB
TEST_C_0005	2024-11-20 오전 9:12	Microsoft Excel 실...	1,255KB
TEST_C_0006	2024-11-20 오전 9:14	Microsoft Excel 실...	1,255KB
TEST_C_0007	2024-11-20 오전 9:12	Microsoft Excel 실...	1,255KB
TEST_C_0008	2024-11-20 오전 9:14	Microsoft Excel 실...	1,255KB
TEST_C_0009	2024-11-20 오전 9:14	Microsoft Excel 실...	1,255KB
TEST_C_0010	2024-11-20 오전 9:15	Microsoft Excel 실...	1,255KB
TEST_C_0011	2024-11-20 오전 9:15	Microsoft Excel 실...	1,255KB
TEST_C_0012	2024-11-20 오전 9:15	Microsoft Excel 실...	1,255KB
TEST_C_0013	2024-11-20 오전 9:13	Microsoft Excel 실...	1,255KB
TEST_C_0014	2024-11-20 오전 9:15	Microsoft Excel 실...	1,255KB
TEST_C_0015	2024-11-20 오전 9:11	Microsoft Excel 실...	1,255KB
TEST_C_0016	2024-11-20 오전 9:12	Microsoft Excel 실...	1,255KB

<관망 구조 C 데이터>

2919개의 관망 구조 C 측정 데이터

- TimeStamp : 1개
- Q1 ~ Q8(유량계) : 8개
- M1 ~ M10(개폐여부) : 10개
- P1 ~ P8(압력계) : 8개
- Anomly(이상여부) : 1개
- P1 flag ~ P8 flag : 8개

<관망 구조 D 데이터>

2737개의 관망 구조 D에서의 측정 데이터

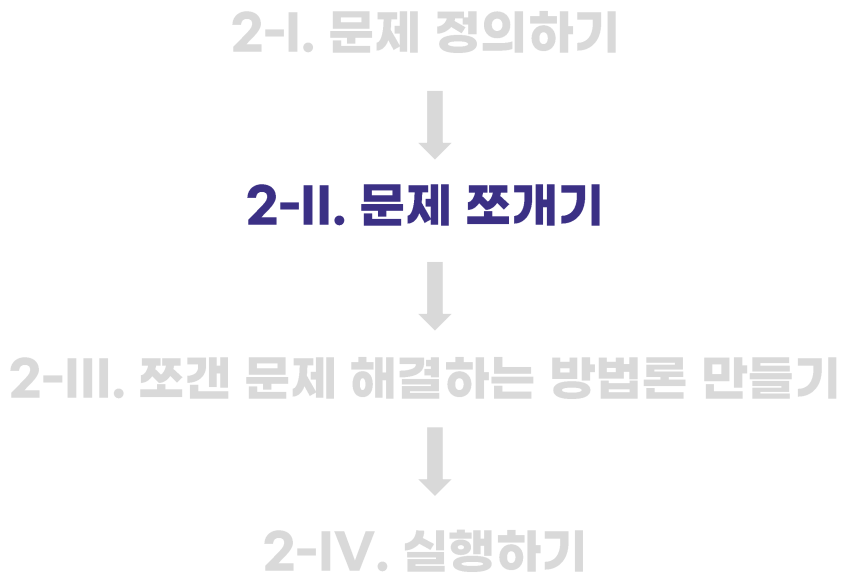
<문제 정의>

전혀 다른 관망구조를 가진
(= 데이터 구조가 다른) A, B 데이터를 통해

C, D 관망구조의 2919 + 2737개의
상황에서 각 상황 후반부에
이상현상이 일어났는지 판단하고,

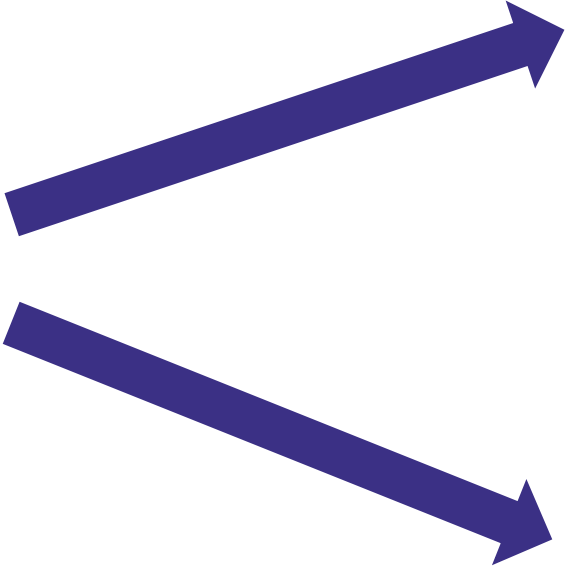
그들 중에서 어느 압력계가 문제인지 찾기

2-II. 문제 쪼개기



<문제 정의>

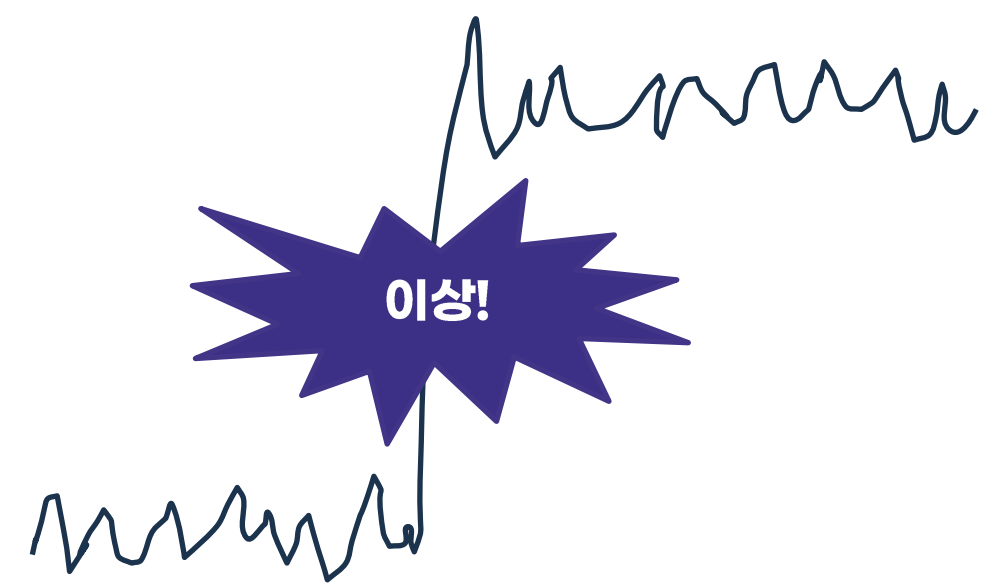
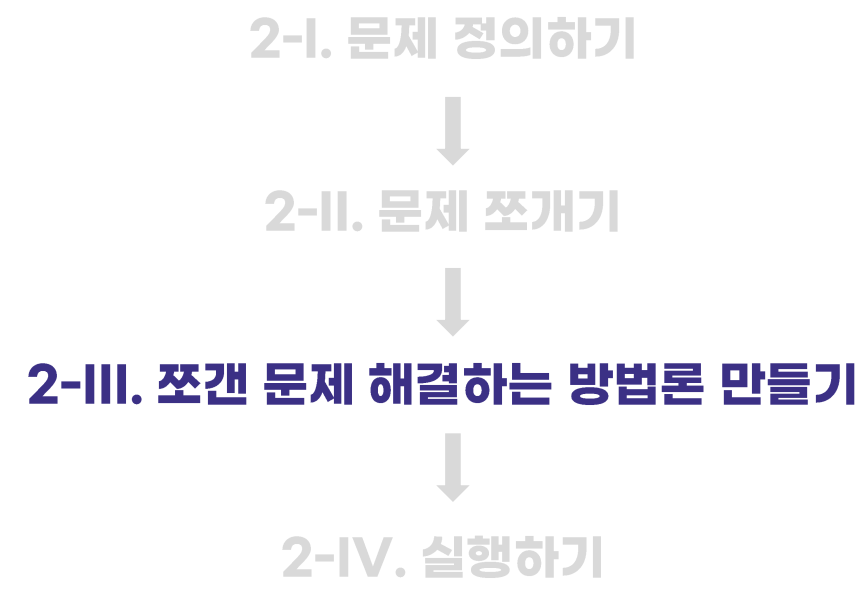
전혀 다른 관망구조를 가진
(= 데이터 구조가 다른) A, B 데이터를 통해
C, D 관망구조의 2919 + 2737개의 상황에서
각 상황 후반부에
이상현상이 일어났는지 판단하고,
그들 중에서 어느 압력계가 문제인지 찾기



- 1) 5656개의 상황 중에서
이상치라고 판단되는 상황 찾기
- 2) 이상치라고 판단된 부분들 중에서
어느 위치에서 문제가 일어났는지 찾기

2-III. 쪼갬 문제 해결하는 방법론 만들기

“1번 문제. 5656개의 상황 중에서 이상치라고 판단되는 상황 찾기”



급격하게 상승하는 구간이 있다!



점차 상승하는 구간이 있다!

2-III. 조건 문제 해결하는 방법론 만들기

“1번 문제. 5656개의 상황 중에서 이상치라고 판단되는 상황 찾기”

관망구조 A 데이터에는 이상치 데이터와 정상 상태 데이터가 모두 있다!

AP	AQ	AR	AS	AT	AU	AV	AW	AX	AY	AZ	BA	BB	BC	BD	F
P22	P23	P24	P25	P26	anomaly	P1_flag	P2_flag	P3_flag	P4_flag	P5_flag	P6_flag	P7_flag	P8_flag	P9_flag	F
2.7738	2.8075	2.5213	2.4331	2.395	1	0	0	0	0	0	0	0	1	1	
2.7587	2.7525	2.5213	2.4331	2.395	1	0	0	0	0	0	0	0	1	1	
2.7525	2.7462	2.5219	2.43	2.3937	1	0	0	0	0	0	0	0	1	1	
2.7375	2.755	2.5219	2.43	2.3937	1	0	0	0	0	0	0	0	1	1	
2.7513	2.735	2.5219	2.4194	2.3806	1	0	0	0	0	0	0	0	1	1	
2.745	2.7562	2.5219	2.4194	2.3806	1	0	0	0	0	0	0	0	1	1	
2.725	2.7188	2.5075	2.4206	2.3819	1	0	0	0	0	0	0	0	1	1	
2.7387	2.74	2.5	2.43	2.3731	1	0	0	0	0	0	0	0	1	1	
2.7438	2.7438	2.5106	2.4306	2.3863	1	0	0	0	0	0	0	0	1	1	
2.7375	2.7387	2.5	2.4088	2.3763	1	0	0	0	0	0	0	0	1	1	
2.7487	2.7438	2.5	2.4088	2.3763	1	0	0	0	0	0	0	0	1	1	
2.72	2.7375	2.4869	2.42	2.3606	1	0	0	0	0	0	0	0	1	1	
2.75	2.7625	2.4869	2.42	2.3606	1	0	0	0	0	0	0	0	1	1	
2.72	2.735	2.4881	2.4156	2.3613	1	0	0	0	0	0	0	0	1	1	
2.7413	2.7338	2.4944	2.41	2.3844	1	0	0	0	0	0	0	0	1	1	
2.7375	2.725	2.4944	2.41	2.3844	1	0	0	0	0	0	0	0	1	1	
2.735	2.7475	2.4869	2.41	2.3688	1	0	0	0	0	0	0	0	1	1	
2.7462	2.75	2.4869	2.41	2.3688	1	0	0	0	0	0	0	0	1	1	
2.7362	2.7313	2.4962	2.41	2.3588	1	0	0	0	0	0	0	0	1	1	
2.7375	2.7562	2.4962	2.41	2.3588	1	0	0	0	0	0	0	0	1	1	
2.76	2.7625	2.49	2.415	2.3575	1	0	0	0	0	0	0	0	1	1	
2.7137	2.7262	2.4919	2.4169	2.3569	1	0	0	0	0	0	0	0	1	1	
2.7275	2.7425	2.4944	2.4181	2.36	1	0	0	0	0	0	0	0	1	1	
2.7275	2.7338	2.49	2.4113	2.3588	1	0	0	0	0	0	0	0	1	1	
2.7125	2.7287	2.49	2.4113	2.3588	1	0	0	0	0	0	0	0	1	1	
2.7113	2.7225	2.4887	2.4188	2.3581	1	0	0	0	0	0	0	0	1	1	
2.71	2.7213	2.4887	2.4188	2.3581	1	0	0	0	0	0	0	0	1	1	
2.7287	2.7362	2.4725	2.4037	2.3606	1	0	0	0	0	0	0	0	1	1	
2.7025	2.7375	2.4919	2.3963	2.3606	1	0	0	0	0	0	0	0	1	1	
2.7338	2.7662	2.4919	2.3963	2.3606	1	0	0	0	0	0	0	0	1	1	
2.98	2.9875	2.8981	2.8175	2.7762	0	0	0	0	0	0	0	0	0	0	
2.98	2.9875	2.8994	2.8163	2.775	0	0	0	0	0	0	0	0	0	0	
2.98	2.9875	2.8912	2.8144	2.775	0	0	0	0	0	0	0	0	0	0	
2.98	2.9875	2.8956	2.8125	2.7775	0	0	0	0	0	0	0	0	0	0	
2.98	2.9875	2.8956	2.8125	2.7775	0	0	0	0	0	0	0	0	0	0	

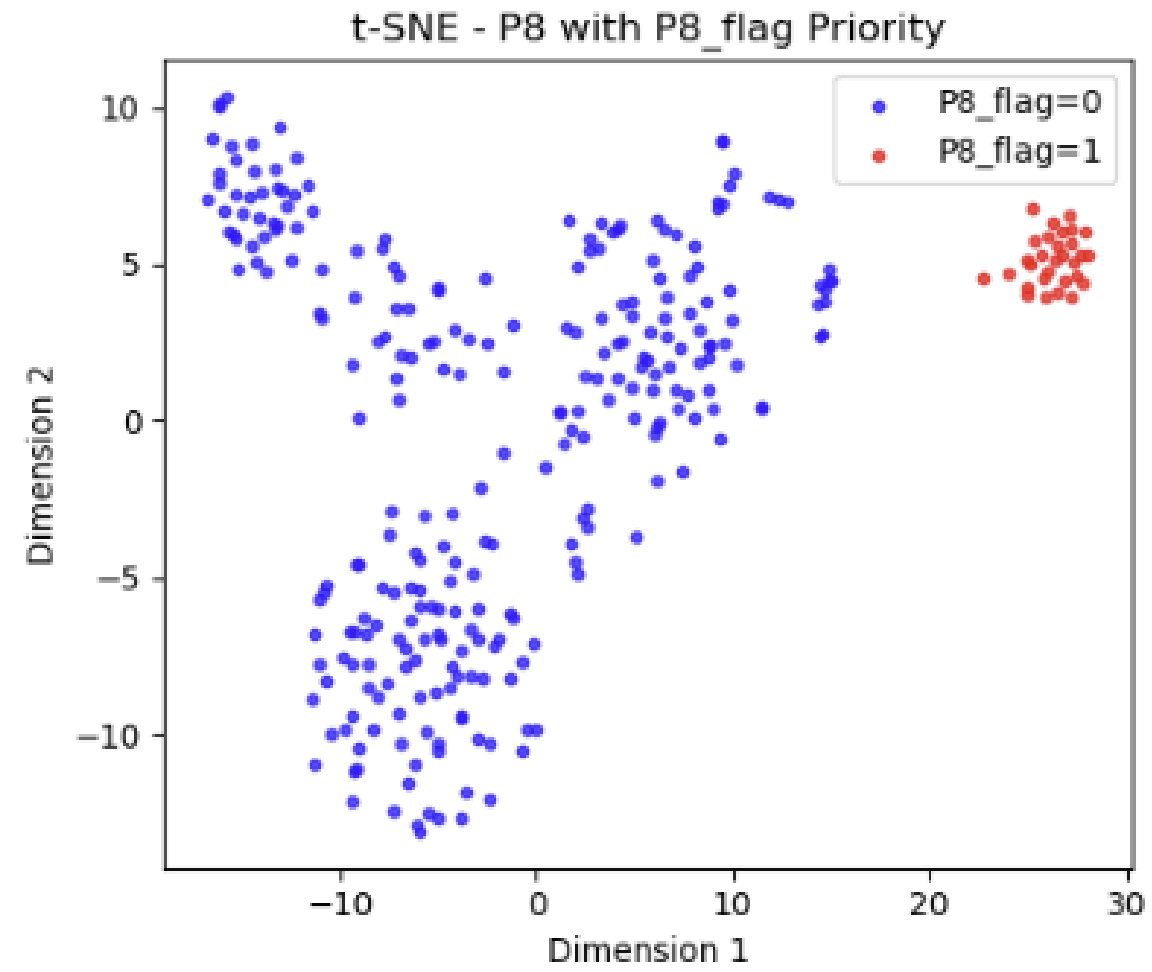
비정상 데이터 : 30개

정상 데이터 : 4407개

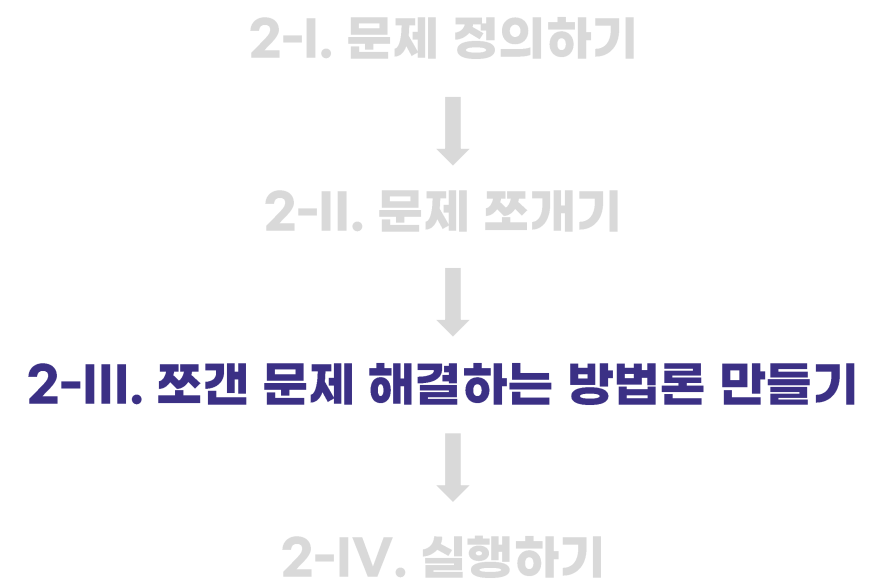
비정상 데이터 : 30개
비정상 바로 직전 정상 데이터 : 300개

두 상황을 2차원 상에 표현하면 구분되지 않을까?

- P, Q 데이터를 활용하여 2차원 상에 표현하기

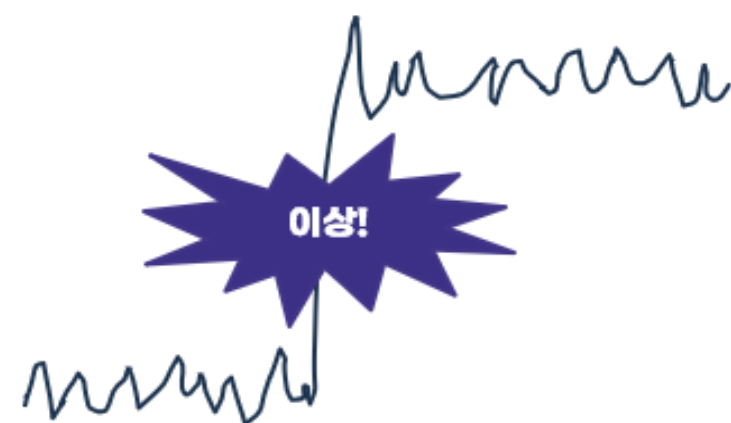


- 정상/비정상 P, Q 데이터를 활용하여 2차원 상에 표현했는데, 구분이..된다..!
- 그룹이 명확하게 구분되면, 구분된 소수 그룹은 곧 비정상이다.



2-III. 쪼갬 문제 해결하는 방법론 만들기

“2번 문제. 이상치라고 판단된 부분들 중에서 어느 위치에서 문제가 일어났는지 찾기”



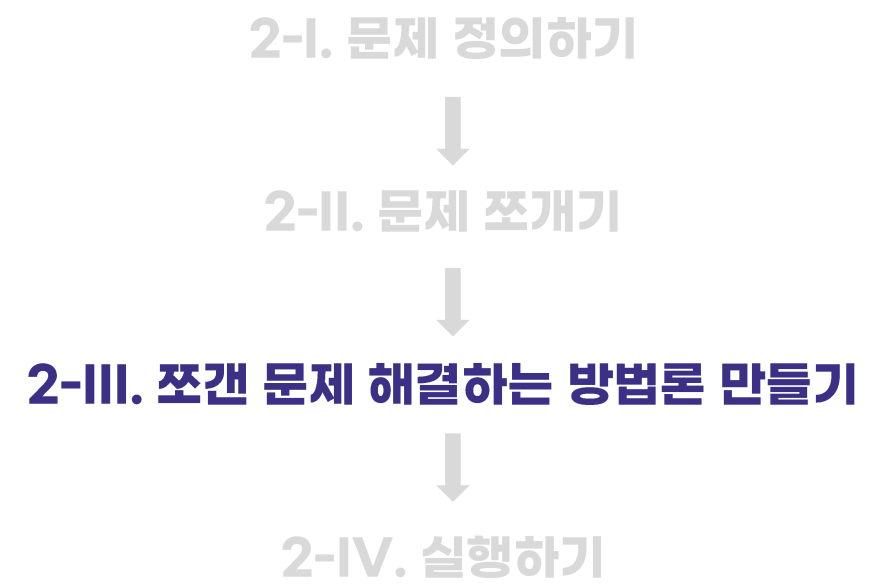
급격하게 상승하는 구간이 있다!



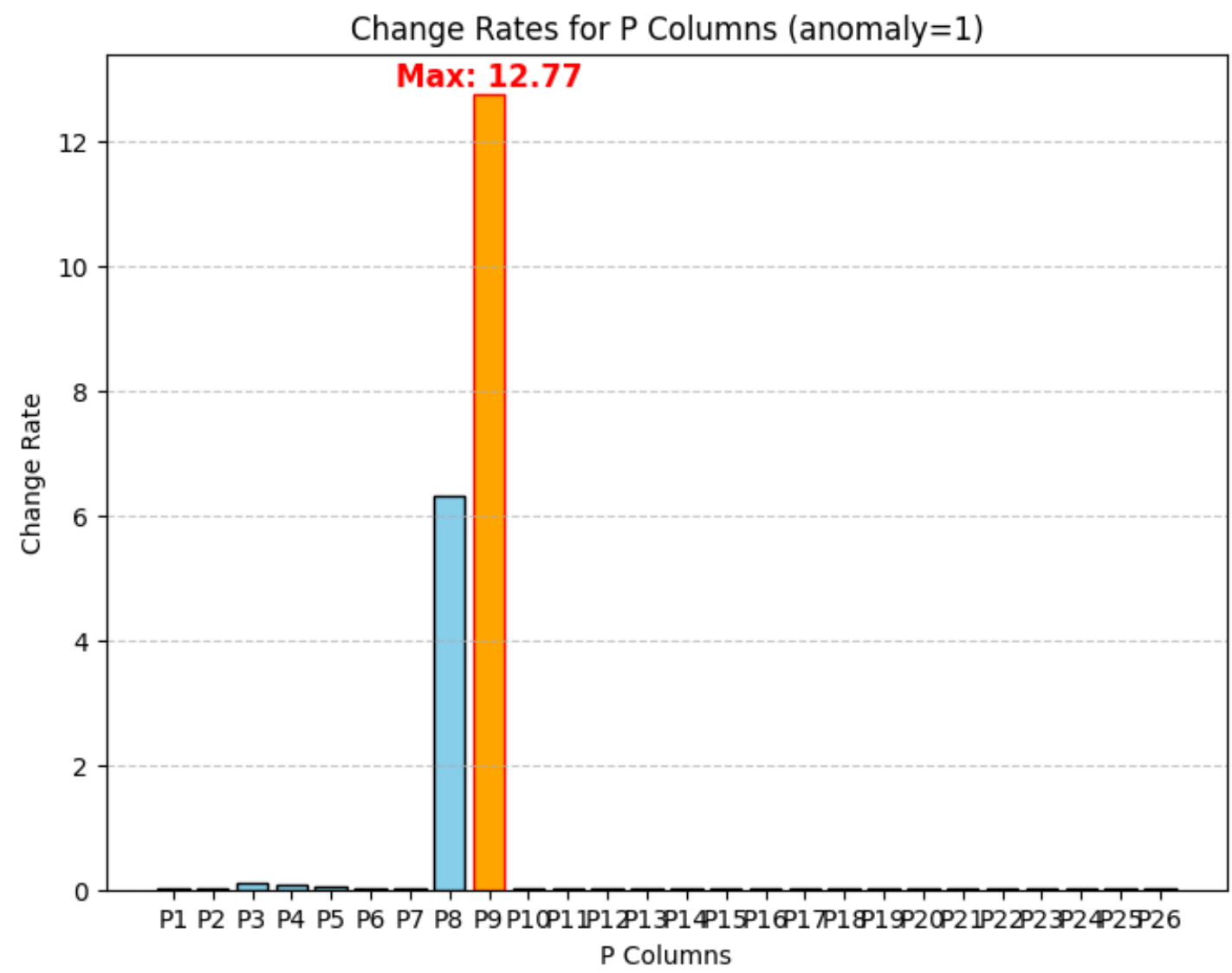
점차 상승하는 구간이 있다!

AY	AZ	BA	BB	BC	BD	BE	BF	BG	P13
flag	P5_flag	P6_flag	P7_flag	P8_flag	P9_flag	P10_flag	P11_flag	P12_flag	P13
0	0	0	0	1	1	0	0	0	0
0	0	0	0	1	1	0	0	0	0
0	0	0	0	1	1	0	0	0	0
0	0	0	0	1	1	0	0	0	0
0	0	0	0	1	1	0	0	0	0
0	0	0	0	1	1	0	0	0	0
0	0	0	0	1	1	0	0	0	0
0	0	0	0	1	1	0	0	0	0
0	0	0	0	1	1	0	0	0	0
0	0	0	0	1	1	0	0	0	0
0	0	0	0	1	1	0	0	0	0
0	0	0	0	1	1	0	0	0	0
0	0	0	0	1	1	0	0	0	0
0	0	0	0	1	1	0	0	0	0
0	0	0	0	1	1	0	0	0	0
0	0	0	0	1	1	0	0	0	0
0	0	0	0	1	1	0	0	0	0
0	0	0	0	1	1	0	0	0	0

실제 데이터 결과와 유사한 결과 확보



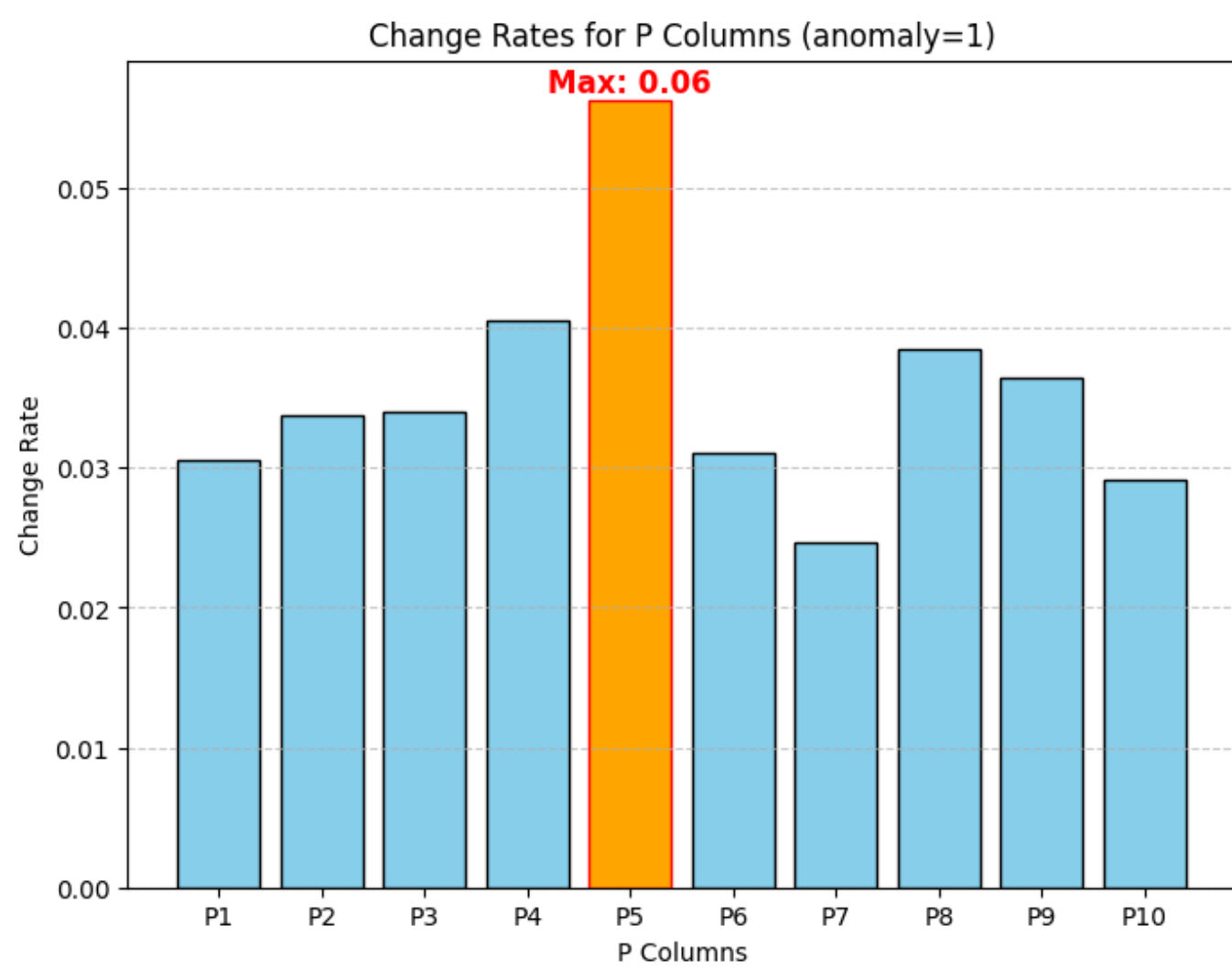
관망구조에서 P 데이터의 변화를 잘 살펴보면, 큰 변화가 있는 부분에서 문제가 일어나지 않을까?



2-III. 조건 문제 해결하는 방법론 만들기

“2번 문제. 이상치라고 판단된 부분들 중에서 어느 위치에서 문제가 일어났는지 찾기”

관망구조 B에서도 성립해야 한다!

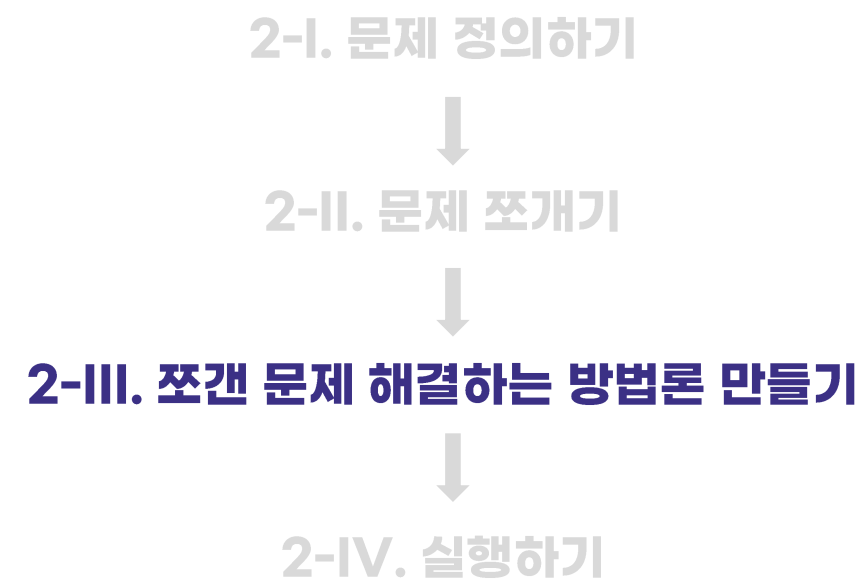


≠

U	V	W	X	Y	Z
P5_flag	P6_flag	P7_flag	P8_flag	P9_flag	P10_flag
0	0	1	1	1	0
0	0	1	1	1	0
0	0	1	1	1	0
0	0	1	1	1	0
0	0	1	1	1	0
0	0	1	1	1	0
0	0	1	1	1	0
0	0	1	1	1	0
0	0	1	1	1	0
0	0	1	1	1	0
0	0	1	1	1	0
0	0	1	1	1	0
0	0	1	1	1	0
0	0	1	1	1	0
0	0	1	1	1	0
0	0	1	1	1	0

관망구조 B에서는 성립하지 않는다 ππ

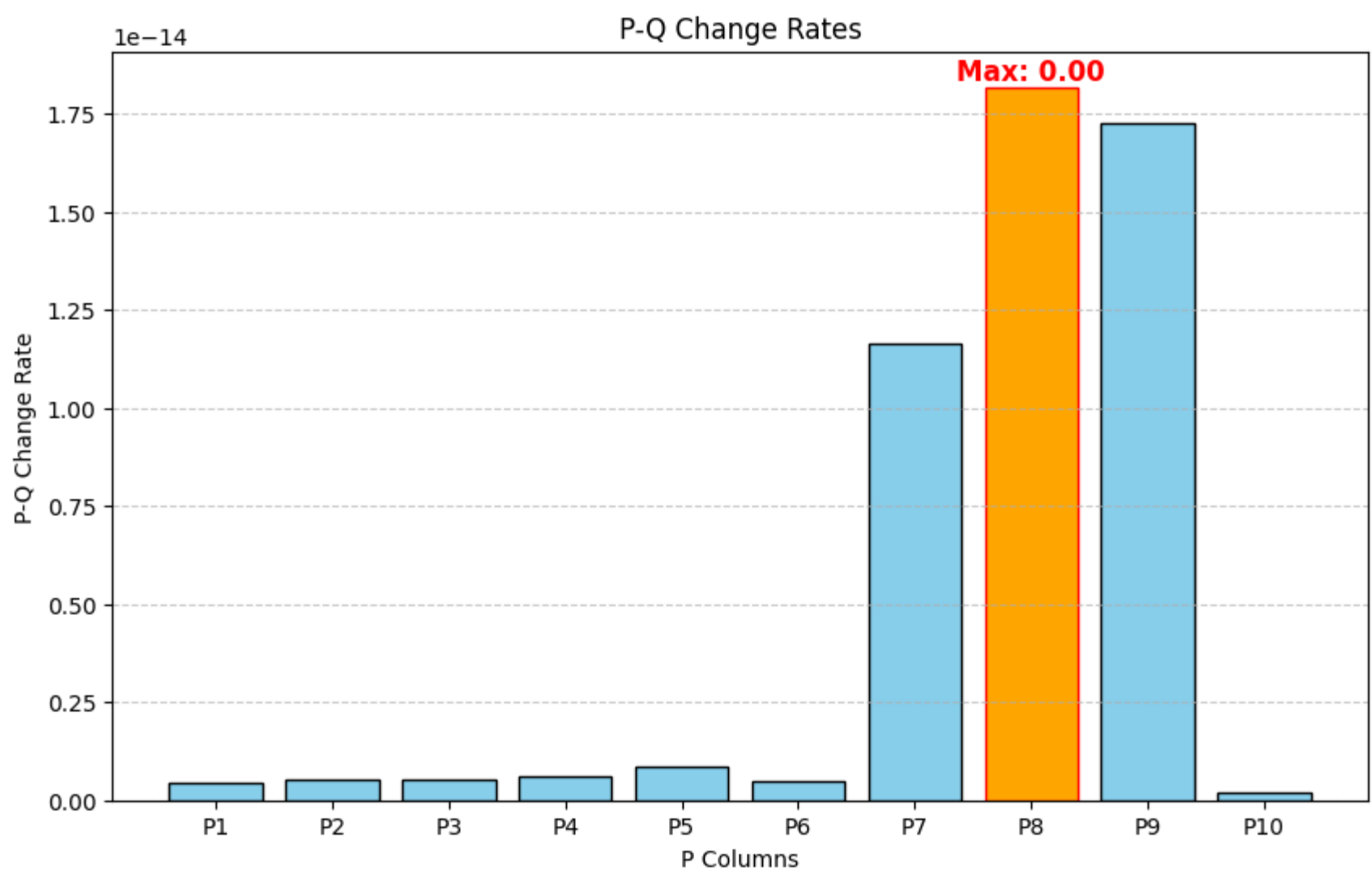
Q 데이터를 추가해서 생각해보자



2-III. 쪼갬 문제 해결하는 방법론 만들기

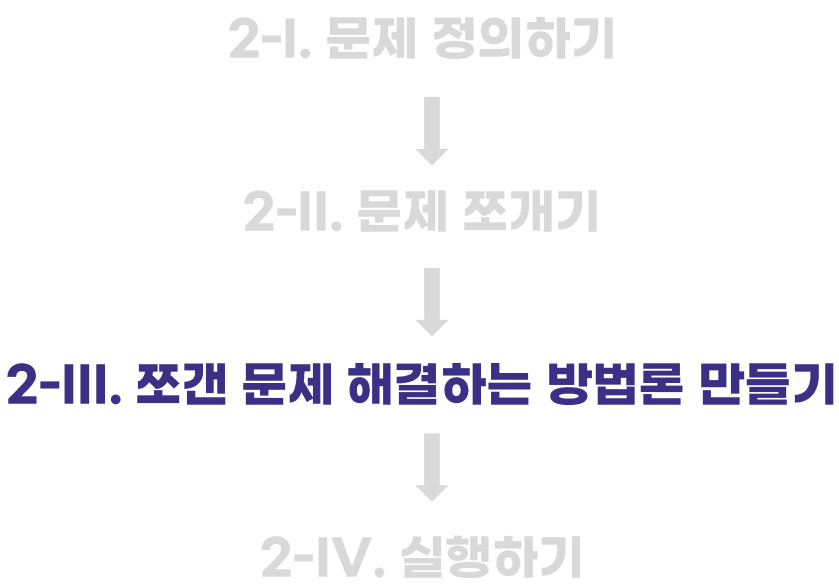
“2번 문제. 이상치라고 판단된 부분들 중에서 어느 위치에서 문제가 일어났는지 찾기”

Q 데이터를 추가해서 생각해보자



(가중치 1) * P 데이터 영향 + (가중치 2) * Q 데이터 영향

어찌저찌 P와 Q를 적절히 섞어서 P7,8,9를 분리할 수 있는 식을 찾긴 했는데...



2-IV. 실행하기 (GPT 쓰기)

“GPT와 함께하는 코딩은 두렵지 않다!”

'sample_submission.csv'의 'flag_list' 열에는 길이가 서로 다른 리스트가 [0,0,0,0], [0,0,0,0,0]과 같은 형태로 1천여개 저장되어 있다. 이 'sample_submission.csv'를 pd.read_csv로 읽어오면서, 'flag_list' 열의 데이터들을 string이 아닌 list 형태로 반환받고자 한다. 함수 선언하지 않고 반복문만을 이용하여 list 형태로 변환된 'flag_list' 열의 값들을 'scores' 변수에 저장하게끔 하는 코드를 작성하라.

11초 동안 생각

```
python

import pandas as pd

# CSV 파일을 읽어옵니다.
df = pd.read_csv('sample_submission.csv')

# 'flag_list' 열의 데이터를 리스트로 변환하여 저장할 변수 초기화
scores = []

# 반복문을 사용하여 각 행의 'flag_list' 데이터를 처리합니다.
```

데이터프레임에는 데이터가 T- 시간에 대해 오름차순으로 정렬되어있고, Q1~Qn 칼럼과 P1~Pn 칼럼이 존재해. 우리의 목적은 각각의 Q에 관련 있는 P값들을 분류하는거야. 예를 들어 Q1에 P1과 P2가 연관되어있는지 확인하는 방식이야. 어떤 방법을 사용할 수 있을까?

Q와 P 간의 연관성을 확인하는 것은 데이터 분석 및 통계적 방법을 사용하여 상관성 또는 종속성을 평가하는 문제로 볼 수 있습니다. 몇 가지 접근 방법을 아래에 제안합니다:

이 문제는 시계열 데이터를 기반으로 미래의 유량 및 압력을 예측하는 작업입니다. 각 펌프의 개폐 여부는 예측하지 않아도 되며 이전 값을 그대로 사용하므로, 유량과 압력을 중심으로 모델을 구축하면 됩니다. 아래는 이 문제를 해결하기 위한 Python 코드를 작성한 예제입니다.

2-I. 문제 정의하기

2-II. 문제 쪼개기

2-III. 쪼갠 문제 해결하는 방법론 만들기

2-IV. 실행하기

수도관 데이터의 유량과 압력을 예측하는 모델을 제작해야 해.

이때 주어지는 데이터는 시간에 따른 특정 위치들에서의 유량과 특정 위치들에서의 압력, 그리고 각 펌프의 개폐 여부야.

유량은 Q1,Q2,... 칼럼들로 나타내어지고, 시간은 timestamp 칼럼으로 나타내져. 압력은 P1,P2,P3... 칼럼들로 나타내어지고, 각 펌프의 개폐 여부는 M1,M2,M3... 칼럼들로 나타내져.

산출해야 할 결과값은 마지막 Timestamp 값에 +1을 한 시간에서 각 위치들의 유량들과, 압력들이고, 각 펌프의 개폐 여부 값은 이전의 것을 따라 작성돼.

이러한 문제를 해결할 수 있는 코드를 작성해줄 수 있어?

2-V. 결과

제출물 0점 처리 관련 질문드립니다.

 **temmietaxi**  **연현** 공동작성자

2024.12.09 00:34 427 조회

안녕하세요,

제출파일의 0점처리와 관련하여 궁금한 사항이 있어 질문드립니다.

이전 제출물의 개선사항을 확인하고 코드 수정 후 얻은 csv 결과물을 다시 제출하였는데,

점수가 완전히 0점으로 들어가는 것이 어떤 이유에서인지 궁금하여 질문드립니다.

타 대회 사례를 찾아보니 0점으로 검사가 되는 것은 대부분 한글 라벨이 붙어 있는 경우

ANSI 인코딩 오류 혹은 한글자모 분리 문제로 인해 데이터 인식이 제대로 되지 않을 때로 보이는데,

본 대회 제출물 양식에는 한글이 전혀 포함되어 있지도 않고, 내용 상 문제도 없어 보입니다만

어떻게 0점으로 처리된 것인지를 알 수 없어 질문을 드립니다.

운영진님께, 어떠한 부분이 문제가 되는 것인지 정답과 관련한 것이라면

세세히 답변하시기 어려운 점 이해합니다.

그러나 기존 제출물은 0점이 아니었는데

개선한 버전의 점수가 완전히 0점 처리 되는 것은 무슨 이유에서인지

팀 내부에서 도저히 실마리를 찾지 못하여 질문을 남깁니다.

혹시 저와 같은 문제를 겪고 계신 혹은 해결하신 분들께,

이 글에 반응 남겨 주시면 감사하겠습니다, 또한 해결 방법을 찾으셨다면

어떤 부분에서 문제를 찾으셨는지도 언질 주시면 무척 감사하겠습니다.

배움이 짧은 관계로 많은 도움을 부탁드립니다.

감사합니다.



DH.BU 2024.12.09 01:22

```
import ast
import pandas as pd
import numpy as np
```

```
submission = pd.read_csv("제출물 파일 이름.csv")
print((submission[submission['ID'].str.contains('C')]['flag_list'].apply(lambda x: len(np.array(ast.literal_eval(x))))==8).sum() == len(submission[submission['ID'].str.contains('C')]))
print((submission[submission['ID'].str.contains('D')]['flag_list'].apply(lambda x: len(np.array(ast.literal_eval(x))))==6).sum() == len(submission[submission['ID'].str.contains('D')]))
```

둘다 True가 출력되는 지 확인해보시면 좋을 것 같습니다

답글 달기



temmietaxi 2024.12.09 01:30

DH.BU님께, 먼저 답변 주셔서 정말 감사합니다. 확인 결과, 둘 다 True로 나옵니다. 혹시 이 외에 짐작 가시는 원인이 또 있으시다면 말씀 주시면 시험해 보겠습니다.



DH.BU 2024.12.09 01:51

```
print(len((submission[submission['ID'].str.contains('C')]['ID']).unique()) == 2920)
print(len((submission[submission['ID'].str.contains('D')]['ID']).unique()) == 2738)
```

이것도 확인해보시겠어요?



temmietaxi 2024.12.09 02:20

DH.BU님께, 확인해 본 결과 정상 출력됩니다.(둘 다 True) 직접 .csv 내용을 열람도 해보았고 위 두 코드 내용을 고려해 보았을 때 flag_list의 인식도 정상적으로 되는 것 같은데 참 의아하네요. ㅜㅜ 아무리 생각해도 정확하게 0점이 뜰 정도로 벗어나는 건 불가능할 듯한데 희한한 일입니다.



어비스 2024.12.09 16:17

혹시 대회 끝나고 간단하게 리뷰나 공유 가능하실까요??
감을 못잡겠네용

1151명(486팀) 중에서 “132”등 기록

2-VI. 번외편

IMEN358 - 품질공학 00분반 정은택 학생 12월 9일 논문리뷰 영상 문이드립니다. -산업경영공학부 2023170833 정은택

받은편지함



나 12월 9일

교수님, 안녕하십니까, 저는 교수님의 <품질공학>(IMEN358(00))...



Seoung Bum Kim 12월 9일

받는사람: 나

은택,

영상 첨부합니다. 영상 앞부분은 Transformer에 대한 설명이고, 34분 이후 Anomaly Transformrer와 TranAD 관련 내용입니다.

<https://youtu.be/MJYBdTCwxDY?si=Vf3zqso9h-rF5Cyx>

도움이 되길 바랍니다.

고맙습니다.

김성범 드림

-----Original Message-----

Subject : IMEN358 - 품질공학 00분반 정은택 학생 12월 9일 논문리뷰 영상 문이드립니다. -산업경영공학부 2023170833 정은택

Date : 2024-12-09 17:02:54

From : 정은택 <05temtxi21@gmail.com>

To : sbkim1@korea.ac.kr

Cc :

교수님께 조언을 구함

- 이상치 탐지 모델을 활용한 2번 문제 방법론 구상

TranAD: Deep Transformer Networks for Anomaly Detection in Multivariate Time Series Data

Shreshth Tuli
Imperial College London
London, UK
s.tuli20@imperial.ac.uk

Giuliano Casale
Imperial College London
London, UK
g.casale@imperial.ac.uk

Nicholas R. Jennings
Loughborough University
London, UK
n.r.jennings@lboro.ac.uk

Abstract

Efficient anomaly detection and diagnosis in multivariate time-series data is of great importance for modern industrial applications. However, building a system that is able to quickly and accurately pinpoint anomalous observations is a challenging problem. This is due to the lack of anomaly labels, high data volatility and the demands of ultra-low inference times in modern applications. Despite the recent developments of deep learning approaches for anomaly detection, only a few of them can address all of these challenges. In this paper, we propose TranAD, a deep transformer network

increasing data volatility creates the requirement for significant amounts of data for accurate inference. However, due to the rising federated learning paradigm with geographically distant clusters synchronizing databases across devices is expensive, causing limited data availability for training [48, 57]. Further, next-generation applications need ultra-fast inference speeds for quick recovery and optimal Quality of Service (QoS) [6, 49, 50]. Time-series databases are generated using several engineering artifacts (servers, robots *etc.*) that interact with the environment, humans or other systems. As a result, the data often displays both stochastic and temporal trends [45]. It thus becomes crucial to distinguish outliers due to

Transformer-based multivariate time series anomaly detection using inter-variable attention mechanism

Hyeongwon Kang, Pilsung Kang *

Department of Industrial & Management Engineering, Korea University, 126-16 Anam-dong 5-ga, Seongbuk-gu, Seoul, Republic of Korea

ARTICLE INFO

Keywords:
Multivariate time-series
Anomaly detection
Transformer
XAI
Attention mechanism

ABSTRACT

The primary objective of multivariate time-series anomaly detection is to spot deviations from regular patterns in time-series data compiled concurrently from various sensors and systems. This method finds application across diverse industries, aiding in system maintenance tasks. Capturing temporal dependencies and correlations between variables simultaneously is challenging due to the interconnectedness and mutual influence among variables in multivariate time-series. In this paper, we propose a unique method, the Variable Temporal Transformer (VTT), which utilizes the self-attention mechanism of transformers to effectively understand the temporal dependencies and relationships among variables. This proposed model performs anomaly detection by employing temporal self-attention to model temporal dependencies and variable self-attention to model variable correlations. We use a recently introduced evaluation metric after identifying potential overestimations in the performance of traditional time series anomaly detection methods using the point adjustment protocol evaluation metric. We confirm that our proposed method demonstrates cutting-edge performance through this new metric. Furthermore, we bring forth an anomaly interpretation module to shed light on anomalous data, which we verify using both synthetic and real-world industrial data.

2-I. 문제 정의하기

2-II. 문제 쪼개기

2-III. 쪼갠 문제 해결하는 방법론 만들기

2-IV. 실행하기

Variational transformer-based anomaly detection approach for multivariate time series

Xixuan Wang ^a, Dechang Pi ^{a,*}, Xiangyan Zhang ^b, Hao Liu ^a, Chang Guo ^a

^a College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing, China
^b Beijing Institute of Spacecraft System Engineering, Beijing, China

ARTICLE INFO

Keywords:
Telemetry data
Transformer
Variational autoencoder
Multivariate time series
Anomaly detection

ABSTRACT

Due to the strategic importance of satellites, the safety and reliability of satellites have become more important. Sensors that monitor satellites generate lots of multivariate time series, and the abnormal patterns in the multivariate time series may imply malfunctions. The existing anomaly detection methods for multivariate time series have poor effects when processing the data with few dimensions or sparse relationships between sequences. This paper proposes an unsupervised anomaly detection model based on the variational Transformer to solve the above problems. The model uses the Transformer's self-attention mechanism to capture the potential correlations between sequences and capture the multi-scale temporal information through the improved positional encoding and up-sampling algorithm. Then, the model comprehensively considers the extracted features through the residual variational autoencoder to perform effective anomaly detection. Experimental results on a real dataset and two public datasets show that the proposed method is superior to the mainstream and state-of-the-art methods.

오후 11:05

GO&STOP 의사결정이 필요한 시기일듯한다..ㅎㅎ

님덜

기말고사 시험 기간 이슈로 잠정 중단

3. 실패한 프로젝트 의미있게 만들기

	[Public 2위, Pritvate 3위] SKKU brAln Solution
	[PRIVATE 4위 AIME한테 DM해~] 코드 및 PPT 공유
	팀 PQM 코드 및 PPT 공유

```
level3_sigma = 2
level2_sigma = 1
Q_ratio_diff_window_size = 9
Q_drop_threshold = -0.03
Q_jump_threshold = 0.03
anomaly_lookback_window_size = 2
rolling_window_size = 5
top_k = 1


# C structure
input_Q_C = ['Q1']
output_Q_C = ['Q2','Q3','Q4','Q5','Q6','Q7','Q8']
target_P_list_C = [f"P{i}" for i in range(1,9)]
group_P_list_C = [['P1'], ['P2'], ['P2','P4'], ['P1','P3'], ['P5','P8'], ['P6'], ['P7']]
data_num_C = 2920


# Inference C structure
for number in tqdm(range(data_num_C)):
    file_name = "TEST_C_{0:04d}".format(number)
    test_df = pd.read_csv(f"./test/C/{file_name}.csv")
    submission.loc[submission["ID"]==file_name, 'flag_list'] = algorithm(test_df, input_Q_C, output_Q_C, target_P_list_C, group_P_list_C, level3_sigma)

# D structure
input_Q_D = ['Q1']
output_Q_D = ['Q2','Q3','Q4','Q5']
target_P_list_D = [f"P{i}" for i in range(1,7)]
group_P_list_D = [['P1'], ['P2'], ['P3'], ['P5'], ['P4','P6']]
data_num_D = 2738
```

대회가 끝난 후, 상위권분들의 결과물 공유

비결이 궁금한 분들을 위해 미리 공유하는 몇가지 내용

 Statistics

 공동작성자

2024.12.16 11:56 466 조회

👍 15 💬 9

오랜만에 순위권에 들어서 정말 기쁩니다.
여러 대회에 참여하다보면 Score의 벽을 만나는 경우가 많은데, "도대체 저 위에 있는 사람들은 어떻게 한 거지?"하는 의문이 들 때가 많았습니다.
이번 대회에도 큰 점수의 벽이 있어서, 지난 대회들의 저 같이 진짜 순수하게 궁금한 분들을 위해 미리 몇가지 내용을 공유합니다.

1. 외부데이터 vs 알고리즘 vs 데이터 처리

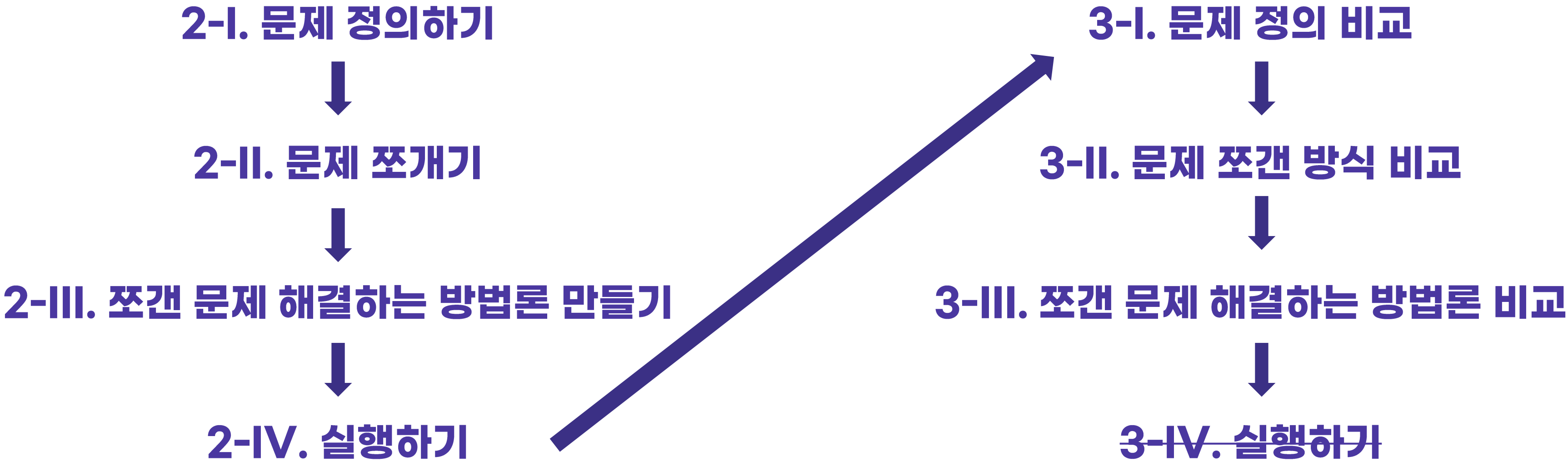
일반적으로 점수 벽이 생기는 이유는 외부 데이터 활용, 사용한 알고리즘의 종류와 특성, 파생변수 생성 등 데이터 처리 등으로 나뉩니다.
먼저 이 대회에서는 외부 데이터를 활용할 수 없습니다.
남은 것은 알고리즘과 데이터 처리인데, 대회 제목인 "이상 감지"가 함정입니다.

2. "관망 이상"의 정의

많은 팀이 이상 감지를 위해 anomaly detection에 특화된 알고리즘을 활용했을 것 같습니다.
autoencoder류로 모델을 적합하고 관망A, B를 활용해 임계값을 설정하는 방향이었을 것 같네요.
혹은 LSTM 류의 시계열 모델을 적합하고, 예측값과 실제값의 차이로 이상 지점을 찾는 것도 가능합니다.

대회의 이상향, 큰 꿈은 그런 방향이 맞지만 현실적으로는 실제 데이터에서 "anomaly", "flag"의 값이 1인 관측치를 설명하는 것이 목표입니다.
현실적인 대회 주제는 "K-water가 설정한 anomaly의 기준 찾기"라고 생각할 수 있습니다.

3. 실패한 프로젝트 의미있게 만들기



3-I. 문제 정의 비교

3-I. 문제 정의 비교

1. 공모 주제 및 분석 목표 설정

2024 제4회 K-water AI 경진대회



3-II. 문제 조건 방식 비교

3-III. 조건 문제 해결하는 방법론 비교

개요

목표

- 상수도 관망의 이상 시점 및 누수 발생 구간 탐지

제약

- 1분 단위로 구성된 데이터
- T 시점에서 주어진 직전 1주일(60*24*7=10,080분)간 압력계/유량계 값 추이로 T+1 시점에서의 관망 이상 및 누수 발생 구간 탐지
- 관망구조별 상이한 압력계/유량계의 위치, 압력계/유량계의 개수, 펌프의 개수 및 유무
- 특정 관망구조에 유리하지 않은 범용적인 알고리즘 개발
- 이상 시점 평가 지표 - 자카드 유사도(Jaccard Similarity)
- 누수 발생 구간 평가 지표 - 변형된 F1 Score

우수작 발표 PPT

문제정의 Not bad

3-II. 문제 조건 방식 비교

모델링 - 이상 시점 탐지



1. 마분위수 기반 임계값 설정

- IQR을 사용해 기본 임계값과 상위 임계값 설정.
- (유입량 - 유출량)이 임계값들을 초과하는지 확인.

2. 연속된 이상치 탐색

- 데이터를 역순으로 탐색하며 연속된 이상치 구간 길이와 허용된 False 값 확인.

3. 추가 조건 확인

- 이상치 구간 내 상위 임계값 초과 여부와 음수 값 존재 여부 확인.

4. 최종 판단

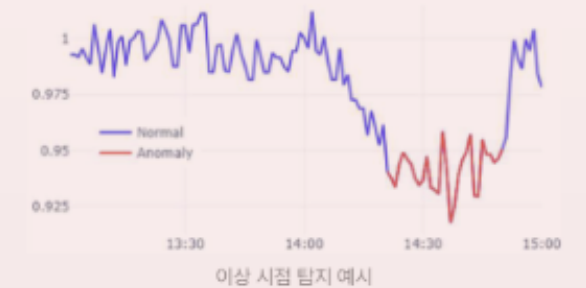
- 연속된 이상치 길이, 임계값 초과 조건, 펌프 개수 변화 값 및 마지막 값을 종합해 이상치 여부를 결정.

4. 파생 변수 생성 및 규칙 설정

이상 시점 및 누수 발생 구간 탐지 조건 설정

이상 시점 탐지

- 이상 시점 조건1
[이동 평균 유수율 97.5% 이하 4분 지속] & [유수율 97.5% 이하 6분 지속]
- 이상 시점 조건2
[유수율 95% 이하 3분 지속]
- 정상 시점 조건(이상 예외 조건)
[최근 6분 이내 최대 유수율이 100% 이상]



이상 시점 탐지 예시

누수 발생 구간 탐지

- 누수 구간 탐지 선행 작업
이상 시점 직전 30분 평균 압력 대비 이상 시점의 평균 압력 비율(변동비) 계산
- 누수 발생 구간 조건
[전체 평균 변동비 대비 해당 압력계 변동비가 97.5% 이하]
[위의 조건에 해당하는 압력계가 없을 경우 변동비 최솟값 구간으로 설정]



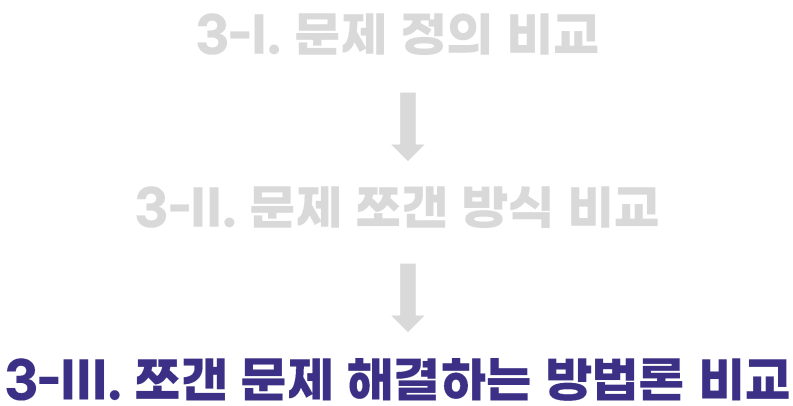
누수 발생 구간 탐지 예시

- * Test 데이터의 경우 최종 6분 데이터를 활용하여 T 시점이 이상 조건에 해당할 경우 T+1 시점도 이상이라고 판단
- * 조건 설정 임계값은 학습 데이터의 A 관망을 참고하고, Public Score를 기준으로 시행착오(trial & error)를 통해 설정

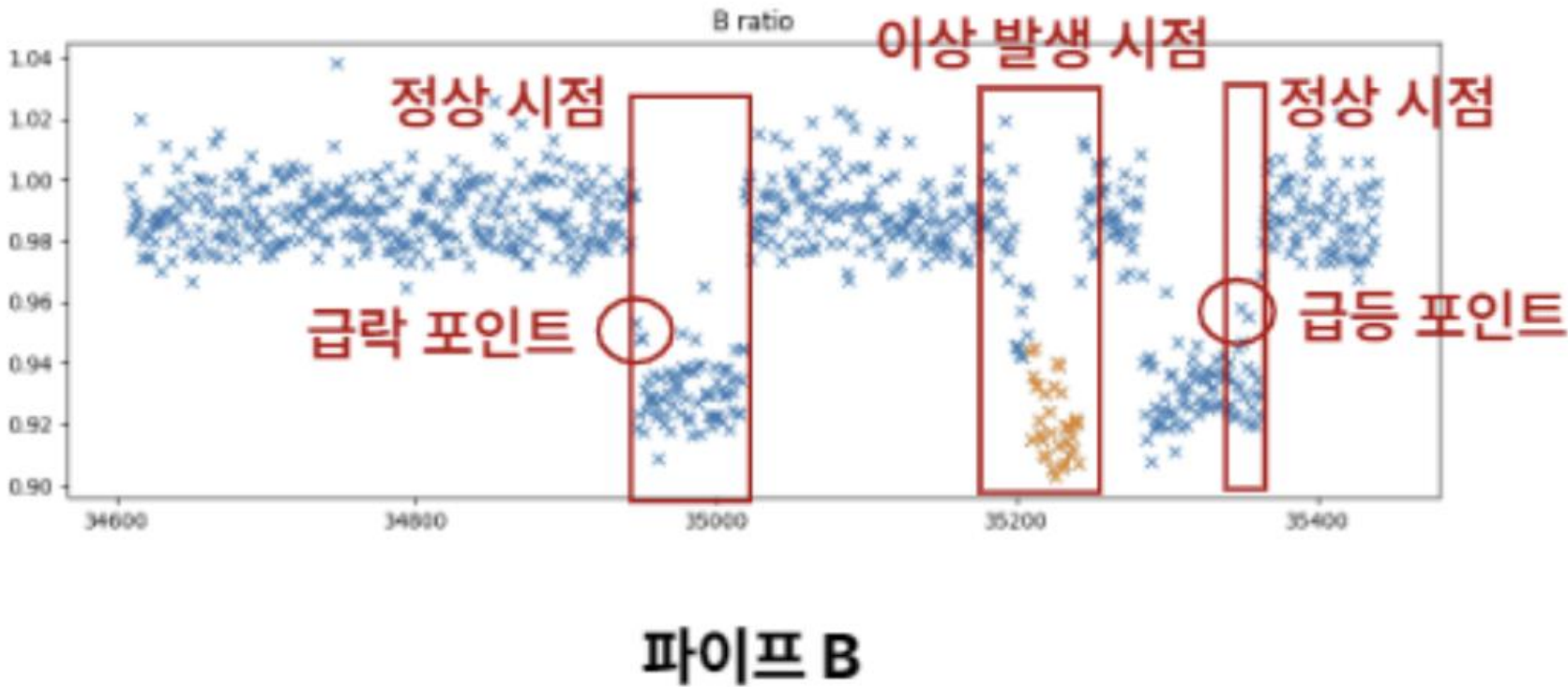
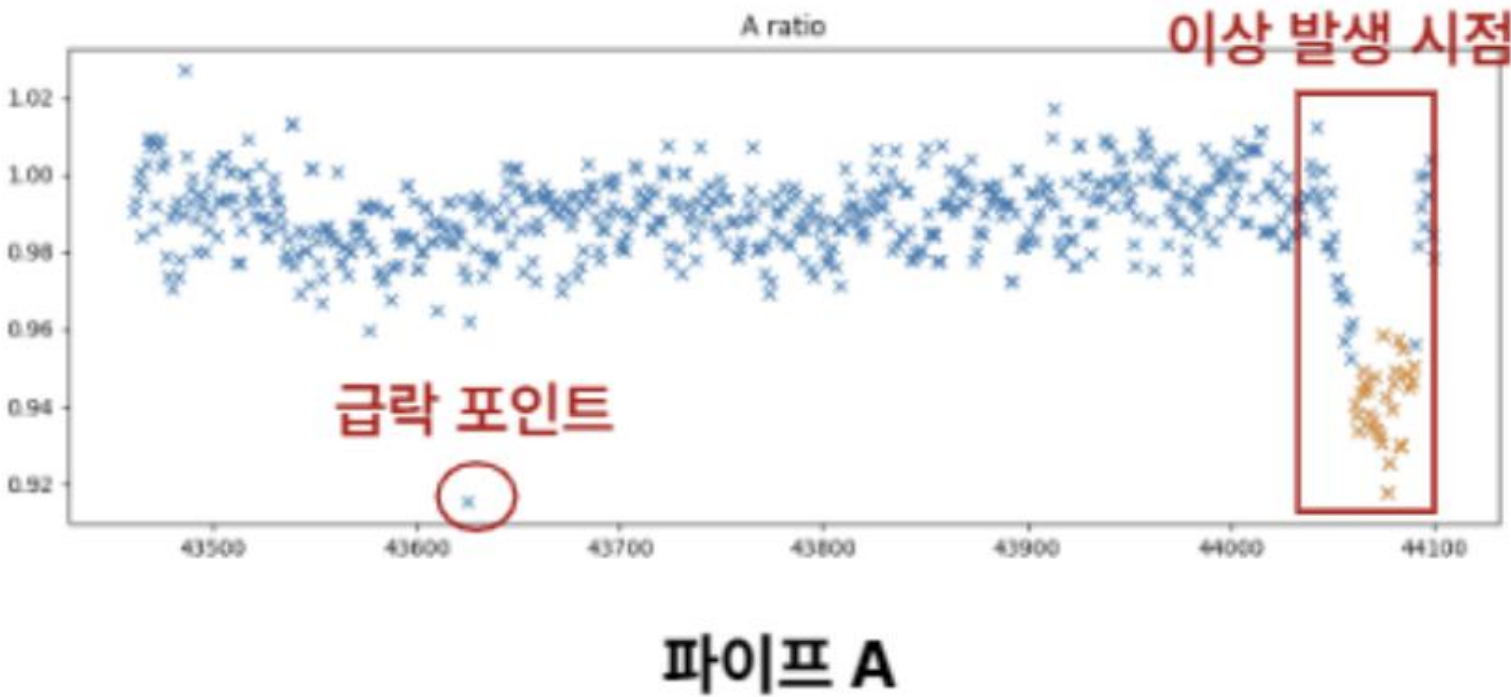
문제 쪼개기 Not bad

3-III. 쪼갬 문제 해결하는 방법론 비교

보완점 1. 급락하는 상황은 이상 상황이 아니다.



EDA 결과



탐색적 데이터 분석(EDA)을 조금 더 깊이 있게 했다면...

3-III. 쪼갬 문제 해결하는 방법론 비교

보완점 2. Q를 단순히 그 자체로 보는 것이 아닌 관계성을 부여해야 한다.

3-I. 문제 정의 비교



3-II. 문제 쪼갬 방식 비교



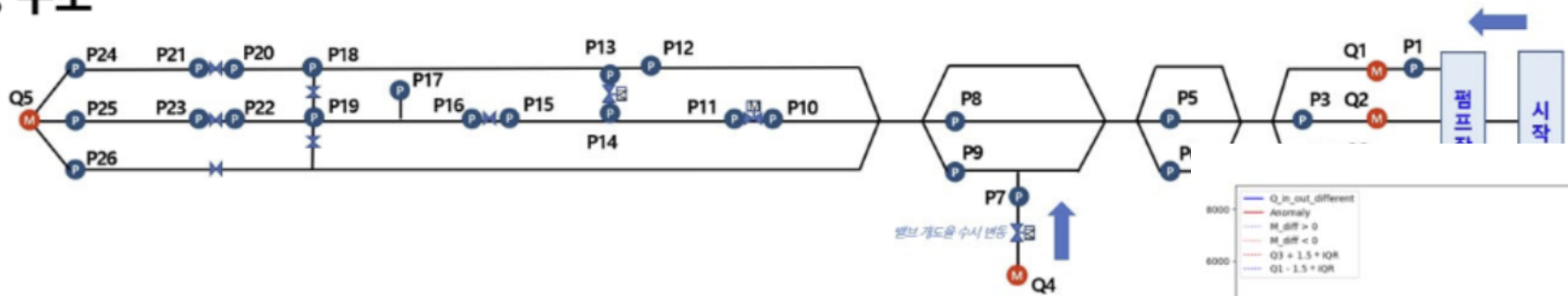
3-III. 쪼갬 문제 해결하는 방법론 비교

[배경]

본 경진대회는 상수도 관망의 이상 시점과 누수 발생 구간을 정확하게 탐지할 수 있는 범용 AI 알고리즘 개발을 목표로 하고 있습니다.

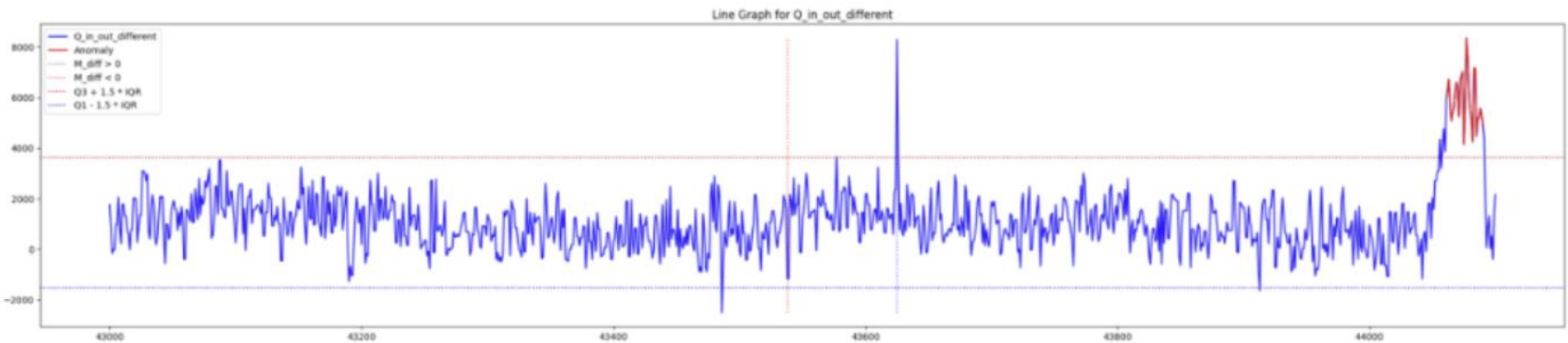
관망구조 기반 그룹화 - 유량계 결과

A 관망 구조



그룹화 결과

입력: {1,2,3,4}	출력: {5}
---------------	---------



(유입량 - 유출량)에 대한 연 그래프

문제 상황에서 “누수 ” 라는 키워드에 조금 더 집중했다면...

3-III. 쪼갬 문제 해결하는 방법론 비교

좋았던 점 1. P 변화율을 잘 체크했다는 점

좋았던 점 2. 이상치 탐지 모델이 아닌 Rule-Based로 진행한 점

3-I. 문제 정의 비교



3-II. 문제 쪼갬 방식 비교



3-III. 쪼갬 문제 해결하는 방법론 비교

4. 질문에 대한 우리의 답

“일단 대회는 찾았는데, 어떻게 프로젝트를 시작하지?”

**제공받은 데이터로 EDA를 해보자!
이를 통해 몰랐던 도메인 지식을 발굴하고,
프로젝트 진행 방향을 잡을 수 있다**

“프로젝트는 어떤 진행 과정으로 진행하지?”

다양한 방법을 시도하면서 데이터를 보는 초점을 맞춰 나가는 연습을 해보자!

“이 프로젝트에 얼마만큼 투자를 하고, 나에게 얼마나 의미가 있을까?”

**처음 시작하는 입장에서는 재미있을 만큼만 만져보자!
하나의 프로젝트에서 깊게 고민한 경험과 정답을 비교해보는 과정만으로도
큰 의미를 갖지 않을까…?**

감사합니다.