

누수 탐지 기업 비즈니스 모델 제안서

Business Model Proposal

01

문제 정의 및
제안 프로세스

02

비용 구조 분석

03

분류 모델 제작 및
성능 검정

04

비즈니스 모델
전환 제안

Executive Summary

1. 문제 정의



- 비용구조 분석을 통한 비즈니스 모델 전환의 효용성 파악
- 기존 방안 대비 **제시 방안**의 비용 절감 효과 및 기대효과 고려 후, 센서 시스템 도입 여부를 결정

2. 제안 프로세스

- **비용 구조 분석 : 발생하는 비용을 세 가지 대분류로 구분 후 비용 산정**
 - 인건비: 노동자의 노동에 대한 대가, 혹은 노동과 관련하여 발생하는 비용 (ex. 보수, 식대, 신규 직원 채용 및 교육비 등)
 - 서비스 운영비: 업체가 누수 탐지 및 수도관 수리 서비스를 제공하는 과정에서 발생하는 비용 (ex. 교통지원금, 누수 보상비, 컨설팅 비용 등)
 - 장비 구매비용: 업체가 누수 탐지 및 수도관 수리 서비스를 제공하기 위해 필요한 장비 구매에 사용되는 비용 (ex. 센서 구매비, 수도관 교체비 등)
- **분류 모델 제작: 세 단계로 진행**
 - 평가지표 지정: 분류모델의 Accuracy, 사전에 정의한 비용함수, FNR를 기반으로 평가지표 도출
 - 최적 센서 조합 선택: 센서 개수별 조합 생성 후 사전 정의한 평가지표로 높은 정확도 및 가격경쟁력의 최적 조합 파악
 - 하이퍼 파라미터 조정: 최적 조합의 분류 모델 성능 향상을 위해 파라미터 조정
- **비즈니스 모델 선택**
 - 총 비용을 토대로 비즈니스 전환의 효용성 제시
- **선택한 모델의 기대효과 추산**

비용 구조 분석

> 센서 도입으로 가져올 변화에 따라 비용을 분류하였고, 변화 정도는 비즈니스 모델 전환의 효용성에 큰 영향을 불러올 것으로 보임

상황 가정

센서

- 센서 자체의 고장 발생 x
- 누수 시에만 새 센서로 교체

배터리

- 배터리 방전 가능성 배제
- 배터리만 교체하는 일 x

누수

- 한 달 누수건수: 100건

오탐지

- 누수라고 잘못 탐지
- 누수가 아니라고 잘못 탐지

- FNR, FPR으로 오탐지 건 수 파악

비용 구조



- 센서 도입 시 변경되는 비용
- 센서 도입 시 추가로 나가게 되는 비용
- 센서 도입 여부 상관없이 지출되는 비용

분류 모델 제작 및 결과

> 높은 정확도의 조합을 대상으로 총 비용이 낮은 최적 센서 조합을 선정함. 총 비용은 누수 배상금에 민감하기에 FNR이 중요함

비용 함수

기존 비즈니스 모델

총 비용=(보수+식대+4대보험비+상여금+차량리스비*0.1+출장숙소비+교통지원금)*4주*1000명+수도관교체*100+ 수도관 부품교체*4주
=₩3,450,040,000

신 비즈니스 모델

총 비용 =(보수+식대+4대보험비+상여금+ 차량리스비*0.1+**신규직원채용*0.1**)*4주* 250명+ 교통지원금*250+ **퇴직금*775**+ 출장숙소비*250+ **누수보상비*100*FN**+ **건당컨설팅비*5000**+수도관교체*100+ 수도관 부품교체*4주+ **센서가격합계*5000/12+ 센서가격합계 *100 + 센서 오작동으로 인해 발생한 출장비 *FP**

Test Accuracy + FNR + 도입 후 총 비용 고려

- 최적 조합: 동일 모델 하에서 구한 조합들의 accuracy가 비슷한 경우, **비용이 가장 낮은 조합**
- FNR: 누수일 때 누수가 아니라고 탐지한 비율

	로지스틱 회귀분석			KNN classifier		SVM	
조합	A[12, 17, 34, 42, 44]	B[10, 26, 36,45]	C[10, 36, 42, 45]	D[10, 13, 17, 37, 48]	E[10, 13, 17, 37, 48, 49]	F[11, 16, 32]	G[10, 11, 12, 16,19,22,36,37,38,43,45,46, 47, 48]
정확도	0.9048	0.8810	0.8810	0.9048	0.9048	0.8333	0.8333
가격	61,800	26,400	32,400	53,700	64,800	37,800	188,400
FNR	0.0741	0.1111	0.1111	0.1111	0.1111	0.0476	0.0476
도입 후 총 비용	₩ 1,000, 848,799	₩ 1,001 ,058,799	₩ 1,004, 158,799	₩ 1,014, 755,699	₩ 1,020, 490,699	₩ 978, 573,965	₩ 1,056, 383,965

Kernel: RBF Kernel: Linear

Test Accuracy 기준 우수 조합: A, D, E, G
 총 비용 고려 후 최종 최적센서 조합: F

F[11, 16, 32]의
 분류모델 성능 개선을 위해
 하이퍼파라미터 조정

비즈니스 모델 전환 제안

〉 신 비즈니스 모델 도입은 총비용 감소의 효과를 불러올 것이고, 이는 투자 영역 및 사업 확장으로 이어질 수 있음

비즈니스 모델 간 비용 비교

기존 비즈니스

₩3,450,040,000

신 비즈니스

₩ 978,573,965

- **인력 감축**
 - 인건비 절감 (보수, 식대, 4대보험, 상여금)
- **파견 횟수 감소**
 - 출장 숙소 비용 · 교통지원금 절감
 - 차량 운용 및 관리유지비 (리스비) 절감
- **신 비즈니스 모델 도입 시**
 - 총 비용은 센서 구매 비용보다 누수에 대한 보상 비용에 더 큰 영향을 받음.
 - 누수 오염지 비율 (FNR)을 크게 반영

신 비즈니스 모델의 활용 제안

인력자원 투자

- 신규 직원 채용
 - 센서 관련 전문가 채용

R&D 투자

- 센서 기술 확보
 - 현재: 외주업체로부터 센서 구매
 - 기존 인건비를 센서 개발에 투자하여 기술력 확보
 - 미래: 자체 생산으로 센서 구매 불필요

사업 확장

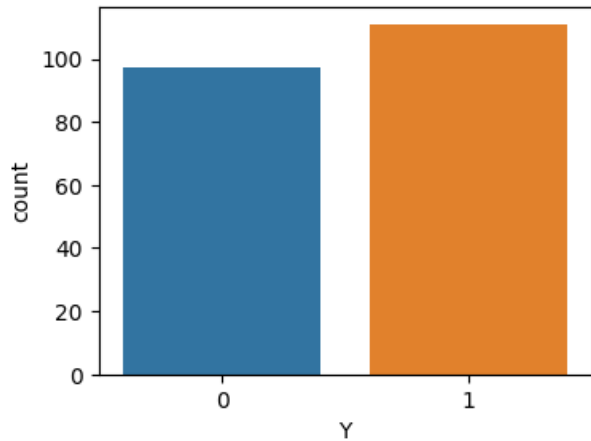
- 탐지 규모 및 지역 확대
 - 현재: 일부 지역의 5000개 수도관 관리
 - 미래: 지부 설립을 통한 전국구 사업 확대

APPENDIX.

EDA

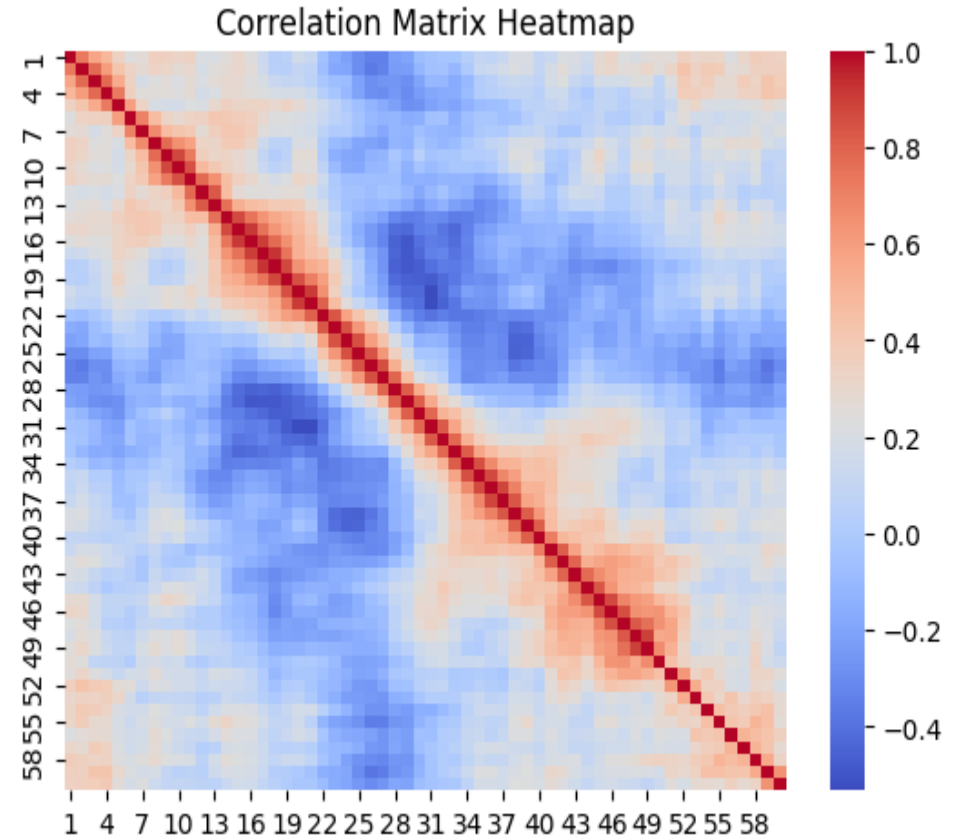
실제 누수여부 시각화

	1	2	3	4	5	6	7	8	9	10	...	52	53	54	55	56	57	58	59	60	Y
1	0.0200	0.0371	0.0428	0.0207	0.0954	0.0986	0.1539	0.1601	0.3109	0.2111	...	0.0027	0.0065	0.0159	0.0072	0.0167	0.0180	0.0084	0.0090	0.0032	0
2	0.0453	0.0523	0.0843	0.0689	0.1183	0.2583	0.2156	0.3481	0.3337	0.2872	...	0.0084	0.0089	0.0048	0.0094	0.0191	0.0140	0.0049	0.0052	0.0044	0
3	0.0262	0.0582	0.1099	0.1083	0.0974	0.2280	0.2431	0.3771	0.5598	0.6194	...	0.0232	0.0166	0.0095	0.0180	0.0244	0.0316	0.0164	0.0095	0.0078	0
4	0.0100	0.0171	0.0623	0.0205	0.0205	0.0368	0.1098	0.1276	0.0598	0.1264	...	0.0121	0.0036	0.0150	0.0085	0.0073	0.0050	0.0044	0.0040	0.0117	0
5	0.0762	0.0666	0.0481	0.0394	0.0590	0.0649	0.1209	0.2467	0.3564	0.4459	...	0.0031	0.0054	0.0105	0.0110	0.0015	0.0072	0.0048	0.0107	0.0094	0
...
204	0.0187	0.0346	0.0168	0.0177	0.0393	0.1630	0.2028	0.1694	0.2328	0.2684	...	0.0116	0.0098	0.0199	0.0033	0.0101	0.0065	0.0115	0.0193	0.0157	1
205	0.0323	0.0101	0.0298	0.0564	0.0760	0.0958	0.0990	0.1018	0.1030	0.2154	...	0.0061	0.0093	0.0135	0.0063	0.0063	0.0034	0.0032	0.0062	0.0067	1
206	0.0522	0.0437	0.0180	0.0292	0.0351	0.1171	0.1257	0.1178	0.1258	0.2529	...	0.0160	0.0029	0.0051	0.0062	0.0089	0.0140	0.0138	0.0077	0.0031	1
207	0.0303	0.0353	0.0490	0.0608	0.0167	0.1354	0.1465	0.1123	0.1945	0.2354	...	0.0086	0.0046	0.0126	0.0036	0.0035	0.0034	0.0079	0.0036	0.0048	1
208	0.0260	0.0363	0.0136	0.0272	0.0214	0.0338	0.0655	0.1400	0.1843	0.2354	...	0.0146	0.0129	0.0047	0.0039	0.0061	0.0040	0.0036	0.0061	0.0115	1



누수 Y(1)과 누수 N(0)는
비슷한 분포를 보임.

Correlation Matrix

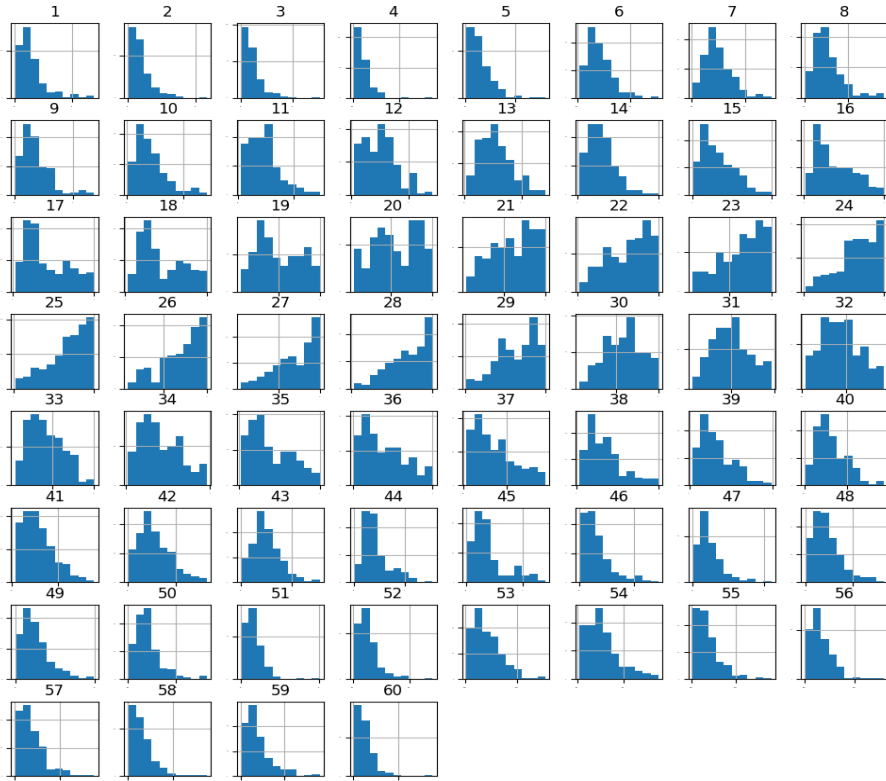


각 센서간 correlation 알아보기 위해 히트맵 생성

APPENDIX.

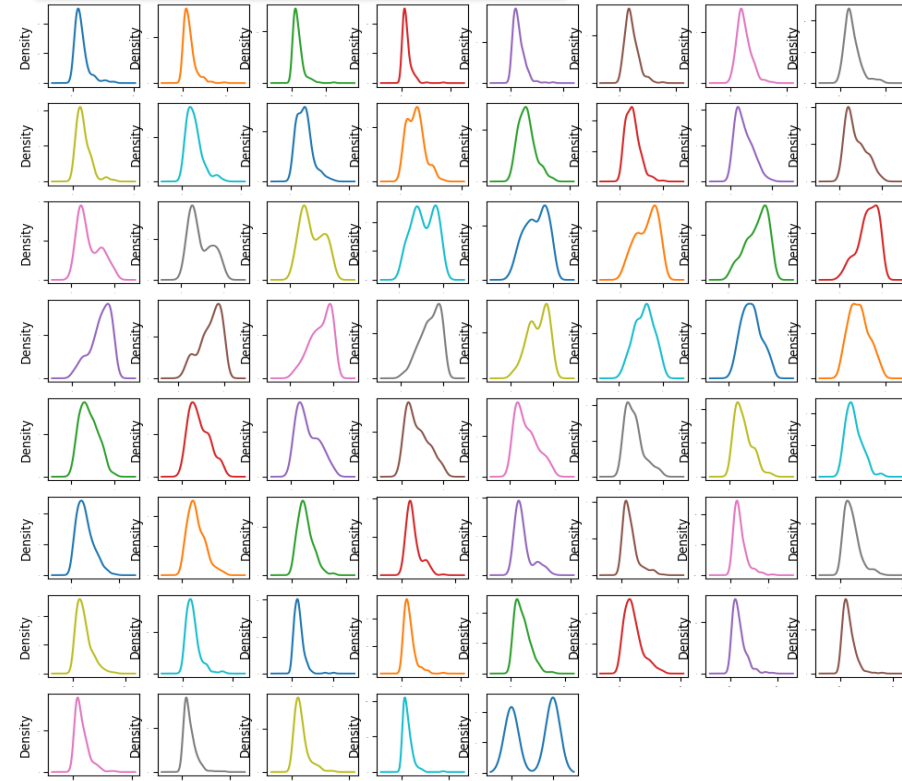
EDA

Histogram



- 개별센서와 수도관에서의 누수음 주파수 관계
- 오른쪽으로 꼬리가 긴 분포
- = 가장 높은 주파수와 가장 낮은 주파수 관측치 간 차이 ↑

Density

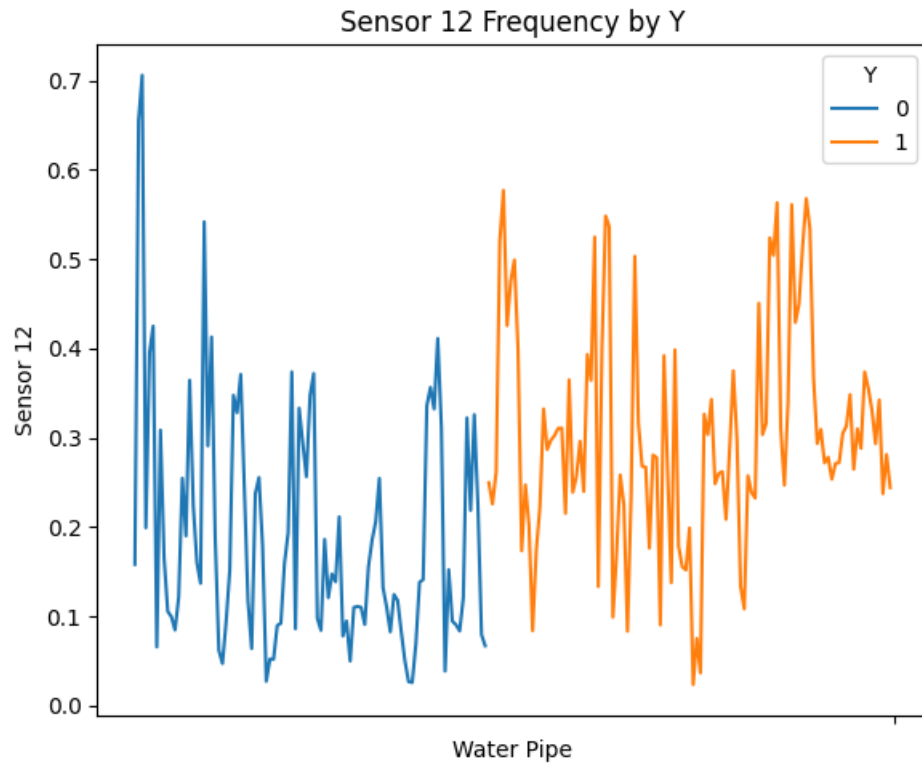


- 개별센서와 수도관에서의 누수음 주파수 관계

APPENDIX.

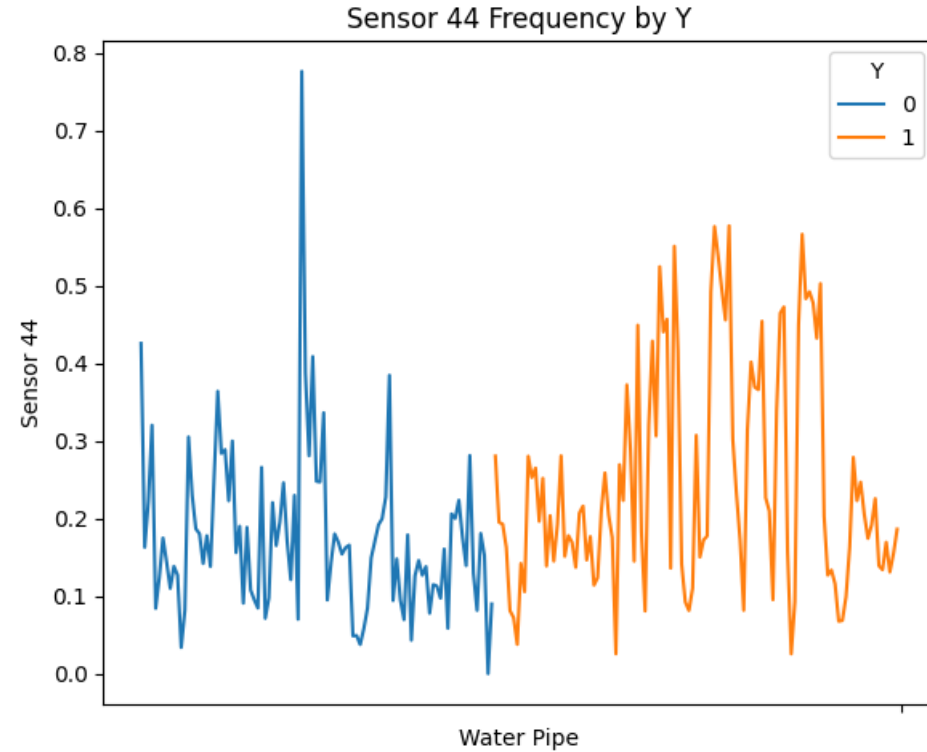
> EDA

Sensor 12 Frequency



- 0.0236~0.706사이에서 다양하게 포진
- 전반적으로 Y=1일 때, Y=0에 비해 높은 주파수 분포를 보임
- 다만, 주파수 중 가장 높은 주파수는 Y=0일 때 관측

Sensor 44 Frequency



- 0~0.7762사이에서 다양하게 포진
- 0.7762 관측치를 제외, Y=0 주파수의 분포가 Y=1에 비해 변동폭이 작음
- 주파수 중 가장 높은 주파수는 Y=0일 때 관측

APPENDIX.

› Logistic Regression

1차 센서 제거 by Accuracy

제거 센서	정확도
19	0.47619
21	0.47619
22	0.5
23	0.47619
29	0.452381
39	0.47619

정확도 0.5 이하 센서 제거

정확도가 0.5 => 0, 1 중 하나를 랜덤하게 선택할 확률과 같음

1차 센서 제거 by FNR

제거 센서	FNR
19 (1번 과정과 중복)	0.592593
20	0.518519
21 (1번 과정과 중복)	0.518519
23 (1번 과정과 중복)	0.518519
39 (1번 과정과 중복)	0.518519
44	0.518519

FNR 0.5 이상 센서 제거

*FNR: 실제로 1(누수)인데 0(누수없음)으로 예측한 비율


APPENDIX.

> Logistic Regression

2차 센서 후보군 선정

```
df_no = pd.DataFrame(columns=['Sensors', 'Accuracy', 'FNR'])
for i in range(1, 61):
    df_name = f'df_{i}'
    df_name = lr_sensors1(i)
    df_no = pd.concat([df_no, df_name], ignore_index=True)
df_no
```

센서 60개 모든 조합으로 모델 성능 평가 시도하였으나 시간多



Sensors	Accuracy	FNR
(10, 11, 36, 44)	0.880952	0.111111
(10, 26, 36, 45)	0.880952	0.111111
(10, 36, 42, 45)	0.880952	0.111111
(12, 17, 34, 43)	0.880952	0.111111
(12, 33, 43, 46)	0.880952	0.111111
(12, 35, 43, 48)	0.880952	0.111111

센서 4개 조합까지 성능 평가 후 Test Accuracy 최대 센서후보군 선정
>10, 11, 36, 44, 26, 45, 42, 12, 17, 34, 43, 33, 46, 35, 48

2차 센서 후보군 성능 평가

Sensors	Accuracy	FNR
(44, 42, 12, 17, 34)	0.904762	0.074074
(10, 11, 36, 44, 45, 42, 12, 17, 34, 33)	0.904762	0.111111

첫번째 조합: Accuracy 상승(0.88→0.90), FNR 감소(0.1→0.07)

두번째 조합: Accuracy 상승(0.88→0.90), FNR 유지(0.1→0.1)

6900 9000 28500 7500 9900
11100 23400 8100 6900 4200 9000 28500 7500 9900 15900

>각 센서 조합의 가격

APPENDIX.

› Logistic Regression

최종 센서 조합 선정

2차 센서 후보군 선정 과정에서의 조합들의 Accuracy, FNR, 각 센서당 비용을 고려해 최적 조합 후보 3가지 도출

센서 조합	정확도	FNR	가격
12,17,34,42,44	0.905	0.074	28500, 7500, 9900, 9000, 6900
10,26,36,45	0.881	0.111	11100, 3000, 8100, 4200
10,36,42,45	0.881	0.111	11100, 7500, 9900, 7800

```
price.describe()
```

```
count      60.000000
mean      11640.000000
std       5309.064842
min       2700.000000
25%       8100.000000
50%      11100.000000
75%      14400.000000
max       28500.000000
Name: 가격, dtype: float64
```

최종 센서 조합 성능 출력

Sensors	Accuracy_train	Accuracy_test	FNR	FPR	AUC	Price Total	Loss
[12, 17, 34, 42, 44]	0.7108	0.9048	0.0741	0.0741	0.8617	61800.0	31930000.0
[10, 26, 36, 45]	0.7590	0.8810	0.1111	0.0769	0.8667	26400.0	13640000.0
[10, 36, 42, 45]	0.7470	0.8810	0.1111	0.0769	0.8519	32400.0	16740000.0

비용 함수 적용해 최종 조합 선정

12,17,34,42,44 (FNR이 0.074로 매우 낮음!!)

해당 조합의 가격이 더 비싸더라도 선택한 이유

- 기업에 자문 후, 한달 100건의 누수가 발생한다고 가정하고 시작
- 센서의 구매 및 교체 비용 << FN시의 500만원 보상이 총비용에 더 큰 영향 미침

APPENDIX.

› Logistic Regression

참고) Bayesian Optimization

```
#Bayesian Optimization을 위한 하이퍼파라미터 공간 정의
param_space = {
    'C': Real(1e-3, 1e+3, prior='log-uniform'), #규제 강도.
                                                #log-uniform 분포는 큰 범위에서의 샘플링에 유용
                                                #Real: 범위 내에서 실수값 샘플링 가능
    'max_iter': Integer(1000, 10000, prior='log-uniform')}

opt = BayesSearchCV(model, param_space, cv=10, n_jobs=-1)
opt.fit(X_train1, y_train1)
```

베이지안 최적화 기법 사용 (cv=10)

- C (규제 강도): $1e-3 \sim 1e+3$
- max_iter (최대 반복 횟수): 1000 ~ 10000
- CV=10으로 지정

APPENDIX.

> KNN classifier

1차 센서 제거 by Accuracy

버릴 센서	정확도
3	0.452381
18	0.476190
19	0.476190
21	0.5
22	0.5
23	0.5
29	0.5
30	0.5
31	0.476190
33	0.5
35	0.476190
39	0.5
41	0.380952
53	0.476190
54	0.380952
56	0.476190
57	0.452381
60	0.5

센서 1개의 조합에서 **정확도 0.5**
이하 센서 제거
정확도가 0.5 => 0, 1 중 하나를
랜덤하게 선택할 확률과 같음

1차 센서 제거 by FNR

버릴 센서	fnr
14	0.555556
16	0.518519
28	0.555556
45	0.518519

FNR 0.5 이상 센서 제거

*FNR: 실제로 1(누수)인데 0(누수없음)으로 예측한 비율

APPENDIX.

> KNN classifier

2차 센서 제거 by Accuracy/ FNR

센서 2개 조합 중

정확도가 0.5 이하, FNR이 0.5 이상인 조합은 제거

=> 버릴 센서: 1, 2, 4, 5, 6, 7, 8, 9, 15, 20, 24, 25, 26, 27, 32, 34, 38, 40, 42, 43, 44, 47, 50, 51, 52, 55, 58, 59

2차 센서 후보군 성능 평가

	Sensors	Accuracy	FNR	K
347	(10, 11, 12, 37, 49)	0.928571	0.074074	{'n_neighbors': 3}
426	(10, 13, 17, 37, 48)	0.904762	0.111111	{'n_neighbors': 5}
615	(10, 11, 12, 37, 48, 49)	0.904762	0.074074	{'n_neighbors': 3}
695	(10, 13, 17, 37, 48, 49)	0.904762	0.111111	{'n_neighbors': 5}

남은 센서들에 대해서 센서 3개 조합 ~ 센서 10개 조합에 대해 로지스틱 적합 후 성능 도출

-> Test Accuracy가 0.881이상인 조합 추출.

APPENDIX.

> KNN classifier

최종 센서 조합 선정

2차 센서 후보군 선정 과정에서의 조합들의 Accuracy, FNR, 각 센서당 비용을 고려해 최적 조합 후보 2가지 도출

센서 조합	정확도	FNR	가격
10,13,17,37,48	0.905	0.111	11100,10500,7500,2700,21900
10,13,17,37,48,49	0.905	0.111	11100,10500,7500,2700,21900,11100

```
price.describe()
count      60.000000
mean     11640.000000
std       5309.064842
min       2700.000000
25%       8100.000000
50%      11100.000000
75%      14400.000000
max      28500.000000
Name: 가격, dtype: float64
```

최종 센서 조합 성능 출력

Sensors	Accuracy_train	Accuracy_test	FNR	FPR	AUC	Price Total	Loss
[10, 13, 17, 37, 48]	0.8675	0.9048	0.1111	0.04	0.9210	53700.0	27745000.0
[10, 13, 17, 37, 48, 49]	0.8675	0.9048	0.1111	0.04	0.9148	64800.0	33480000.0



비용 함수 적용해 최종 조합 선정

10,13,17,37,48 선정

- FNR 0.1111, FPR 0.04로 모두 낮고, 가격은 10,13,17,37,48 조합에서 더 낮음

APPENDIX.

› KNN classifier

참고) Bayesian Optimization

```
#Bayesian Optimization을 위한 하이퍼파라미터 공간 정의
param_space = {
    'n_neighbors': Integer(3, 20),
    'p': Integer(1, 3),
    'weights': Categorical(['uniform', 'distance'])
}

opt = BayesSearchCV(model, param_space, cv=10, n_jobs=-1)
opt.fit(X_train1, y_train1)
```

베이지안 최적화 기법 사용 (cv=10)

- n_neighbors (분류에 사용할 이웃 데이터 포인트 수): 3 ~ 20
- p (거리 계산 방법): 1, 2, 3
- weights (가중치): uniform, distance

APPENDIX.

> SVM

SVM 모델 코드

```
kernels = ['linear']
C_values = [0.0001, 0.001, 0.01, 0.1, 1, 10, 100, 1000, 10000]
```

Linear커널의 선택

*C값을 다양하게 적용해보기 위해 C값을 한정하지 않았음

```
# 상위 5개의 유니크한 조합을 선택
results_df = pd.DataFrame(results)
results_df.sort_values('Accuracy', ascending=False, inplace=True)
top_results = pd.DataFrame()

# 유니크한 조합들을 추출
seen_combinations = set()
for _, row in results_df.iterrows():
    features_key = row['Selected Features']
    if features_key not in seen_combinations:
        top_results = top_results.append(row, ignore_index=True)
        seen_combinations.add(features_key)
    if len(top_results) >= 5:
        break

final_results = pd.concat([final_results, top_results], axis=0)
```

상위 5개의 유니크한 조합(기준:정확도)을 선택해 추출함
Seen_combinations을 활용해 중복된 선택을 방지하고자 하였음

```
def calculate_total_cost(row):
    fnr = row['FNR']
    fpr = row['FPR']
    total_price = row['Total Price']

    # fpr가 1에 매우 가까운 경우를 처리
    if fpr >= 0.999: # fpr가 0.999 이상일 경우, fpr를 조금 줄여서 0으로 나누는 상황 방지
        fpr = 0.999

    total_cost = 5000000 * 100 * fnr + 110000 * (100 - 100 * fnr) * fpr / (1 - fpr) + total_price * 5000 / 12
    return total_cost
```

Total cost(총비용)을 마지막 칼럼에 추가

*총비용 계산시에는 FPR을 0.999로 제한하였음

: FPR(False Positive Rate)이 1에 가까워 0으로 나누는 것을
방지하기 위해

APPENDIX.

> SVM

상위조합 선정

C	Selected Features	Accuracy	FPR	FNR	Total Price	Total Cost ▲
10.0	11, 37, 47	0.7857142857142857	0.3333333333333333	0.09523809523809523	41100.0	69720238.09523809
10.0	11, 47	0.7142857142857143	0.47619047619047616	0.09523809523809523	38400.0	72666666.66666666
1.0	11, 12, 45	0.7380952380952381	0.42857142857142855	0.09523809523809523	56100.0	78458333.33333334
100.0	3, 4, 11, 49	0.7142857142857143	0.47619047619047616	0.09523809523809523	61200.0	82166666.66666666
10.0	4, 11, 37, 38, 47	0.7142857142857143	0.47619047619047616	0.09523809523809523	65700.0	84041666.66666666
1.0	11, 12, 36, 45, 47	0.8095238095238095	0.2857142857142857	0.09523809523809523	79200.0	84600000.0
1.0	11, 12, 36, 43, 45, 47	0.8095238095238095	0.2857142857142857	0.09523809523809523	87000.0	87850000.0
10.0	11	0.7142857142857143	0.42857142857142855	0.14285714285714285	23400.0	88249999.99999999
10000.0	1, 12, 49, 51, 53, 57	0.7142857142857143	0.47619047619047616	0.09523809523809523	88500.0	93541666.66666666
1.0	11, 12, 36, 45	0.8095238095238095	0.23809523809523808	0.14285714285714285	64200.0	101124999.99999999
10.0	11, 37, 38, 47	0.6904761904761905	0.47619047619047616	0.14285714285714285	53400.0	102249999.99999999
100.0	3, 4, 8, 11, 49	0.7619047619047619	0.3333333333333333	0.14285714285714285	65700.0	103517857.14285713
1.0	10, 11, 12, 16, 19, 22, 36, 37, 38, 43, 45, 46, 47, 48	0.8333333333333334	0.2857142857142857	0.047619047619047616	188400.0	106500000.0

비용 함수 적용해 최종 조합 선정

10,11,12,16,19,22,36,37,38,43,45,46,47,48

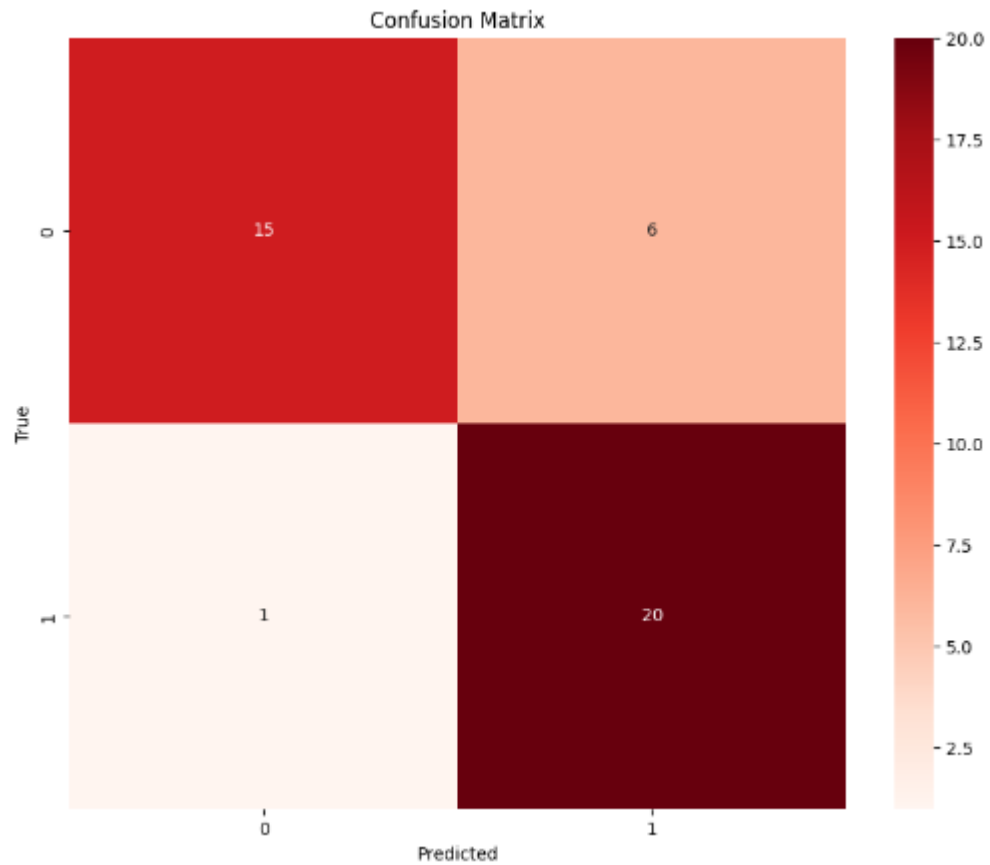
SVM모델의 경우 정확도가 타 모델들에 비해 높지 않아 최대한 높은 정확도를 지닌 조합 선정

➤ FNR은 0.048 정도로 매우 낮고, SVM모델 내 다른 조합과 비교시 FPR도 0.286정도로 낮은 수치를 보여줌

APPENDIX.

> SVM

최적 조합 Confusion Matrix



- F[11, 16, 32] 센서 선택
- Kernel function: rbf

› 비용 함수

분류	색	비용 분류	단가(₩/주)	센서 도입 전 비용 (₩/월)	센서 도입 후 비용 (₩/월)	상세설명
인건비		보수	₩ 500,000 (명)	₩2,000,000,000	₩500,000,000	주휴수당, 개인 장비 지원 포함
		식대	₩ 50,000 (명)	₩200,000,000	₩50,000,000	
		4대 보험	₩ 50,000 (명)	₩200,000,000	₩50,000,000	근로자 소득의 10%
		상여금	₩ 125,000(명)	₩500,000,000	₩125,000,000	약 300%로 가정 1500,000/12=
		퇴직금 (₩ 일 평균임금 × 30(일) × (재직일수/365))	₩41,100(명)	₩0	₩138,013,699	센서 도입 후 기존 대비(1000명) 25%로 인원 감축 첫달에만 드는 비용
		신규 직원 채용	₩ 35,000(명)	₩0	₩3,500,000	센서 관련 기술자(25명) 채용
서비스 운영비 (정기 점검)		차량 운용 및 관리유지비(리스)	₩ 250,000(대)	₩100,000,000	₩25,000,000	전체 인원의 10%에 대해 차량 지원
		출장 숙소 비용	₩60,000(명)	₩240,000,000	₩15,000,000	
		교통지원금	₩ 50,000(명)	₩200,000,000	₩12,500,000	출장지 주차비용+ 국도 이용 대금+유류비
		누수에 대한 보상 비용	₩ 5,000,000(건)	₩0	₩45,000,000	누수 발생 시
		컨설팅비용	₩ 400(건)	₩0	₩2,000,000	연락망 시스템 체계화하는데 필요한 비용 (전산망 체계화, 시스템 구축 등의 초기 IT 관련 비용)
장비 관련 비용		수도관 교체	₩ 100,000 (건)	₩10,000,000	₩10,000,000	누수 시 수도관 교체로 발생하는 비용
		수도관 부품 교체	10000 (주)	₩40,000	₩40,000	점검 시 수도관 파열 발견, 수도관 수리를 위한 부품 (누수 시에만 수도관 전체 교체 가정.)
		센서 구매 및 주문 비용 (₩ 최적 센서 조합 가격 × 5000)	₩	₩0		초기 수도관 5000개에 센서 부착
		센서 교체 비용 (₩ 최적 센서 조합 가격 × 누수건수)	₩	₩0	₩5,820,000	누수 시 새 센서 교체
		실제 누수가 발생하지 않았으나 센서가 올린 경우 발생할 출장비	₩ 110000(명)(건)	₩0	₩582,615	출장 숙소 비용+교통지원금
비용 총계			₩3,450,040,000	₩982,456,314		

APPENDIX.

> 비용 함수

appendix				
발생하는 비용은 맞으나, 세세하여 부록으로 따로 첨가함				
수리에 필요한 장비(드라이버 등) 구매 비용	<< 인건비에 장비 지원으로 포함			
보수 도구 감가상각비	<< 감가상각비 제외 가정			
누수된 수도관 처리가 제대로 되지 않았을 경우 발생하는 재파견 비용 + 그 시간에 다른 곳을 점검 가거나 새 수도관을 설치하러 갔다면 벌 수 있었을 수익에 대한 기회비용				
계약 파기로 인한 기회비용	<< 누수가 발생하였으나 센서가 울리지 않아 신뢰성 문제로 계약이 파기된 경우			
센서 고장 시 보수비용(고쳐서 다시 쓸 것인지 버리고 새로 살 것인지도 결정하는 비용함수 필요할 듯)				
이 오작동된 센서가 붙어있는 수도관을 점검하러 갈 시간에 벌 수 있었을 다른 보수에 대한 기회비용				
센서 점검	<< 인건비에 포함			

비용함수 가설

- (1)수리에 필요한 장비(드라이버)등의 구매 비용은 인건비에 장비 지원금액으로 포함시킴
- (2)보수 도구 감가상각비의 경우에는 residual연한을 알아야 감가상각을 산정한다는 어려움이 있어 감가상각 가정 배제
- (3)누수된 수도관 처리가 제대로 되지 않았을 경우 발생하는 재파견 비용 및 해당 시간 중 다른 곳 점검 및 설치 시 벌 수 있었던 수익에 대한 기회비용도 마찬가지로 정확한 산정 금액 어려움이 있어 배제
- (4)계약파기로 인한 기회비용 및 오작동센서 점검 기회비용은 금액 산정 어려움이 있어 배제
- (5)센서 고장 시 보수비용의 경우에는 고쳐서 다시 쓸 것인지 혹은 새로 구매할 것인지에 대한 자세한 가정이 필요할 것 같음
- (6)센서를 점검하는 비용은 인건비가 주로 될 것이라 인건비에 포함시킴