

[신재생e 설비(접속반) 건전성 평가 모델 분석]

Golden Time을 잡아라

이원

이유송

임혁진

목차 소개

01

제안 배경

02

분석 내용 및 방법

03

분석 결과

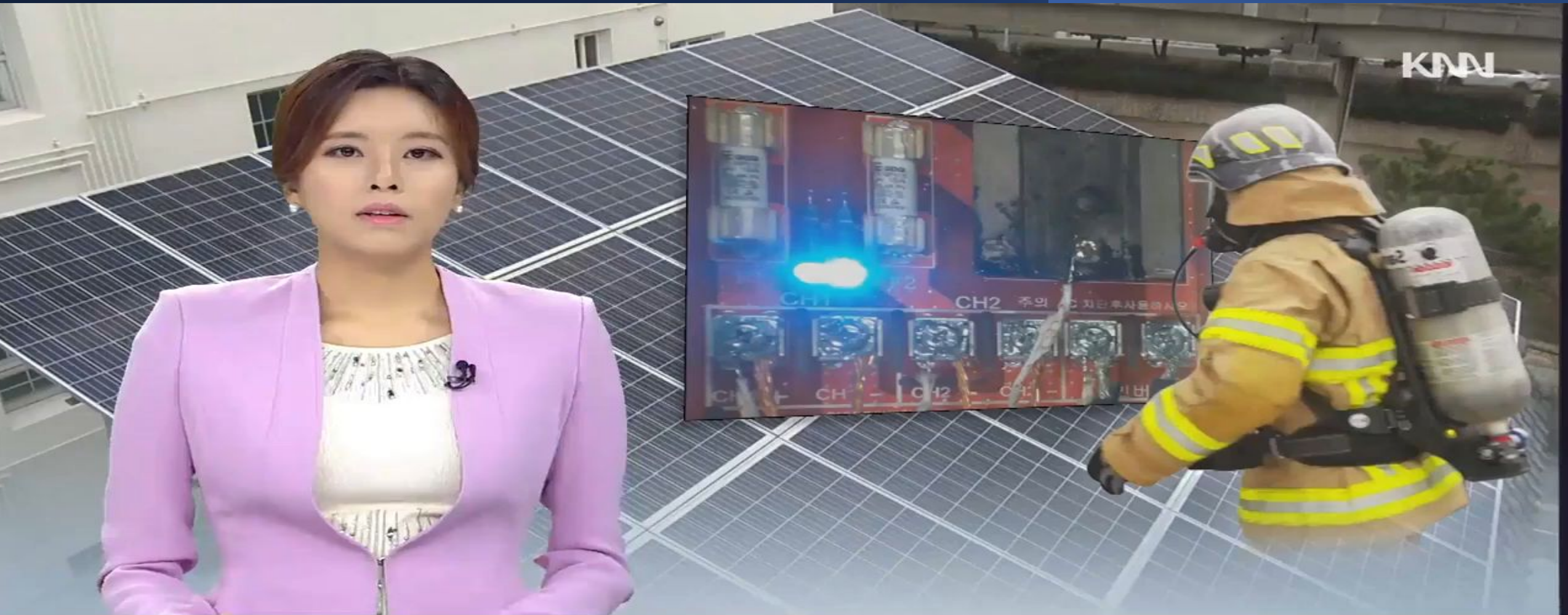
04

활용방안 및 한계점

제안 배경

1. 제안 배경

공공데이터 활용 BI공모전
Golden time을 잡아라



집이나 가게에 태양광발전 설비를 갖추는 곳이 늘어나고 있죠.

1. 제안 배경

공공데이터 활용 BI공모전
Golden time을 잡아라

	분전반 및 접속반	배선 (접속단자)	인버터	모듈 및 어레이	기타	합계
2016년	36	12	14	11	0	73
2017년	44	26	10	6	0	86
2018년	67	35	28	13	2	145
합계	147	73	52	30	2	304
비율	48%	24%	17%	10%	1%	

태양광 발전 설비 화재의 발화지점은 분전반 및 접속함, 접속반 내부의 스파크 발생이 화재 원인으로 추정

태양광 발전소 증가는 화재사고 증가로 이어져 태양광 설비에 대한 화재의 관리 지표 개발이 필요.

분석 내용 및 방법

- 분석 목표
- 가설
- 데이터 셋 훈련
- 분석모델 생성

2. 분석 내용 및 방법

공공데이터 활용 BI공모전
Golden time을 잡아라

- 분석 목표
- 접속반 내의 단자대 온도를 활용한 건전성 판단 지표 개발
- 분당 접속반 데이터건수가 400만 건이 넘어 분석에 필요한 대표 접속반 선정을 위한 기술통계
- 접속반 온도에 영향을 미치는 요소 상관 분석
- 현재, 5분 전, 10분 전 데이터로 10분, 20분, 30분, 40분, 50분 후 접속반 온도 예측모델을 구현
- 40분 후 예측 접속반 온도에 대하여 건전성 판단 지표 개발

2. 분석 내용 및 방법

공공데이터 활용 BI공모전
Golden time을 잡아라

- 분석 환경
- 환경 : MS-Azure
- 분석도구 : Python(numpy, pandas, sklearn, matplotlib) , Excel
- 분석 데이터

Set	설명	건수	수집주기	기간
th_connectbox_day	시간당 접속반	92,044	1분	일자 : 21/01/01 ~ 22/03/30 시간 : 5:00~21:00
th_connectbox	분당 접속반	4,227,340	1분	
th_digitalmeter	분당 차단기	1,300,970	1분	
th_digitalmeter_day	시간당 차단기	28,766	1시간	
th_event	이벤트	5,463	발생시	
th_inverter	분당 인버터	878,236	1분	
th_inverter_day	시간당 인버터	15,133	1시간	
th_temp	분당 변압기	845,468	1분	
th_temp_day	시간당 변압기	15,133	1시간	
th_weather	분당 기상	422,734	1분	
th_weather_day	시간당 기상	15,133	1시간	

2. 분석 내용 및 방법

공공데이터 활용 BI공모전
Golden time을 잡아라

- 접속반 선정



- 접속반 건전성 평가에 연관된 데이터
- 분당 접속반 데이터 중 대표 접속반 데이터와 동일한 주기로 수집되는 기상데이터를 활용
- 설비 이벤트를 활용하려 하였으나 **2022년** 데이터만 제공되어 활용 불가

2. 분석 내용 및 방법

공공데이터 활용 BI공모전
Golden time을 잡아라

- 데이터 셋 생성
- 분석 데이터 셋 생성을 위한 데이터 클리징
- 접속반 생성일자와 접속반 번호 기준으로 **2021년** 데이터의 중복을 제거
- 그 후 접속반 온도 기술통계를 통하여 대표 접속반 ***CB15**를 선택

* CB(Connect Box) : 접속반

	CB8	CB9	CB10	CB11	CB12	CB13	CB14	CB15	CB16	CB17	전체평균
건수	422,734	422,734	422,734	422,734	422,734	422,734	422,734	422,734	422,734	422,734	422,734
평균	20.34	20.52	19.90	20.29	19.50	20.25	20.13	20.33	20.41	20.70	20.24
표준 편차	12.52	12.84	12.59	12.45	11.92	12.23	12.12	12.38	12.34	12.45	12.38
최소	-8.80	-8.30	-7.90	-7.20	-8.30	-7.60	-7.90	-7.80	-7.60	-8.00	-7.94
25%	10.60	10.50	10.00	10.50	10.20	10.60	10.60	10.60	10.70	10.90	10.52
50%	20.00	19.70	18.70	19.60	19.00	19.70	19.60	19.90	20.00	20.30	19.65
75%	28.80	29.20	28.40	28.70	27.80	28.60	28.40	28.80	28.90	29.30	28.69
최대	52.70	54.40	52.70	52.50	48.30	51.50	51.10	52.10	52.10	52.60	52.00

2. 분석 내용 및 방법

공공데이터 활용 BI공모전
Golden time을 잡아라

- 데이터 셋 생성
- 접속반 데이터와 기상 데이터를 생성일시를 기준으로 융합
- 데이터 셋에 존재하는 각 데이터의 변수 간 상관계수 산출
- 첨부한 표를 살펴보면 접속반 온도와 외기온도, 발전전류 등이 0.5 이상의 상관관계를 보임

융합기준	접속반데이터			기상데이터			
생성일시	단자대온도	발전전압	발전전류	경사일사량	수평일사량	모듈온도	외기온도

2. 분석 내용 및 방법

공공데이터 활용 BI공모전
Golden time을 잡아라

표 6 실험 조건 및 결과

	표준 온도 (°C)	표준 전압 (kV)	표준 전류 (A)	표준 에너지 (MJ/m ²)	표준 에너지 (MJ/m ²)	표준 온도 (°C)	표준 온도 (°C)
표준 조건	1.000						
표준 조건	0.362	1.000					
표준 조건	0.601	0.432	1.000				
표준 조건	0.599	0.429	0.969	1.000			
표준 조건	0.555	0.449	0.967	0.993	1.000		
표준 조건	0.938	0.268	0.423	0.427	0.383	1.000	
표준 조건	0.935	0.412	0.750	0.753	0.727	0.901	1.000

2. 분석 내용 및 방법

공공데이터 활용 BI공모전
Golden time을 잡아라

- 가설
- 데이터 셋을 이용하여 설정한 시간 이후를 예측
- 접속반 온도의 설정시간 이후를 예측, 골든타임 내에 경고
- 분석 데이터 셋을 현재 기준 설정시간, 설정시간 5분전, 설정시간 10분전 외기온도, 접속반 온도, 발전전류를 독립변수로 하여 설정 시간 후의 접속반 온도(종속변수)를 예측

기준일시	설정시간			설정시간+5분전		설정시간+10분전	
예측 일시**	기상	접속반		접속반		접속반	
	외기온도	발전전류	단자대온도	발전전류	단자대온도	발전전류	단자대온도

** 예측일시 기준 설정시간 후의 접속반 온도 값을 Label로 활용

2. 분석 내용 및 방법

공공데이터 활용 BI공모전
Golden time을 잡아라

- 접속반 통계치 기반 건전성 판단 지표 생성

- 정상 상태인 경우 접속반 온도 범위를 10등분하여 10구간을 얻었고 각 구간의 크기를 6°C이다 .
- 실측 접속반 온도 최대치인 52.1°C에서 정상 동작한 바 이를 최소의 기준으로 삼고, 여기서 한 구간 씩(6°C) 증가시켜 주의, 경고, 위험 판단 지표를 설정한다.

단차대 온도(°C)	위치(%)	위치간 온도차(°C)	Signal
실측	-7.9	0%	정상
	-1.9	10%	
	4.1	20%	
	10.1	30%	
	16.1	40%	
	22.1	50%	
	28.1	60%	
	34.1	70%	
	40.1	80%	
	46.1	90%	
	52.1	100%	
예측	58.1	110%	주의
	64.1	120%	경고
	70.1	130%	위험

2. 분석 내용 및 방법

공공데이터 활용 BI공모전
Golden time을 잡아라

- 분석모델 생성을 위한 데이터 셋 생성과 훈련
- 분석 데이터셋의 경우 상관계수에 따라 분당 접촉반 상태와 함께 접촉반 온도와 관련이 큰 기상 데이터를 포함하였다.
- 분석 데이터 셋의 경우 예측 시간을 기준으로 생성되며 설정시간의 기상과 접촉반 상태, 설정시간의 5분 전의 접촉반과 기상, 10분 전의 접촉반과 기상으로 데이터셋을 생성하였다.
- `create_date`는 설정시간에 따라 시작시간이 변경된다.

	out_temp_p	in_temp_p	a_total_p	out_temp_5m	in_temp_5m	a_total_5m	out_temp_10m	in_temp_10m	a_total_10m	y_in_temp
create_date										
2021-01-01 05:50:00	-2.299999	0.0	0.0	-2.299999	0.0	0.0	-2.299999	0.0	0.0	-0.1
2021-01-01 05:51:00	-2.299999	0.0	0.0	-2.299999	0.0	0.0	-2.299999	0.0	0.0	-0.1
2021-01-01 05:52:00	-2.299999	-0.1	0.0	-2.200001	0.0	0.0	-2.400000	0.0	0.0	-0.2
2021-01-01 05:53:00	-2.299999	-0.1	0.0	-2.299999	0.0	0.0	-2.299999	0.0	0.0	-0.2
2021-01-01 05:54:00	-2.299999	-0.1	0.0	-2.299999	0.0	0.0	-2.299999	0.0	0.0	-0.2
...
2022-03-30 09:50:00	9.799999	10.8	8.8	10.100000	10.7	2.4	10.200001	10.5	7.6	11.8
2022-03-30 09:51:00	9.700001	10.9	9.2	10.000000	10.7	8.0	10.200001	10.6	3.2	11.8
2022-03-30 09:53:00	9.600000	11.0	8.3	9.900000	10.8	8.7	10.100000	10.6	6.1	11.8
2022-03-30 09:54:00	9.600000	11.0	7.6	9.799999	10.8	8.8	10.100000	10.7	0.8	11.8
2022-03-30 09:55:00	9.600000	11.0	7.6	9.799999	10.8	8.8	10.100000	10.7	2.4	11.9

2. 분석 내용 및 방법

공공데이터 활용 BI공모전
Golden time을 잡아라

- 학습 셋/ 테스트 셋 분리
- 분석모델(LinearRegression, Radomforest) 학습을 위한 독립변수(X)와 종속변수(y)를 생성하였다.
- 점속반 온도를 예측하기 위해 [표8]에서 사용된 변수로 구성된 데이터 셋을 학습 셋(X_train_in, y_train_in)과 테스트 셋(X_test_in, y_test_in)으로 분리해준다. (총 387,897건을 8:2 비율인 310,317건:77,580건으로 분리)

```
# Split data 80%-20% into training set and test set
X_train_in, X_test_in, y_train_in, y_test_in = train_test_split(X, y, test_size=0.20, random_state=0)

print ('Training Set: %d rows\nTest Set: %d rows' % (X_train_in.shape[0], X_test_in.shape[0]))
```

```
Training Set: 310317 rows
Test Set: 77580 rows
```

- X값은 각 시간대의 실제 점속반 온도값, 외기온도, 전류를, y값에는 각 시간대에 예측한 점속반의 온도값이다.
- 이때 학습에서 사용된 각 독립변수는 "out_temp_p", "in_temp_p", "a_total_p", "out_temp_5m", "in_temp_5m", "a_total_5m", "out_temp_10m", "in_temp_10m", "a_total_10m"이다.

```
# Separate features and labels
X, y = df_15[['out_temp_p', 'in_temp_p', 'a_total_p', 'out_temp_5m', 'in_temp_5m', 'a_total_5m', 'out_temp_10m', 'in_temp_10m', 'a_total_10m']].values, df_15['y_in_temp'].values
print('Features:', X[1387897], '\nLabels:', y[1387897], sep='\n')
```

```
Features:
[[-2.2999992  0.          0.          ... -2.2999992  0.
  0.          ]
 [-2.2999992  0.          0.          ... -2.2999992  0.
  0.          ]
 [-2.2999992 -0.1        0.          ... -2.3999996  0.
  0.          ]
 ...
 [ 9.6000004 11.         8.3000002 ... 10.1        10.6
  6.0999999 ]
 [ 9.6000004 11.         7.5999999 ... 10.1        10.7
  0.8000001 ]
 [ 9.6000004 11.         7.5999999 ... 10.1        10.7
  2.4000001 ]]
```

```
Labels:
[-0.1 -0.1 -0.2 ... 11.8 11.8 11.9]
```

2. 분석 내용 및 방법

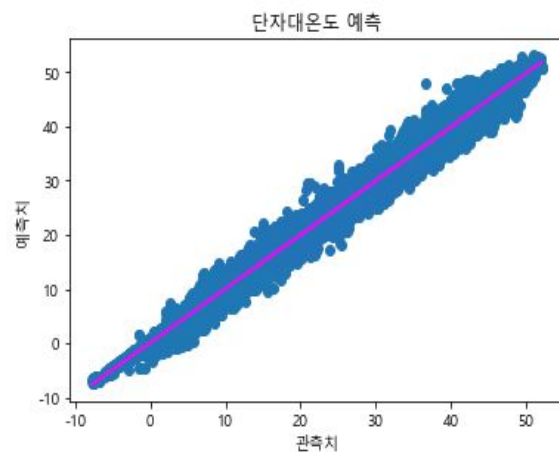
공공데이터 활용 BI공모전
Golden time을 잡아라

- 분석모델 생성
- Linear Regression 모델 학습하기
- Python의 sklearn Lib를 활용하여 Linear Regression모델을 학습시키고 모델 결과는 statsmodels Lib를 사용하여 해석하였다.
- 현재시간과 예측시간과의 간격을 40분으로 설정한 선형회귀 모델 생성

```
# Fit a linear regression model on the training set
model = LinearRegression().fit(X_train_in, y_train_in)
print(model)

LinearRegression()
```

```
MSE: 1.031097911226328
RMSE: 1.0154299144826924
R2: 0.9932441239859163
절편: 1.0233406125945343
```



OLS Regression Results						
Dep. Variable:	y	R-squared (uncentered):	0.998			
Model:	OLS	Adj. R-squared (uncentered):	0.998			
Method:	Least Squares	F-statistic:	1.707e+07			
Date:	Sat, 20 Aug 2022	Prob (F-statistic):	0.00			
Time:	01:16:07	Log-Likelihood:	-4.6602e+05			
No. Observations:	310317	AIC:	9.321e+05			
Df Residuals:	310308	BIC:	9.322e+05			
Df Model:	9					
Covariance Type:	nonrobust					
	coef	std err	t	P> t	[0.025	0.975]
x1	0.2978	0.003	107.842	0.000	0.292	0.303
x2	4.3980	0.019	234.581	0.000	4.361	4.435
x3	0.0272	0.000	152.579	0.000	0.027	0.028
x4	-0.1574	0.004	-37.027	0.000	-0.166	-0.149
x5	-1.4881	0.035	-42.498	0.000	-1.557	-1.420
x6	-0.0126	0.000	-63.574	0.000	-0.013	-0.012
x7	-0.0881	0.002	-35.392	0.000	-0.093	-0.083
x8	-1.9565	0.018	-110.733	0.000	-1.991	-1.922
x9	-0.0183	0.000	-93.007	0.000	-0.019	-0.018
Omnibus:	35634.979	Durbin-Watson:	1.967			
Prob(Omnibus):	0.000	Jarque-Bera (JB):	310541.789			
Skew:	-0.213	Prob(JB):	0.00			
Kurtosis:	7.882	Cond. No.	2.66e+03			

- R^2 는 모델의 설명력을 나타내는 지표인 결정계수로 1에 매우 가까운 0.99로 모델의 성능이 높다고 생각할 수 있다.
- coef(회귀계수)는 X가 한 단위 증가할 때의 y 변화량으로 x1~x3까지는 X가 상승할 때 y가 같이 상승함을, x4~x9에서는 X가 상승할 때 y가 감소함을 알 수 있다. (각 x값들의 순서는 독립변수의 순서와 동일)

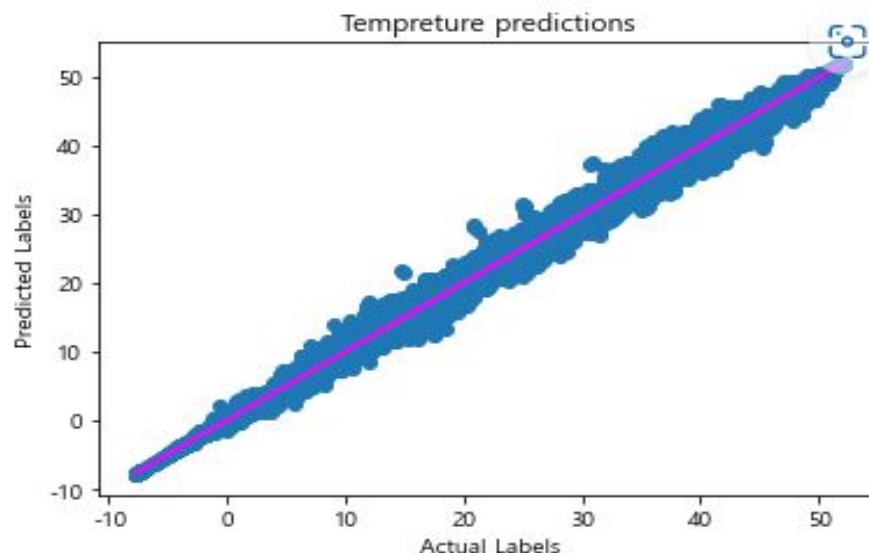
2. 분석 내용 및 방법

공공데이터 활용 BI공모전
Golden time을 잡아라

- 랜덤 포레스트 모델 학습하기
 - Python의 sklearn Lib.를 활용하여 앙상블 알고리즘 중 하나인 Random forest 모델을 학습시켰다.
 - 학습 셋과 테스트 셋은 Linear Regression 모델 학습과 동일한 셋을 사용하였다.
 - 기본적으로 MSE, RMSE, R^2 모두 선형 회귀 보다 우수한 값을 가졌다.

```
model_r = RandomForestRegressor().fit(X_train_in, y_train_in)
print(model_r, "\n")
RandomForestRegressor()
```

```
MSE: 0.3005734288798261
RMSE: 0.5482457741559219
R2: 0.9980306071843119
```



분석 결과

- 분석 모델 평가
- 분석 모델 선택

3. 분석 결과

공공데이터 활용 BI공모전
Golden time을 잡아라

- 분석모델 평가
- 회귀분석 결과값

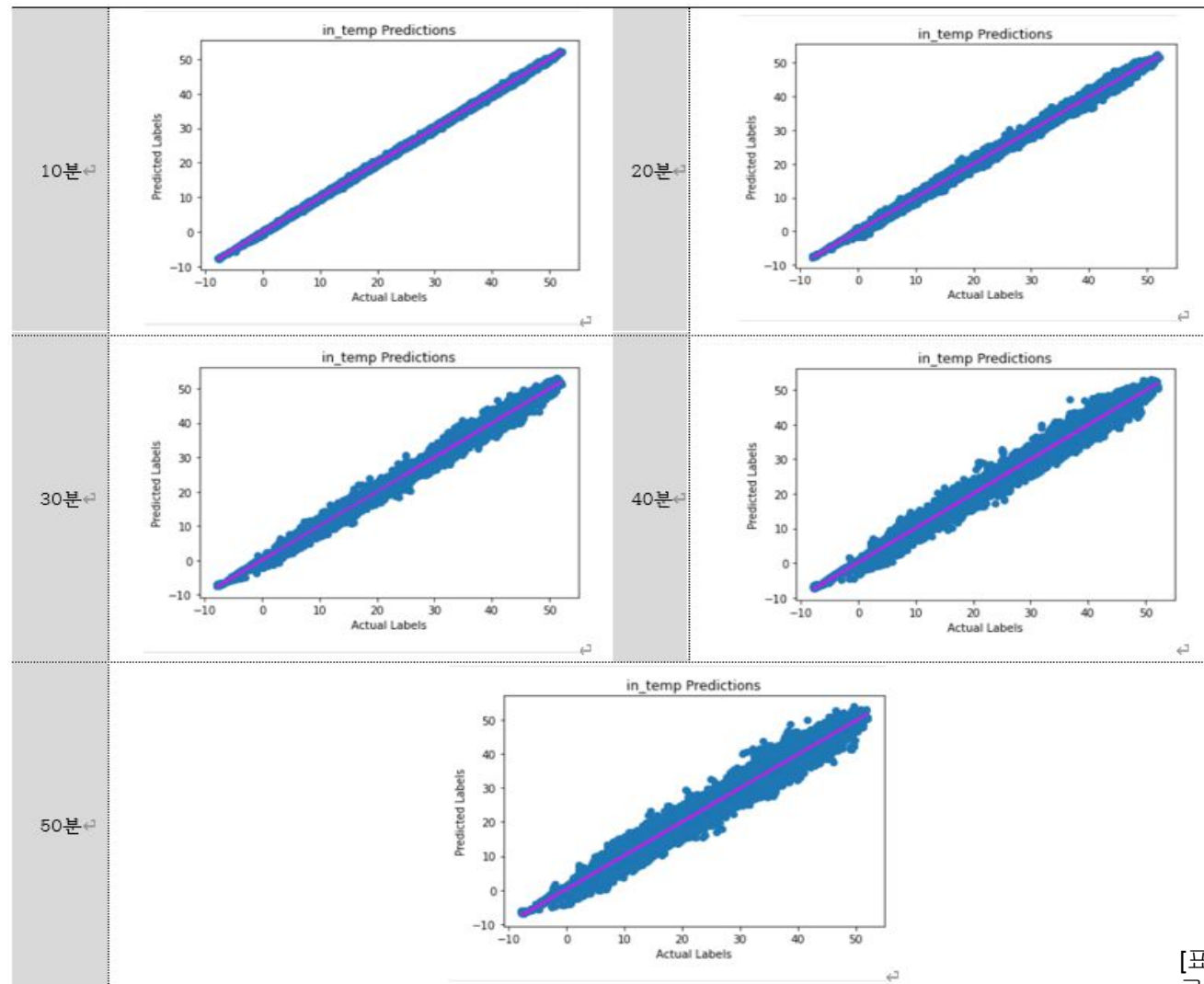
설정시간		10분	20분	30분	40분	50분
오차율	MSE	0.034697	0.19211	0.523333	1.048125	1.764183
	RMSE	0.186271	0.438303	0.723418	1.023779	1.328225
설명력		0.999771	0.998738	0.996579	0.993132	0.9884

[표 10] 설정시간(iTime)에 따른 회귀분석 모델 검증 결과값

- [표 10]에 나타난 바와 같이, 설정시간(iTime)을 10분에서 50분으로 증가시킬 수록 오차지표인 MSE와 RMSE 값은 증가하고, R^2 값은 낮아졌다. 이는 가까운 미래(10분)를 예측하는 모형이 먼 미래(50분)를 예측하는 모형보다 성능이 우수함을 의미한다.
- 그러나 설비 이상 감지 목적을 고려하면 이상 상황을 예측한 후에 사고를 방지할 대응시간이 필요할 것이므로, R^2 값이 0.99보다 크면서 온도 예측 평균 오차는 1°C정도에 불과한 40분을 설정시간(iTime)으로 선택하였다.
- 즉, [주의], [경고], [위험] 알람에 대처할 수 있는 골든 타임을 40분(iTime)으로 설정하였다. (표 11 참조)

3. 분석 결과

공공데이터 활용 BI공모전
Golden time을 잡아라



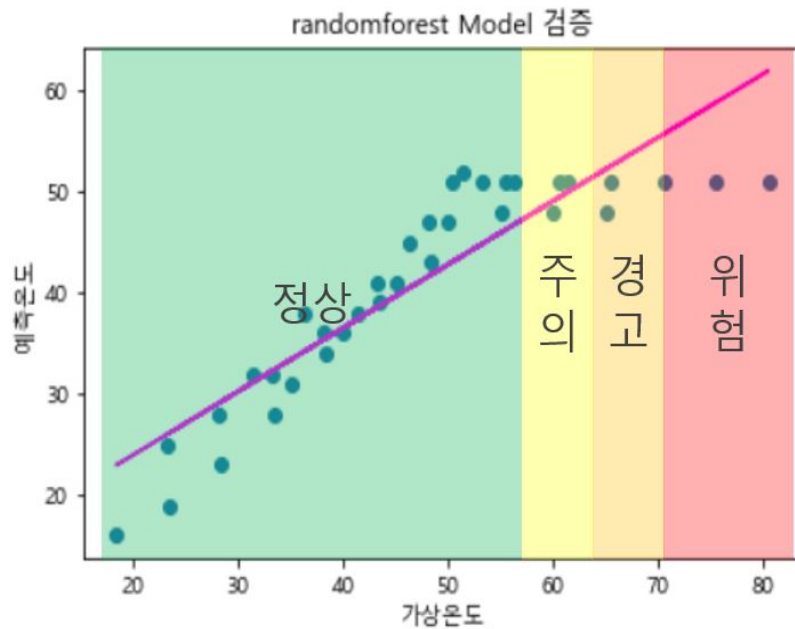
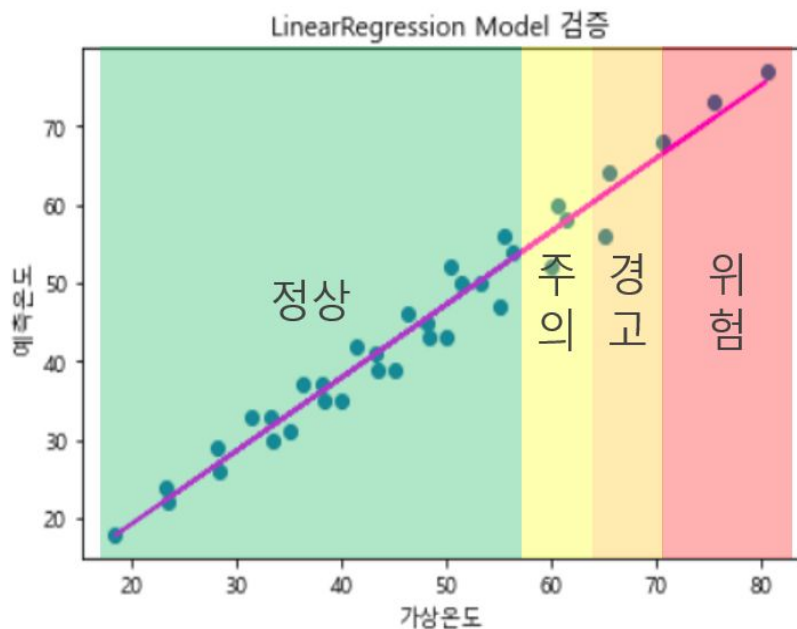
[표 11] 설정시간(iTime)에 따른 예측치와 관측치 비교
그래프

3. 분석 결과

공공데이터 활용 BI공모전
Golden time을 잡아라

• 분석 모델 선택

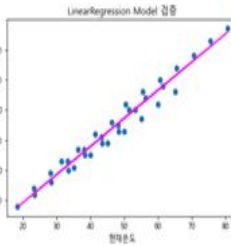
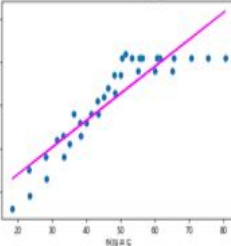
- 테스트셋에 포함되지 않은 가상의 새로운 테스트셋을 만들어 Linear Regression 모델과 Random Forest 모델의 예측 성능을 테스트하였다.
- 이를 위해 기존 테스트셋에서 데이터 5건을 랜덤 추출한 후, 외기온도와 발전전류는 고정하고 초기 접속반 온도만 5°C부터 40°C까지 5°C씩 증가시켜 예외적인 테스트 셋을 생성한 후 각 모델의 예측치를 비교하였다.



3. 분석 결과

공공데이터 활용 BI공모전
Golden time을 잡아라

- 검증 그래프를 보면 데이터의 최댓값이 되는 52.1°C 가 넘으면 **Random Forest**가 예측하지 못하는 것을 볼 수 있다.
- 이는 학습할 때 포함된 y값들의 범위 내에서만 예측치를 생성할 수 있는 **Random Forest** 알고리즘의 특성(의사결정 트리)에 기인한다.
- 따라서 학습 셋의 y값 최댓값인 52.1°C 를 넘는 예외적인 상황이 나왔을 때를 대처할 수 없다.
- 반면 **Linear Regression** 모델은 y값이 52.1°C 를 넘는 범위에서도 잘 예측하고 있음을 볼 수 있다.
- 오차지표와 설명력 면에서 **Random Forest** 모델이 **Linear Regression** 모델보다 우수하였음에도 불구하고, 학습되지 않은 예외적인 상황을 예측할 경우에는 활용이 어렵다는 결론에 도달하여 예측을 위한 분석 모델로 **Linear Regression** 모델을 선택하였다.

Model	RMSE	MSE	R-Squared	검증
LinearRegression	1.0154	1.0311	0.9932	
RandomForest	0.5482	0.3006	0.9980	

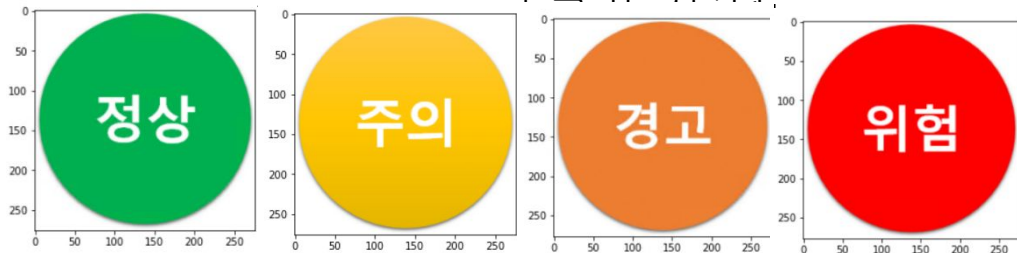
활용방안 및 한계점

4. 활용방안 및 한계점

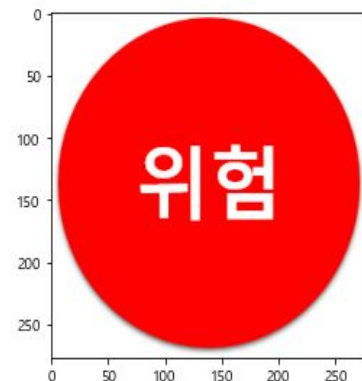
공공데이터 활용 BI공모전
Golden time을 잡아라

• 활용방안

- 알람이 발생하면 이에 대처할 수 있는 골든타임 **40분**을 확보하기 위하여 설정시간(iTime)을 40분으로 설정하여 활용하기를 제안한다.
- Linear Regression 모형을 활용하여 현재 및 5, 10분 전 외기온도, 발전전압, 접속반 온도를 입력 받아 40분 후의 접속반 온도를 예측한다.
- 이 예측된 온도를 [표9]에 정한 기준에 따라 정상/주의/경고/위험 상태로 구분하고, 각 상태에 적당한 색을 할당하여 40분 후의 상태를 시간전으로 파다한 수 있도록 구축한다.



```
현재 외기 온도(ex : 12.9) : 34
현재 단자대온도(ex : 23.1) : 55
현재 발전 전류(ex : 0.4) : 22
5분전 외기 온도(ex : 12.9) : 33
5분전 단자대온도(ex : 23.1) : 48
5분전 발전 전류(ex : 0.4) : 23
10분전 외기 온도(ex : 12.9) : 33
10분전 단자대온도(ex : 23.1) : 42
10분전 발전 전류(ex : 0.4) : 21
Input Test Data Set : [[34.0, 55.0, 22.0, 33.0, 48.0, 23.0, 33.0, 42.0, 21.0]]
LinearRegression Model 40분후 단자대 예측온도 : [81.40359783]
```



4. 활용방안 및 한계점

공공데이터 활용 BI공모전
Golden time을 잡아라

- 한계점

- 야간 데이터 부재 : 사고나 화재는 시간을 구분하지 않는데 제공 데이터는 태양광 발전시간인 05:00~21:00까지만 제공되어 21:00~05:00까지의 사각지대가 존재하였다.
- 이상 상태 부재 : 건전성을 평가하기 위해서는 이상 사례와 정상 사례가 모두 있어야 판단기준을 설정할 수 있으나, 제공받은 데이터에는 정상적인 운영 데이터만 존재하여 이상치 기준을 설정하는 데 있어 어려움이 있었다.
- 사전지식 부족 : 태양광 발전에 대한 사전지식이 부족하여 대용량의 데이터 (5GB) 가공, 데이터 선택부터 난관이 있었다. 사전에 과제나 데이터에 대한 설명회나 온라인 동영상 제공했으면 한다.

감사합니다.

이원
이유송
임혁진