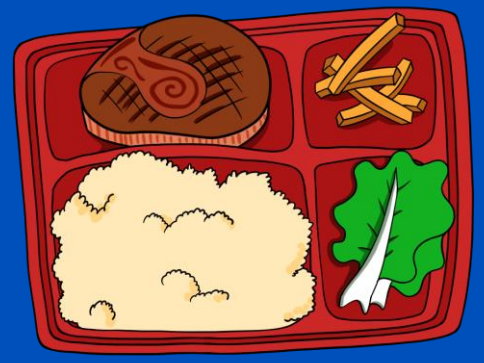
구내식당 식수 인원 예측 AI 경진대회

-김정진, 김하영, 최소연 (학식의 신)

목차

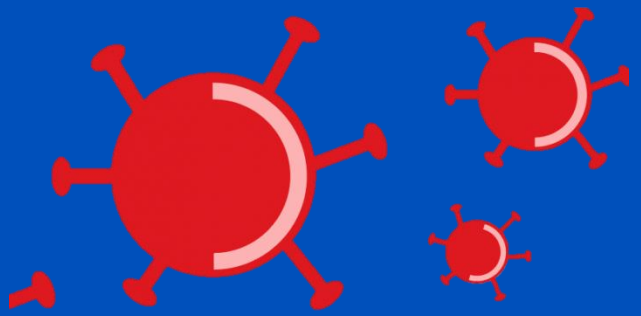


- 문제 제기 및 대회의 필요성 소개
- 대회 및 데이터 소개
- 간략한 EDA
- 데이터 분석에 사용한 프로그래밍 언어 및 모델링 소개
- 시도 A: 주어진 데이터 일부 변형 + ‘공휴일 전후 ' 도입
- 시도 B: ‘자기계발의 날’ 도입
- 시도 C: 코로나 외부데이터 사용 + 일부 변수 수학적 변형
- 최종 모델링 결과 발표
- 본 문제에 대한 해결책 및 모델링 성능 향상을 위한 제언 제시

문제 제기 및 대회의 필요성 소개

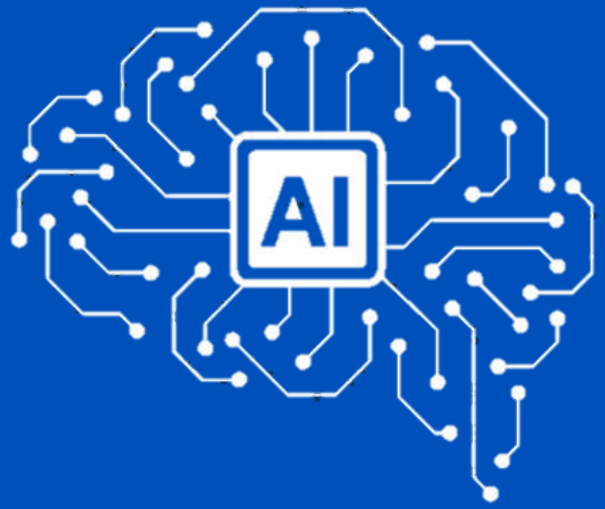
국민 1인당 연간
음식물 쓰레기
배출량 = 134kg
=> 이로 인해
1인당 222kg의
온실가스 배출

전 세계 온실가스
배출의 10%를 차지하는
음식물 쓰레기 ->
기후변화 부추김



코로나-19로 재택근무,
원격수업 등이 증가하며
잔반량이 증가했을
가능성 존재

문제 제기 및 대회의 필요성 소개



인공지능을 이용해
음식물 쓰레기를 줄이는
시스템이 개발됨
향후 구글에서 프로젝트
델타를 통해 식량
낭비와 음식물 쓰레기
감소 방법을 연구 중

=======〉

인공지능을 활용한
정확한 식량 수요
예측으로 잔반 감소가
가능하고, 그것이
필요한 사회가 도래함

대회 평가 방식 및 데이터 칼럼 소개

평가 방식: MAE (Mean Absolute Error): |예측값 – 실제값| 의 평균으로, MSE(Mean Square Error)보다 이상치에 덜 예민하여 많은 분야에서 선호됨.

대회 평가 방식 및 데이터 칼럼 소개

train 데이터 칼럼 소개:

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1205 entries, 0 to 1204
Data columns (total 12 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   일자              1205 non-null   object
 1   요일              1205 non-null   object
 2   본사정원수           1205 non-null   int64
 3   본사휴가자수          1205 non-null   int64
 4   본사출장자수          1205 non-null   int64
 5   본사시간외근무명령서승인건수  1205 non-null   int64
 6   현본사소속재택근무자수     1205 non-null   float64
 7   조식메뉴            1205 non-null   object
 8   중식메뉴            1205 non-null   object
 9   석식메뉴            1205 non-null   object
 10  중식계             1205 non-null   float64
 11  석식계             1205 non-null   float64
dtypes: float64(3), int64(4), object(5)
memory usage: 89.5+ KB
```

1205개의 데이터 & 12개의 컬럼

test 데이터 칼럼 소개:

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 50 entries, 0 to 49
Data columns (total 10 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   일자              50 non-null     object
 1   요일              50 non-null     object
 2   본사정원수           50 non-null     int64
 3   본사휴가자수          50 non-null     int64
 4   본사출장자수          50 non-null     int64
 5   본사시간외근무명령서승인건수  50 non-null     int64
 6   현본사소속재택근무자수     50 non-null     float64
 7   조식메뉴            50 non-null     object
 8   중식메뉴            50 non-null     object
 9   석식메뉴            50 non-null     object
dtypes: float64(1), int64(4), object(5)
memory usage: 4.0+ KB
```

50개의 데이터 & 10개의 칼럼

EDA - 도입

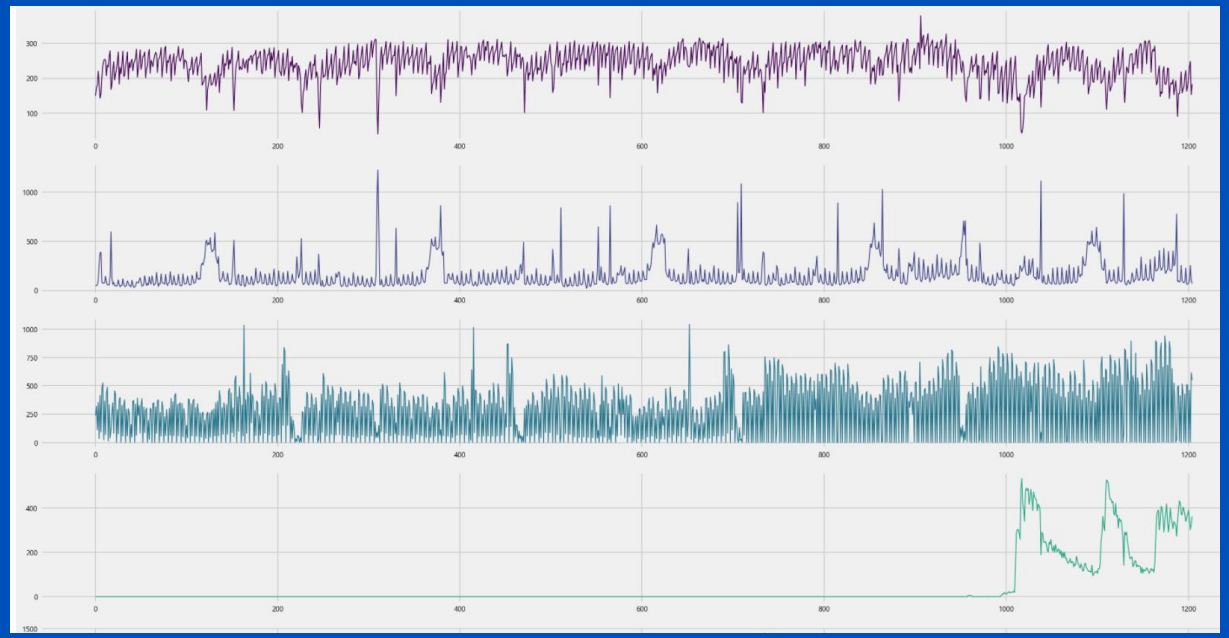
1) 변수 추가 전 주로 사용한 변수 – ‘월, ‘주’, ‘요일‘, ‘출근’, ‘본사휴가자수’, ‘본사출장자수‘, ‘야근비율’, ‘휴가비율’

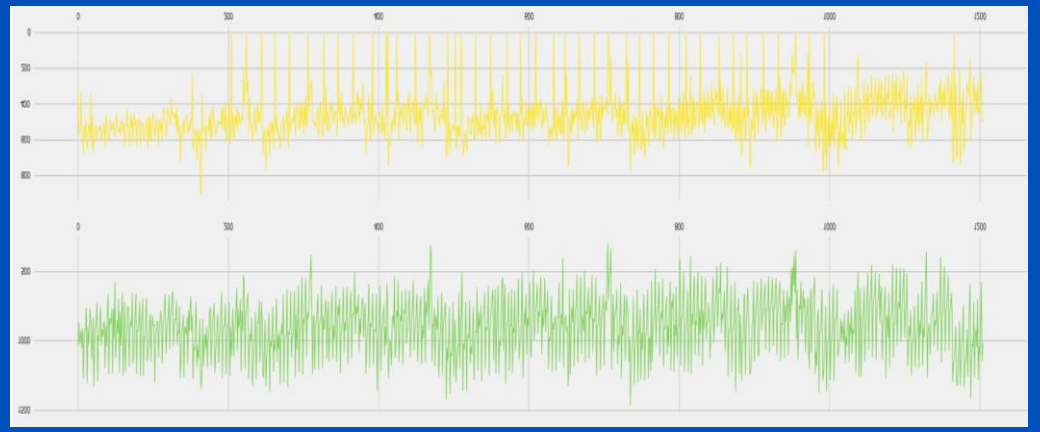
	중식계	석식계
요일	-0.734273	-0.313240
본사정원수	-0.115529	-0.173852
본사휴가자수	-0.391975	-0.316894
본사출장자수	-0.512680	-0.188164
본사시간외근무명령서승인건수	0.535611	0.571168
현본사소속재택근무자수	0.076509	-0.057534
중식계	1.000000	0.508287
석식계	0.508287	1.000000
년	-0.078804	-0.194792
월	-0.154664	-0.127142
일	-0.097392	-0.185565
주	-0.135008	-0.117561
출근	0.286810	0.172373
휴가비율	-0.388266	-0.308355
출장비율	-0.442041	-0.119128
야근비율	0.535956	0.572467
재택비율	0.076757	-0.056949

: 절대적인 수를 나타내는 본사시간외근무명령서승인건수, 본사출장자수, 본사휴가자수, 현본사소속재택근무자수 등을 비율로 변환하여 예측력을 향상시키고자 함. 특히 코로나-19 이전에 0건에 수렴하던 재택근무가 코로나-19 이후 크게 늘어난 것의 영향력을 ‘재택비율’로 치환함

EDA - 도입

1) 변수 추가 전 주로 사용한 변수 – ‘월, ‘주’, ‘요일‘, ‘출근’, ‘본사휴가자수’, ‘본사출장자수‘, ‘야근비율’, ‘휴가비율’

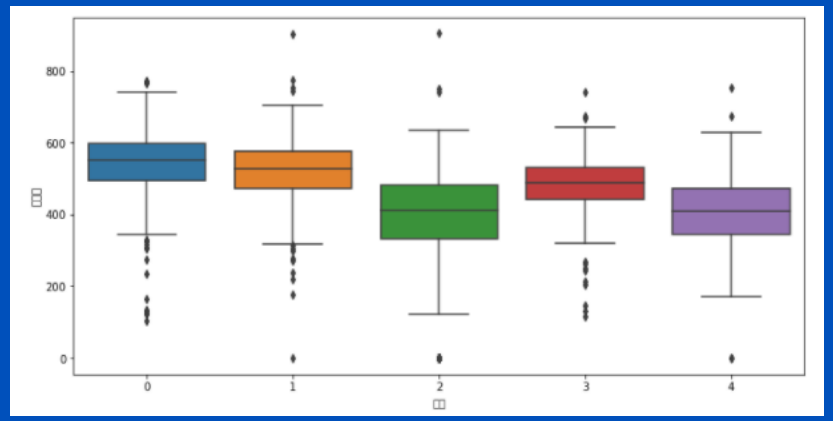


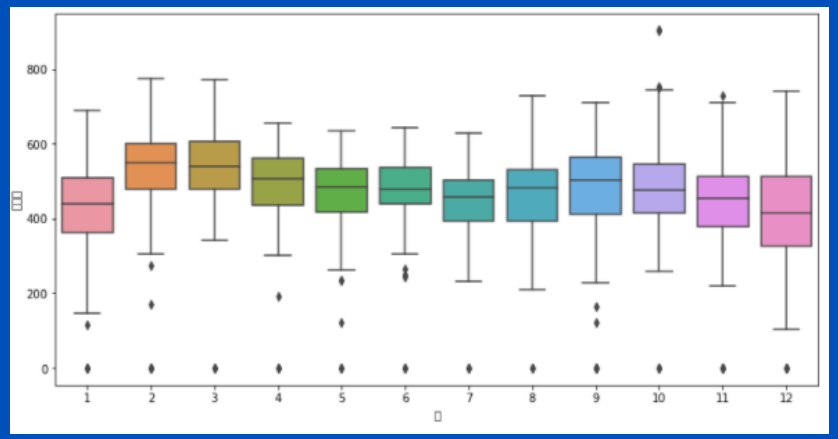


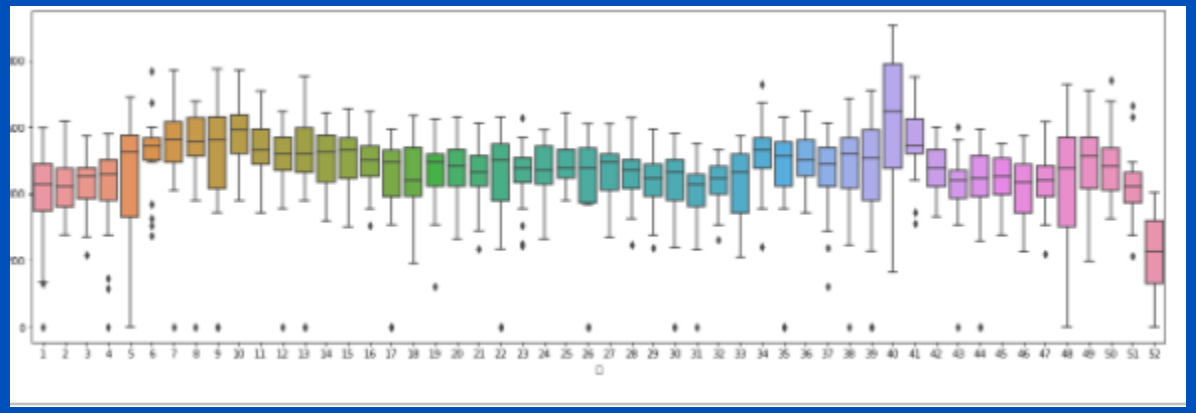
: 그러면서도 동시에 절대적인 숫자의 증감을 시계열로 분석한 결과, 데이터의 시간의 흐름에 따른 양상에 패턴이 있다고 판단하여 원 변수도 병행하여 사용함 (순서대로 "본사출장자수", "본사휴가자수", '본사시간외근무명령서승인건수', '현본사소속재택근무자수’ / '중식계','석식계’)

EDA - 도입

1) 변수 추가 전 주로 사용한 변수 – '월, '주', '요일', '출근', '본사휴가자수', '본사출장자수', '야근비율', '휴가비율'







: 한편, 요일과 주, 월 등의 시간적 변수에 따라 중식계와 석식계에 일정한 영향력을 행사하고 있는 것으로 분석하여 데이터 분석에 이와 같은 컬럼들을 사용함

대회에 사용한 프로그래밍 언어 및 모델링 소개

프로그래밍 언어: Python 3.7

사용한 모델링: 파이썬 pycaret으로 총 18개의 모델의 MAE를 한 번에 비교

(compare_model() 사용)

-> 정확도, AUC 등 다양한 기준으로 모델별 성능 평가 가능

대부분의 상황에서 가장 우수한 성능을 기록했던 catboost:

기존의 부스팅 방식과 유사하지만, catboost는 기존의 부스팅 모델이 일괄적으로 모든 훈련데이터에 잔차 계산을 한 것과 달리 일부에만 적용해 이를 모델로 만들고 이 모델로 예측한 값으로 그 뒤의 잔차 구함 -> 최적화되기 쉬움

대회에 사용한 프로그래밍 언어 및 모델링 소개

모델별 평가 후, 단일모델을 사용했을 때보다 복수의 모델을 앙상블하여 사용하는 것이 더 효율적인 것으로 드러나 앙상블 기법을 통해 모델을 혼합하여 최종 MAE를 도출함. 이때 도출을 위한 모델의 개수는 가장 작은 MAE를 기록한 모델 5개로 고정함.

```
best_5_d = compare_models(sort='MAE', n_select=5)
```

	Model	MAE	MSE	RMSE	R2	RMSLE
catboost	CatBoost Regressor	51.7430	6.005706e+03	76.7437	0.6856	0.8678
gbr	Gradient Boosting Regressor	54.0599	6.200097e+03	78.0725	0.6741	0.8741
et	Extra Trees Regressor	55.1447	6.933416e+03	82.2538	0.6368	0.8673
lightgbm	Light Gradient Boosting Machine	55.5024	6.478303e+03	80.0597	0.6557	0.8909
rf	Random Forest Regressor	55.7227	6.960151e+03	82.4774	0.6390	0.8908
huber	Huber Regressor	59.6062	7.779317e+03	87.5913	0.5891	0.7689

```
blended_d = blend_models(estimator_list = best_5_d, fold = 5, optimize = 'MAE')
pred_holdout = predict_model(blended_d)
final_model_d = finalize_model(blended_d)
pred_esb_d = predict_model(final_model_d, test_mer)
```

	MAE	MSE	RMSE	R2	RMSLE	MAPE
0	51.8571	5213.7837	72.2065	0.6949	0.7675	0.1163
1	54.8277	6940.2643	83.3082	0.7289	1.2014	0.1183
2	49.6369	5520.2121	74.2981	0.6560	0.6683	0.1059
3	47.3177	5189.7851	72.0402	0.6995	0.9541	0.0944
4	54.9595	6085.0141	78.0065	0.7085	0.9637	0.1159
Mean	51.7198	5789.8119	75.9719	0.6976	0.9110	0.1101
SD	2.9628	659.5771	4.2520	0.0238	0.1836	0.0090

	Model	MAE	MSE	RMSE	R2	RMSLE	MAPE
0	Voting Regressor	49.2925	5635.9881	75.0732	0.7046	0.8173	0.1122

시도 A: 주어진 변수를 단순하게 변형

1) 기존의 '본사시간외근무명령서승인건수', '본사휴가자수', '본사출장자수' 등을 '본사정원수'를 활용하여 '야근비율', '휴가비율', '출장비율' 등으로 변경,

```
train['출근'] = train['본사정원수']-(train['본사휴가자수']+train['본사출장자수']+train['현본사소속재택근무자수'])
train['휴가비율'] = train['본사휴가자수']/train['본사정원수']
train['출장비율'] = train['본사출장자수']/train['본사정원수']
train['야근비율'] = train['본사시간외근무명령서승인건수']/train['출근']
train['재택비율'] = train['현본사소속재택근무자수']/train['본사정원수']
```

```
test['출근'] = test['본사정원수']-(test['본사휴가자수']+test['본사출장자수']+test['현본사소속재택근무자수'])
test['휴가비율'] = test['본사휴가자수']/test['본사정원수']
test['출장비율'] = test['본사출장자수']/test['본사정원수']
test['야근비율'] = test['본사시간외근무명령서승인건수']/test['출근']
test['재택비율'] = test['현본사소속재택근무자수']/test['본사정원수']
```

2) 출장자와 휴가자 등 본사에 부재한 사람들의 수를 빼 '식사가능자수', '출근' 등의 변수 추가

```
train['식사가능자수'] = train['본사정원수'] - train['본사휴가자수'] - train['현본사소속재택근무자수']
test['식사가능자수'] = test['본사정원수'] - test['본사휴가자수'] - test['현본사소속재택근무자수']
```

시도 A: 주어진 변수를 단순하게 변형

3) train과 test 데이터의 날짜를 조사하여 주말을 제외한 공휴일 전날과 다음날을 표시하는 'date' 데이터 만들어 기존의 데이터와 merge, '공휴일전', '공휴일후', '공휴일합' 변수 도입

	일자	공휴일전	공휴일후	공휴일합	개수	선호	계발
0	2016-02-01	0	0	0	4.0	1.0	0
1	2016-02-02	0	0	0	4.0	1.0	0
2	2016-02-03	0	0	0	4.0	1.0	0
3	2016-02-04	0	0	0	3.0	1.0	0
4	2016-02-05	1	0	1	4.0	1.0	0

```
1  date.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1255 entries, 0 to 1254
Data columns (total 7 columns):
 #   Column  Non-Null Count  Dtype
---  ------  --------------  -----
 0   일자      1255 non-null   datetime64[ns]
 1   공휴일전    1255 non-null   int64
 2   공휴일후    1255 non-null   int64
 3   공휴일합    1255 non-null   int64
 4   개수      1205 non-null   float64
 5   선호      1205 non-null   float64
 6   계발      1255 non-null   int64
dtypes: datetime64[ns](1), float64(2), int64(4)
memory usage: 68.8 KB
```

시도 A: 주어진 변수를 단순하게 변형

결과: 중식계 예측 약 69.3584 (모델 비교 시 최소 MAE), 70.7853 (최소 MAE를 기록한 5개의 모델을 앙상블)

석식계 예측 약 62.6378, 61.5169

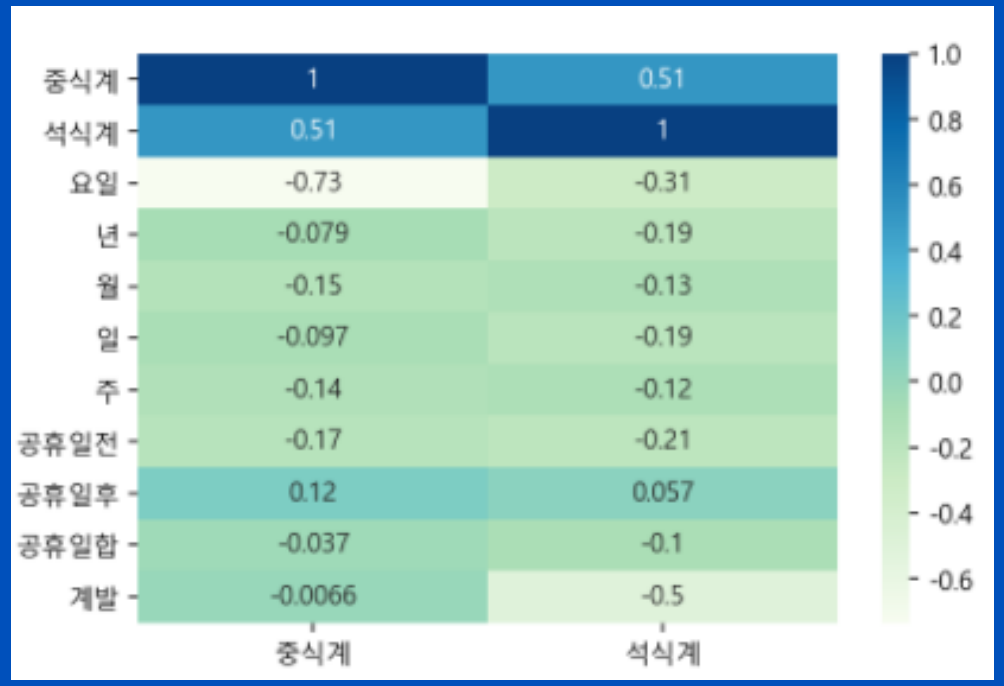
	Model	MAE	MSE	RMSE	R2	RMSLE	MAPE	TT (Sec)
catboost	CatBoost Regressor	69.3584	8.772097e+03	93.0287	0.7935	0.1188	0.0862	1.784
Mean		70.7853	9102.8977	95.2856	0.7873	0.1207	0.0879	

	Model	MAE	MSE	RMSE	R2	RMSLE	MAPE	TT (Sec)
catboost	CatBoost Regressor	62.6378	9433.6015	96.1890	0.4680	1.0337	0.1216	1.716
Mean		61.5169	9414.9311	96.4679	0.4716	1.0596	0.1162	

```
train_lunch = train_mer[['월', '주', '요일', '식사가능자수', '휴가비율', '출장비율', '재택비율',
'공휴일후','출근비율' ,'중식계' ]]
train_dinner = train_mer[['월', '주', '요일', '식사가능자수', '휴가비율', '출장비율', '재택비율',
'야근비율', '공휴일전', '석식계']]
```

AI도 B: '자기계발의 날' 도입

1) 매달 마지막 주 수요일을 '자기계발의 날 ' 로 선언, 직원들의 여가생활을 위해 조기퇴근을 장려하고 있음을 알게 되어 'date' 데이터의 '계발' 변수 도입



'계발' 변수가 석식에 특히 큰 영향을 주고 있는 것을 알 수 있음

시도 B: '자기계발의 날' 도입

결과: 중식계 예측 약 67.3379 (모델 비교 시 최소 MAE), 65.0302 (최소 MAE를 기록한 5개의 모델을 앙상블)

석식계 예측 약 55.0823, 55.1860 로 크게 개선됨

	Model	MAE	MSE	RMSE	R2	RMSLE	MAPE
catboost	CatBoost Regressor	67.3379	7.980525e+03	88.6466	8.185000e-01	0.1120	0.0832
Mean		65.0302	7839.5226	88.2099	0.8269	0.1141	0.0813

	Model	MAE	MSE	RMSE	R2	RMSLE	MAPE	TT (Sec)
catboost	CatBoost Regressor	55.0823	6.440361e+03	79.6626	0.6321	0.8228	0.1206	1.692
Mean		55.1860	6696.1588	81.5689	0.6347	0.8581	0.1207	

```
train_lunch = train_merge[['월', '주', '요일', '출근','휴가비율', '재택비율',
'공휴일후','공휴일합','출근비율' ,'중식계' ]]
train_dinner = train_mer[['월', '주', '요일', '식사가능자수', '휴가비율', '출장비율', '재택비율',
'야근비율', '공휴일전', '석식계' , '계발' ]]
```

시도 C: 코로나 확진자 관련 외부데이터 도입, 변수의 수학적 변형

1) 2020년 3월부터 심해진 코로나바이러스-19 상황을 반영하기 위해 공공데이터 포털에 있는 '보건복지부 코로나19 감염 현황 ' 을 사용함. 코로나-19 이후 재택근무가 급증하고 출장 수가 적어짐에 따라 중식과 석식계 인원의 변화가 있을 것으로 생각

2) 위의 외부데이터에서 다음과 같은 컬럼들을 사용

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 503 entries, 0 to 502
Data columns (total 17 columns):
 #   Column    Non-Null Count  Dtype
---  ------    --------------  -----
 0   누적확진률     503 non-null    float64
 1   누적검사      503 non-null    int64
 2   누적검사완료    503 non-null    int64
 3   치료중       502 non-null    float64
 4   격리해제      503 non-null    int64
 5   사망자       503 non-null    int64
 6   확진자       503 non-null    int64
 7   검사진행      503 non-null    int64
 8   음성        503 non-null    int64
 9   기준일       503 non-null    int64
 10  기준날짜      503 non-null    object
 11  년         503 non-null    int64
 12  월         503 non-null    int64
 13  일         503 non-null    int64
 14  전날대비확진자  503 non-null    int64
 15  확진자증감     503 non-null    int64
 16  확진s       503 non-null    float64
dtypes: float64(3), int64(13), object(1)
memory usage: 64.9+ KB
```

신규로 만든 데이터 컬럼

전날대비확진자: 당일 기준 전날 대비 확진자 증감수

확진자증감: 당일 확진자-전날확진자

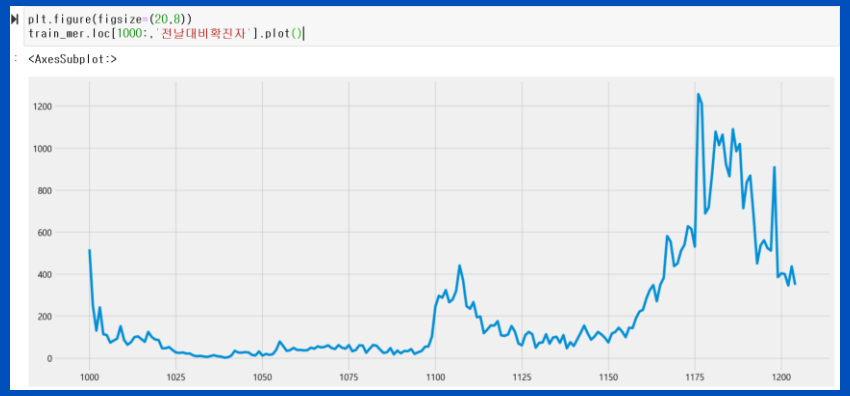
확진s: (당일확진자 – 그 전날 확진자)/전날 확진자*100

→ 매출액증가율의 공식을 응용

시도 C: 코로나 확진자 관련 외부데이터 도입, 변수의 수학적 변형

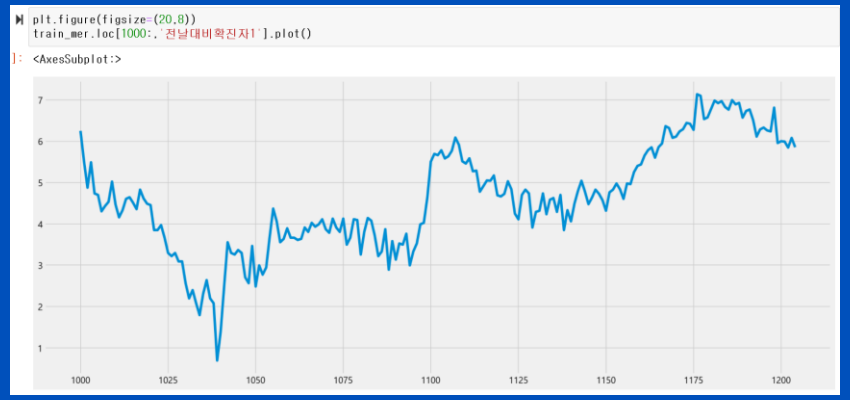
3) 일부 변수들을 로그변환, 지수변환하여 최종 변수 완성

```
plt.figure(figsize=(20,8))
train_mer.loc[1000:,'전날대비확진자'].plot()
```



▲ 전날대비확진자

```
plt.figure(figsize=(20,8))
train_mer.loc[1000:,'전날대비확진자1'].plot()
```



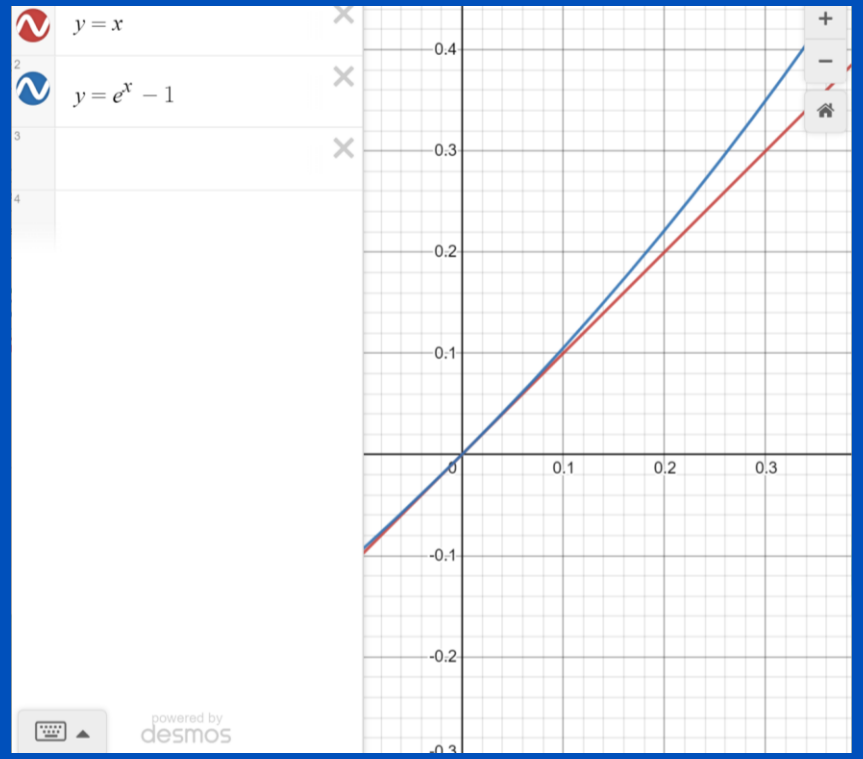
▲ 전날대비확진자1 (로그변환)

[전날대비확진자1]에서의 로그변환:
확진자가 누적임을 감안하여 시간이 지날수록 단위가 매우 커지고, 전염병이므로 증가하는 폭이 커짐을 반영하여 스케일 조정을 하기 위해 log변환을 시행

```
covid['전날대비확진자1'] = np.log(covid['전날대비확진자'])
```

시도 C: 코로나 확진자 관련 외부데이터 도입, 변수의 수학적 변형

3) 일부 변수들을 로그변환, 지수변환하여 최종 변수 완성



[재택비율1]에서의 지수변환:

비율은 0과 1사이의 값임.

그 사이에서 변환하지 않은 일반 그래프와 증가폭이 유사하면서 약간의 차이를 유도하는 점이 중식, 석식계 예측에 도움이 될 것이라고 생각함

```
train_mer['재택비율1'] =np.exp(train_mer['재택비율'])
```

최종 모델링 결과 발표

결과: 중식계 예측 65.6596 (앙상블 기준),
석식계 예측 50.5329 (앙상블 기준)

DACON Private 기준 : MAE 109. 36348을 기록함

〈중식〉

	MAE	MSE	RMSE	R2	RMSLE	MAPE
0	64.6522	7355.3979	85.7636	0.8400	0.1116	0.0825
1	71.2018	8301.4926	91.1125	0.8063	0.1057	0.0820
2	64.3310	7232.5565	85.0444	0.8312	0.1018	0.0756
3	60.5164	6110.0507	78.1668	0.8292	0.1044	0.0762
4	67.5964	8139.3798	90.2185	0.8260	0.1180	0.0854
Mean	65.6596	7427.7755	86.0612	0.8265	0.1083	0.0804
SD	3.5693	780.9622	4.6096	0.0111	0.0058	0.0038

〈석식〉

	MAE	MSE	RMSE	R2	RMSLE	MAPE
0	49.9025	5550.5483	74.5020	0.6597	0.6658	0.1086
1	50.2634	4887.0097	69.9072	0.7798	0.9543	0.1131
2	50.8692	5896.3622	76.7878	0.6607	0.6803	0.1188
3	52.2339	5267.3396	72.5764	0.7181	0.8769	0.1098
4	49.3956	5013.6843	70.8074	0.7873	1.1801	0.1004
Mean	50.5329	5322.9888	72.9161	0.7211	0.8715	0.1101
SD	0.9767	365.8918	2.4948	0.0552	0.1903	0.0060

기대효과와 문제 해결책 제시

자원을 효율적으로 사용하여
환경오염 개선 및 식수 절약에
이바지할 수 있음

구내식당의 시간적, 자원적
효율적 이용 가능

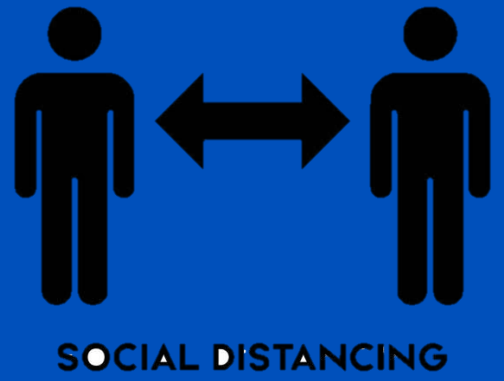
=======〉

이를 위해 회사에서 구내식당으로의
임직원들의 야근 및 출장 등의 특수한
근로형태를 최대한 빨리 공지하는 것이 필요

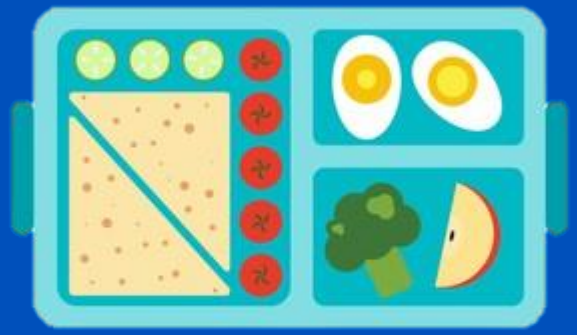
모델링 성능 향상을 위한 제언

중식, 석식 메뉴에 대한 구체적 분류 (ex. 찌개류, 구이류)
-> 직원들의 메뉴 선호도를 통한 더욱 정확한 예측 가능

직원들이 정상 출근을 했음에도 외부 식당에서 식사를 하는 경우를 파악하고자 회사에 주변 식당 간의 거리, 주 메뉴 및 식당 평점 등의 데이터를 고려할 필요 있음



본사가 위치하는 경상남도 진주시의 구체적인 확진자 증가 양상 및 그에 따른 사회적 거리두기 단계 변화 등의 측정이 필요해 보임



THANK YOU!