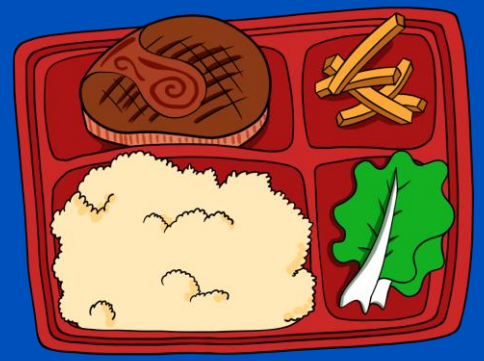
구내식당 식수 인원 예측 AI 경진대회

학식의 신

목차



- 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

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.

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

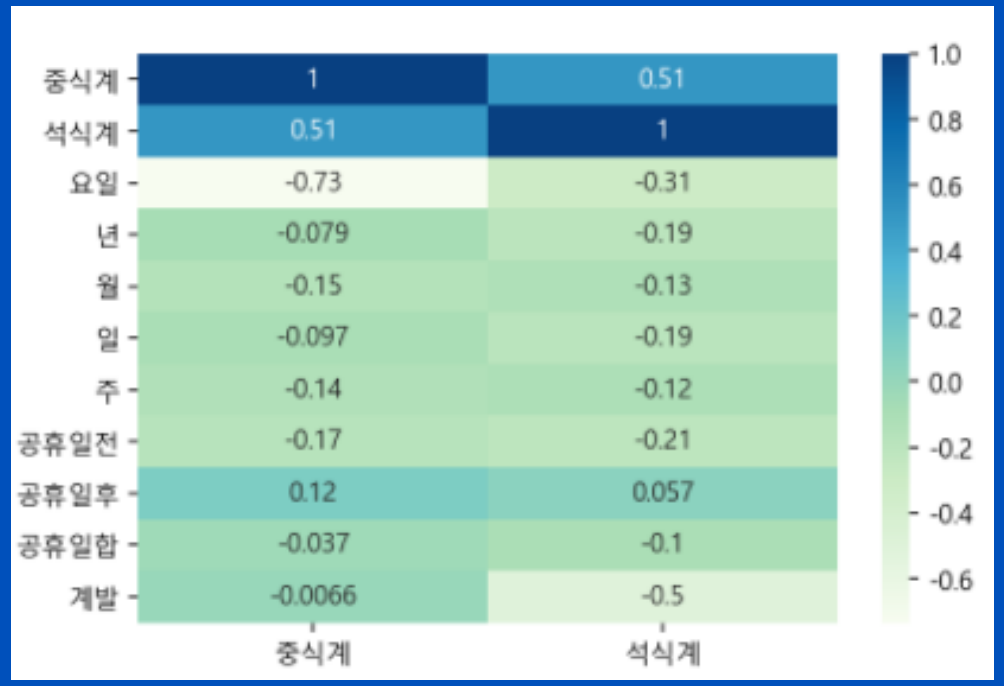
	일자	공휴일전	공휴일후	공휴일합	개수	선호	계발
0	2016-02-01	0	0	0	4.0	1.0	0
1	2016-02-02	0	0	0	4.0	1.0	0
2	2016-02-03	0	0	0	4.0	1.0	0
3	2016-02-04	0	0	0	3.0	1.0	0
4	2016-02-05	1	0	1	4.0	1.0	0

```
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
```

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 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

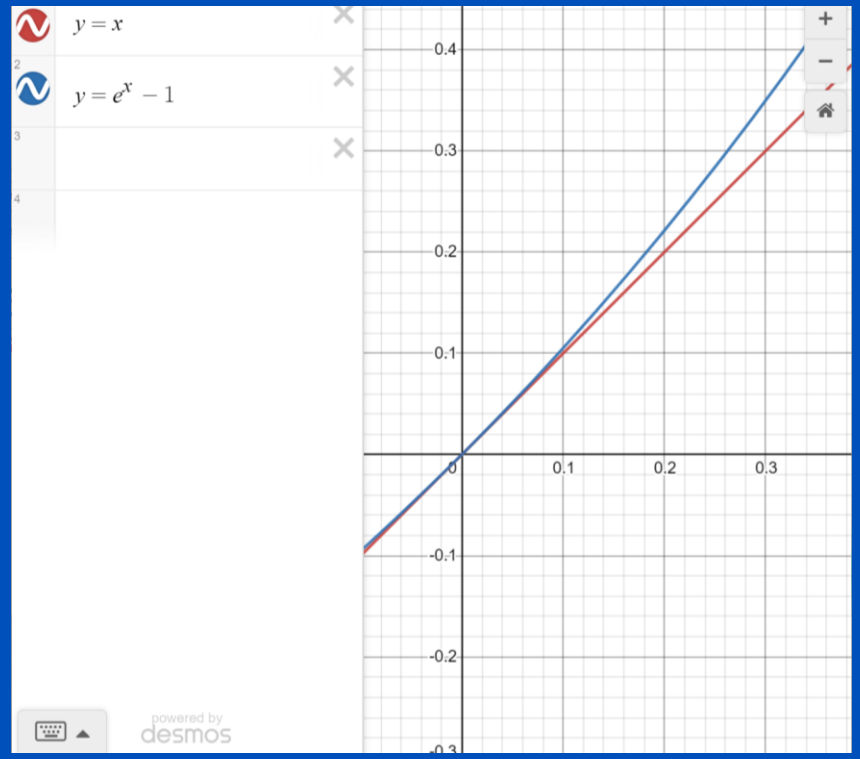
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 503 entries, 0 to 502
Data columns (total 17 columns):
 #   Column    Non-Null Count  Dtype
---  ------    --------------  -----
 0   누적확진률     503 non-null    float64
 1   누적검사      503 non-null    int64
 2   누적검사완료    503 non-null    int64
 3   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



지수변환의 이유: 변환하지 않은 일반 그래프와 증가폭이 유사하면서 약간의 차이를 유도하는 점이 중식, 석식계 예측에 도움이 될 것이라고 생각함

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

최종 모델링 결과 발표

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

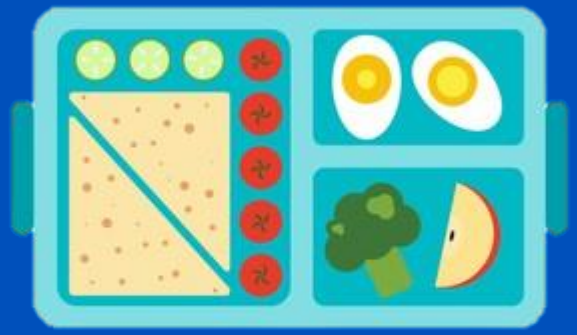
DACON Private 기준 : MAE 109. 36348을 기록함

〈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



THANK YOU!