TIME SERIES
PREDICTION COMPETITION

AI+X E조 PRESENTATION

휴먼지능정보공학전공 19 최가원
휴먼지능정보공학전공 20 박유진
컴퓨터과학전공 20 박은희
융합전자공학전공 20 윤은수
휴먼지능정보공학전공 18 이정곤

CONTENTS

팀원 소개

최가원

박유진

박은희

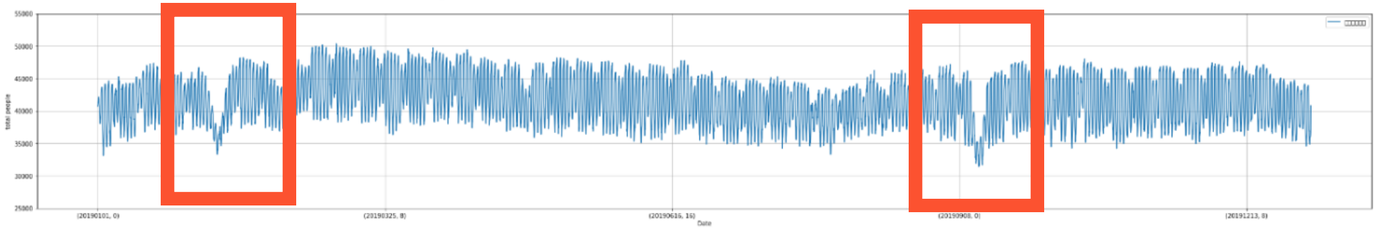
윤은수

이정곤

[접근법 및 결과] 최가원 팀원

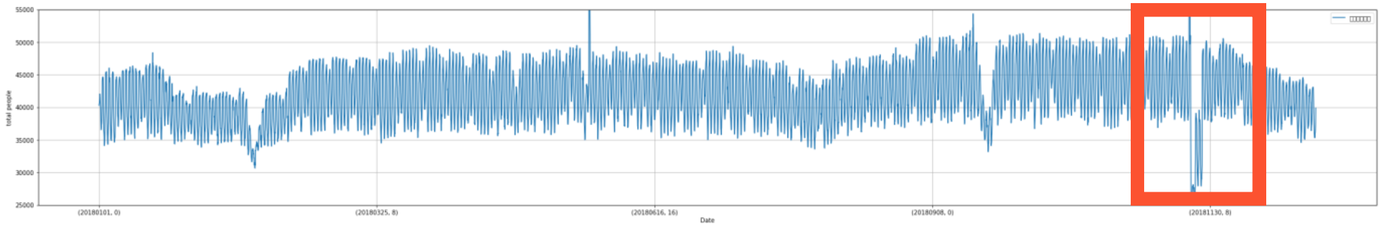
데이터 전처리

2019년도 timestemp별 총생활인구 그래프



- 모든 연도에서 1월말-2월초에 총생활인구수가 급락하는 곳 존재 -> **설 연휴 기간**
- 모든 연도에서 9월말-10월초에 총생활인구수가 급락하는 곳 존재 -> **추석 연휴 기간**

2018년도 timestemp별 총생활인구 그래프



2018년도 11월 말에 총생활인구수가 급등 후 급락하는 곳 존재
-> 원인 파악 불가, **이상치로 판단**

원인이 파악되는 규칙적인 변동점

설 연휴 | 추석 연휴

공휴일 당일 하루의 인구수 변동폭보다
연휴 기간의 인구수 변동폭이 더 의미있다고 판단

-> 설 연휴와 추석 연휴엔 holiday=1,
아닐 경우 holiday=0 로 feature 생성

이상치

2018년도 11월 말의 급등 후 급락

-> 원인 파악 불가

원인 불명의 이상치-> 학습에 악영향
따라서 **제거**하기로 결정
: 그 전주치의 총생활인구수를 가져와
이상치 주간의 총생활인구수에 덮어씀

[접근법 및 결과] 최가원 팀원

모델 학습

* features

	59d	3m	4m	5m	6m	12m	holiday
0	42835.2302	37381.2564	44253.0104	41619.5618	43364.9747	40541.4503	0
1	42944.8698	37302.1402	43955.5296	41532.3859	43302.3004	40542.5039	0
2	42934.3707	37207.5155	43881.8257	41600.4954	43179.2150	40817.7914	0
3	42886.7606	37260.9731	43836.3020	41601.8505	43358.0279	40934.7898	0
4	42747.8772	37465.9758	43836.5062	41762.3010	43449.2461	41087.2801	0

: 2 - 6 month SHIFT, 365day SHIFT, holiday(연휴기간) 여부

* model

```
1 # GradientBoosting
2 from sklearn.ensemble import GradientBoostingRegressor as GBR
3
4 gbr_reg = GBR()
5 gbr_reg.fit(train_x, train_y)
```

: Gradinent Boosting 모델

최종 SCORE

1026.68577

[접근법 및 결과] 박유진 팀원

데이터 전처리

6887	20191014	23
6888	20191028	0

2019.10.15. - 2019.10.27. 2주치 데이터가 비어 있음
-> 2021. 10. 15. - 2021.10.27. 데이터를 사용하여 결측치 처리

최종 SCORE

1458.99459

모델 학습

* features

	총생활인구수	59d	60d	61d	62d	63d	64d	65d	66d	67d	68d
0	43995.2837	45206.4166	45255.0588	43446.0870	43302.5443	44576.4163	45381.1852	45455.3554	45109.1336	45599.5255	43335.6902
1	44289.1658	45448.7924	45385.8499	43404.6941	43480.5513	44738.0849	45928.5120	45680.9697	45613.2005	45616.7501	43808.1562
2	44850.4454	45481.9503	45487.1695	43543.2808	43886.8839	44776.3651	46230.0548	45637.7142	44703.8624	45722.2274	43119.5238
3	44809.9306	45787.0853	45479.7469	43743.4724	43972.9278	44768.0974	46353.7221	46583.8557	44830.0410	45778.9722	43633.2521
4	44855.8697	46092.3177	45673.7311	43961.3010	44301.8534	44945.7866	46719.7912	46892.5482	45186.2041	45888.3440	43782.4048

: 59일 - 63일 전 총생활인구수 SHIFT

* model

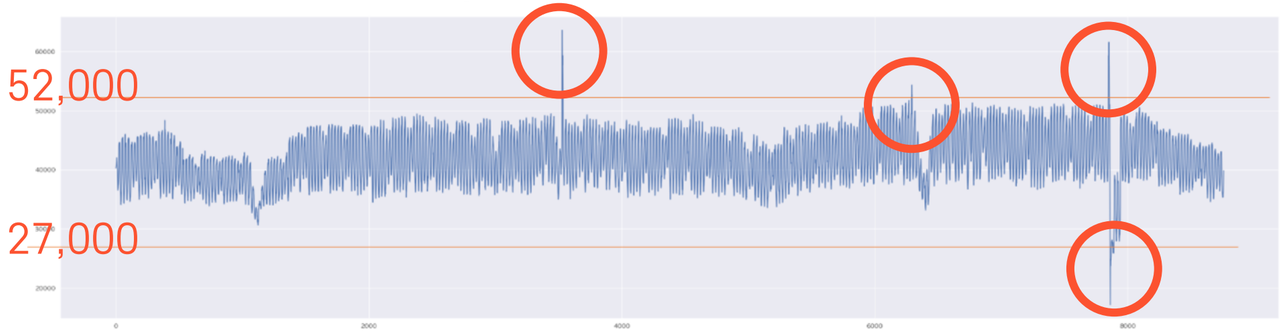
```
LR_reg = LinearRegression()
LR_reg.fit(train_x3, train_y3)

test_y2 = LR_reg.predict(test_x2)
```

: 선형 회귀 모델

[접근법 및 결과] 박은희 팀원

데이터 전처리

2018년도 timestemp별 총생활인구 그래프



2018년도의 데이터가 다른 연도에 비해 편차가 심하여, **주위값을 대입하여 편차 제거**

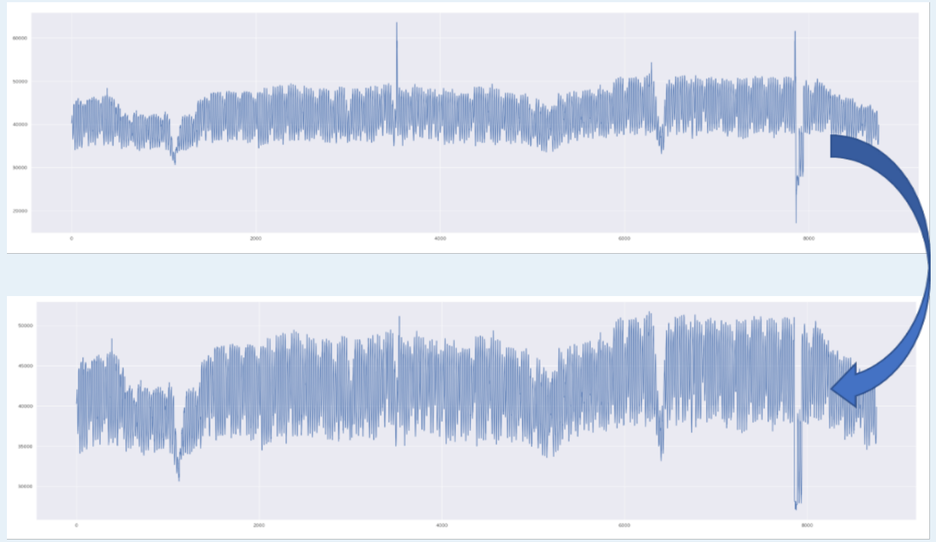
```
df18[df18['총생활인구수'] > 52000]
```

```
m = (df18.loc[3527,'총생활인구수'] + df18.loc[3535,'총생활인구수'])/2
for i in range(3528, 3535):
    df18.loc[i,'총생활인구수'] = m

mm= (df18.loc[7847,'총생활인구수']+df18.loc[7856,'총생활인구수'])/2
for i in range(7848, 7856):
    df18.loc[i,'총생활인구수'] = mm

df18.loc[6291,'총생활인구수'] = df18.loc[6290,'총생활인구수']
k= (df18.loc[6292,'총생활인구수'] + df18.loc[6296,'총생활인구수'])/2
for i in range(6293, 6296):
    df18.loc[i,'총생활인구수'] = k
```

```
df18[df18['총생활인구수'] < 27000]
```

```
mmm = (df18.loc[7861,'총생활인구수'] + df18.loc[7871,'총생활인구수'])/2
for i in range(7862, 7871):
    df18.loc[i,'총생활인구수'] = mmm

kk = (df18.loc[7882,'총생활인구수'] + df18.loc[7892,'총생활인구수'])/2
for i in range(7883, 7892):
    df18.loc[i,'총생활인구수'] = kk
```

전처리 전후



[접근법 및 결과] 박은희 팀원

모델 학습

* features

```
df_total['month'] = df_total['총생활인구수'].shift(1512)
```

구해야하는 예측일: 2022.01 - 2022.02월 (59일)
-> 이때 2022.12.31. 부터 59일전은 화요일,
2022.01.01. 부터 59일전은 토요일
토요일 - 화요일은 4일이기 때문에 (59+4)*24 = 1512
따라서 **1512만큼 shift** 해준 feature를 선택

* model

```
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error as mse
from sklearn.metrics import r2_score as r2
from math import sqrt

LR_reg = LinearRegression()
LR_reg.fit(train_x, train_y)
```

```
▾ LinearRegression
LinearRegression()
```

: 선형 회귀 모델

* test set

```
# 1월 31일 <-> 2월 11일
for i in range(30*24,33*24):
    k = i+11*24
    test_x.loc[i,'총생활인구수']= df21.loc[k,'총생활인구수']
    test_x.loc[k,'총생활인구수']=df21.loc[i,'총생활인구수']
```

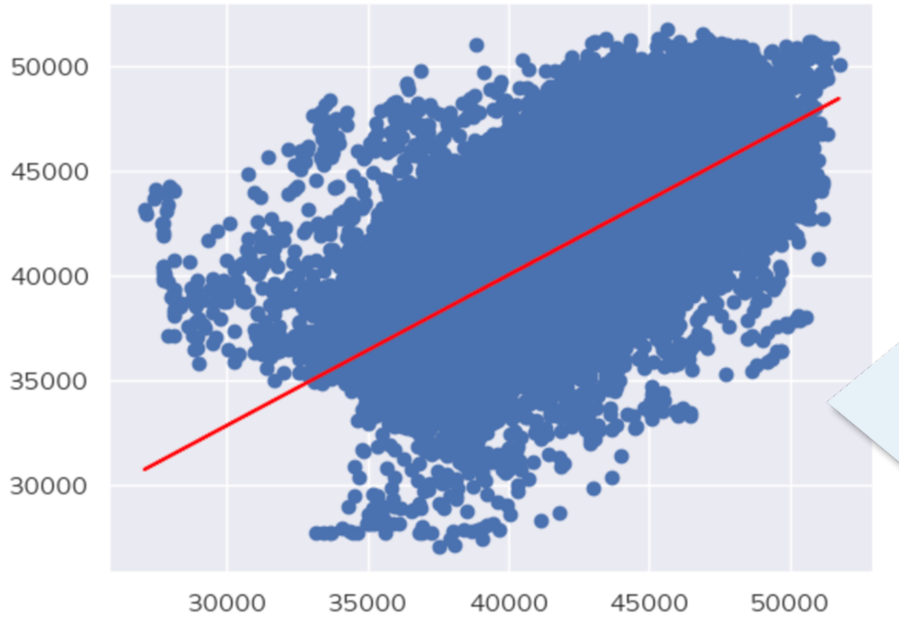
2021.01. - 2021.02에서 총생활인구수를 가져와,
설날의 총생활인구수를 조정하여 test_x로 사용

```
tt=[]
for i in range(0,1416):
    tt.append(i)
    y_pred_LR[i]-=700
```

연도별 그래프를 확인했을 시,
코로나의 여파로 평균적으로 300-1000명 가량 감소한 것을 확인
-> 약 700명 정도를 줄인 뒤 최종 test_x 완성

[접근법 및 결과] 박은희 팀원

모델 학습



R-squared

0.7172

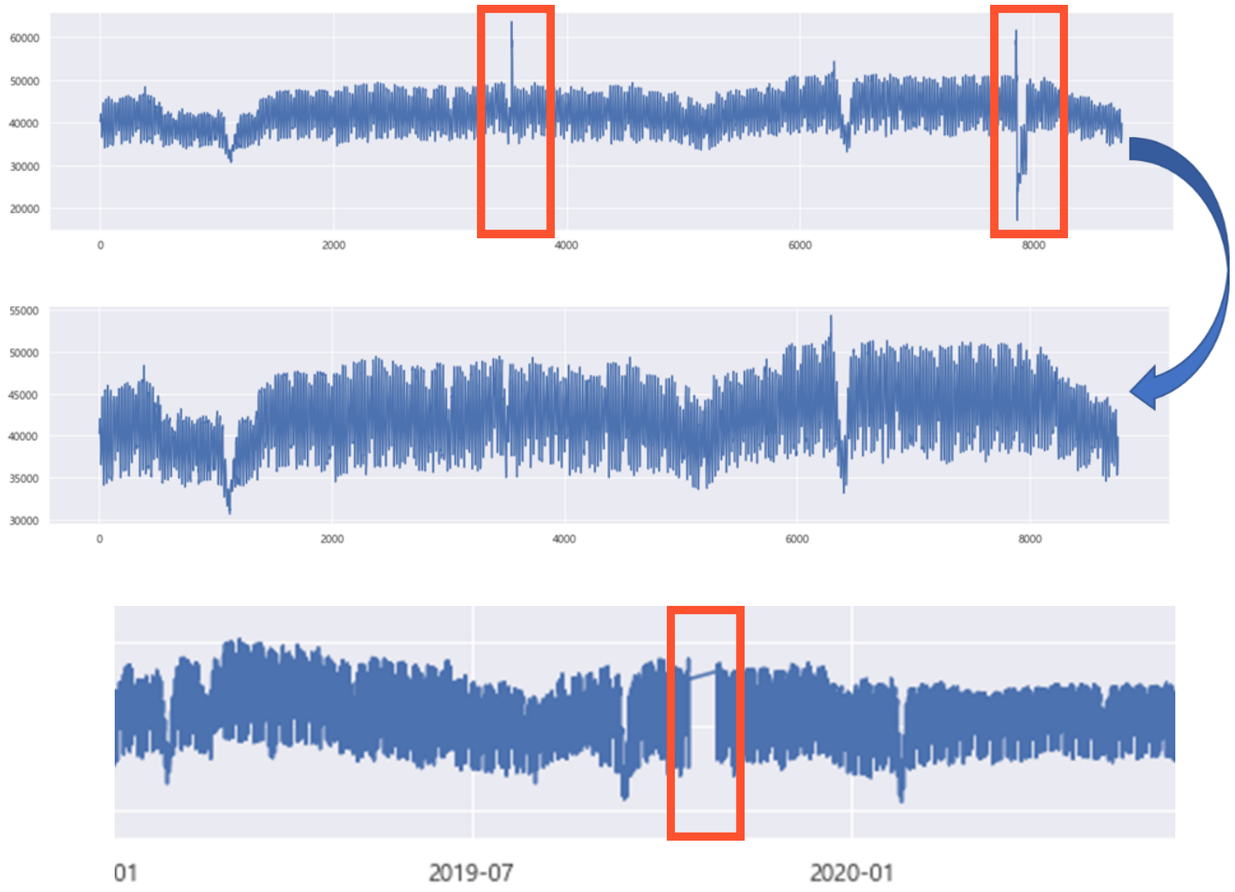
X(1512 shift)는
y 를 71.7%
설명헐 수 있다

최종 SCORE

1374.67255

[접근법 및 결과] 윤은수 팀원

데이터 전처리



이상치

2018년도 과도하게 큰 값, 작은 값

1) 2018-05-28(토요일)의 데이터를
2018-05-29(일요일)의 데이터로 대체

2) 2018-11-23~2018-11-27의 데이터를
2018-11-16~2018-11-20의 데이터로 대체

결측치

2019.10.15 - 10.27의 데이터가 비어 있어
2021.10.15 - 10.27의 데이터로 채워줌

[접근법 및 결과] 윤은수 팀원

모델 학습

* features

	-59d	-60d	-61d	-62d	-63d	-64d	-65d	-66d	-67d	-68d	-69d	-70d	
0	48312.5225	48480.0756	48673.0793	48891.3083	49456.6294	48083.9542	48293.8007	46068.1115	46962.2829	46914.9958	46930.1410	47624.4334	45
1	48449.6203	48691.2718	48795.2500	49011.4144	49479.4266	48344.0680	48496.8126	46820.3263	47126.8971	46977.7567	46997.3777	47477.7608	45
2	48494.9168	48836.7618	49086.9065	49203.7630	49353.0137	48382.8773	48594.0275	47326.0443	46989.3424	47090.3631	46939.4629	47356.0664	45
3	48803.7467	48999.7458	49021.9699	49458.1467	49545.7792	48531.6801	48877.8940	46959.0033	47071.0136	46992.4299	47182.6315	47449.2047	45
4	48721.7953	49434.1657	49210.9543	49788.0857	49771.8941	48903.7631	49398.5974	47371.1520	47367.9363	47195.3218	47368.2774	47715.7001	45

feature: 59일 이전의 20일치의 총생활인구수
-> 59일 SHIFT, 60일 SHIFT ... 79일 SHIFT

* model

```
import xgboost

xgb_reg = xgboost.XGBRegressor(n_estimators=100, learning_rate=0.08, gamma=0,
                               subsample=0.75, colsample_bytree=1, max_depth=7)
xgb_reg.fit(train_x, train_y)
y_pred = xgb_reg.predict(test_x)
```

: XGBoost Regression 모델

최종 SCORE

1369.17881

[접근법 및 결과] 이정곤 팀원

* 데이터 전처리

따로 수행하지 않음

* model

선형 회귀 모델

* features

59일*24만큼 shift한
feature 사용

최종 SCORE

1715.40668

향후 방향성

RNN으로 예측 수행해보기

해당 방안으로 접근해보고 싶은 이유

연속적인 input에 대해 recurrent 하게 예측하므로
시계열 데이터에 대해 일반 DNN보다 더 우수한 예측을 보일 것으로 예상

시도해보지 않은 이유

INPUT FORMAT

INPUT TENSOR

향후 방향성

전염병의 경향성을 feature로 대입해보기

해당 방안으로 접근해보고 싶은 이유

2017-2019	2020	2021
45,000 - 50,000	45,000 ↓	40,000 내외

총생활인구수의 증감과
전염병 여부에 상관관계가
있을 것으로 추정됨

시도해보지 않은 이유

BOOL VALUE

전염병O	전염병X
1	0

경향성

전염병 발생 -> 확산 ->
규제 -> 규제완화 -> 일상복귀

전염병에 따른 경향성을
반영하기 위해선
충분한 조사 및 계산 필요
시간 관계상 이번에는 생략 결정

소감

감사합니다

AI+X E조 PRESENTATION