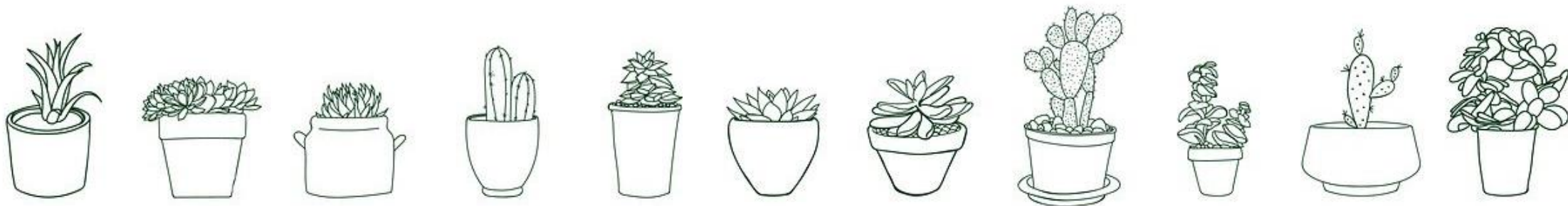


제주도 음식물 쓰레기양 예측을 통한 배출량 감소 방안 도출

팀명 : 에코탐라

팀장 : 김소은(trit1268@naver.com)

팀원 : 김서린(tjfls96@naver.com), 신정아(alalwjddk2@naver.com), 이윤지(pioneer0305@naver.com)





INDEX

01 서론

- 1.1 분석 배경
- 1.2 제공데이터 추이 확인
- 1.3 분석 내용 요약

02 데이터 전처리

- 2.1 구분변수
- 2.2 설명변수
- 2.3 반응변수

03 활용 알고리즘

- 3.1 상관분석
- 3.2 회귀분석
- 3.3 XGBoost
- 3.4 ARIMA model

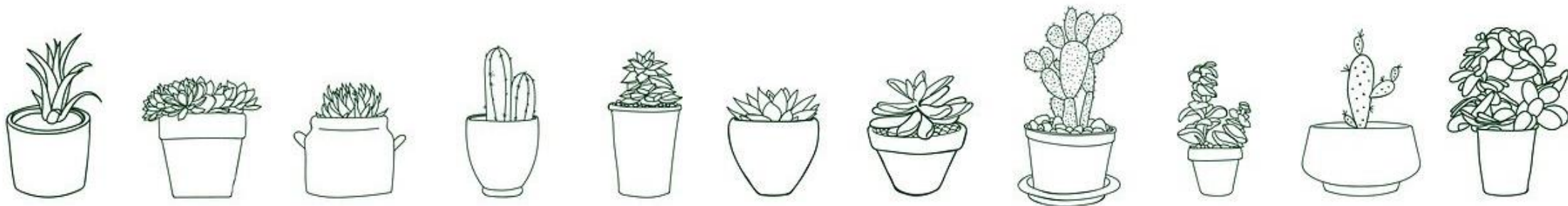
04 최종예측 결과

- 4.1 행정동별 예측 결과
- 4.2 행정동 "알 수 없음"
데이터 예측 결과

05 결론

- 5.1 분석결과 활용 및 시사점

01 서론 ♡



01 서론

1.1 분석 배경



대한민국 유일한 유네스코 3관왕 지역, but 1인당 생활 폐기물 배출량 전국 1위

01 서론

1.1 분석 배경



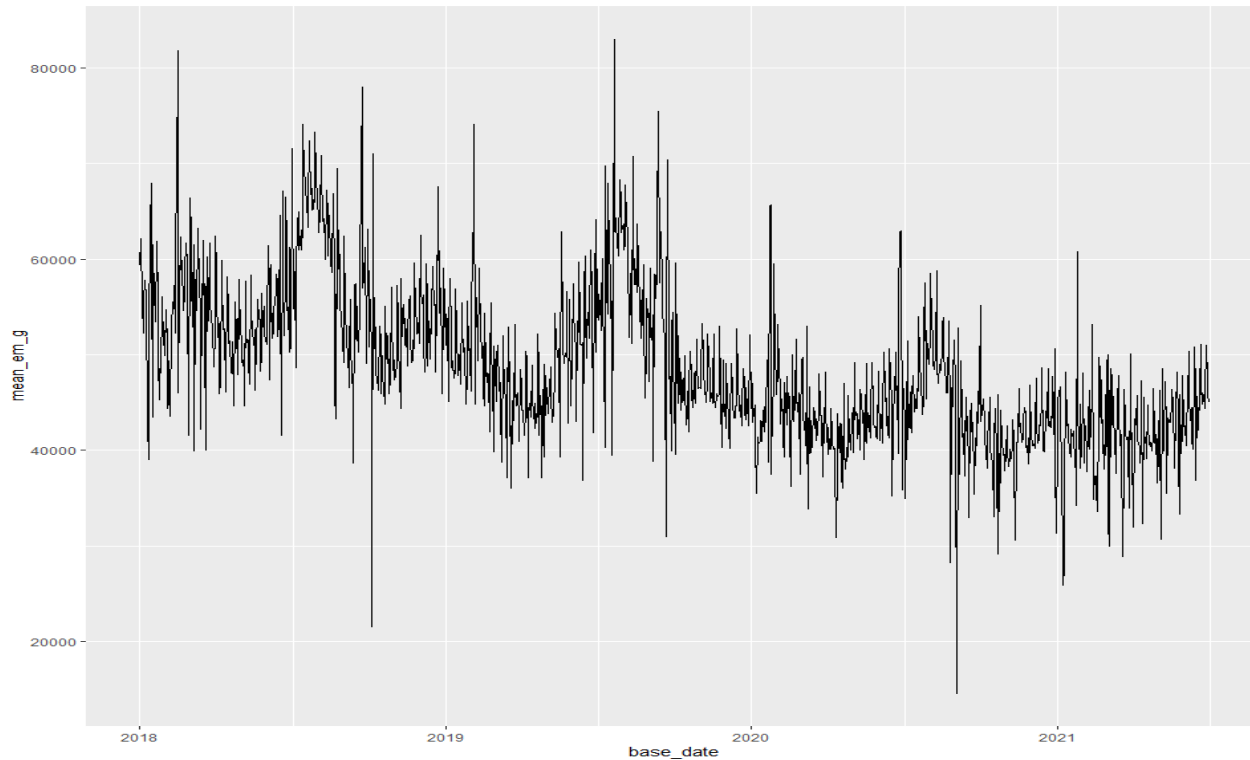
제주도 음식물 쓰레기 배출량 감소 방안 필요

1. 음식물 쓰레기 배출의 주요 요인 찾기
2. 음식물 쓰레기 배출량 예측
3. 음식물 쓰레기 감소 방안 모색

01 서론

1.2 제공데이터 추이 확인

1 음식물 쓰레기 배출량(g) 일일 그래프

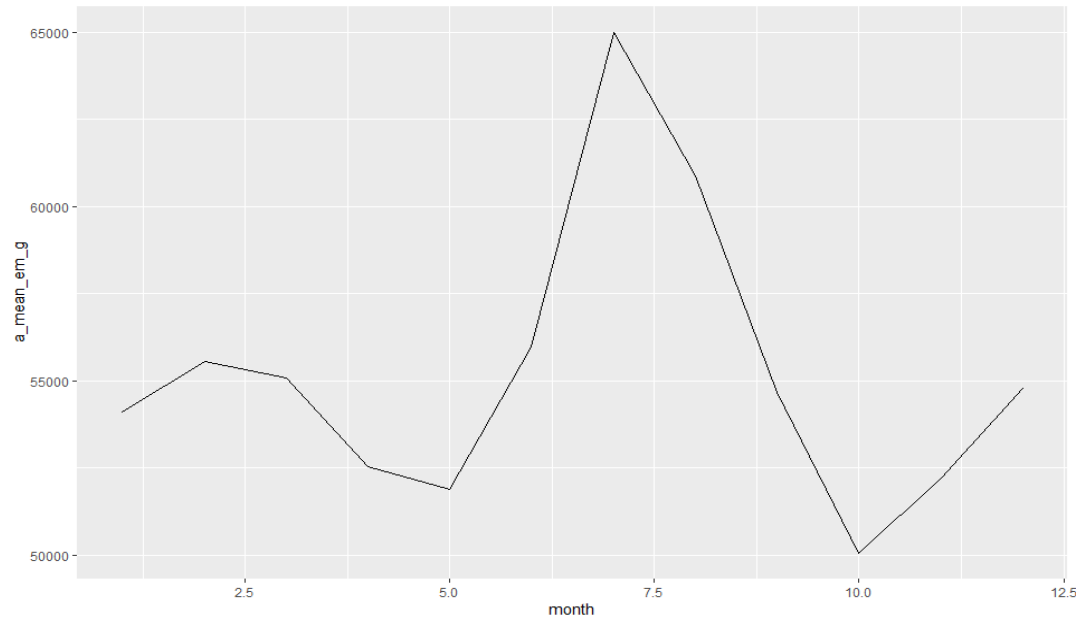


- 2018년부터 지속적으로 감소하는 추세
- 코로나가 발발한 2020년부터 크게 감소

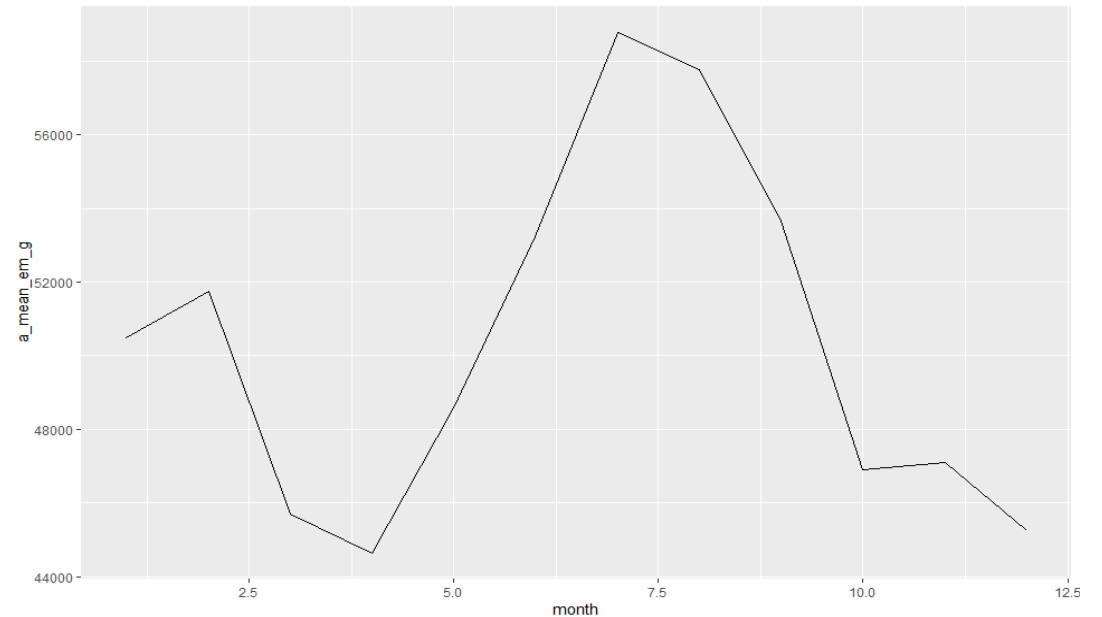
01 서론

1.2 제공데이터 추이 확인

2 음식물 쓰레기 배출량(g) 월별 그래프



2018년

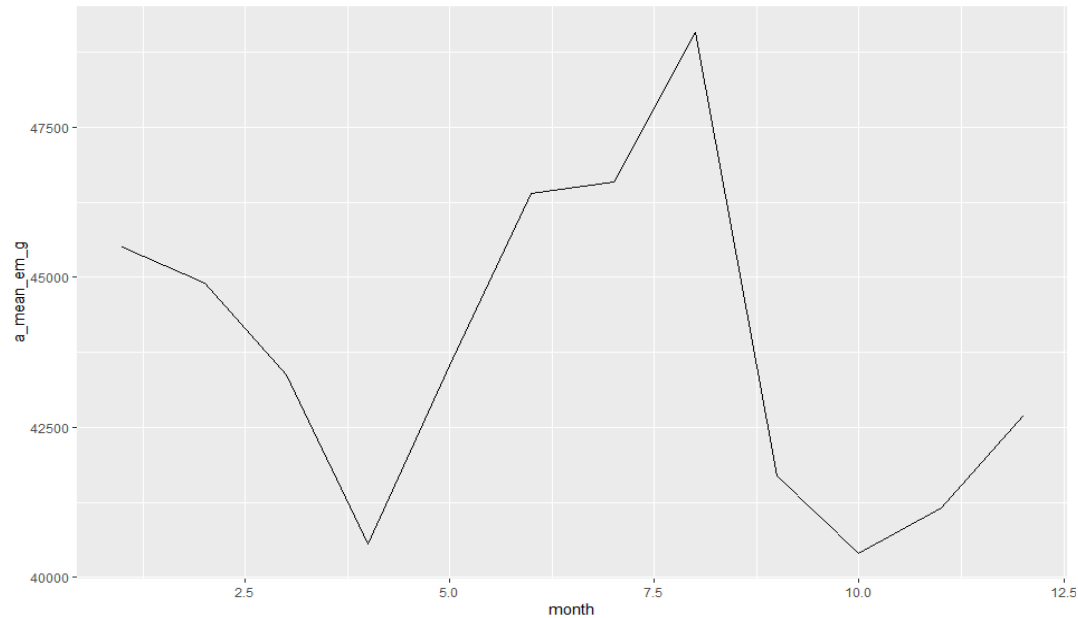


2019년

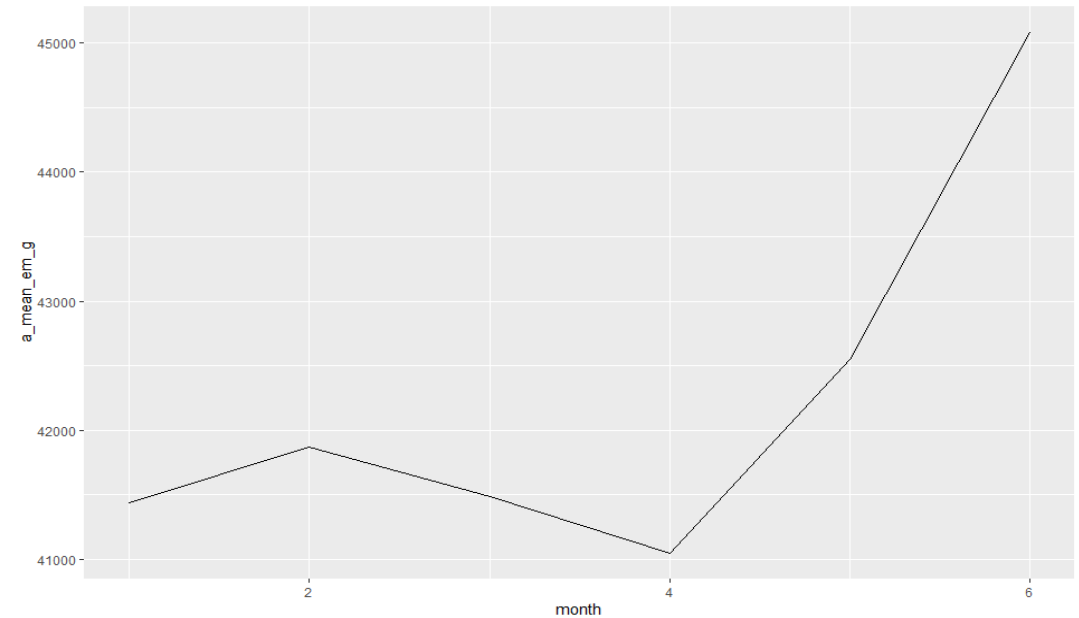
01 서론

1.2 제공데이터 추이 확인

2 음식물 쓰레기 배출량(g) 월별 그래프



2020년



2021년 (1~6월)

01 서론

1.2 제공데이터 추이 확인

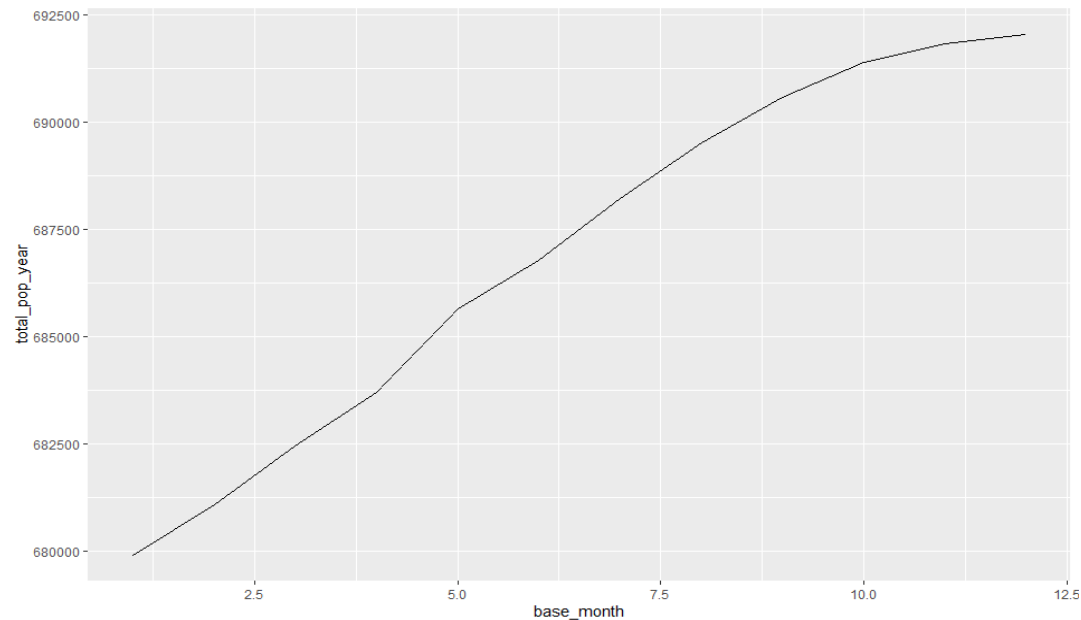
2 음식물 쓰레기 배출량(g) 월별 그래프

- 여름 휴가철인 6, 7, 8월에 급증
- 겨울 휴가철인 12, 1, 2월에도 증가
- 따라서 휴가철에 음식물 쓰레기 배출량이 많음을 알 수 있음

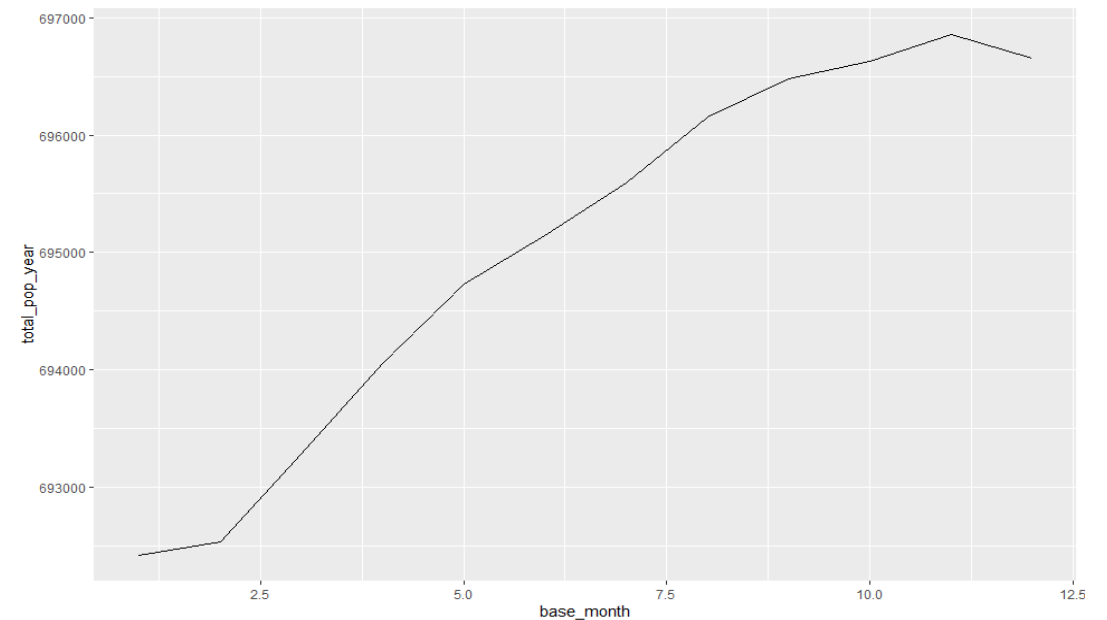
01 서론

1.2 제공데이터 추이 확인

3 제주 총 거주 인구수(명) 월별 그래프



2018년

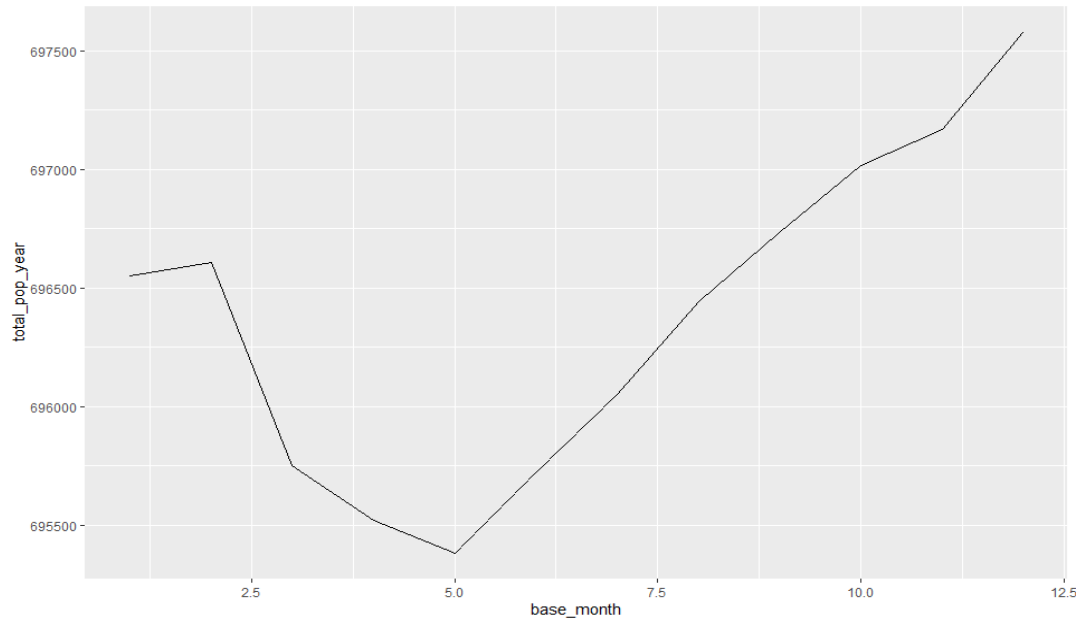


2019년

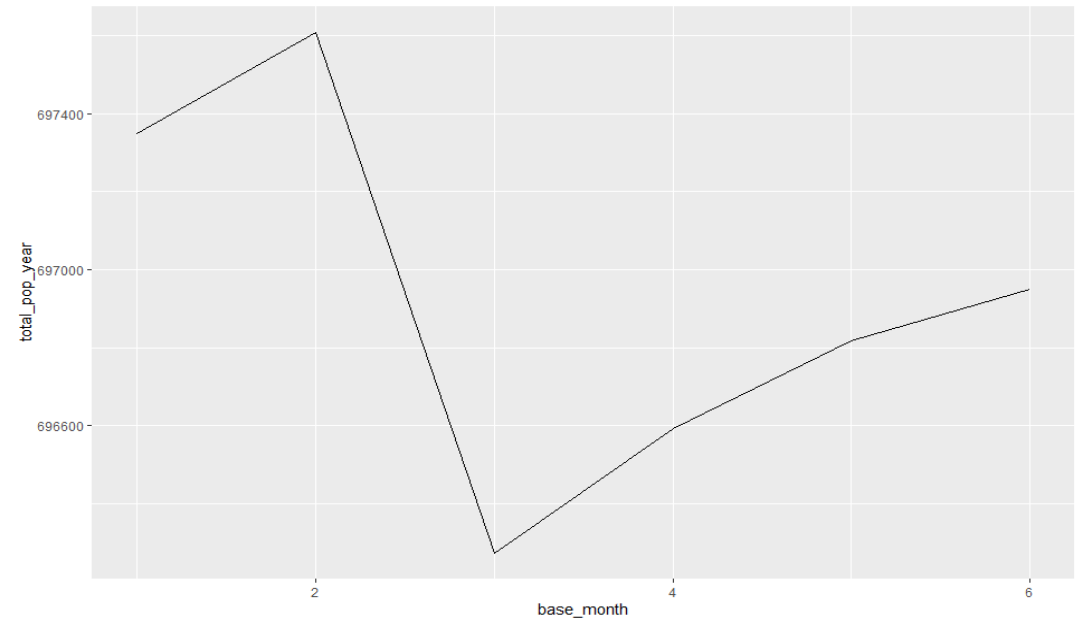
01 서론

1.2 제공데이터 추이 확인

3 제주 총 거주 인구수(명) 월별 그래프



2020년



2021년 (1~6월)

01 서론

1.2 제공데이터 추이 확인

3 제주 총 거주 인구수(명) 월별 그래프

- 2018, 2019년에 지속적으로 증가
- 2020년 초부터 감소, 증가를 반복

01 서론

1.2 제공데이터 추이 확인

- 2020년 : 음식물 쓰레기 배출량이 확연히 감소하기 시작
- 2020년 : 계속해서 증가하던 총 거주 인구수가 감소하기 시작
따라서 두 변수 사이에 관련성이 있을 것으로 예상됨
- 분석목표
이와 같이 음식물 쓰레기 배출량과 다른 변수들 사이에 관련성이 있을 것으로 예상되므로, 관련성을 확인하고 최종적으로 배출량 감소방안 강구

01 서론

1.3 분석 내용 요약

1. 제공데이터 추이 확인

- 분석 목적 설정
- 분석 방향 설정

2. 데이터 전처리

- 외부 데이터
불러오기
- 각 변수를
행정동마다의
월별 데이터로 변환

3. 상관분석

- 외부데이터의
적합성 확인

4. 회귀분석

- 외부데이터와
반응변수 사이의
인과관계 확인

01 서론

1.3 분석 내용 요약

5. XGBoost

- 파라미터 조절해
최적 모델 선정
- 주요 변수 선정

6. ARIMA model

- 시계열 데이터의
정상성 확인
- 독립변수의 2021년
7, 8월 값 예측

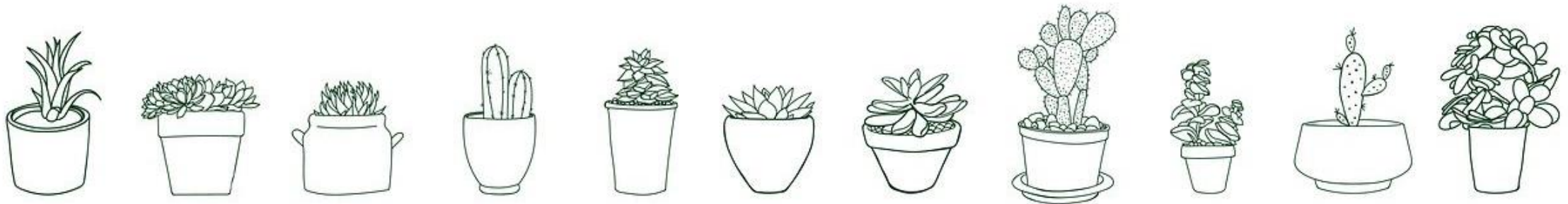
7. 최종 예측

- XGBoost로 선정한 주요
변인을 ARIMA를 통해
2021년 7, 8월 값 예측
- Qgis를 이용해 음식물쓰레기
배출량(y)의 예측값
시각화
- 행정동 "알 수 없음"의
데이터 예측

8. 결론

- 최종 예측을 통해
제주도의 음식물쓰레기
배출량을 줄일 수
있는 방안 강구

02 데이터 전처리 ♡



02 데이터 전처리

2.1 구분변수

1

base_date (배출일자)

- 타입 : DATETIME
- YYYY-MM-DD
- 분석에 사용될 각 설명변수들을 base_date를 이용해 월 단위 데이터로 변환해 사용

	base_date
2018-01-31	2018-01-31
2018-02-28	2018-02-28
2018-03-31	2018-03-31
2018-04-30	2018-04-30
2018-05-31	2018-05-31
...	...
2021-02-28	2021-02-28
2021-03-31	2021-03-31
2021-04-30	2021-04-30
2021-05-31	2021-05-31
2021-06-30	2021-06-30
1806 rows × 3 columns	

02 데이터 전처리

2.1 구분변수

2

emd_cd (행정동 코드)

- 타입 : STRING
- 43개 행정동 코드 + 알 수 없음
- 분석에 사용될 각 데이터를 emd_cd별로 정리한 후 분석
- "알 수 없음" 데이터는 따로 모아서 분석

No	행정등코드	행정등명
0	1	50110250 한림읍
1	2	50110253 애월읍
2	3	50110256 구좌읍
3	4	50110259 초천읍
4	5	50110310 한경면
5	6	50110320 추자면
6	7	50110330 우도면
7	8	50110510 원도1동
8	9	50110520 원도2동
9	10	50110530 이도1동
10	11	50110540 이도2동
11	12	50110550 삼도1동
12	13	50110560 삼도2동
13	14	50110570 용담1동
14	15	50110580 용담2동
15	16	50110590 건암동
16	17	50110600 화북동
17	18	50110610 삼양동
18	19	50110620 봉개동
19	20	50110630 아라동
20	21	50110640 오라동
21	22	50110650 연동
22	23	50110660 노형동
23	24	50110670 외도동
24	25	50110680 이호동
25	26	50110690 도두동

26	27	50130250 대정읍
27	28	50130253 남원읍
28	29	50130259 성산읍
29	30	50130310 안덕면
30	31	50130320 표선면
31	32	50130510 송산동
32	33	50130520 정방동
33	34	50130530 중앙동
34	35	50130540 천치동
35	36	50130550 효돈동
36	37	50130560 영천동
37	38	50130570 동흥동
38	39	50130580 서흥동
39	40	50130590 대륜동
40	41	50130600 대천동
41	42	50130610 중문동
42	43	50130620 예래동
43	44	XXXXXXX 알수없음

02 데이터 전처리

2.2 설명변수

1

korean (내국인 유동인구)

- 거주인구(res_pop_cnt) + 근무인구(work_pop_cnt) + 방문인구(visit_pop_cnt)
- 타입 : FLOAT
- 해당 시각 정각에 측정한 거주, 근무, 방문인구(명)의 합
- 거주/근무/방문인구 : 1~24시 해당 시간 정각에 측정한 인구
(거주지/근무지/방문지와 근무지 외 지역에 머문 시간(분) / 60분)
- 시간별 내국인 유동인구 데이터를 월별 내국인 유동인구 데이터로 정리해 사용
- 행정동별 월 내국인 유동인구 수로 변환 후 분석에 사용

	base_date	emd_cd	pop_cnt
2018-01-31	2018-01-31	50110250	7.209306e+06
2018-02-28	2018-02-28	50110250	5.579125e+06
2018-03-31	2018-03-31	50110250	7.615021e+06
2018-04-30	2018-04-30	50110250	8.498349e+06
2018-05-31	2018-05-31	50110250	8.916782e+06
...
2021-02-28	2021-02-28	50130620	4.321265e+06
2021-03-31	2021-03-31	50130620	4.744309e+06
2021-04-30	2021-04-30	50130620	5.236706e+06
2021-05-31	2021-05-31	50130620	5.805413e+06
2021-06-30	2021-06-30	50130620	5.705196e+06

1806 rows × 3 columns

02 데이터 전처리

2.2 설명변수

2 long_term_frqn (장기체류 외국인 유동인구)

- 거주인구(res_pop_cnt) + 근무인구(work_pop_cnt) + 방문인구(visit_pop_cnt)
- 타입 : FLOAT
- 해당 시각 정각에 측정한 거주, 근무, 방문인구(명)의 합
- 거주/근무/방문인구 : 1~24시 해당 시간 정각에 측정한 인구
(거주지/근무지/방문지와 근무지 외 지역에 머문 시간(분) / 60분)
- 시간별 장기체류 외국인 유동인구 데이터를 월별 장기체류 외국인 유동인구 데이터로 정리해 사용
- 행정동별 월 장기체류 유동인구 수로 변환 후 분석에 사용

	base_date	emd_cd	pop_cnt
2018-01-31	2018-01-31	50110250	427986.3363
2018-02-28	2018-02-28	50110250	427240.8119
2018-03-31	2018-03-31	50110250	513638.2631
2018-04-30	2018-04-30	50110250	474601.3510
2018-05-31	2018-05-31	50110250	395262.5483
...
2021-02-28	2021-02-28	50130620	109325.4866
2021-03-31	2021-03-31	50130620	118258.8891
2021-04-30	2021-04-30	50130620	129227.7480
2021-05-31	2021-05-31	50130620	144213.2650
2021-06-30	2021-06-30	50130620	146956.6848

1806 rows × 3 columns

02 데이터 전처리

2.2 설명변수

3 short_term_frgn (단기체류 외국인 유동인구)

- 타입 : FLOAT
- 해당 시각 정각에 측정한 방문인구(명)
- 시간별 단기체류 외국인 유동인구 데이터를 월별 단기체류 외국인 유동인구 데이터로 정리해 사용
- 행정동별 월 단기체류 유동인구 수로 변환 후 분석에 사용

	base_date	emd_cd	visit_pop_cnt
2018-01-31	2018-01-31	50110250	58090.5960
2018-02-28	2018-02-28	50110250	69710.8394
2018-03-31	2018-03-31	50110250	98026.6381
2018-04-30	2018-04-30	50110250	103289.7891
2018-05-31	2018-05-31	50110250	115255.9750
...
2021-02-28	2021-02-28	50130620	280.6159
2021-03-31	2021-03-31	50130620	1475.8404
2021-04-30	2021-04-30	50130620	1283.0997
2021-05-31	2021-05-31	50130620	1212.2652
2021-06-30	2021-06-30	50130620	1194.0627

1806 rows × 3 columns

02 데이터 전처리

2.2 설명변수

4

resident (총 거주인구)

- 타입 : INT
- 행정동별 총 거주인구
- 주민등록 거주인구(resid_reg_pop)와 외국인 거주 인구(foreign_pop)의 합
- 행정동별 월 총 거주인구 수 데이터 그대로 사용

	base_date	emd_cd	total_pop
2018-01-31	2018-01-31	50110250	24419
2018-02-28	2018-02-28	50110250	24427
2018-03-31	2018-03-31	50110250	24532
2018-04-30	2018-04-30	50110250	24606
2018-05-31	2018-05-31	50110250	24715
...
2021-02-28	2021-02-28	50130620	4173
2021-03-31	2021-03-31	50130620	4178
2021-04-30	2021-04-30	50130620	4161
2021-05-31	2021-05-31	50130620	4178
2021-06-30	2021-06-30	50130620	4152

1806 rows × 3 columns

02 데이터 전처리

2.2 설명변수

	base_date	emd_cd	use_cnt	use_amt
2018-01-31	2018-01-31	50110250	215.79479	8171132.98371	2021-02-28	2021-02-28	50130620	65.21600	2463788.71200
2018-02-28	2018-02-28	50110250	213.71326	8605532.33333	2021-03-31	2021-03-31	50130620	67.70000	2572585.70000
2018-03-31	2018-03-31	50110250	228.40134	8517373.71572	2021-04-30	2021-04-30	50130620	68.64234	2473174.90146
2018-04-30	2018-04-30	50110250	239.89597	8640048.18456	2021-05-31	2021-05-31	50130620	69.33333	2864493.94898
2018-05-31	2018-05-31	50110250	250.34839	9384709.80968	2021-06-30	2021-06-30	50130620	69.89299	2404632.73432
...	1806 rows × 4 columns				

5 card_cnt (음식관련 카드 결제건수)

- 타입 : INT
- 단위 : 건
- 일별 음식관련 카드 결제건수 데이터를 월별 음식관련 카드 결제건수 데이터로 정리해 사용
- 행정동별 월 음식관련 카드 결제건수로 변환 후 분석에 사용

6 card_amt (음식관련 카드 결제금액)

- 타입 : INT
- 단위 : 원
- 일별 음식관련 카드 결제금액 데이터를 월별 음식관련 카드 결제금액 데이터로 정리해 사용
- 행정동별 월 음식관련 카드 결제금액으로 변환 후 분석에 사용

02 데이터 전처리

2.2 설명변수

7

waste_cnt (음식물 쓰레기 배출건수)

- 타입 : INT
- 배출거점지역 음식물 쓰레기 배출건수(건)
- 일별 음식물 쓰레기 배출건수 데이터를 월별 음식물 쓰레기 배출건수 데이터로 정리해 사용
- 행정동별 월 음식물 쓰레기 배출건수로 변환 후 분석에 사용

	base_date	emd_cd	em_cnt
2019-11-30	2019-11-30	50110250	620
2019-12-31	2019-12-31	50110250	8178
2020-01-31	2020-01-31	50110250	13042
2020-02-29	2020-02-29	50110250	12264
2020-03-31	2020-03-31	50110250	14316
...
2021-02-28	2021-02-28	50130620	4563
2021-03-31	2021-03-31	50130620	4920
2021-04-30	2021-04-30	50130620	5256
2021-05-31	2021-05-31	50130620	6346
2021-06-30	2021-06-30	50130620	6328

1634 rows × 4 columns

02 데이터 전처리

2.2 설명변수

	base_date	emd_cd	detached	apt	town	multiplex	commercial_building
0	2018-01-31	50110250	59279	56760	17641	27893	3647
1	2018-01-31	50110253	59279	56760	17641	27893	3647
2	2018-01-31	50110256	59279	56760	17641	27893	3647
3	2018-01-31	50110259	59279	56760	17641	27893	3647
4	2018-01-31	50110310	59279	56760	17641	27893	3647
...

...
1801	2021-06-30	50130580	852	1758	512	625	66
1802	2021-06-30	50130590	1844	2143	1688	397	81
1803	2021-06-30	50130600	1612	2529	797	214	39
1804	2021-06-30	50130610	1484	1667	1152	615	88
1805	2021-06-30	50130620	1023	0	219	39	29
1806 rows × 7 columns							

8

detached(단독주택), apt(아파트), town(연립주택), multiplex(다세대 주택), commercial_building(비거주용 건물 내 주택)

- 타입 : INT
- 외부 데이터 KOSIS의 '주택의 종류별 주택 -읍면동(2015, 2020), 시군구(2016~2019) 이용
- 2018년 : 제주도시, 서귀포시의 년 단위 주택 수, 12개월을 같은 주택 수라고 가정
- 2019년 : 제주도시, 서귀포시의 년 단위 주택 수, 12개월을 같은 주택 수라고 가정
- 2020년 : 행정동별 년 단위 주택 수, 12개월을 같은 주택 수라고 가정
- 2021년 : 2020년 데이터 사용, 6개월을 같은 주택 수라고 가정

02 데이터 전처리

2.2 설명변수

9 distancing (사회적 거리두기 단계)

- 타입 : FLOAT
- 외부 데이터 'covid19.jeju'와 기사를 통해 자료 수집
- 제주의 사회적 거리두기 단계를 일별로 정리한 후 월평균으로 구하여 사용
- 사회적 거리두기 단계 체계가 등장하기 이전은 0 단계라고 가정

	base_date	emd_cd	distancing
0	2018-01-31	50110250	0.0
1	2018-01-31	50110253	0.0
2	2018-01-31	50110256	0.0
3	2018-01-31	50110259	0.0
4	2018-01-31	50110310	0.0
...
1801	2021-06-30	50130580	2.0
1802	2021-06-30	50130590	2.0
1803	2021-06-30	50130600	2.0
1804	2021-06-30	50130610	2.0
1805	2021-06-30	50130620	2.0

1806 rows × 3 columns

02 데이터 전처리

2.2 설명변수

	base_date	emd_cd	temp	rain
2018-01-31	2018-01-31	50110250	186.067742	3190.0
2018-02-28	2018-02-28	50110250	220.710714	5336.0
2018-03-31	2018-03-31	50110250	580.093548	9555.5
2018-04-30	2018-04-30	50110250	849.023333	11614.5
2018-05-31	2018-05-31	50110250	1031.838710	14833.5
...

...
2018-08-31	2018-08-31	50130620	1451.683871	10063.0
2018-09-30	2018-09-30	50130620	1204.370000	19198.0
2018-10-31	2018-10-31	50130620	904.706452	10889.5
2018-11-30	2018-11-30	50130620	719.200000	2711.5
2018-12-31	2018-12-31	50130620	427.983871	3291.5
505 rows × 4 columns				

10

temp (월평균기온 총합)

- 타입 : FLOAT
- 외부 데이터 기상청과 제주데이터허브를 통해 자료 수집
- 행정동별 일 평균기온을 행정동별 월평균 기온의 총합으로 정리 후 사용

11

rain (월별 강수량 총합)

- 타입 : FLOAT
- 외부 데이터 기상청과 제주데이터허브를 통해 자료 수집
- 행정동별 일 강수량을 행정동별 월강수량의 총합으로 정리 후 사용

02 데이터 전처리

2.3 반응변수

1

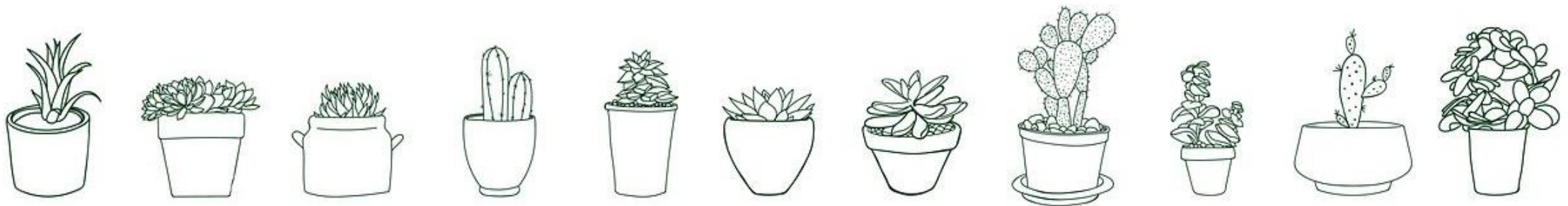
waste_amt (음식물 쓰레기 배출량)

- 타입 : INT
- 배출거점지역 음식물 쓰레기 총 배출량(g)
- 일별 음식물 쓰레기 배출량 데이터를 월별 음식물 쓰레기 배출량 데이터로 정리해 사용
- 행정동별 월 음식물 쓰레기 총 배출량으로 변환 후 분석에 사용

	base_date	emd_cd	em_g
2019-11-30	2019-11-30	50110250	2279550
2019-12-31	2019-12-31	50110250	31493650
2020-01-31	2020-01-31	50110250	49632850
2020-02-29	2020-02-29	50110250	46197050
2020-03-31	2020-03-31	50110250	52561750
...
2021-02-28	2021-02-28	50130620	15566250
2021-03-31	2021-03-31	50130620	17300600
2021-04-30	2021-04-30	50130620	17646850
2021-05-31	2021-05-31	50130620	19494750
2021-06-30	2021-06-30	50130620	18753350

1634 rows × 3 columns

03 활용 알고리즘 🌱



03 활용 알고리즘

3.1 상관분석

1

상관분석의 의미와 선정 이유

- 상관분석(correlation analysis)이란?

- 두 변수 간에 어떤 선형적 또는 비선형적 관계를 갖는지 분석하는 방법으로 상관계수를 이용해 측정

- 상관계수(correlation coefficient)이란?

- 두 변수 사이의 선형 관계 정도를 수치화한 계수
- '+1'은 완벽한 양의 선형 상관관계, '0'은 선형 상관관계 없음, '-1'은 완벽한 음의 선형 상관관계를 의미한다.

03 활용 알고리즘

3.1 상관분석

1

상관분석의 의미와 선정 이유

- 외부 데이터 temp(월 평균기온의 총합) & waste_amt(음식물 쓰레기 배출량)
- rain(월별 강수량의 총합) & waste_amt(음식물 쓰레기 배출량)

각각의 관계의 정도를 알아보기 위해 상관분석을 실시

03 활용 알고리즘

3.1 상관분석

2

em_g와 temp, rain간의 상관분석

- em_g와 temp의 상관계수 : -0.02
 - em_g와 rain의 상관계수 : 0.13
- => 상관계수가 0에 가까운 값.
따라서 음식물 쓰레기 배출량의 주
요 요인으로 적합하지 않다고 판단

```
corr=food_waste_sort_201819.corr(method='pearson')  
corr
```

	emd_cd	em_cnt	em_g	temp	rain
emd_cd	1.000000	-0.451129	-0.382820	0.053998	-0.070709
em_cnt	-0.451129	1.000000	0.986177	-0.004420	0.173686
em_g	-0.382820	0.986177	1.000000	-0.022951	0.129840
temp	0.053998	-0.004420	-0.022951	1.000000	0.333577
rain	-0.070709	0.173686	0.129840	0.333577	1.000000

- 회귀분석(regression analysis)이란?

- 관찰된 연속형 변수들에 대해 변수 간의 인과관계를 밝히고 모형을 적합하여 관심 있는 변수를 예측하거나 추론하기 위한 분석방법
- 설명변수로 반응변수가 반응변수에 미치는 영향을 일반화
- 선형회귀분석은 선형성, 등분산성, 독립성, 비상관성, 정규성을 가정

03 활용 알고리즘

3.2 회귀분석

1 회귀분석의 의미와 선정 이유

- 외부 데이터 temp(월 평균기온의 총합), rain(월별 강수량의 총합)과 waste_amt(음식물 쓰레기 배출량) 사이의 인과관계를 살펴보기 위해 회귀분석 실시
=> 이는 매일의 날씨에 따라 음식 배달 빈도가 달라지며,
음식물 쓰레기 배출량 역시 달라질 것이란 생각에서 출발했다.

03 활용 알고리즘

3.2 회귀분석

2

temp와 waste_amt의 회귀분석

OLS Regression Results

Dep. Variable:	em_g	R-squared (uncentered):	0.208	
Model:	OLS	Adj. R-squared (uncentered):	0.207	
Method:	Least Squares	F-statistic:	229.1	
Date:	Thu, 09 Sep 2021	Prob (F-statistic):	4.02e-46	
Time:	23:42:53	Log-Likelihood:	-17139.	
No. Observations:	873	AIC:	3.428e+04	
Df Residuals:	872	BIC:	3.428e+04	
Df Model:	1			
Covariance Type:	nonrobust			
	coef	std err	t P> t [0.025 0.975]	
temp	1.21e+05	7994.329	15.135 0.000	1.05e+05 1.37e+05
Omnibus:	190.241	Durbin-Watson:	0.101	
Prob(Omnibus):	0.000	Jarque-Bera (JB):	355.314	
Skew:	1.287	Prob(JB):	6.99e-78	
Kurtosis:	4.773	Cond. No.	1.00	

- waste_amt 변동의 약 20%만이 temp 변동에 의해 설명됨을 의미
- temp와 waste_amt 사이의 회귀식의 정확도는 매우 낮으므로 설명변수 temp로서의 역할을 하기 어렵다고 판단해 제거

03 활용 알고리즘

3.2 회귀분석

3

rain와 waste_amt의 회귀분석

OLS Regression Results

Dep. Variable:	em_g	R-squared (uncentered):	0.327
Model:	OLS	Adj. R-squared (uncentered):	0.327
Method:	Least Squares	F-statistic:	424.6
Date:	Thu, 09 Sep 2021	Prob (F-statistic):	3.62e-77
Time:	23:44:11	Log-Likelihood:	-17068.
No. Observations:	873	AIC:	3.414e+04
Df Residuals:	872	BIC:	3.414e+04
Df Model:	1		
Covariance Type:	nonrobust		

	coef	std err	t	P> t	[0.025	0.975]
rain	1.603e+05	7780.776	20.605	0.000	1.45e+05	1.76e+05

Omnibus:	138.458	Durbin-Watson:	0.392
Prob(Omnibus):	0.000	Jarque-Bera (JB):	260.269
Skew:	0.948	Prob(JB):	3.04e-57
Kurtosis:	4.888	Cond. No.	1.00

- waste_amt 변동의 32.70%만이 rain 변동에 의해 설명됨을 의미
- rain과 waste_amt 사이의 회귀식의 정확도는 매우 낮으므로 설명 변수 rain으로서의 역할을 하기 어렵다고 판단해 제거

03 활용 알고리즘

3.2 회귀분석

4

temp, rain와 waste_amt의 회귀분석

OLS Regression Results

Dep. Variable:	em_g	R-squared:	0.022			
Model:	OLS	Adj. R-squared:	0.020			
Method:	Least Squares	F-statistic:	2.69			
Date:	Thu, 09 Sep 2021	Prob (F-statistic):	6.86e-05			
Time:	23:26:57	Log-Likelihood:	-16915.			
No. Observations:	873	AIC:	3.384e+04			
Df Residuals:	870	BIC:	3.385e+04			
Df Model:	2					
Covariance Type:	nonrobust					
	coef	std err	t	P> t	[0.025	0.975]
const	5.977e+07	3.25e+06	18.409	0.000	5.34e+07	6.61e+07
temp	-1.817e+04	8667.493	-2.096	0.036	-3.52e+04	-1156.920
rain	4.275e+04	9829.308	4.350	0.000	2.35e+04	6.2e+04
Omnibus:	265.817	Durbin-Watson:	0.170			
Prob(Omnibus):	0.000	Jarque-Bera (JB):	608.572			
Skew:	1.669	Prob(JB):	7.08e-133			
Kurtosis:	5.363	Cond. No.	654.			

- waste_amt 변동의 2%만이 temp와 rain 변동에 의해 설명됨을 의미
- temp, rain과 waste_amt 사이의 회귀식의 정확도는 매우 낮으므로 설명변수 temp, rain으로서의 역할을 하기 어렵다고 판단해 제거

03 활용 알고리즘

3.3 XGBoost

1 XGBoost의 개념과 선정이유

- 머신러닝 기법 중 의사결정나무(Decision tree)를 기반으로 한 앙상블 방법
- Boosting 기반
- 효율성, 유연성, 휴대성의 장점
- 유연하게 여러 파라미터를 조절해가며 최적의 모델을 만드는 러닝 시스템
- 과적합(over-fitting)을 방지
- 빠른 학습 및 예측 가능
- 높은 성능

03 활용 알고리즘

3.3 XGBoost

2 XGBoost 통해 최적 모델 찾기

- 하이파라미터 설정

- 전체 데이터셋을 학습용 80%, 테스트용 20%로 분할
- max_depth(트리당 최대 깊이) = 7
- eta(학습률) = 0.1
- 목적함수(objective) = reg:linear (회귀)
- 부스팅 반복횟수 = 1000
- 조기종단을 위한 최소 반복횟수 = 100

03 활용 알고리즘

3.3 XGBoost

2 XGBoost 통해 최적 모델 찾기

- 모델 비교

- 설명변수 waste_amt (음식물쓰레기 배출량, g)의 범위가 매우 크기 때문에 RMSE 대신 RMSLE 사용
- RMSLE : 예측값과 실제값에 로그를 씌운 후 차이를 비교하는 방법

03 활용 알고리즘

3.3 XGBoost

2 XGBoost 통해 최적 모델 찾기

- Ver_1

- 제공데이터의 변수만을 이용해 XGBoost 모델링
- 설명변수(x) :

korean (내국인 유동인구)	card_cnt (음식 관련 카드 결제건수)
long_term_frgn (장기체류 외국인 유동인구)	card_amt (결제금액)
short_term_frgn (단기체류 외국인 유동인구)	waste_cnt (매출건수)
resident (총 거주인구)	

03 활용 알고리즘

3.3 XGBoost

- Ver_1

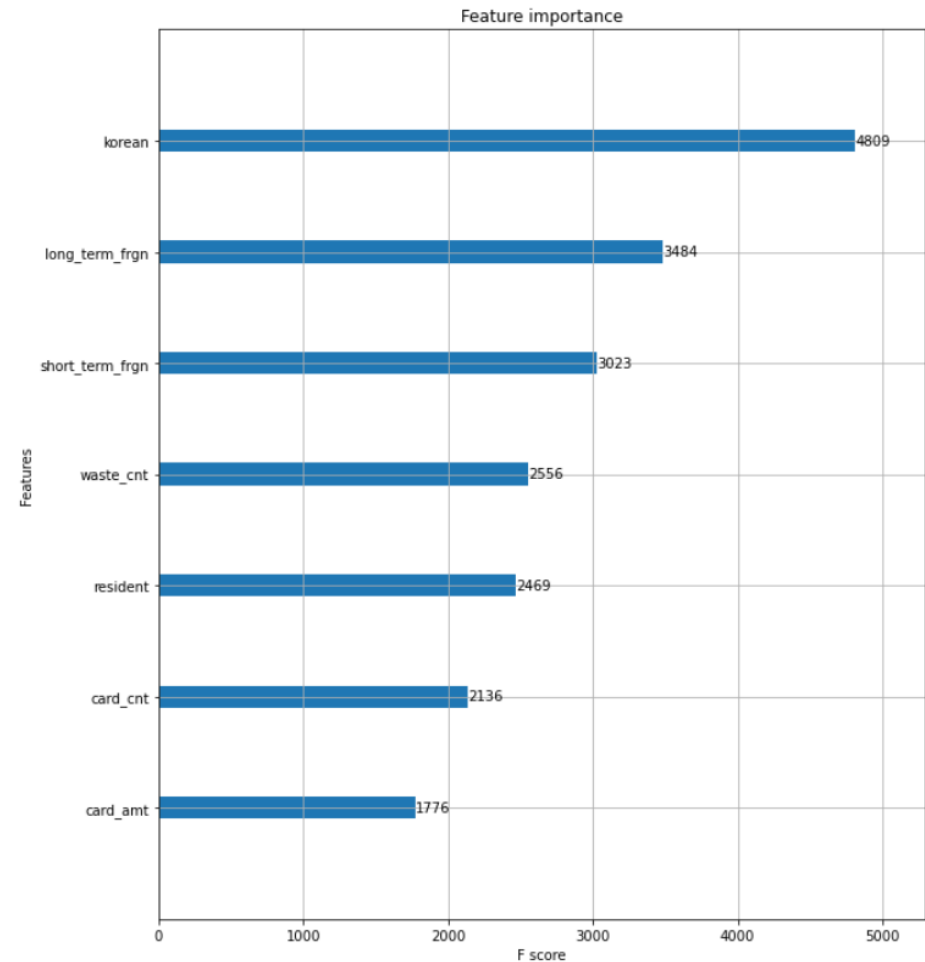
Model explanation

- Feature Importance top3

- korean
- long_term_frqn
- short_term_frqn

결과 : eval- RMSLE = 0.09863

2 XGBoost 통해 최적 모델 찾기



03 활용 알고리즘

3.3 XGBoost

2 XGBoost 통해 최적 모델 찾기

- Ver_2-1

- 제공데이터의 변수만을 이용해 XGBoost 모델링
 - ver_1의 변수 중 **korean, long_term_frgn** -> 거주/근무/방문인구로 분류
 - ver_1의 변수 중 **resident** -> 주민등록/외국인 거주인구, 남성/여성으로 분류
 - ver_1의 변수 중 **card_cnt, card_amt**
- > home(배달, 간식, 식품, 마트/슈퍼마켓/농축수산물) / out(한식, 패스트푸드, 주점 및 주류 판매, 양식, 뷔페, 아시아음식) 으로 분류

03 활용 알고리즘

3.3 XGBoost

2

XGBoost 통해 최적 모델 찾기

- 설명변수 (x) :

korean_resd (내국인 유동인구, 거주인구)	short_term_frgn_visit (단기체류 외국인 유동인구)	card_amt_home (결제금액, 가정)
korean_work (내국인 유동인구, 근무인구)	resident_fe_korean (총 거주인구, 여성, 주민등록 거주인구)	card_cnt_out (음식 관련 카드 결제건수, 외식)
korean_visit (내국인 유동인구, 방문인구)	resident_fe_frgn (총 거주인구, 여성, 외국인 거주인구)	card_amt_out (결제금액, 외식)
long_term_frgn_resd (장기체류 외국인 유동인구, 거주인구)	resident_m_korean (총 거주인구, 남성, 주민등록 거주인구)	waste_cnt (배출건수)
long_term_frgn_work (장기체류 외국인 유동인구, 근무인구)	resident_m_frgn (총 거주인구, 남성, 외국인 거주인구)	
long_term_frgn_visit (장기체류 외국인 유동인구, 방문인구)	card_cnt_home (음식 관련 카드 결제건수, 가정)	

03 활용 알고리즘

3.3 XGBoost

2

XGBoost 통해 최적 모델 찾기

- Ver_2-1

Model explanation

- Feature Importance top3

korean_resid

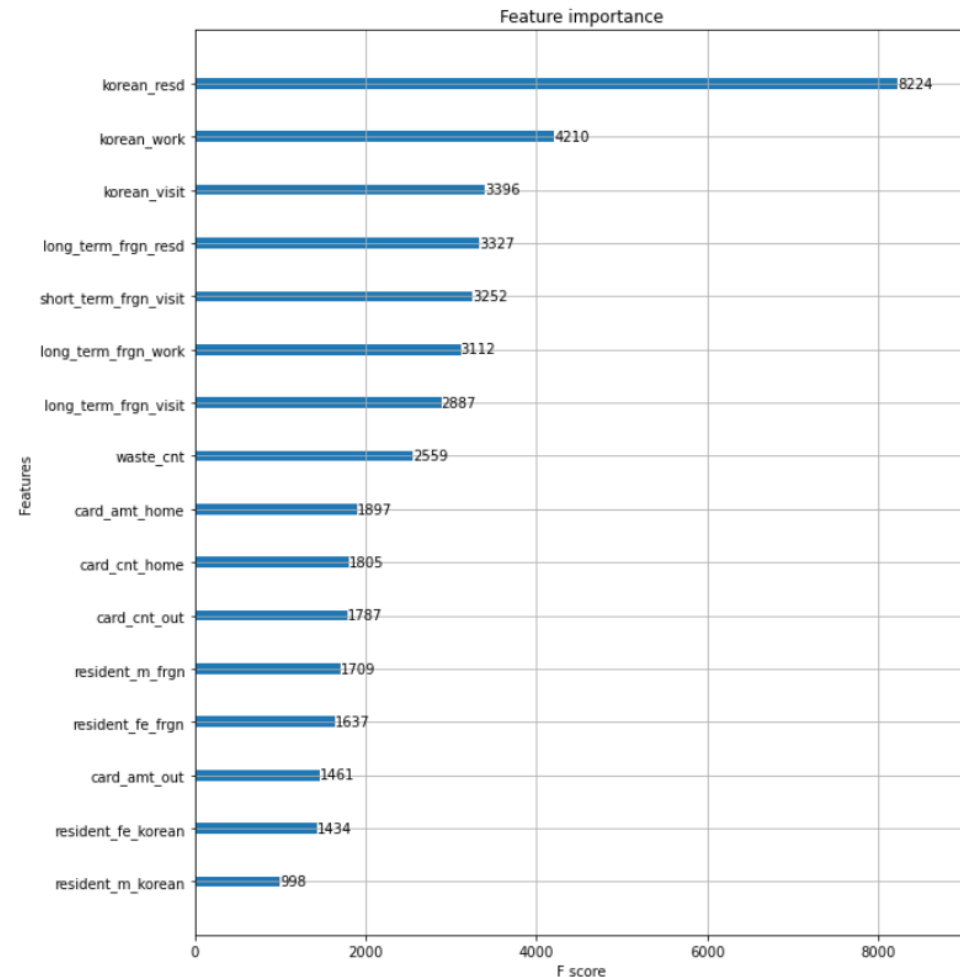
korean_work

korean_visit

- 결과 eval-rmsle = 0.08959

- ver_1(0.09862) > ver_2-1(0.08959)

- ver_2-1 선택



03 활용 알고리즘

3.3 XGBoost

2 XGBoost 통해 최적 모델 찾기

- Ver_2-2
 - 제공데이터의 변수만을 이용해 XGBoost 모델링
 - ver_2-1의 변수 중 korean_resd, korean_work, korean_visit을 각각
성별, 연령대별로 분류

03 활용 알고리즘

3.3 XGBoost

2

XGBoost 통해 최적 모델 찾기

- 설명변수 (x) :

korean_resd_ (내국인 유동인구, 거주인구)	f0 ~ f80 (여성, 0대 ~ 80대) m0 ~ m80 (남성, 0대 ~ 80대)	short_term_frqn_visit (단기체류 외국인 유동인구)	card_amt_home (결제금액, 가정)
korean_work_ (내국인 유동인구, 근무인구)	f0 ~ f80 (여성, 0대 ~ 80대) m0 ~ m80 (남성, 0대 ~ 80대)	resident_fe_korean (총 거주인구, 여성, 주민등록 거주인구)	card_cnt_out (음식 관련 카드 결제건수, 외식)
korean_visit_ (내국인 유동인구, 방문인구)	f0 ~ f80 (여성, 0대 ~ 80대) m0 ~ m80 (남성, 0대 ~ 80대)	resident_fe_frqn (총 거주인구, 여성, 외국인 거주인구)	card_amt_out (결제금액, 외식)
long_term_frqn_resd (장기체류 외국인 유동인구, 거주인구)		resident_m_korean (총 거주인구, 남성, 주민등록 거주인구)	waste_cnt (배출건수)
long_term_frqn_work (장기체류 외국인 유동인구, 근무인구)		resident_m_frqn (총 거주인구, 남성, 외국인 거주인구)	
long_term_frqn_visit (장기체류 외국인 유동인구, 방문인구)		card_cnt_home (음식 관련 카드 결제건수, 가정)	

03 활용 알고리즘

3.3 XGBoost

2

XGBoost 통해 최적 모델 찾기

- Ver_2-2

Model explanation

- Feature Importance top3

korean_resid_f0

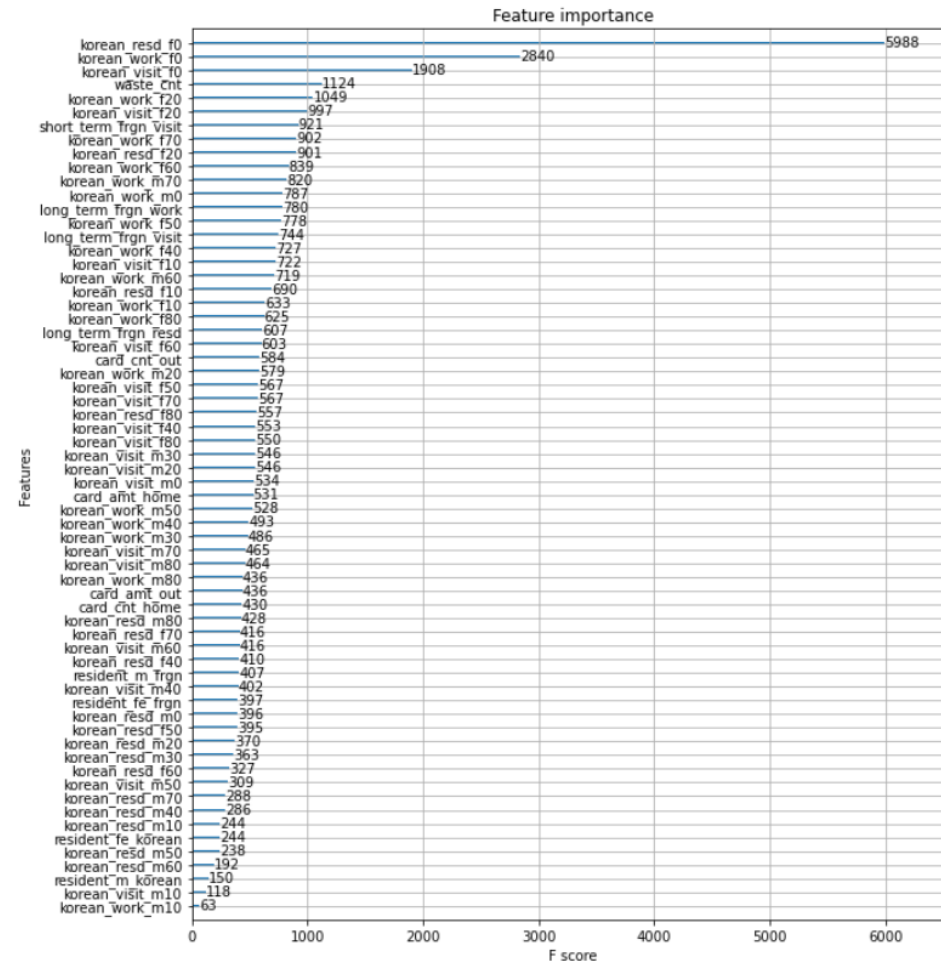
korean_work_f0

korean_visit_f0

- 결과 eval-rmsle = 0.08853

- ver_2-1(0.08959) > ver_2-2(0.08853)

- ver_2-2 선택



03 활용 알고리즘

3.3 XGBoost

2 XGBoost 통해 최적 모델 찾기

- Ver_3-1

- ver_1의 제공데이터 변수
- + 외부 데이터 total_house(총 주택 수)의 종류(단독주택, 아파트, 연립주택, 다세대주택, 비거주용 건물내 주택)
- + distancing(사회적 거리두기 단계) 사용

03 활용 알고리즘

3.3 XGBoost

2 XGBoost 통해 최적 모델 찾기

- 설명변수 (x) :

korean (내국인 유동인구)	card_amt (결제금액)	multiplex (다세대주택)
long_term_frqn (장기체류 외국인 유동인구)	waste_cnt (배출건수)	commercial_building (비거주용 건물내 주택)
short_term_frqn (단기체류 외국인 유동인구)	detached (단독주택)	distancing (사회적 거리두기 단계)
resident (총 거주인구)	apt (아파트)	
card_cnt (음식 관련 카드 결제건수)	town (연립주택)	

03 활용 알고리즘

3.3 XGBoost

2

XGBoost 통해 최적 모델 찾기

- Ver_3-1

Model explanation

- Feature Importance top3

korean

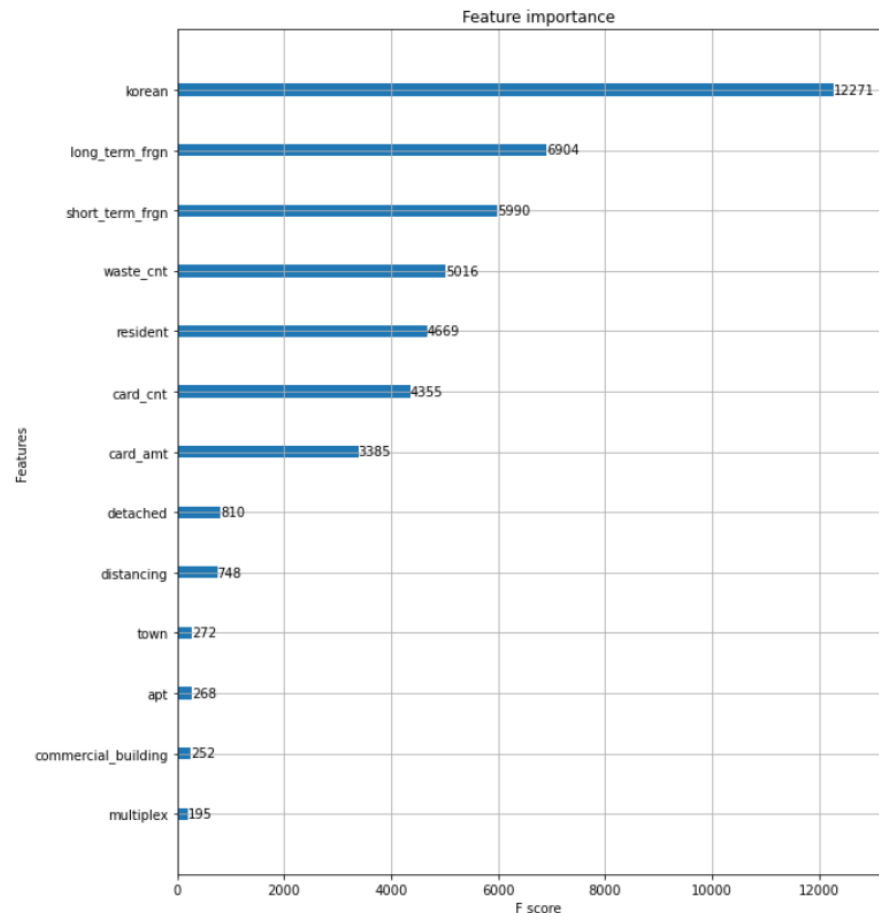
long_term_frgn

short_term_frgn

- **결과** eval-rmsle = 0.07914

- ver_3-1(0.08853) > ver_3-1(0.07914)

- ver_3-1 선택



03 활용 알고리즘

3.3 XGBoost

2 XGBoost 통해 최적 모델 찾기

- Ver_3-2

- 시계열 ARIMA 모델링 결과 short_term_frgn(단기체류 외국인 유동인구)의 2021년 7, 8월 값 중 음수(-) 다수 존재

- ver_3-1의 변수 중 short_term_frgn(단기체류 외국인 유동인구) 제거

03 활용 알고리즘

3.3 XGBoost

2 XGBoost 통해 최적 모델 찾기

- 설명변수(x) :

korean (내국인 유동인구)	detached (단독주택)
long_term_frqn (장기체류 외국인 유동인구)	apt (아파트)
resident (총 거주인구)	town (연립주택)
card_cnt (음식 관련 카드 결제건수)	multiplex (다세대주택)
card_amt (결제금액)	commercial_building (비거주용 건물내 주택)
waste_cnt (배출건수)	distancing (사회적 거리두기 단계)

03 활용 알고리즘

3.3 XGBoost

2

XGBoost 통해 최적 모델 찾기

- Ver_3-2

Model explanation

- Feature Importance top3

korean

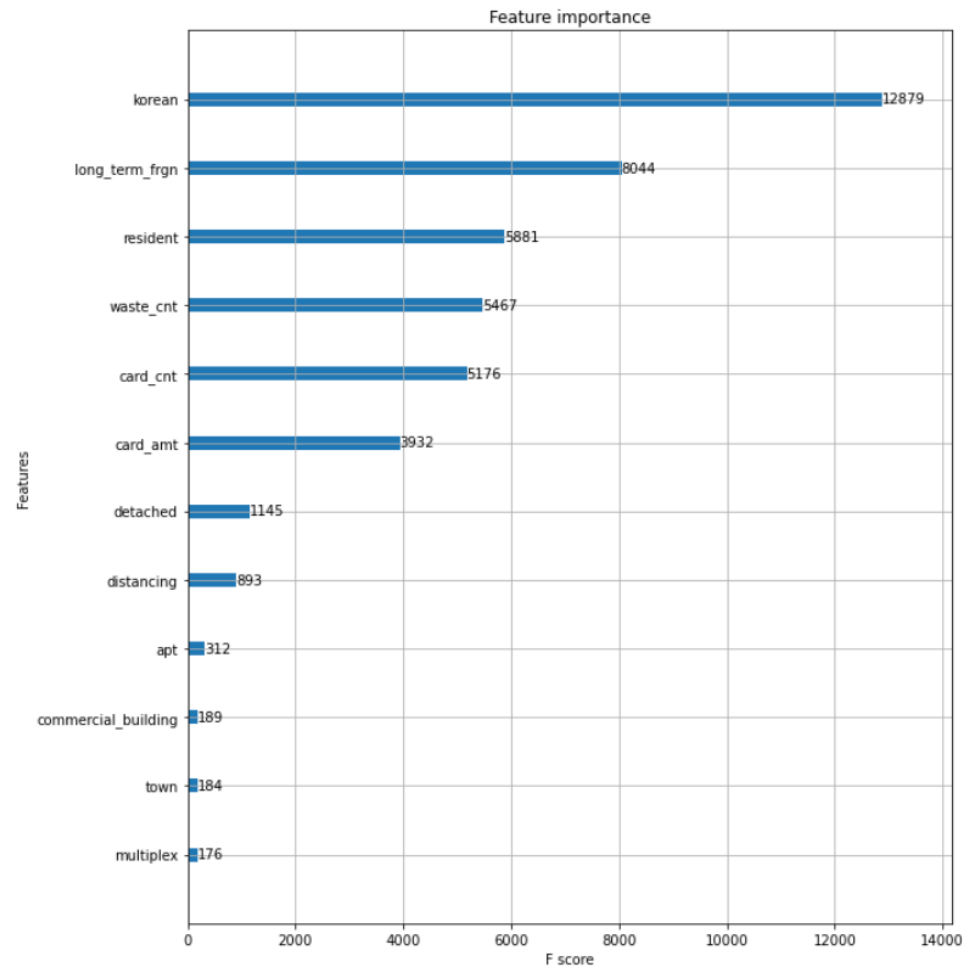
long_term_frqn

resident

- 결과 eval-rmsle = 0.07985

- ver_3-1(0.07914) < ver_3-2(0.07985)

- ver_3-1 선택



03 활용 알고리즘

3.3 XGBoost

2 XGBoost 통해 최적 모델 찾기

- Ver_3-2

- ver_3-1의 total_house(총 주택 수)의 모든 종류를 통합한 total_house만 사용
- ver_1의 제공데이터 변수 + 외부 데이터 total_house(총 주택 수) + distancing(사회적 거리두기 단계) 사용

03 활용 알고리즘

3.3 XGBoost

2 XGBoost 통해 최적 모델 찾기

- 설명변수(x) :

korean (내국인 유동인구)	card_amt (결제금액)
long_term_frgn (장기체류 외국인 유동인구)	waste_cnt (배출건수)
short_term_frgn (단기체류 외국인 유동인구)	total_house (총 주택 수)
resident (총 거주인구)	distancing (사회적 거리두기 단계)
card_cnt (음식 관련 카드 결제건수)	

03 활용 알고리즘

3.3 XGBoost

2

XGBoost 통해 최적 모델 찾기

- Ver_3-3

Model explanation

- Feature Importance top3

korean

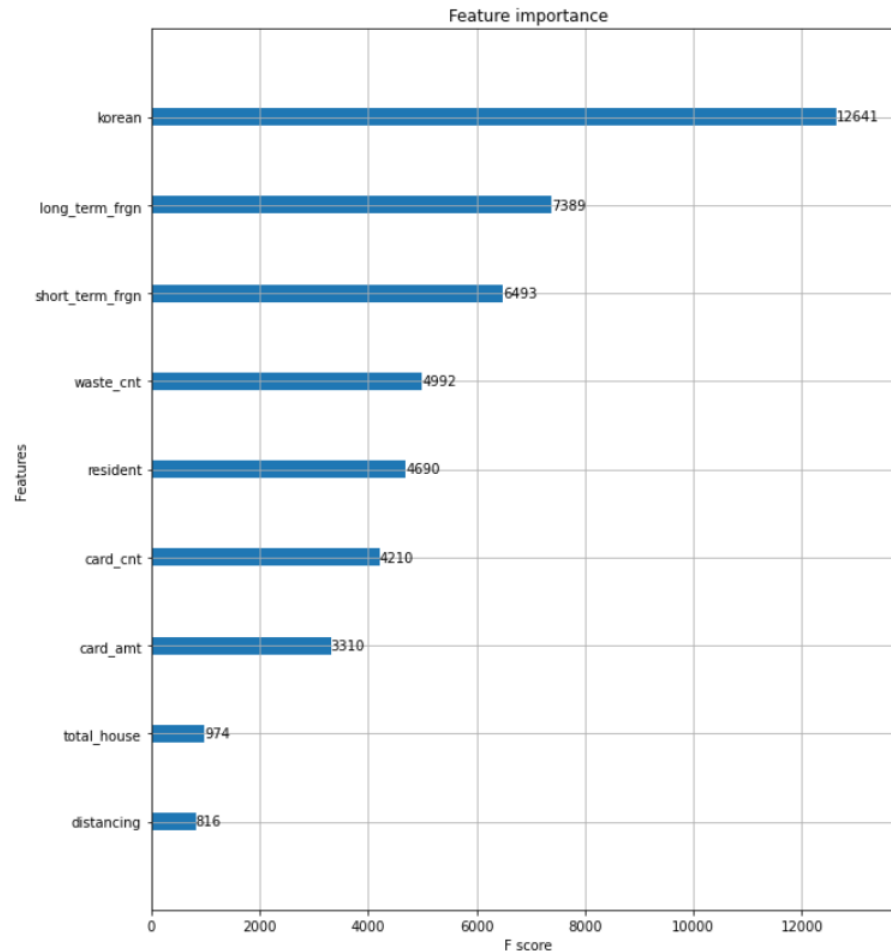
long_term_frqn

short_term_frqn

- 결과 eval-rmsle = 0.08281

- ver_3-1(0.07914) < ver_3-1(0.08281)

- ver_3-1 선택



03 활용 알고리즘

3.3 XGBoost

2 XGBoost 통해 최적 모델 찾기

- 최종 모델 선택

- 최종적으로 다음 변수를 설명변수로 이용한 XGBoost 선택

korean (내국인 유동인구)	card_amt (결제금액)	multiplex (다세대주택)
long_term_frgn (장기체류 외국인 유동인구)	waste_cnt (배출건수)	commercial_building (비거주용 건물내 주택)
short_term_frgn (단기체류 외국인 유동인구)	detached(단독주택)	distancing (사회적 거리두기 단계)
resident (총 거주인구)	Apt (아파트)	
card_cnt (음식 관련 카드 결제건수)	Town (연립주택)	

03 활용 알고리즘

3.4 ARIMA model

1 ARIMA model 의 의미와 선정 이유

1. ARIMA model 이란?

- 시계열 모형
- AR(자기상관)모형 + MA(이동평균)모형 + 추세(trend) 반영

2. ARIMA model 선정 이유

- 설명변수(x)들이 추세와 계절적 요인을 지님
- 과거 데이터를 사용해 미래 값 예측

03 활용 알고리즘

3.4 ARIMA model

1 ARIMA model 의 의미와 선정 이유

3. 예외

1) 종류별 주택 수 : detached(단독주택), apt(아파트), town(연립주택), multiplex(다세대주택), commercial_building(비거주용 건물 내 주택)

- 시계열보다 최근 데이터와 유사
- KOSIS가 제공하는 가장 최근 데이터인 2020년 데이터를 이용

2) distancing (사회적 거리두기 단계)

- 시계열 데이터X , 질병관리청에서 지정

03 활용 알고리즘

3.4 ARIMA model

2 시계열 데이터의 정상성 확인하기

- ARIMA 모델은 데이터의 정상성을 가정
- 정상성 : 시계열 데이터의 특성이 사건의 흐름에 따라 변하지 않는 것
- 우리 데이터의 대부분 변수들은 비정상적
- 전처리 필요

03 활용 알고리즘

3.4 ARIMA model

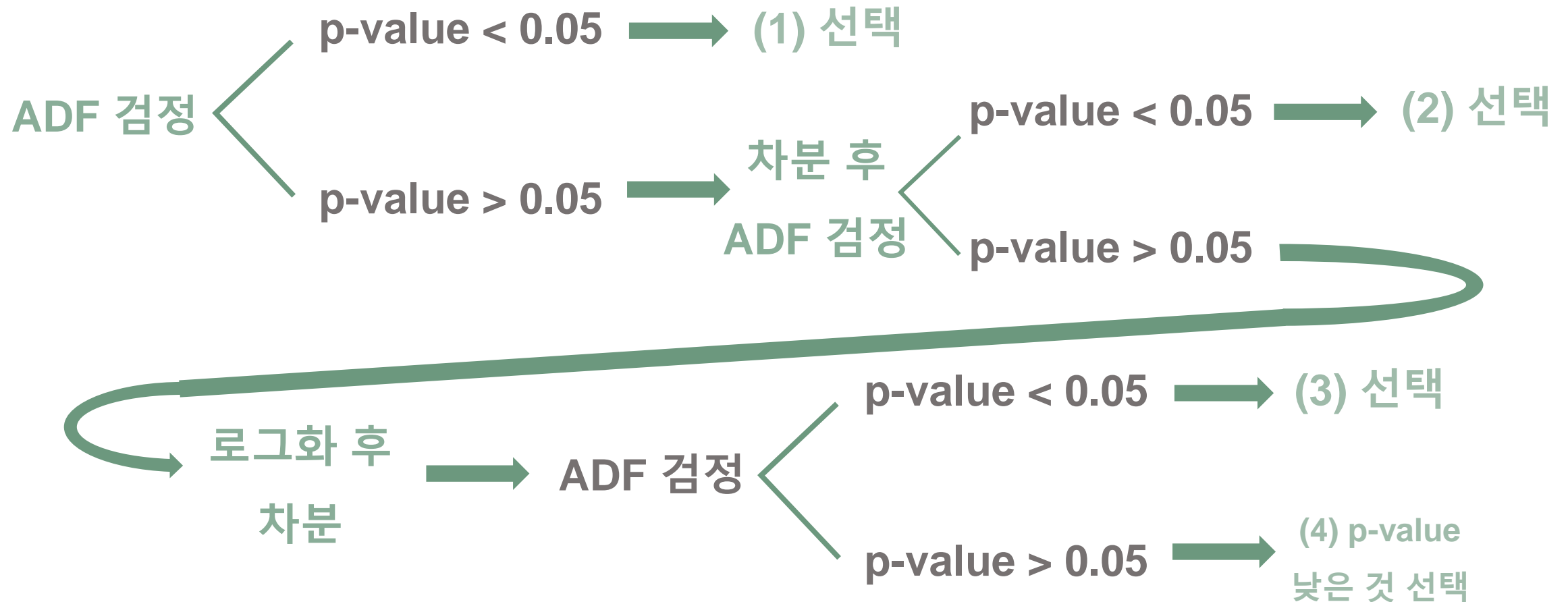
2 시계열 데이터의 정상성 확인하기

- ADF 검정을 통해 정상성 만족 여부 확인
귀무가설 : 데이터가 정상성을 만족하지 않는다
대립가설 : 데이터가 정상성을 만족한다
- 다음 페이지의 방법에 따라 정상성을 만족하는 데이터를 선택

03 활용 알고리즘

3.4 ARIMA model

2 시계열 데이터의 정상성 확인하기



03 활용 알고리즘

3.4 ARIMA model

3 최적화 ARIMA 모형 찾기

- ARIMA 모형은 3가지 파라미터
p(AR모형의 lag), d(차분 횟수),
q(MA 모형의 lag) 필요
- 자동으로 최적화된 모형을 선정해주는
auto_arima 라이브러리 사용

```
ARIMA(0,0,0)(0,0,0)[1] intercept : AIC=-64.907, Time=0.04 sec
ARIMA(0,0,1)(0,0,0)[1] intercept : AIC=-68.786, Time=0.13 sec
ARIMA(0,0,2)(0,0,0)[1] intercept : AIC=-68.664, Time=0.08 sec
ARIMA(0,0,3)(0,0,0)[1] intercept : AIC=-71.097, Time=0.16 sec
ARIMA(0,0,4)(0,0,0)[1] intercept : AIC=-69.414, Time=0.28 sec
ARIMA(0,0,5)(0,0,0)[1] intercept : AIC=-68.843, Time=0.45 sec
ARIMA(1,0,0)(0,0,0)[1] intercept : AIC=-70.618, Time=0.06 sec
ARIMA(1,0,1)(0,0,0)[1] intercept : AIC=-69.002, Time=0.25 sec
ARIMA(1,0,2)(0,0,0)[1] intercept : AIC=-67.615, Time=0.47 sec
ARIMA(1,0,3)(0,0,0)[1] intercept : AIC=-68.574, Time=0.32 sec
ARIMA(1,0,4)(0,0,0)[1] intercept : AIC=-67.488, Time=0.34 sec
ARIMA(2,0,0)(0,0,0)[1] intercept : AIC=-69.221, Time=0.13 sec
ARIMA(2,0,1)(0,0,0)[1] intercept : AIC=-67.210, Time=0.36 sec
ARIMA(2,0,2)(0,0,0)[1] intercept : AIC=-65.905, Time=0.24 sec
ARIMA(2,0,3)(0,0,0)[1] intercept : AIC=-66.456, Time=0.52 sec
ARIMA(3,0,0)(0,0,0)[1] intercept : AIC=-67.231, Time=0.26 sec
ARIMA(3,0,1)(0,0,0)[1] intercept : AIC=-65.057, Time=0.19 sec
ARIMA(3,0,2)(0,0,0)[1] intercept : AIC=-63.197, Time=0.45 sec
ARIMA(4,0,0)(0,0,0)[1] intercept : AIC=-67.376, Time=0.57 sec
ARIMA(4,0,1)(0,0,0)[1] intercept : AIC=-65.203, Time=0.17 sec
ARIMA(5,0,0)(0,0,0)[1] intercept : AIC=-65.433, Time=0.29 sec
```

```
Best model: ARIMA(0,0,3)(0,0,0)[1] intercept
Total fit time: 5.764 seconds
```

```
ARIMA(order=(0, 0, 3), scoring_args={}, seasonal_order=(0, 0, 0, 1),
      suppress_warnings=True)
```

03 활용 알고리즘

3.4 ARIMA model

3 최적화 ARIMA 모형 찾기

- 최적화 방법을 통해 선정한
최적 모델로 ARIMA 모델 피팅

ARMA Model Results						
Dep. Variable:	y	No. Observations:	42			
Model:	ARMA(0, 3)	Log Likelihood	40.548			
Method:	css-mle	S.D. of innovations	0.091			
Date:	Sat, 11 Sep 2021	AIC	-71.097			
Time:	01:37:08	BIC	-62.408			
Sample:	0	HQIC	-67.912			
	coef	std err	z	P> z	[0.025	0.975]
const	16.6630	0.028	585.399	0.000	16.607	16.719
ma.L1.y	0.4970	0.147	3.373	0.001	0.208	0.786
ma.L2.y	0.1383	0.168	0.822	0.411	-0.192	0.468
ma.L3.y	0.4401	0.132	3.322	0.001	0.180	0.700
Roots						
	Real	Imaginary	Modulus	Frequency		
MA.1	-1.1192	-0.0000j	1.1192	-0.5000		
MA.2	0.4025	-1.3668j	1.4248	-0.2044		
MA.3	0.4025	+1.3668j	1.4248	0.2044		

03 활용 알고리즘

3.4 ARIMA model

4 각 변수의 2021년 7, 8월 값 예측하기

```
In [10]: 1 # 2단위 이후의 예측결과
          2 fore = model_fit.forecast(steps=2)
          3 print(fore)

(array([16.72822029, 16.66296014]), array([0.09093318, 0.10154359]), array([[16.54999453, 16.90644605],
                                     [16.46393836, 16.86198191]]))
```

```
In [11]: 1 print( np.exp(16.72822029) - 1 )
          2 print( np.exp(16.66296014) - 1 )
```

```
18406607.02683503
17243745.94875778
```

- ARIMA 모델로 2021년 7, 8월 값 예측하기

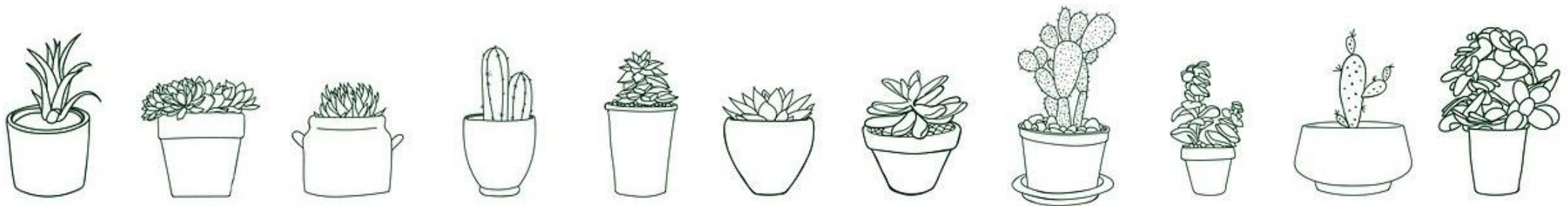
03 활용 알고리즘

3.4 ARIMA model

5 최종 예측값

Index	korean	long_term_frgn	short_term_frgn	resident	card_amt	card_cnt	waste_cnt
1	8684899.70	469672.43	387.93	24457.77	10681271.81	292.31	23313.07
2	8084718.33	487440.03	0	24460.11	10741037.02	294.14	29166.39
3	18406607.03	767679.83	24078.52	38348.80	19944449.09	612.17	43546.58
4	17243745.95	738378.19	20402.22	38397.63	20061968.59	616.38	44514.03
5	7144596.61	171265.97	564.44	15952.53	6808693.19	201.47	14990.38
...

04 최종예측 결과 ♡



04 최종예측 결과

4.1 행정동별 예측 결과

korean (내국인 유동인구)	detached (단독주택)
long_term_frgn (장기체류 외국인 유동인구)	apt (아파트)
short_term_frgn (단기체류 외국인 유동인구)	town (연립주택)
resident (총 거주인구)	multiplex (다세대주택)
card_cnt (음식관련 카드 결제건수)	commercial_building (비거주용 건물 내 주택)
card_amt (음식관련 카드 결제금액)	distancing (사회적 거리두기 단계)
waste_cnt (음식물 쓰레기 배출건수)	

- XGBoost ver3_1의 모델 채택
- 학습용 : 18.01. ~ 21.06.
테스트용 : 21.07. ~ 21.08.
- 이 때 테스트용 데이터의
X_feature 값들은
ARIMA 모형으로 예측한 값임

04 최종예측 결과

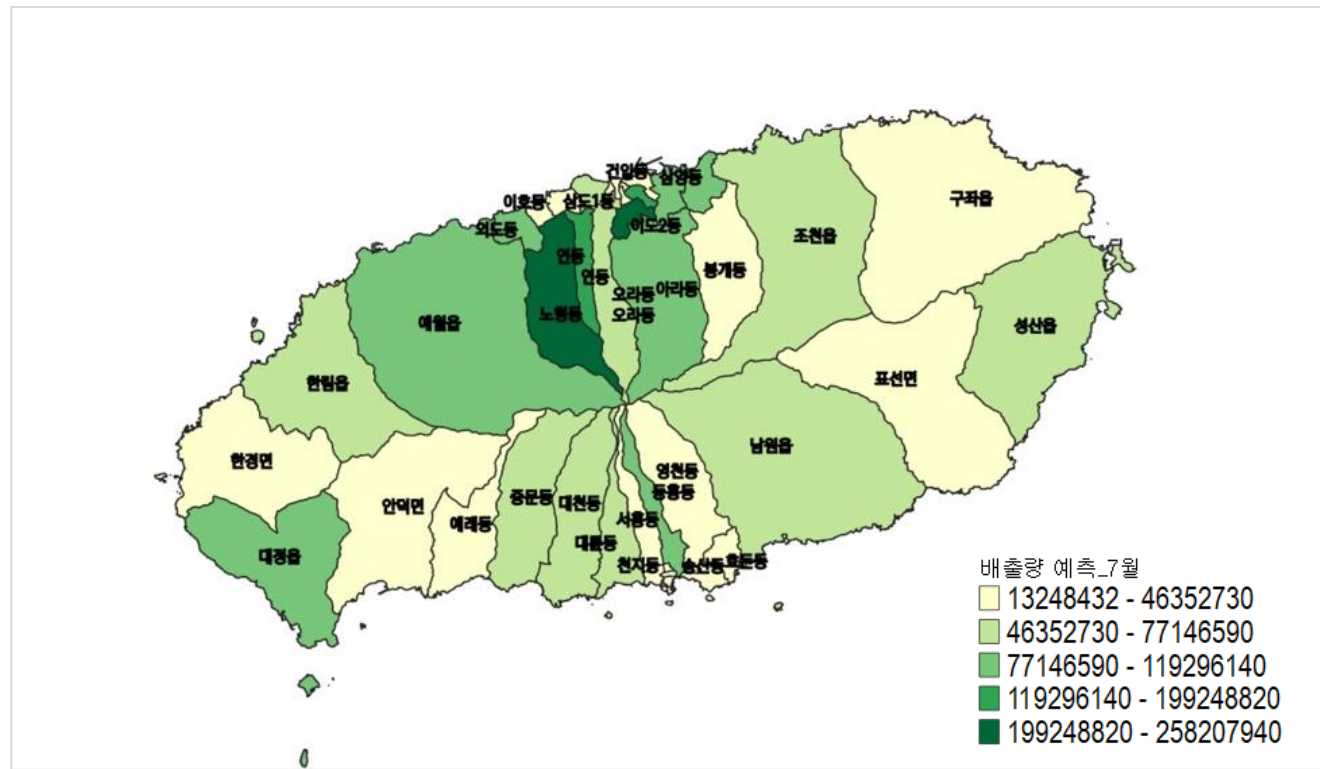
4.1 행정동별 예측 결과

emd_cd \ base_date	2021-07-31	2021-08-31
50110250	76529910.0	9.700360e+07
50110253	107665100.0	1.074393e+08
50110256	46352730.0	4.719103e+07
50110259	75931544.0	8.967203e+07
50110310	29249542.0	2.863940e+07
...

04 최종예측 결과

4.1 행정동별 예측 결과

- 제주도 음식물쓰레기 배출량 최종 예측
- 7월 예측 배출량 읍면동별 시각화



04 최종예측 결과

4.1 행정동별 예측 결과

- 제주도 음식물쓰레기 배출량 최종 예측

- Qgis 앱을 사용해 배출량을 5단계로 나눠 단계 구분 시각화 수행
- 예측 배출량(g) top5 읍면동 :

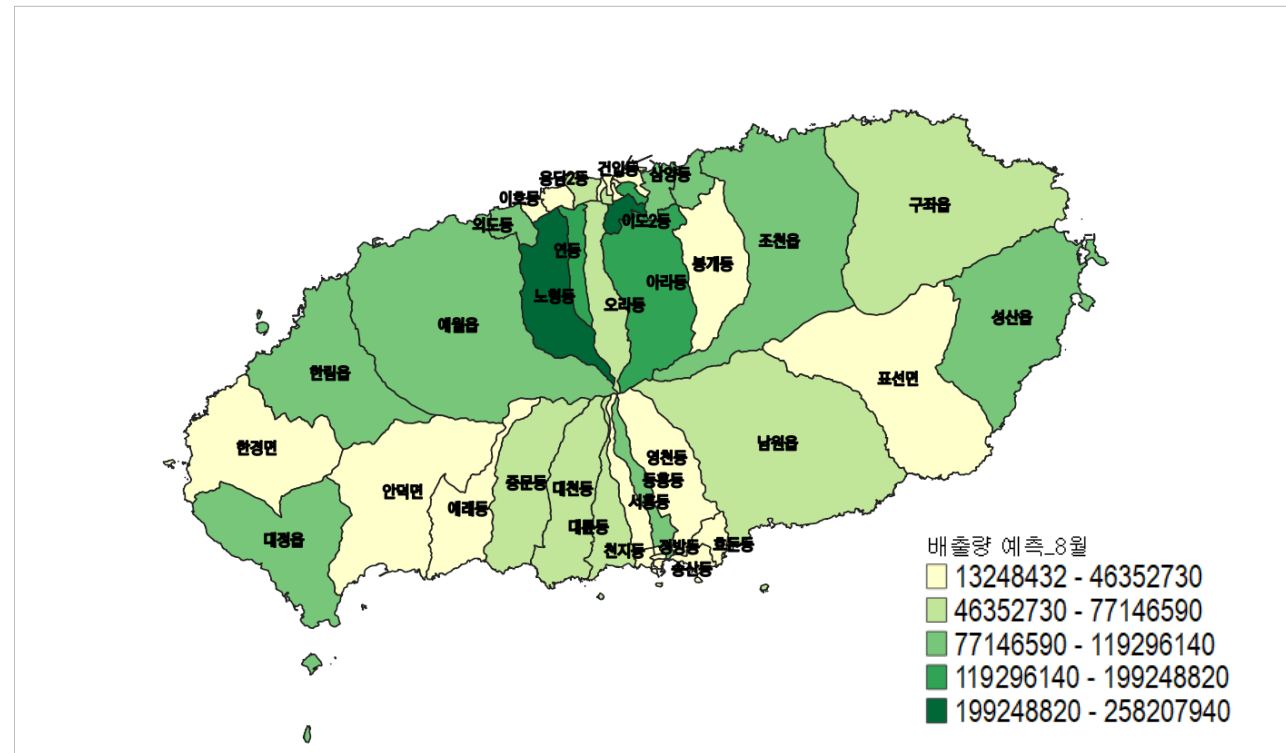
노형동	258207940
이도2동	239246770
연동	199248820
일도2동	165704100
아라동	119296140

→ 노형동, 연동, 이도 2동 등 제주도 시가지 지역을 중심으로 배출량이 많을 것으로 예측

04 최종예측 결과

4.1 행정동별 예측 결과

- 제주도 음식물쓰레기 배출량 최종 예측
- 8월 예측 배출량 읍면동별 시각화



04 최종예측 결과

4.1 행정동별 예측 결과

- 제주도 음식물쓰레기 배출량 최종 예측

- 예측 배출량(g) top5 읍면동 :

노형동	257021470
이도2동	235867870
연동	193611630
일도2동	166047710
아라동	132510850

→ 7월 배출량에 비해 전체적으로 배출량이 증가할 것으로 예측

→ 배출량이 상대적으로 많은 읍면동이 7월과 유사할 것으로 예측

04 최종예측 결과

4.2 행정동 “알 수 없음” 데이터 예측 결과

- 행정동 “알 수 없음” 데이터 : 음식물 쓰레기 배출량, 카드소비량

1 음식물 쓰레기 배출량과 카드결제금액, 카드결제건수 상관관계 파악

	em_g	use_cnt	use_amt
em_g	1.000	-0.112	0.049
use_cnt	-0.112	1.000	0.925
use_amt	0.049	0.925	1.000

- Pearson 상관계수 사용

04 최종예측 결과

4.2 행정동 “알 수 없음” 데이터 예측 결과

2 음식물 쓰레기 배출량과 카드결제금액, 카드결제건수 회귀분석

OLS Regression Results

Dep. Variable:	em_g	R-squared (uncentered):	0.403			
Model:	OLS	Adj. R-squared (uncentered):	0.388			
Method:	Least Squares	F-statistic:	27.66			
Date:	Sat, 11 Sep 2021	Prob (F-statistic):	4.86e-06			
Time:	19:55:13	Log-Likelihood:	-700.24			
No. Observations:	42	AIC:	1402.			
Df Residuals:	41	BIC:	1404.			
Df Model:	1					
Covariance Type:	nonrobust					
	coef	std err	t	P> t	[0.025	0.975]
use_cnt	561.7619	106.821	5.259	0.000	346.033	777.490
Omnibus:	8.807	Durbin-Watson:	0.090			
Prob(Omnibus):	0.012	Jarque-Bera (JB):	6.334			
Skew:	0.815	Prob(JB):	0.0421			
Kurtosis:	2.019	Cond. No.	1.00			

1) 음식물 쓰레기 배출량 & 카드결제건수 회귀분석 결과

- 결정계수 $R^2 = 0.403$
수정결정계수 $R^2 = 0.388$
- waste_amt 변동의 40.30%만이 card_cnt 변동에 의해 설명됨을 의미
- rain과 card_cnt 사이의 회귀식의 정확도는 매우 낮으므로 설명변수 card_cnt로서의 역할을 하기 어렵다고 판단

04 최종예측 결과

4.2 행정동 “알 수 없음” 데이터 예측 결과

2 음식물 쓰레기 배출량과 카드결제금액, 카드결제건수 회귀분석

OLS Regression Results

Dep. Variable:	em_g	R-squared (uncentered):	0.430			
Model:	OLS	Adj. R-squared (uncentered):	0.416			
Method:	Least Squares	F-statistic:	30.88			
Date:	Sat, 11 Sep 2021	Prob (F-statistic):	1.84e-06			
Time:	19:55:45	Log-Likelihood:	-699.28			
No. Observations:	42	AIC:	1401.			
Df Residuals:	41	BIC:	1402.			
Df Model:	1					
Covariance Type:	nonrobust					
	coef	std err	t	P> t	[0.025	0.975]
use_amt	0.0170	0.003	5.557	0.000	0.011	0.023
Omnibus:	8.069	Durbin-Watson:	0.101			
Prob(Omnibus):	0.018	Jarque-Bera (JB):	6.274			
Skew:	0.828	Prob(JB):	0.0434			
Kurtosis:	2.081	Cond. No.	1.00			

2) 음식물 쓰레기 배출량 & 카드결제금액 회귀분석 결과

- 결정계수 $R^2 = 0.430$
수정결정계수 $R^2 = 0.416$
- waste_amt 변동의 43.0%만이 card_amt 변동에 의해 설명됨을 의미
- rain과 card_amt 사이의 회귀식의 정확도는 매우 낮으므로 설명변수 card_amt로서의 역할을 하기 어렵다고 판단

04 최종예측 결과

4.2 행정동 “알 수 없음” 데이터 예측 결과

2 음식물 쓰레기 배출량과 카드결제금액, 카드결제건수 회귀분석

결론

“알 수 없음 ” 으로 분류된 음식물 쓰레기 배출량 데이터에 대해
카드소비량으로 예측하기엔 부적절하다고 판단

04 최종예측 결과

4.2 행정동 “알 수 없음” 데이터 예측 결과

3 음식물 쓰레기 배출량 시계열분석(ARIMA 모델)

- 적당히 예측할 수 있는 변수가 부족해 “알 수 없음”으로 분류된 음식물 쓰레기 배출량 데이터만 가지고 ARIMA 모델로 예측

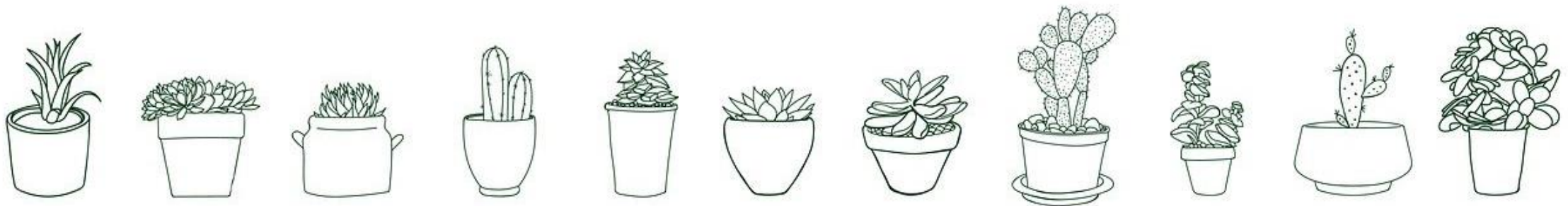
```
In [14]: 1 # 2단위 이후의 예측결과
          2 fore = model_fit.forecast(steps=2)
          3 print(fore)

(array([13.95230321, 13.46703176]), array([4.02614459, 5.07513538]), array([[ 6.06120482, 21.84340161],
[ 3.51994919, 23.41411433]]))

In [16]: 1 print( np.exp(13.95230321) - 1 )
          2 print( np.exp(13.46703176) - 1 )

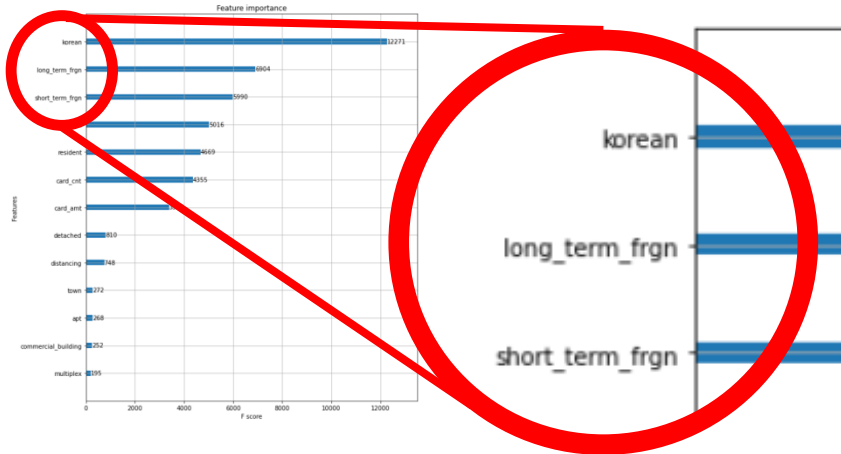
1146589.3806896203
705759.8784326583
```

05 결론 ♡



05 결론

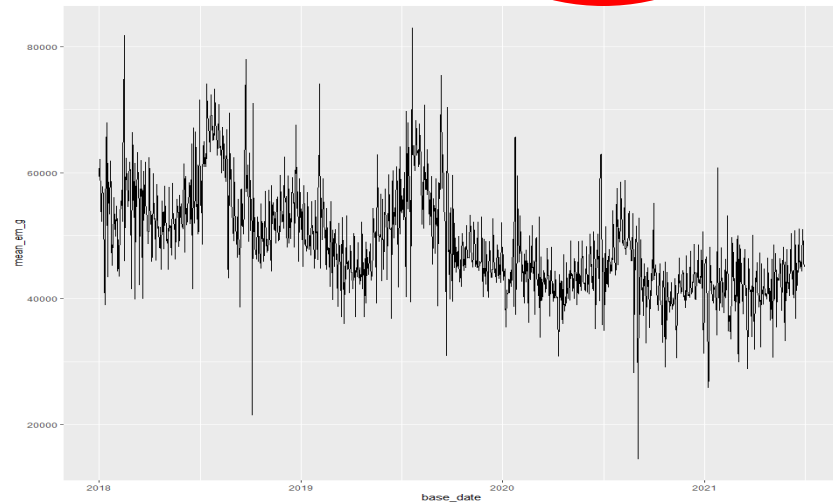
5.1 분석결과 활용 및 시사점



- 유동인구가 주요요인 Top3
- 2020년에 음식물 쓰레기 배출량 감소

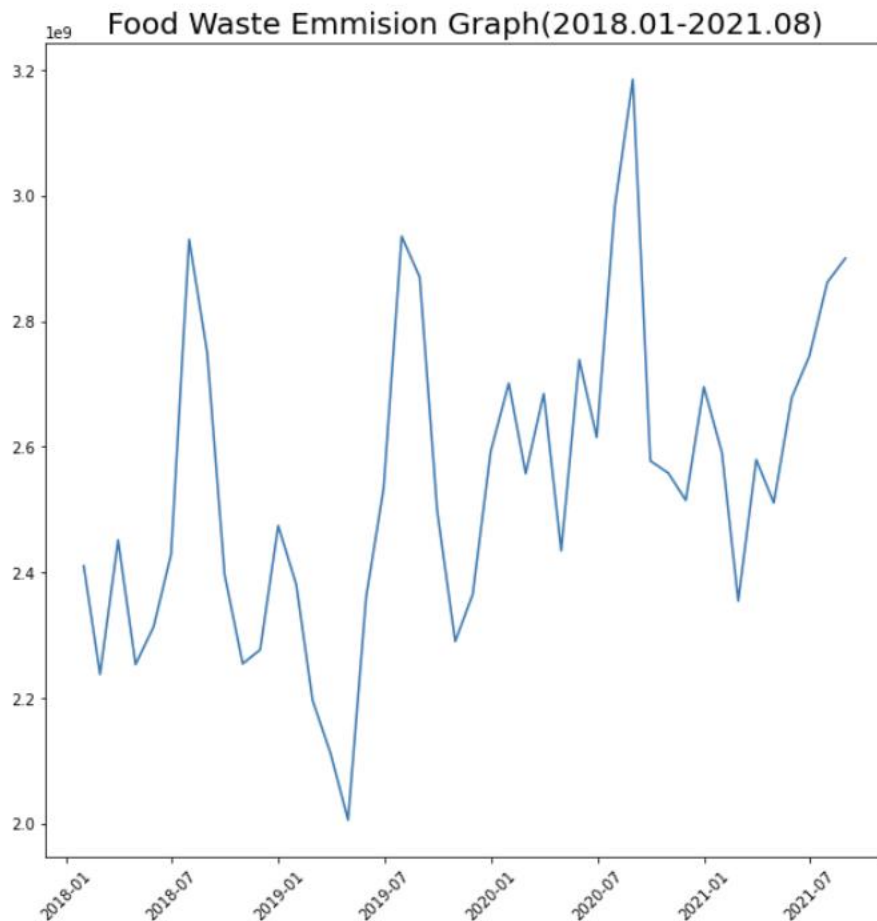


코로나로 인한 여행, 이동이 축소되면서
음식물 쓰레기 배출량이 줄어든 것 !



05 결론

5.1 분석결과 활용 및 시사점



- 특히, 예측한 7, 8월은 배출량이 높음 :
관광 성수기
- 코로나 이후 관광이 활성화되면 다시
음식물 쓰레기 배출량 증가할 것



관광객을 대상으로 한 대책 필요

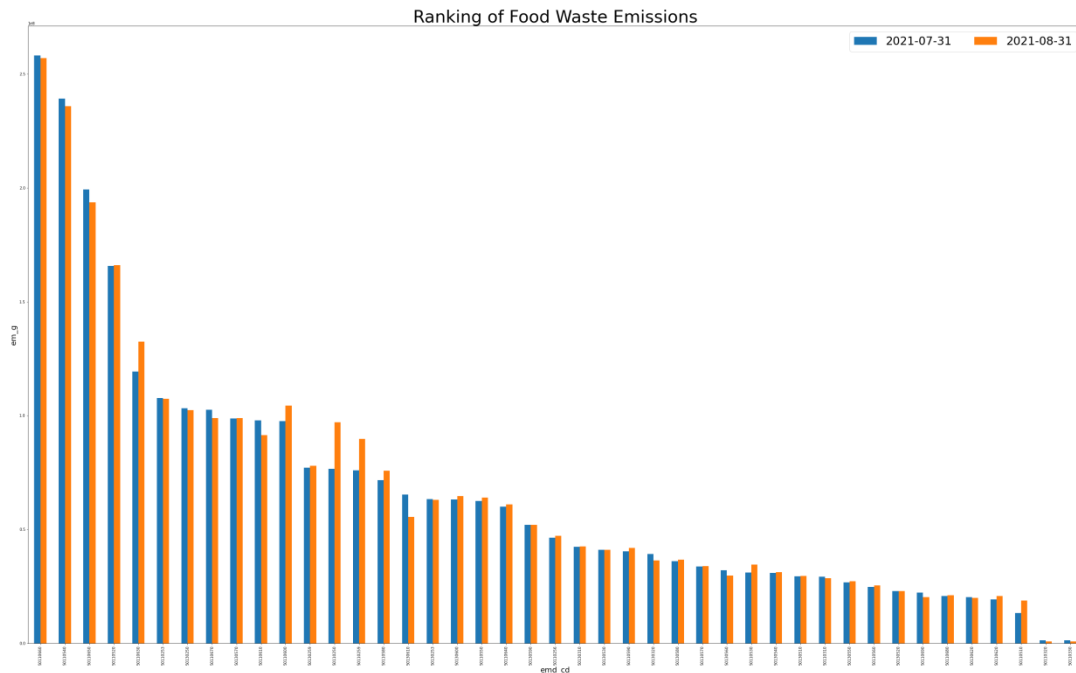
05 결론

5.1 분석결과 활용 및 시사점

- 관광지 주변 음식점의 음식물 쓰레기 배출량 감소를 위한 방안
 - 관광지 곳곳에 음식물 쓰레기 배출에 경각심을 가질 수 있는 문구 부착
 - 외국인 관광객을 위한 다양한 언어의 경고 문구 부착
 - 음식 남김 여부에 따른 이벤트 진행 / 환경부담금 부과

05 결론

5.1 분석결과 활용 및 시사점



- 배출량 자체는 도민들이 많이 사는 시가지 지역이 가장 높음
- 관광객 뿐만 아니라 도민들의 노력도 필요함을 공익광고를 통해 알리기

감사합니다 ☺

팀명 : 에코탐라

팀장 : 김소은(trit1268@naver.com)

팀원 : 김서린(tjfls96@naver.com), 신정아(alalwjddk2@naver.com), 이윤지(pioneer0305@naver.com)

