DACON

주차수요 예측 경진 대회 🚙

gs 동물 (김동화, 이지은, 안현우, 이응진)

목차

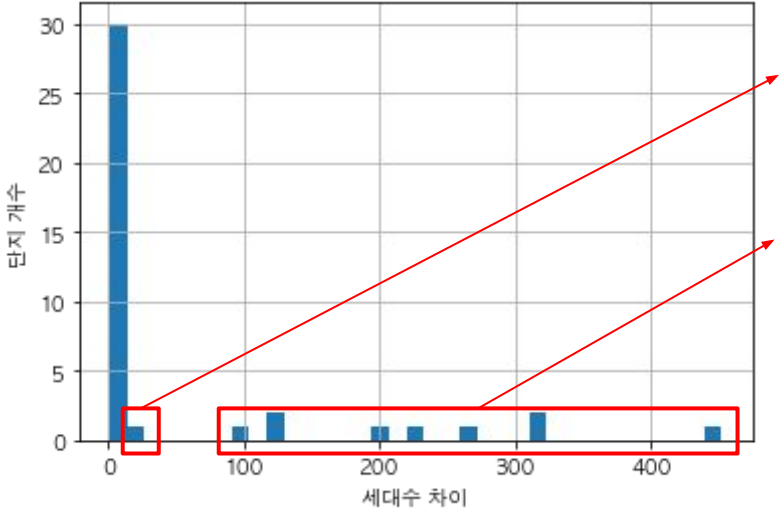
PART I.

데이터 전처리

1. 에러 처리

1) ERROR 1 처리

- 문제 : 총세대수 **!=** 전용면적별 세대수의 총합
- 원인 : 차이는 분양상가 또는 분양아파트가 임대단지 안에 혼합되어 발생한 것
- 분포 : 차이가 14세대 이하인 48개 단지는 **분양상가**, 차이가 94~452세대에 분포하는 10개 단지는 **분양아파트** 혼합단지인 것으로 확인
- 처리 방법 : 차이가 14세대 이하인 단지는 **'분양상가'**로, 차이가 94~452세대인 10개 단지는 **'분양아파트'**로, 나머지는 **'나머지'**로 라벨링을 한 칼럼을 하나 만든다.



분양상가

※ 차이가 94~452세대인 10개 단지 : ['C1490', 'C2497', 'C2620', 'C1344', 'C1024', 'C2470', 'C1206', 'C1740', 'C2405', 'C1804']

분양아파트

※ 별첨 : 차이가 14세대 이하인 48개 단지 - ['C1925', 'C1312', 'C2013', 'C1424', 'C2520', 'C2319', 'C1850', 'C1068', 'C2644', 'C2156', C2453', 'C1910', 'C2139', 'C2508', 'C1695', 'C2556', 'C2362', 'C2568', 'C2245', 'C2549', 'C1584', 'C2298', 'C2225', 'C1218', 'C1970', C1732', 'C2433', 'C1894', 'C1156', 'C2142', 'C2186', 'C2411', 'C1812', 'C1030', 'C1749', 'C1349', 'C2043', 'C1229', 'C2363', 'C1414', C2174', 'C2404', 'C1683', 'C1038', 'C2456', 'C1266', 'C1267', 'C2189']

나머지

1. 에러 처리

2) ERROR 2 처리

- 문제 : 동일한 단지임에도 불구하고 단지코드가 2개 부여됨
- 처리 : 등록차량대수 수정해주고, 단지코드를 'N-'형식으로 변경
- 특이사항 : ERROR2에 해당하는 데이터를 삭제할 경우 에러 데이터가 가지고 있는 단지코드 이외의 피처 레이블들을 사용할 수 없기 때문에 삭제하지 않고 그대로 둠.

단지코드	총세대수		주차면수	등록차량 대수
	코드별	합계		
C2085	969	1339	1590	1595
C1397	370			

단지코드	총세대수		주차면수	등록차량대수	
	코드별	합계		코드별	합계
C2431	472	1047	1066	855	1214
C1649	575			359	

단지코드	총세대수		주차면수	등록차량 대수
	코드별	합계		
C1036	742	1254	1016	1279
C2675	512			

< 코드 예시

'C2085', 'C1397' 처리

```
# 총세대수 처리
train.loc[train['단지코드']=='C2085', "총세대수"] = 1339
train.loc[train['단지코드']=='C1397', "총세대수"] = 1339

# 'C2085', 'C1397' shape 확인
print( train.loc[train['단지코드']=='C2085',:].shape )
print( train.loc[train['단지코드']=='C1397',:].shape )

# 단지코드 처리
train.loc[train['단지코드']=='C2085', "단지코드"] = 'N2085'
train.loc[train['단지코드']=='C1397', "단지코드"] = 'N2085'

# 처리 후 확인
print(train[train['단지코드']=='N2085'].shape)

train[train['단지코드']=='N2085']
```

```
(8, 16)
(6, 16)
(14, 16)
```

1. 에러 처리

3) ERROR 3 처리

- 문제 : 단지코드 등 기입 실수로 데이터 정제 과정에서 매칭 오류 발생
- 처리 : ERROR3에 해당하는 단지코드를 'ERR04-' 형식으로 변경
- 특이사항 : ERROR3에 해당하는 데이터를 삭제할 경우 에러 데이터가 가지고 있는 단지코드 이외의 피처 레이블들을 사용할 수 없기 때문에 삭제하지 않고 그대로 둠.

※ 실수가 발생한 단지 코드 (9개 단지) :
['C2335', 'C1327"C1095', 'C2051', 'C1218', 'C1894', 'C2483', 'C1502', 'C1988']

cf. C2335', 'C1327'는 '테스트 셋 제외 데이터'에 포함되어 있어서 이미 제거됨

< 코드 예시 >

```
# 처리할 데이터의 갯수 확인
print(train[(train['단지코드']=='C1095')|
      (train['단지코드']=='C2051')|
      (train['단지코드']=='C1218')|
      (train['단지코드']=='C1894')|
      (train['단지코드']=='C2483')|
      (train['단지코드']=='C1502')|
      (train['단지코드']=='C1988')
     ].shape)

# 해당 데이터의 단지코드를 'ERR04' 형식으로 처리
train.loc[ train['단지코드']=='C1095',  "단지코드" ] = 'ERR04_1095'
train.loc[ train['단지코드']=='C2051',  "단지코드" ] = 'ERR04_2051'
train.loc[ train['단지코드']=='C1218',  "단지코드" ] = 'ERR04_1218'
train.loc[ train['단지코드']=='C1894',  "단지코드" ] = 'ERR04_1894'
train.loc[ train['단지코드']=='C2483',  "단지코드" ] = 'ERR04_2483'
train.loc[ train['단지코드']=='C1502',  "단지코드" ] = 'ERR04_1502'
train.loc[ train['단지코드']=='C1988',  "단지코드" ] = 'ERR04_1988'

# 처리 후 갯수 확인
print(train.loc[ train['단지코드'].str.contains('ERR'), :].shape)
train.loc[ train['단지코드'].str.contains('ERR'), :]
```

```
(56, 16)
(56, 16)
```

2. 피처 처리

1) test셋 자격유형/train셋 10분내버스정류장수 결측치 처리

- test셋 자격유형 결측치 → 동일한 단지코드의 자격유형을 따름

<코드 예시>

```
print("처리 전 결측치 개수 : ", test['자격유형'].isnull().sum())

test.loc[196, "자격유형"] = "A"
test.loc[258, "자격유형"] = "C"

print("처리 후 결측치 개수 : ", test['자격유형'].isnull().sum())
```

```
처리 전 결측치 개수 :  2
처리 후 결측치 개수 :  0
```

- train셋 10분내버스정류장수 결측치 → 동일한 단지코드의 버스정류장수를 따름

<코드 예시>

```
print("처리 전 결측치 개수 : ", train['버스정류장'].isnull().sum())

null_idx = train[train['버스정류장'].isnull()].index
train.loc[null_idx,"버스정류장"] = 2

print("처리 후 결측치 개수 : ", train['버스정류장'].isnull().sum())
```

```
처리 전 결측치 개수 :  4
처리 후 결측치 개수 :  0
```

2. 피처 처리

2) 범주형 피처* 라벨인코딩 + 원핫인코딩

*범주형 피처 : '임대건물구분','지역',"공급유형","자격유형",'ERROR1'

LabelEncoder, OneHotEncoder를 이용하여 범주형 피처를 인코딩

임대건물구분_lb	상가	아파트	지역_lb	강원도	경기도	경상남도	경상북도	광주광역시	대구광역시	대전광역시	부산광역시	서울특별시	세종특별자치시	울산광역시	전라남도	전라북도	제주특별자치도	충청남도	충청북도	공급유형_lb	공공분양	공공임대(10년)	공공임대(50년)	공공임대(5년)	공공임대(분납)	국민임대	영구임대	임대상가	장기전세	행복주택	자격유형_lb
1	0	1	3	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	5	0	0	0	0	0	1	0	0	0	0	0
1	0	1	3	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	5	0	0	0	0	0	1	0	0	0	0	0

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	ERROR1_lb	나머지	분양상가	분양아파트
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0		0	1	0	0
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0		0	1	0	0

{0: '상가', 1: '아파트'}
{0: '강원도', 1: '경기도', 2: '경상남도', 3: '경상북도', 4: '광주광역시', 5: '대구광역시', 6: '대전광역시', 7: '부산광역시', 8: '서울특별시', 9: '세종특별자치시', 10: '울산광역시', 11: '전라남도', 12: '전라북도', 13: '제주특별자치도', 14: '충청남도', 15: '충청북도'}
{0: '공공분양', 1: '공공임대(10년)', 2: '공공임대(50년)', 3: '공공임대(5년)', 4: '공공임대(분납)', 5: '국민임대', 6: '영구임대', 7: '임대상가', 8: '장기전세', 9: '행복주택'}
{0: 'A', 1: 'B', 2: 'C', 3: 'D', 4: 'E', 5: 'F', 6: 'G', 7: 'H', 8: 'I', 9: 'J', 10: 'K', 11: 'L', 12: 'M', 13: 'N', 14: 'O'}
{0: '나머지', 1: '분양상가', 2: '분양아파트'}

PART II.

임대보증금/임대료 결측치 처리

| 질문1. 임대보증금/임대료 결측치 처리를 어떻게 하면 좋을까? 🤔

```
train.isnull().sum()
```

```
단지코드                          0
총세대수                          0
임대건물구분                        0
지역                            0
공급유형                          0
전용면적                          0
전용면적별세대수                      0
공가수                           0
자격유형                          0
임대보증금                       569
임대료                         569
도보 10분거리 내 지하철역 수(환승노선 수 반영)    211
도보 10분거리 내 버스정류장 수             4
단지내주차면수                       0
등록차량수                         0
dtype: int64
```

```
test.isnull().sum()
```

```
단지코드                          0
총세대수                          0
임대건물구분                        0
지역                            0
공급유형                          0
전용면적                          0
전용면적별세대수                      0
공가수                           0
자격유형                          2
임대보증금                       180
임대료                         180
도보 10분거리 내 지하철역 수(환승노선 수 반영)     42
도보 10분거리 내 버스정류장 수             0
단지내주차면수                       0
dtype: int64
```

아이디어 : 통계치(중앙값, 평균)으로 결측치를 채울까?
문제 : 통계치(중앙값, 평균)으로 결측치를 채우기에는 결측치가 너무 많다.

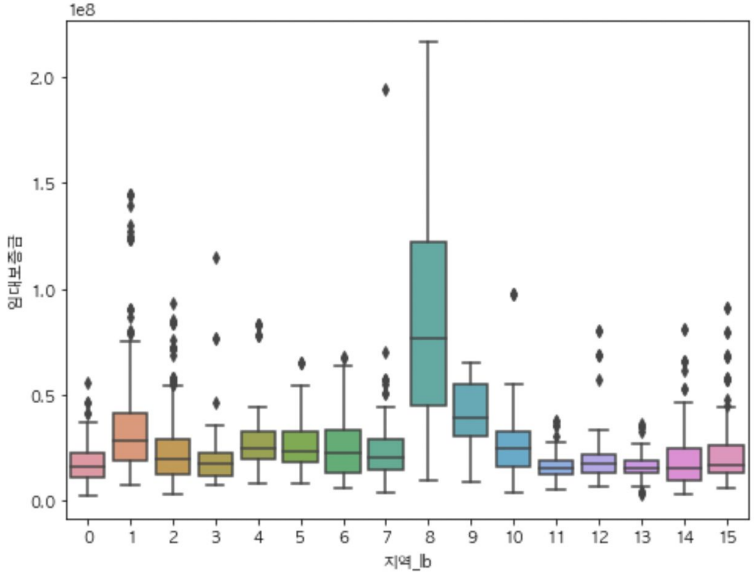
→ 임대보증금과 임대료를 예측하는 모델을 만들자! 👩‍💻💡

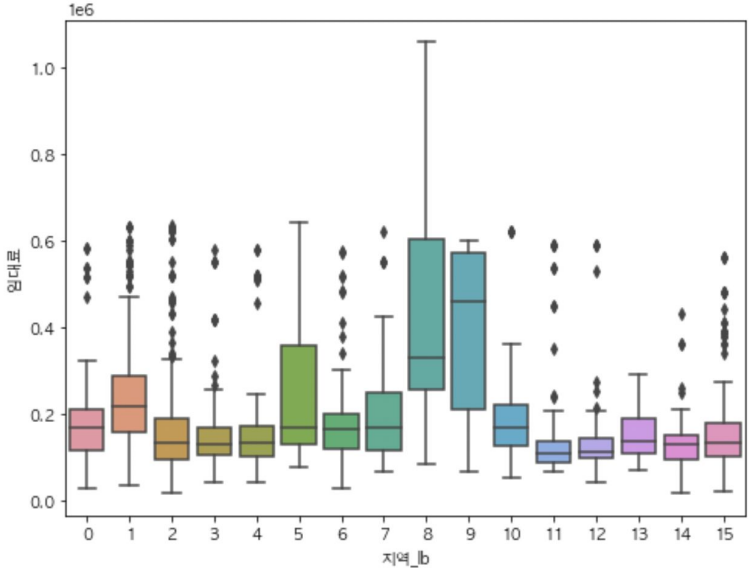
임대보증금/임대료 관련 EDA

피처에 따라 임대보증금과 임대료가 어떻게 차이나는지 확인하고 만약 구분될만한 결과가 보이면 해당 피처를 임대보증금과 임대료를 예측하는 모델을 학습하는데 사용하려고 한다.

| 질문2. 임대보증금/임대료와 관련있을 것 같은 피처는 무엇일까? 🤔

1. 지역에 따른 임대보증금/임대료

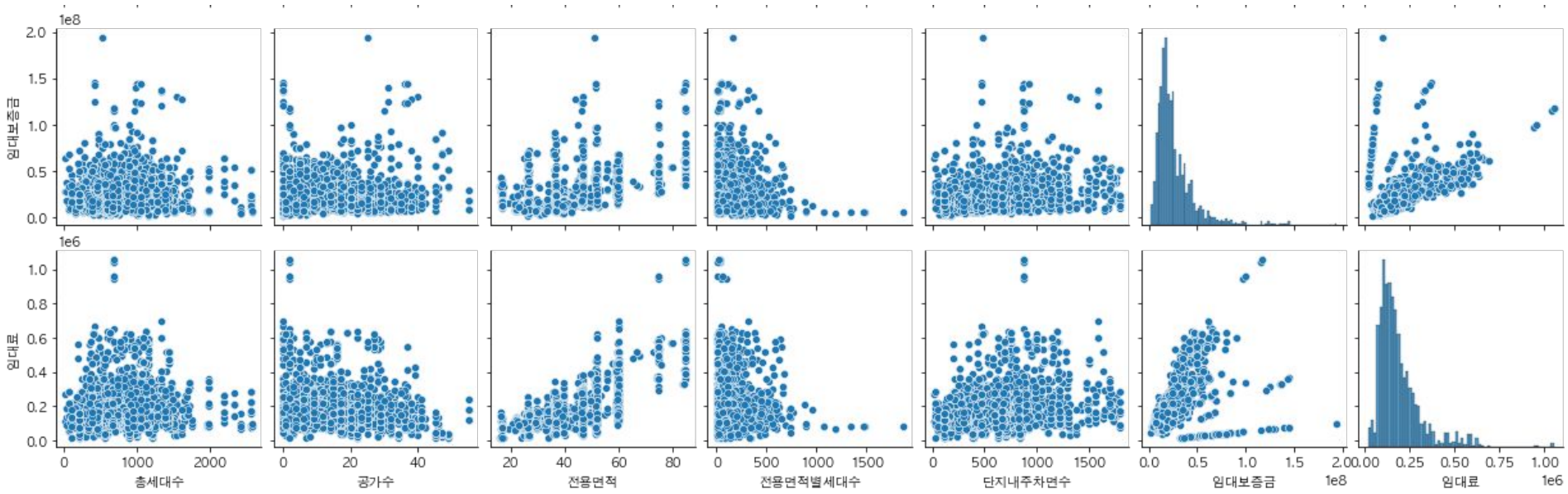




{0: '강원도', 1: '경기도', 2: '경상남도', 3: '경상북도', 4: '광주광역시', 5: '대구광역시', 6: '대전광역시', 7: '부산광역시', 8: '서울특별시', 9: '세종특별자치시', 10: '울산광역시', 11: '전라남도', 12: '전라북도', 13: '제주특별자치도', 14: '충청남도', 15: '충청북도'}

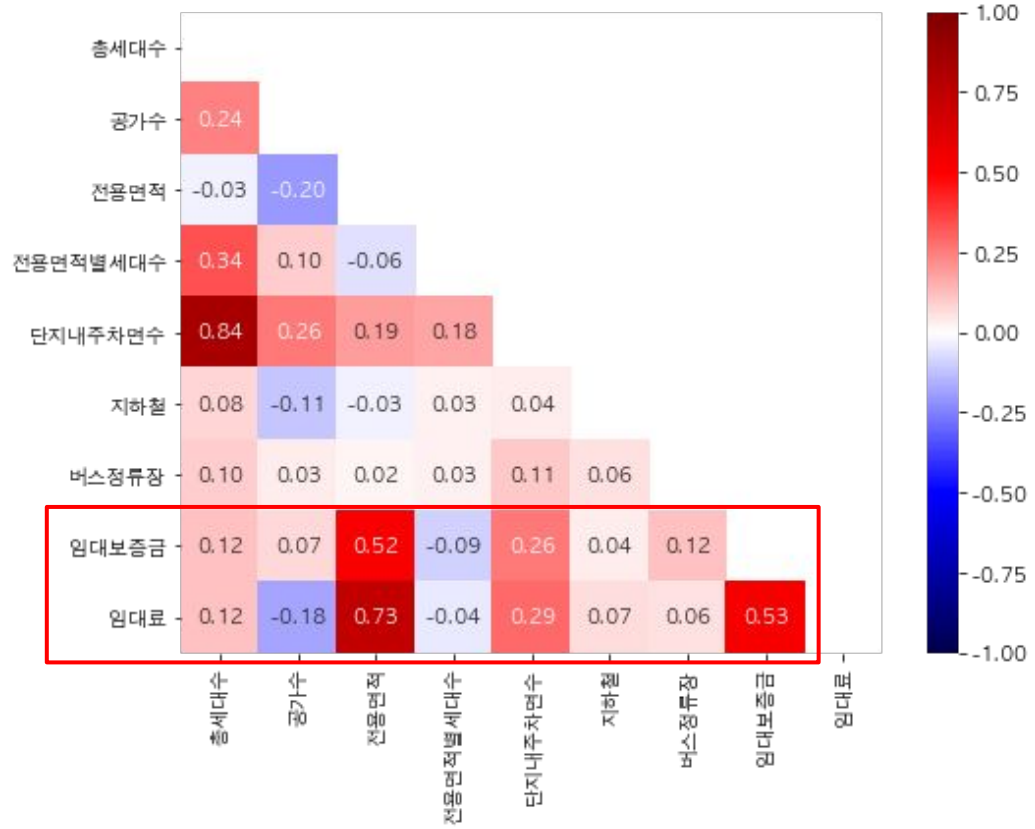
2. 수치형 피처*에 따른 임대보증금/임대료

*수치형 피처 : '총세대수','공가수', '전용면적', '전용면적별세대수', '단지내주차면수', *'지하철','버스정류장'*



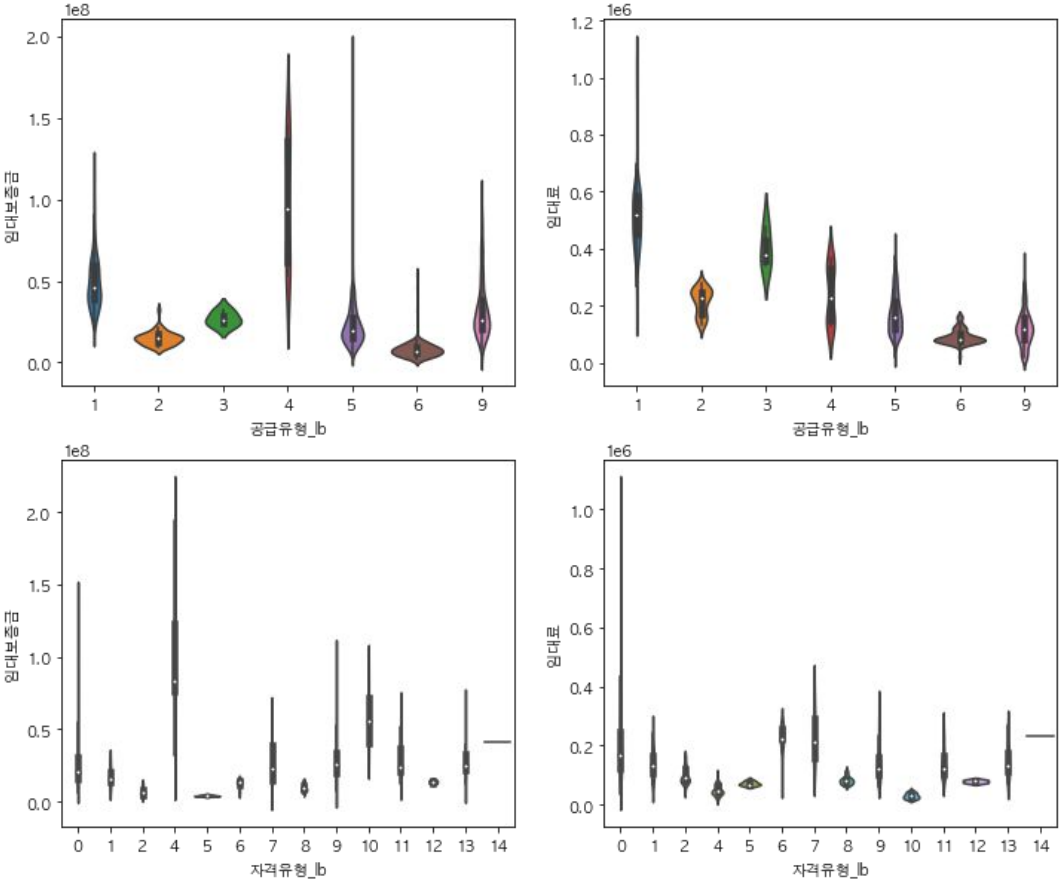
2. 수치형 피처*에 따른 임대보증금/임대료

*수치형 피처 : '총세대수','공가수', '전용면적', '전용면적별세대수', '단지내주차면수', '지하철','버스정류장'
→ 결과 : '총세대수', '전용면적', '단지내주차면수', '버스정류장수' 정도.. 하지만 대부분 강한 상관관계는 아니다.



3. 범주형 피처*에 따른 임대보증금/임대료

*범주형 피처 : '공급유형', '자격유형'



공급유형_lb

{0: '공공분양', 1: '공공임대(10년)', 2: '공공임대(50년)', 3: '공공임대(5년)', 4: '공공임대(분납)', 5: '국민임대', 6: '영구임대', 7: '임대상가', 8: '장기전세', 9: '행복주택'}

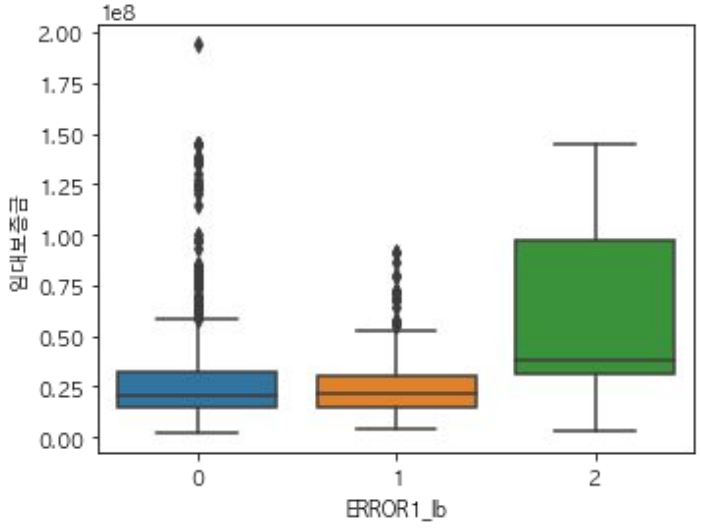
자격유형_lb

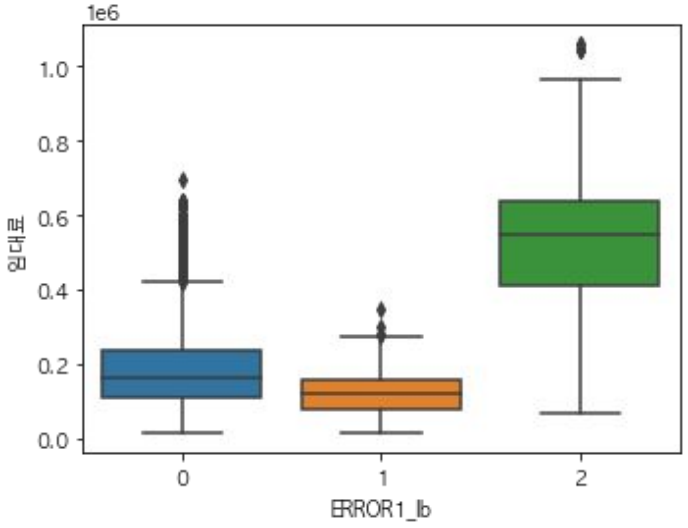
{0: 'A', 1: 'B', 2: 'C', 3: 'D', 4: 'E', 5: 'F', 6: 'G', 7: 'H', 8: 'I', 9: 'J', 10: 'K', 11: 'L', 12: 'M', 13: 'N', 14: 'O'}

4. ERROR1*에 따른 임대보증금/임대료

*ERROR 1 :

- 문제 : 전용면적별 세대수 **!=** 총 세대수
- 원인 : 분양과 임대가 혼합되어 발생, ERROR1의 발생 원인을 살펴보면 '총세대수와 전용면적별 세대수의 총합의 차이에 따라서 분양상가, 분양아파트, 나머지(임대건물)'임을 알 수 있다.





{0: '나머지',
1: '분양상가',
2: '분양아파트'}

임대보증금/임대료 예측을 위한 전처리

|질문3. 데이터가 너무 적은데, 어떻게 하면 가능한 범위에서 데이터를 더 활용할 수 있을까? 🤔

Raw 데이터의 train셋과 test 셋을 위 아래로 합치고,

임대보증금/임대료 중 하나라도 결측치가 있으면 test_new 셋으로, 없으면 train_new 셋으로 저장

train + test (등록차량수 drop)

	단지코드	총세대수	임대건물구분	지역	공급유형	전용면적	전용면적별세대수	공가수	자격유형	임대보증금	임대료	지하철	버스정류장
1000	C1267	675	아파트	경상남도	행복주택	16.94	50	38.0	L	NaN	NaN	0.0	1.0
1001	C1267	675	아파트	경상남도	행복주택	26.85	66	38.0	L	NaN	NaN	0.0	1.0
1002	C1267	675	아파트	경상남도	행복주택	26.85	8	38.0	L	NaN	NaN	0.0	1.0
1003	C1267	675	아파트	경상남도	행복주택	36.77	126	38.0	L	NaN	NaN	0.0	1.0
1004	C2189	382	아파트	전라북도	국민임대	29.19	96	45.0	H	6872000.0	106400.0	0.0	2.0
1005	C2189	382	아파트	전라북도	국민임대	29.19	20	45.0	H	6872000.0	106400.0	0.0	2.0
1006	C2189	382	아파트	전라북도	국민임대	39.45	202	45.0	H	13410000.0	144600.0	0.0	2.0
1007	C2189	382	아파트	전라북도	국민임대	46.23	60	45.0	H	18689000.0	166500.0	0.0	2.0

train_new

1004	C2189	382	아파트	전라북도	국민임대	29.19	96	45.0	H	6872000.0	106400.0	0.0	2.0
1005	C2189	382	아파트	전라북도	국민임대	29.19	20	45.0	H	6872000.0	106400.0	0.0	2.0
1006	C2189	382	아파트	전라북도	국민임대	39.45	202	45.0	H	13410000.0	144600.0	0.0	2.0
1007	C2189	382	아파트	전라북도	국민임대	46.23	60	45.0	H	18689000.0	166500.0	0.0	2.0

test_new

	단지코드	총세대수	임대건물구분	지역	공급유형	전용면적	전용면적별세대수	공가수	자격유형	임대보증금	임대료	지하철	버스정류장
1000	C1267	675	아파트	경상남도	행복주택	16.94	50	38.0	L	NaN	NaN	0.0	1.0
1001	C1267	675	아파트	경상남도	행복주택	26.85	66	38.0	L	NaN	NaN	0.0	1.0
1002	C1267	675	아파트	경상남도	행복주택	26.85	8	38.0	L	NaN	NaN	0.0	1.0
1003	C1267	675	아파트	경상남도	행복주택	36.77	126	38.0	L	NaN	NaN	0.0	1.0

|질문4. 수치형 변수(총세대수)를 어떤 기준으로 범주화를 시킬 수 있을까? 🤔

```
train.loc[train['총세대수'] <=149 ,'총세대수_lb'] = 'A1'
train.loc[(train['총세대수'] > 149) & (train['총세대수'] <= 299),'총세대수_lb' ] = 'B1'
train.loc[(train['총세대수'] > 299) & (train['총세대수'] <= 499), '총세대수_lb' ] = 'C1'
train.loc[(train['총세대수'] > 499) & (train['총세대수'] <= 999), '총세대수_lb' ] = 'D1'
train.loc[(train['총세대수'] >= 1000), '총세대수_lb' ] = 'E1'
```

	단지코드	총세대수	임대건물구분	지역	공급유형	전용면적	전용면적별세대수	공가수	자격유형	임대보증금	...	L	M	N	O	ERROR1_lb	나머지	분양상가	분양아파트	등록차량수	총세대수_lb
0	ERR04_2483	900	아파트	경상북도	국민임대	39.72	134	38.0	A	15667000.0	...	0	0	0	0	0	1	0	0	1015.0	D1
1	ERR04_2483	900	아파트	경상북도	국민임대	39.72	15	38.0	A	15667000.0	...	0	0	0	0	0	1	0	0	1015.0	D1
2	ERR04_2483	900	아파트	경상북도	국민임대	51.93	385	38.0	A	27304000.0	...	0	0	0	0	0	1	0	0	1015.0	D1
3	ERR04_2483	900	아파트	경상북도	국민임대	51.93	15	38.0	A	27304000.0	...	0	0	0	0	0	1	0	0	1015.0	D1
4	ERR04_2483	900	아파트	경상북도	국민임대	51.93	41	38.0	A	27304000.0	...	0	0	0	0	0	1	0	0	1015.0	D1

◇ 세대 규모 클수록 아파트값은 비싸

아파트 관리비가 상대적으로 높은 서울 등 수도권 지역의 세대 규.
대 규모가 클수록 아파트값이 비싼 것으로 나타났다.

1,000세대 이상 아파트는 3.3㎡당 평균 매매가격이 1,937만원으로 가장 높았다. 이어 △500~999세대(1,497만원) △300~499세대(1,425만원) △150~299세대(1,407만원) 순으로 나타났다. 1,000세대 이상 대단지와 소규모 단지의 아파트값이 30% 가까이 차이를 보인 것이다.

기사 출처: https://www.nocutnews.co.kr/news/5126591

임대보증금/임대료 예측을 위한 모델 평가

| 질문5. 어떤 Feature들이 성능을 높이는데 영향을 줄까? 🤔

다양한 변수들을 변경해 보면서 MAE 값을 비교 해 봅니다.

Scaler	기본모델	MAE 점수	사용변수
Standard	LinearRegression	4.78E+17	['총세대수', '버스정류장', '단지내주차면수', '전용면적', '강원도', '경기도', '경상남도', '경상북도', '광주광역시', '대구광역시', '대전광역시', '부산광역시', '서울특별시', '세종특별자치시', '울산광역시', '전라남도', '전라북도', '제주특별자치도', '충청남도', '충청북도', '공공분양', '공공임대(10년)', '공공임대(50년)', '공공임대(5년)', '공공임대(분납)', '국민임대', '영구임대', '임대상가', '장기전세', '행복주택', 'A', 'B','C', 'D', 'E', 'F', 'G', 'H', 'I', 'J', 'K', 'L', 'M', 'N', 'O']
	KNeighborsRegressor	7178019.814	
	RandomForestRegressor	5975579.251	
	DecisionTreeRegressor	7071179.022	
MinMax	LinearRegression	4.00E+17	
	KNeighborsRegressor	6460725.886	
	RandomForestRegressor	5939210.712	
	DecisionTreeRegressor	7126786.782	
Standard	RandomForestRegressor	6710697.309	['총세대수', '버스정류장', '단지내주차면수', '전용면적', '서울특별시', '공공분양', '공공임대(10년)', '공공임대(50년)', '공공임대(5년)', '공공임대(분납)', '국민임대', '영구임대', '임대상가', '장기전세', '행복주택', 'A', 'B','C', 'D', 'E', 'F', 'G', 'H', 'I', 'J', 'K', 'L', 'M', 'N', 'O']
MinMax		6630660.15	
Standard	RandomForestRegressor	8385451.894	['총세대수', '버스정류장', '단지내주차면수', '전용면적', '강원도', '경기도', '경상남도', '경상북도', '광주광역시', '대구광역시', '대전광역시', '부산광역시', '서울특별시', '세종특별자치시', '울산광역시', '전라남도', '전라북도', '제주특별자치도', '충청남도', '충청북도', '공공분양', '공공임대(10년)', '공공임대(50년)', '공공임대(5년)', '공공임대(분납)', '국민임대', '영구임대', '임대상가', '장기전세', '행복주택']
MinMax		8406435.73	

결과 (결측치 처리를 위한 모델 만들기)

|질문6. 임대보증금 예측 모델은 어떤 스케일러와 모델, 변수를 넣었을 때 가장 적합할까? 🤔

Scaler
- Standard
- MinMax

Model
- LinearRegression
- KNeighborsRegressor
- RandomForestRegressor
- DecisionTreeRegressor
- Ridge, Lasso
- xgboost

평가지표
- MAE

Scaler	기본모델	MAE 점수	사용변수
Standard	RandomForestRegressor	6850285.491	['지역_lb', '총세대수', '버스정류장', '단지내주차면수', '전용면적', '공급유형_lb','자격유형_lb']
	RandomForestRegressor	5975579.251	['총세대수', '버스정류장', '단지내주차면수', '전용면적', '강원도', '경기도', '경상남도', '경상북도', '광주광역시', '대구광역시', '대전광역시', '부산광역시', '서울특별시', '세종특별자치시', '울산광역시', '전라남도', '전라북도', '제주특별자치도', '충청남도', '충청북도', '공공분양', '공공임대(10년)', '공공임대(50년)', '공공임대(5년)', '공공임대(분납)', '국민임대', '영구임대', '임대상가', '장기전세', '행복주택', 'A', 'B','C', 'D', 'E', 'F', 'G', 'H', 'I', 'J', 'K', 'L', 'M', 'N', 'O']
MinMax	RandomForestRegressor	5939210.712	
.....			
MinMax	xgboost	5724567.178	['총세대수', '버스정류장', '단지내주차면수', '전용면적', '강원도', '경기도', '경상남도', '경상북도', '광주광역시', '대구광역시', '대전광역시', '부산광역시', '서울특별시', '세종특별자치시', '울산광역시', '전라남도', '전라북도', '제주특별자치도', '충청남도', '충청북도', '공공분양', '공공임대(10년)', '공공임대(50년)', '공공임대(5년)', '공공임대(분납)', '국민임대', '영구임대', '임대상가', '장기전세', '행복주택', 'A', 'B','C', 'D', 'E', 'F', 'G', 'H', 'I', 'J', 'K', 'L', 'M', 'N', 'O']

결과 (결측치 처리를 위한 모델 만들기)

|질문7. 임대료 예측 모델은 어떤 스케일러와 모델, 변수를 넣었을 때 가장 적합할까? 🤔

Scaler
- Standard
- MinMax

Model
- LinearRegression
- KNeighborsRegressor
- RandomForestRegressor
- DecisionTreeRegressor
- Ridge, Lasso
- xgboost

평가지표
- MAE

Scaler	기본모델	MAE 점수	사용변수
Standard	Ridge	42242.73629	['총세대수', '버스정류장', '단지내주차면수', '전용면적', '강원도', '경기도', '경상남도', '경상북도', '광주광역시', '대구광역시', '대전광역시', '부산광역시', '서울특별시', '세종특별자치시', '울산광역시', '전라남도', '전라북도', '제주특별자치도', '충청남도', '충청북도', '공공분양', '공공임대(10년)', '공공임대(50년)', '공공임대(5년)', '공공임대(분납)', '국민임대', '영구임대', '임대상가', '장기전세', '행복주택', 'A', 'B','C', 'D', 'E', 'F', 'G', 'H', 'I', 'J', 'K', 'L', 'M', 'N', 'O'] sel_num = ['총세대수', '버스정류장', '단지내주차면수', '전용면적']
MinMax	Ridge	39594.94581	
.....			
MinMax	RandomForestRegressor	38514.85556	['총세대수', '버스정류장', '단지내주차면수', '전용면적', '강원도', '경기도', '경상남도', '경상북도', '광주광역시', '대구광역시', '대전광역시', '부산광역시', '서울특별시', '세종특별자치시', '울산광역시', '전라남도', '전라북도', '제주특별자치도', '충청남도', '충청북도', '공공분양', '공공임대(10년)', '공공임대(50년)', '공공임대(5년)', '공공임대(분납)', '국민임대', '영구임대', '임대상가', '장기전세', '행복주택', 'A', 'B','C', 'D', 'E', 'F', 'G', 'H', 'I', 'J', 'K', 'L', 'M', 'N', 'O']

결과 (결측치 처리)

임대보증금 결측치 처리

```
X_all_test = np.concatenate((X_poly_test, X_nor_test[:,4:]), axis=1)
print("X_all의 shape :", X_all_test.shape)
pred = xg_reg.predict(X_all_test)
print("pred의 shape :", pred.shape)
print("test의 shape :", test.shape)

print("임대보증금 결측치 처리 전 :", test['임대보증금'].isnull().sum())
test['임대보증금'] = pred
print("임대보증금 결측치 처리 후 :", test['임대보증금'].isnull().sum())
test.head()
```

```
X_all의 shape : (776, 55)
pred의 shape : (776,)
test의 shape : (776, 68)
임대보증금 결측치 처리 전 : 767
임대보증금 결측치 처리 후 : 0
```

임대료 결측치 처리

```
pred = rf.predict(X_nor_test)
print("임대료 결측치 처리 전 :", test['임대료'].isnull().sum())
test['임대료'] = pred
print("임대료 결측치 처리 후 :", test['임대료'].isnull().sum())
test.head()
```

```
임대료 결측치 처리 전 : 776
임대료 결측치 처리 후 : 0
```

train과 test 분리

한계점

- 결론적으로 임대보증금과 임대료 결측치 처리한 값이 도움이 되지 못했다.

train_new에는 아파트 밖에 없었고, test_new에는 상가가 95%를 차지하고 있음
아파트 데이터로 학습을 시키고, 상가 데이터를 예측한 꼴

```
print(train_new['임대건물구분'].value_counts(), '\n')
print(test_new['임대건물구분'].value_counts())
```

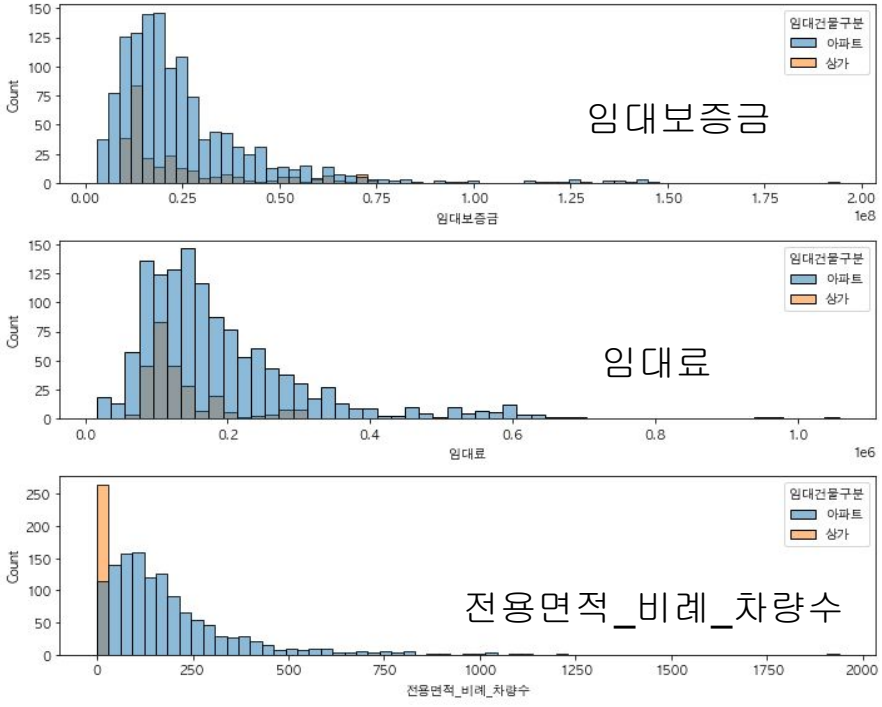
```
아파트    3184
Name: 임대건물구분, dtype: int64

상가      739
아파트      37
Name: 임대건물구분, dtype: int64
```

한계점

- 결론적으로 임대보증금과 임대료 결측치 처리한 값이 도움이 되지 못했다.

임대건물분포가 고르지 못함



한계점

제일 많이 했던 질문...

이 FEATURE를 이렇게 해석해도 되는 건가...?

PART III.

전용면적별
주차대수 모델

질문8. 전용면적이 클수록 등록차량수가 많지 않을까? 🤔

전용면적_비례_차량수

	단지코드	총세대수	임대건물구분	지역	공급유형	전용면적	전용면적별 세대수	공가수	자격유형	임대보증금	임대료	도보 10분거리 내 지하철역 수 (환승노선 수 반영)	도보 10분거리 내 버스정류장 수	단지내주차면수	등록차량수
1803	C1000	566	아파트	충청북도	국민임대	39.57	318	10.0	A	10803000	96030	0.0	1.0	438.0	481.0
1804	C1000	566	아파트	충청북도	국민임대	39.57	26	10.0	A	10803000	96030	0.0	1.0	438.0	481.0
1805	C1000	566	아파트	충청북도	국민임대	39.93	75	10.0	A	10803000	96030	0.0	1.0	438.0	481.0
1806	C1000	566	아파트	충청북도	국민임대	46.70	36	10.0	A	14404000	129640	0.0	1.0	438.0	481.0
1807	C1000	566	아파트	충청북도	국민임대	46.90	36	10.0	A	14404000	129640	0.0	1.0	438.0	481.0
1808	C1000	566	아파트	충청북도	국민임대	51.93	60	10.0	A	15604000	150040	0.0	1.0	438.0	481.0
1809	C1000	566	아파트	충청북도	국민임대	51.93	15	10.0	A	15604000	150040	0.0	1.0	438.0	481.0

1. 전용면적이 넓을수록 실거주자수가 많고 등록차량수가 많지 않을까?
2. 같은 단지코드라면 등록차량수가 같다.
3. 전용면적에 비례하여 등록차량수를 할당하자!

전처리 : 전용면적_비례_차량수 할당

1. 전용면적 소수점 자리가 의미가 없어보인다. → 소수점은 날린다. 동일한 전용면적으로
2. 동일한 단지코드, 전용면적에 대해 전용면적별세대수를 합산
3. 면적_세대수_비율 =면적_세대수_곱 / 면적_세대수_곱_총합
4. 전용면적_비례_차량수 = 면적_세대수_비율 X 단지별 등록차량수

→ ***전용면적이 넓을수록, 전용면적별 세대수가 많을수록 전용면적_비례_차량수가 많다.***

	단지코드	총세대수	임대건물구분	지역	공급유형	전용면적	공가수	자격유형	임대보증금	임대료	...	ERROR1_lb	나머지	분양상가	분양아파트	등록차량수	전용면적별세대수	면적_세대수_곱	면적_세대수_곱_총합	면적_세대수_비율	전용면적_비례_차량수
0	C1000	566	아파트	충청북도	국민임대	39	10.0	A	10803000.0	96030.0	...	0	1	0	0	481.0	419	16341	23478	0.696013	334.782392
1	C1000	566	아파트	충청북도	국민임대	46	10.0	A	14404000.0	129640.0	...	0	1	0	0	481.0	72	3312	23478	0.141068	67.853821
2	C1000	566	아파트	충청북도	국민임대	51	10.0	A	15604000.0	150040.0	...	0	1	0	0	481.0	75	3825	23478	0.162918	78.363787

전용면적_비례_차량수 모델 평가

| 질문9. 어떤 Feature들이 전용면적_비례_차량수 예측 성능을 높이는데 영향을 줄까? 🤔

다양한 변수들을 변경해 보면서 MAE 값을 비교 해 봅니다.

	모델	MAE 점수	사용변수
1차	RandomForestRegressor	34.275589012	['임대료','단지내주차면수','전용면적별세대수', '면적_세대수_곱', '면적_세대수_곱_총합','면적_세대수_비율', '강원도', '경기도', '경상남도', '경상북도', '광주광역시', '대구광역시', '대전광역시', '부산광역시', '서울특별시', '세종특별자치시', '울산광역시', '전라남도', '전라북도', '제주특별자치도', '충청남도', '충청북도', '나머지', '분양상가', '분양아파트']
2차	RandomForestRegressor	38.509349349	['단지내주차면수', '전용면적별세대수', '면적_세대수_비율', '면적_세대수_곱_총합']
	LinearRegression	39.231684829	
	Lasso (alpha=defalt)	42.20182153	
	Ridge (alpha=defalt)	41.507053807	
	RandomForestRegressor	38.422995452	['단지내주차면수', '전용면적별세대수', '면적_세대수_비율', '면적_세대수_곱_총합', '상가','아파트']
	LinearRegression	39.041976671	
	Lasso (alpha=defalt)	42.202134879	
	Ridge (alpha=defalt)	41.312448233	
	RandomForestRegressor	38.326620950	['단지내주차면수', '전용면적별세대수', '면적_세대수_비율', '면적_세대수_곱_총합', '상가','아파트', '공공분양']
	LinearRegression	38.893508699	
	Lasso (alpha=defalt)	42.202134879	
	Ridge (alpha=defalt)	41.25340706	
	RandomForestRegressor	38.447932153	['단지내주차면수', '전용면적별세대수', '면적_세대수_비율', '면적_세대수_곱_총합', '상가','아파트', '공공분양','총세대수lb']
	LinearRegression	39.018547968	
	Lasso (alpha=defalt)	42.201903818	
	Ridge (alpha=defalt)	41.385639428	
	RandomForestRegressor	38.393839834	['단지내주차면수', '전용면적별세대수', '면적_세대수_비율', '면적_세대수_곱_총합', '상가','아파트', '공공분양', '총세대수_lb_A1', '총세대수_lb_B1', '총세대수_lb_C1', '총세대수_lb_D1', '총세대수_lb_E1']
	LinearRegression	5010119289.5	
	Lasso (alpha=defalt)	42.142698275	
	Ridge (alpha=defalt)	41.907967637	
	RandomForestRegressor	0.2361808218	['단지내주차면수','전용면적별세대수', '면적_세대수_곱_총합', '면적_세대수_비율', '강원도', '경기도', '경상남도', '경상북도', '광주광역시', '대구광역시', '대전광역시', '부산광역시', '서울특별시', '세종특별자치시', '울산광역시', '전라남도', '전라북도', '제주특별자치도',
	LinearRegression	0.3110285415	

결과

|질문10. 모델은 어떤 스케일러와 모델, 변수를 넣었을 때 가장 적합할까? 🤔

Scaler
- Standard
- MinMax

Model
- LinearRegression
- KNeighborsRegressor
- RandomForestRegressor
- DecisionTreeRegressor
- Ridge, Lasso
- xgboost

평가지표
- MAE

Scaler	기본모델	MAE 점수	사용변수
Standard	LinearRegression	44.87972014	train_na 임대보증금, 임대료 결측치 처리안한 것 sel = ['임대보증금', '임대료', '단지내주차면수','전용면적별세대수','면적_세대수_곱', '면적_세대수_곱_총합','면적_세대수_비율']
.....			
MinMax	RandomForestRegressor	35.9963295	sel = ['단지내주차면수','전용면적별세대수', '면적_세대수_곱', '면적_세대수_곱_총합','면적_세대수_비율'] + polynomial features(deg=2) 추가
MinMax	RandomForestRegressor	34.30626279	sel = ['임대료','단지내주차면수','전용면적별세대수', '면적_세대수_곱', '면적_세대수_곱_총합','면적_세대수_비율','강원도', '경기도', '경상남도', '경상북도', '광주광역시', '대구광역시', '대전광역시', '부산광역시', '서울특별시', '세종특별자치시', '울산광역시', '전라남도', '전라북도', '제주특별자치도', '충청남도', '충청북도', '나머지', '분양상가', '분양아파트'] + 수치형 피처 ('임대료','단지내주차면수','전용면적별세대수', '면적_세대수_곱', '면적_세대수_곱_총합','면적_세대수_비율')에 대한 polynomial features(deg=2) 추가

DACON 제출 결과

- 최종 점수 : 108.052361점(public), 118.70366점(private)

  과...적합...

PART IV.

개선사항 및 배운점

✅ 개선사항

1. 이상치 분석
   - 이상치를 가진 피처 확인
   - RobustScaler 활용

2. Catboost 모델 사용
   - 모델 이론 학습 및 적용

3. 하이퍼 파라미터 튜닝

✅ 배운점

1. 대회에서 공유하고 있는 코드들을 적극적으로 참고해보기
   - 더 다양한 파생 변수, 모델 등에 대해 학습

2. EDA를 꼼꼼하게 해보기
   - 임대보증금/임대료 예측 모델에 대한 효용성 검증 부재

3. 좋은 피처 만들기
   - Feature Engineering의 중요성

4. 적극적으로 도메인 지식 찾아보기
   - 공급유형, 자격유형의 연결고리 파악 부족

5. 다양한 시도를 해보기

감사합니다 🤦‍♀️