DACON 아파트값 예측 리뷰

19.01.26

김도완

처음 데이터분석 대회를 참여하면서
배운점을 얘기해보려고 한다.

Dacon, 4th competition, 아파트 거래가격 예측 11월~
Kaggle , 타이타닉 생존자 예측 1월~

목차

1. Work Flow에 대한 구체화
2. 모델의 특성
3. 특성에 대한 생각
4. 앙상블의 위력
5. 노가다가 반이다.
6. 현재 진행

1. Work Flow (기존)

1. 데이터 수집 : 데이터를 불러온다
2. 데이터 탐색 : 데이터의 탐색, 점검, 분석
3. 정제 : 이상치 제거와 특성추출 (2번 Loop)
4. 준비 : 모델 적용 가능한 형태로 변환, 학습/평가용 데이터 분할
5. 모델 : 모델 선택 및 학습
6. 평가 : 모델 평가 (5번 Loop)

1. Work Flow (구체화)

1. 데이터 수집

- 데이터가 매우 클때는 먼저 일부분만 떼어낸 표본으로 시작한다.
- train과 test데이터를 불러온 다음 합친다. (특성변환, nan값 제거 등을 한꺼번에)
- test에는 target값이 없기 때문에 EDA나 모델링시에는 다시 분리시켜 진행한다.

2. 데이터 분석 (EDA)과 이상치 제거 (Pre-Processing for Data Cleaning)

- 특성에 대한 충분한 이해가 있어야 특성추출이 수월해진다.
- 이상치(NaN값)은 제거 또는 특정값으로 변환한다.
- Nan값 자체가 의미있는 정보가 될 수 있다.

3. 특성추출 (Feature Engineering and Variable Selection)

- EDA를 바탕으로 target값과 상관성있는 중요한 특성을 선정한다.
- 특성을 조합하거나 변환하여 새로운 특성을 만들어낸다.
- 덧셈, 뺄셈, 곱하기, 나누기, 빈도수, 평균값, lag값

1. Work Flow (구체화)

4. 모델 적용 가능한 형태로 변환

라벨링, 원핫인코딩, 스케일링, 로그변환 등

5. 모델 학습과 평가 (Model Selection, Optimization)

- 모델특성에 대한 이해와 하이퍼파라미터 튜닝에 대한 경험과 노하우가 필요하다.
(1) 처음에는 다양한 단일모델의 성능을 비교해본다.
(2) 교차검증으로 모델의 성능을 확인한다.
(3) 하이퍼파라미터 튜닝을 통해 단일모델성능을 최대한 끌어낸다.
(4) 최고성능을 내는 모델들을 가지고 앙상블을 시도한다.
(5) 피쳐추가, 삭제, 조합을 다르게 하여 위의 과정을 재시도해본다.

[참고] 교차검증

Clip slide

Feature Engineering Part 2

Cross-Validation

KFOLD는 K값 즉 몇 번으로 나눌건지 정하고 K번만큼 Validation을 수행

각 Validation Set이 겹치지 않게 검증하게 되고 결과는 Average하여 성능을 평가

시간이 오래걸림

보통 5회 사용, 많으면 10정도

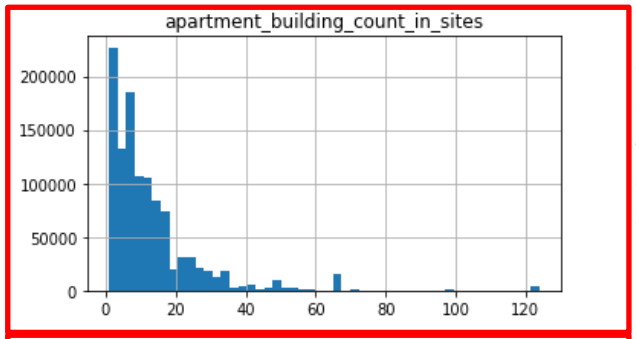
2. 모델의 특성

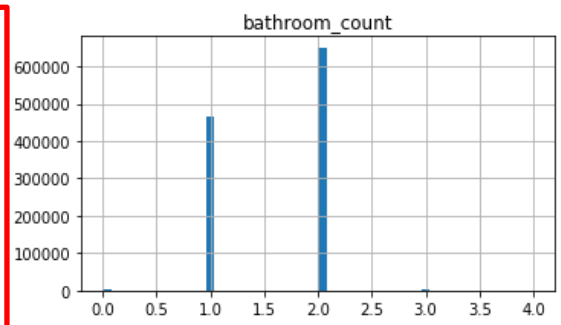
트리기반 모델

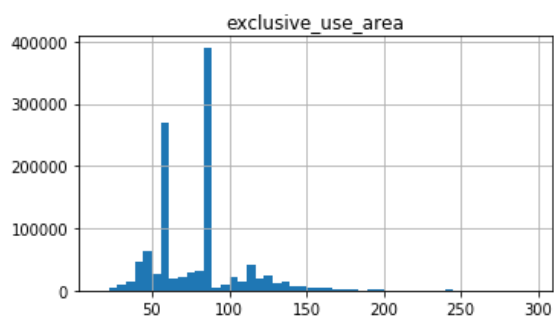
(1) 스케일링 및 로그변환 필요없음.
(2) 원핫인코딩할 필요없음. (문자열은 라벨링까지만)
(3) 연속형 수치에 대해서는 범위를 지정하여 라벨링할때 성능이 높아질때가 있다.
(4) 참고, 타겟값을 평당가격으로 넣는게 성능 높임. (5~10%증가)

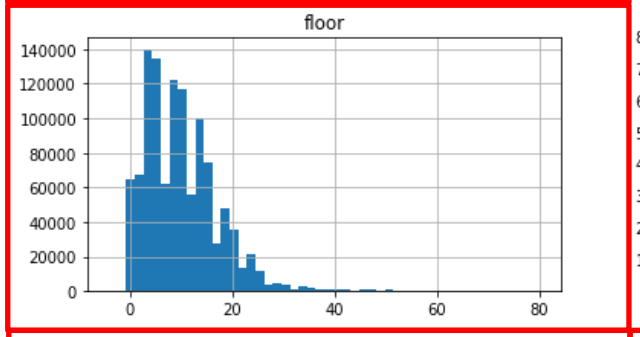
선형모델, SVM, 신경망

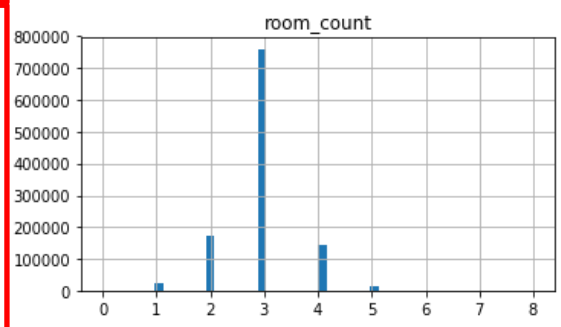
(1) 스케일링과 로그변환은 성능을 올리는데 도움을 준다.
로그변환은 특성의 히스토그램을 종모양으로 만드는 쉽고 효과적인 방법 (역은 exp변환)
(정규분포와 비슷할때 최고의 성능을 낸다.)
(2) 문자열이나 라벨링에 대해 원핫인코딩하면 성능 좋아짐. (하지만 차원이 증가하기 때문에 한계)

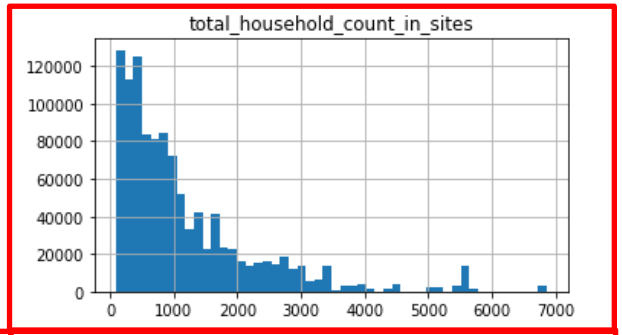

apartment_building_count_in_sites

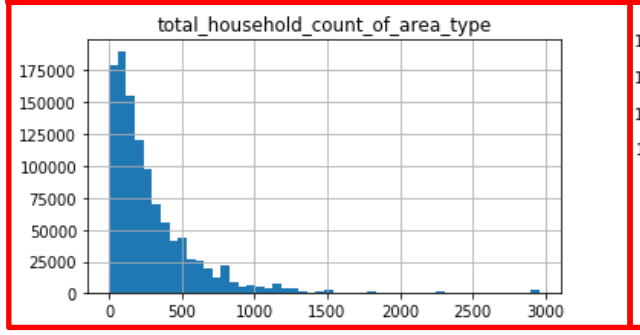

bathroom_count

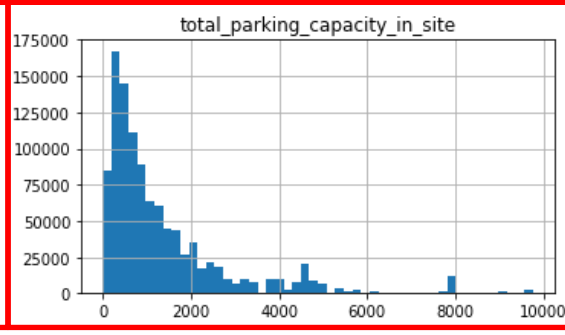

exclusive_use_area

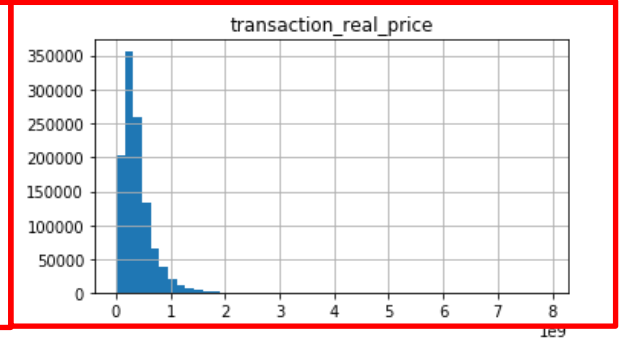

floor

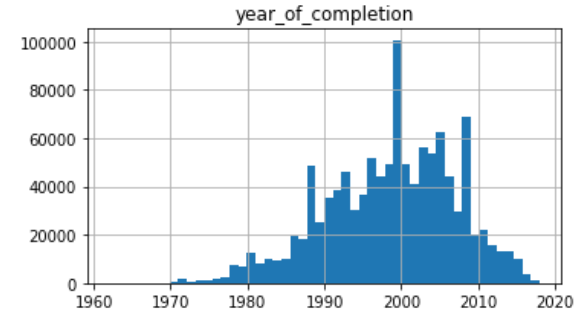

room_count


total_household_count_in_sites


total_household_count_of_area_type


total_parking_capacity_in_site


transaction_real_price


year_of_completion

[참고] 히스토그램

2. 모델의 특성

처리속도
선형모델 > 신경망 > 랜덤포레스트, 아다부스트, 그레디언트부스트

메모리 사용량
선형모델 > 신경망 > 아다부스트, 그레디언트부스트 > 랜덤포레스트

대용량 데이터 적합도
선형모델 > 신경망 > 아다부스트, 그레디언트부스트 > 랜덤포레스트 > SVM

학습율 (오버피팅)
아다부스트, 그레디언트부스트 , 랜덤포레스트, 신경망 > SVM > 선형모델

3. 특성 중요도

특성중요도는 아래와 같이 종합적으로 판단한다.
- 상관계수 히트맵
- 트리모델 특성값 확인
- 선형모델 계수값 확인

약한 특성	강한특성	애매함
복도구조	지역(구,동)	지하철, 학교
난방방식	거래년도	면적비율(전용/공급)
난방연료	전용면적, 공급면적	주차비율(주차수/세대수)
방수, 화장실수	세대수, 동수	층수
	건축년도	
	층수(저층)	

[참고] 상관계수값

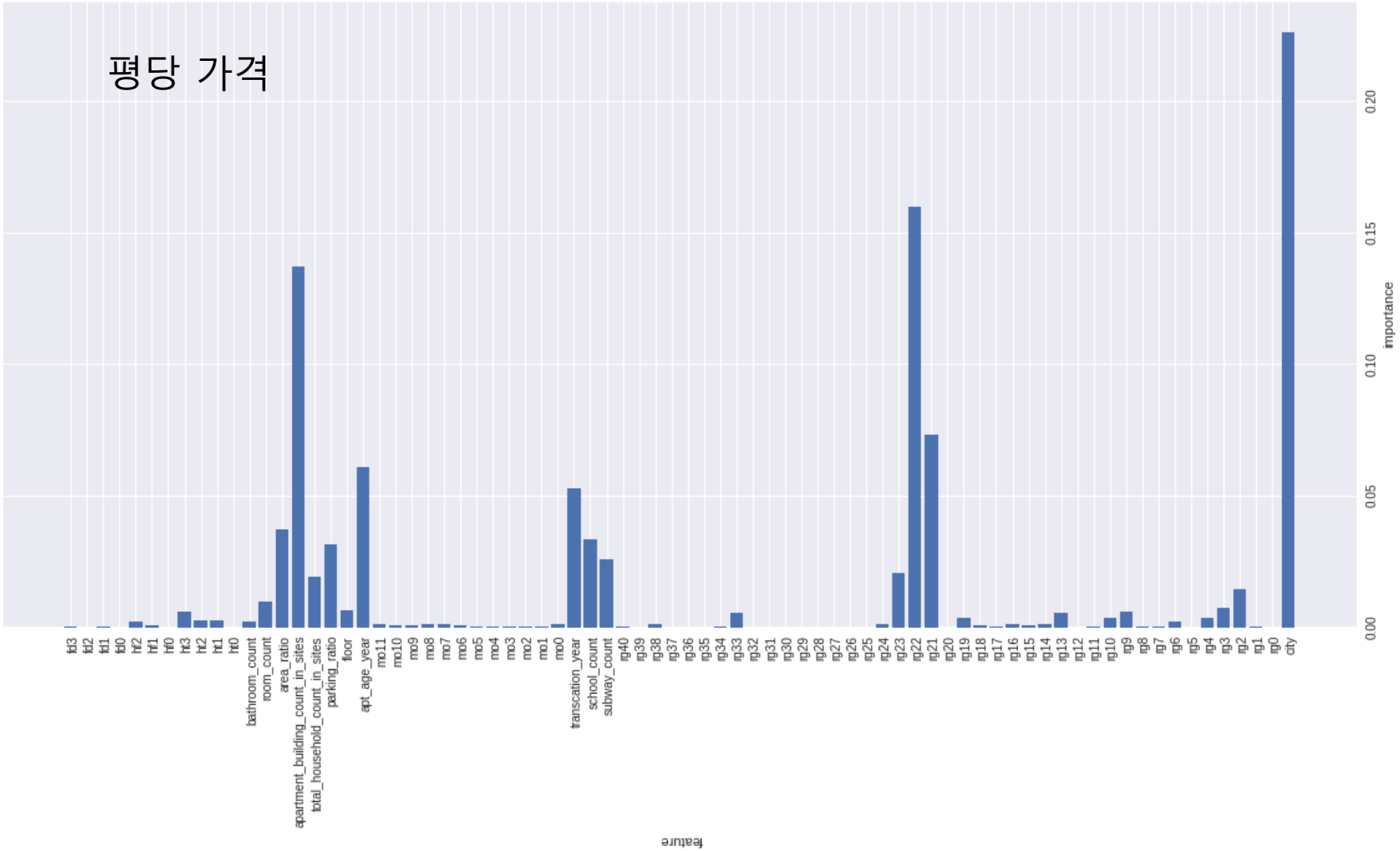
transaction_real_price	1.000000
real_price_by_area	0.825508
supply_area	0.521845
exclusive_use_area	0.518922
parking_ratio	0.400807
city	0.399349
latitude	0.386292
room_count	0.383066
bathroom_count	0.331598
total_parking_capacity_in_site	0.310130
key	0.296135
apartment_building_count_in_sites	0.290689
transaction_year_month	0.285657
transaction_year	0.285481
total_household_count_in_sites	0.216658
tallest_building_in_sites	0.193310
area_ratio	0.192681
apartment_id	0.160393
year_of_completion	0.132442
floor	0.129241
lowest_building_in_sites	0.113520
subway_count	0.078583
transaction_date1	0.011092
transaction_month	0.002283
apt_age_year	-0.004598
total_household_count_of_area_type	-0.030591
room_id	-0.071792
school_count	-0.089539
longitude	-0.389087
address_by_law	-0.394201

real_price_by_area	1.000000
transaction_real_price	0.825508
city	0.546510
latitude	0.532850
apartment_building_count_in_sites	0.398329
key	0.348511
transaction_year_month	0.335251
transaction_year	0.335118
total_parking_capacity_in_site	0.321249
total_household_count_in_sites	0.307601
parking_ratio	0.191207
subway_count	0.115398
apartment_id	0.105852
apt_age_year	0.089271
supply_area	0.083655
exclusive_use_area	0.073694
bathroom_count	0.073111
tallest_building_in_sites	0.067870
year_of_completion	0.058277
floor	0.052648
room_count	0.047320
total_household_count_of_area_type	0.046956
transaction_date1	0.010256
area_ratio	0.009899
lowest_building_in_sites	0.006823
transaction_month	-0.005577
school_count	-0.056439
room_id	-0.136072
longitude	-0.533870
address_by_law	-0.540505

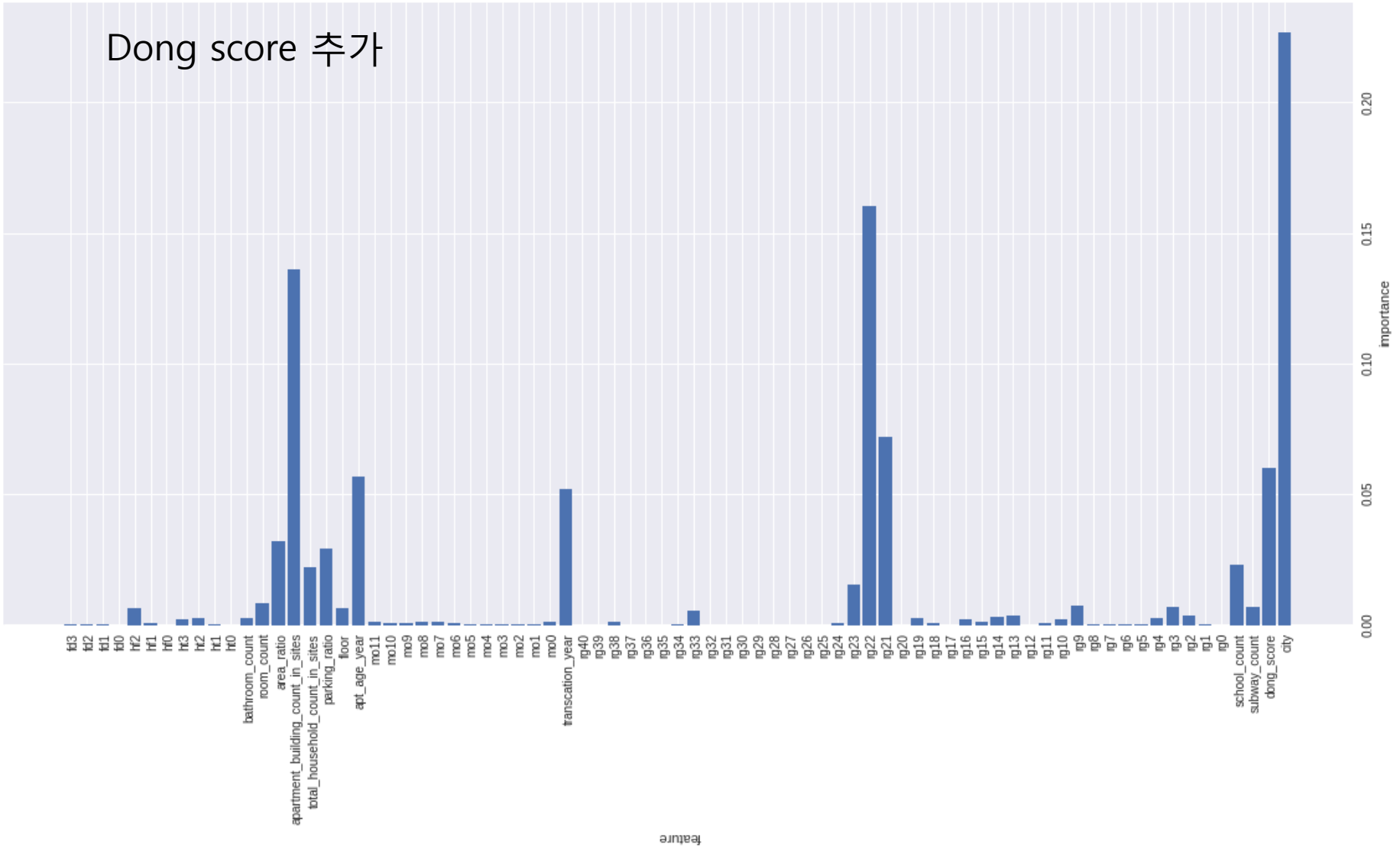
[참고] 트리모델 특성값

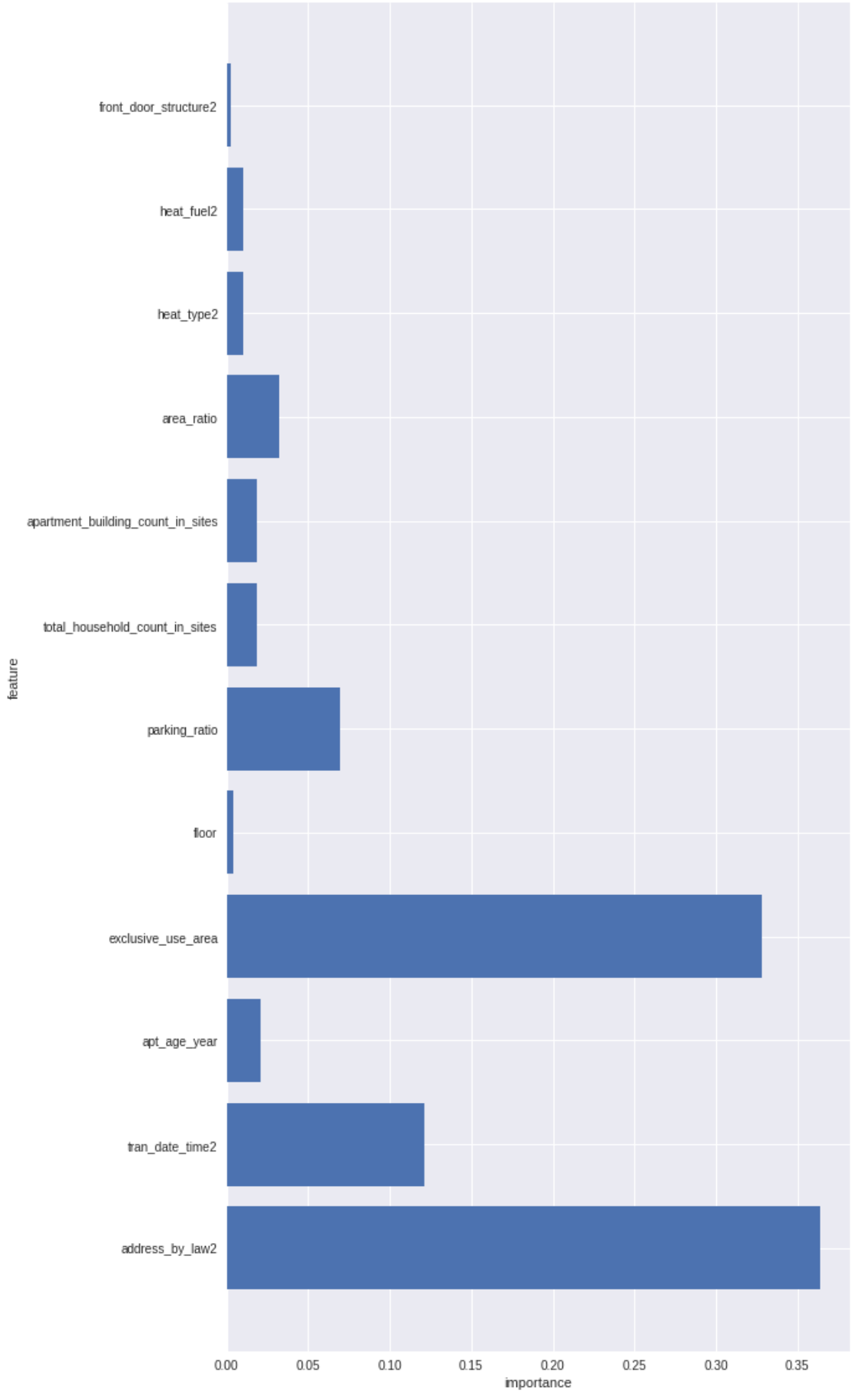

importance
0.00
0.05
0.10
0.15
0.20
0.25
0.30
0.35
feature
td3
td2
td1
td0
hf2
hf1
hf0
ht3
ht2
ht1
ht0
mo11
mo10
mo9
mo8
mo7
mo6
mo5
mo4
mo3
mo2
mo1
mo0
rg40
rg39
rg38
rg37
rg36
rg35
rg34
rg33
rg32
rg31
rg30
rg29
rg28
rg27
rg26
rg25
rg24
rg23
rg22
rg21
rg20
rg19
rg18
rg17
rg16
rg15
rg14
rg13
rg12
rg11
rg10
rg9
rg8
rg7
rg6
rg5
rg4
rg3
rg2
rg1
rg0
bathroom_count
room_count
area_ratio
apartment_building_count_in_sites
total_household_count_in_sites
parking_ratio
floor
exclusive_use_area
apt_age_year
transcation_year
school_count
subway_count
city

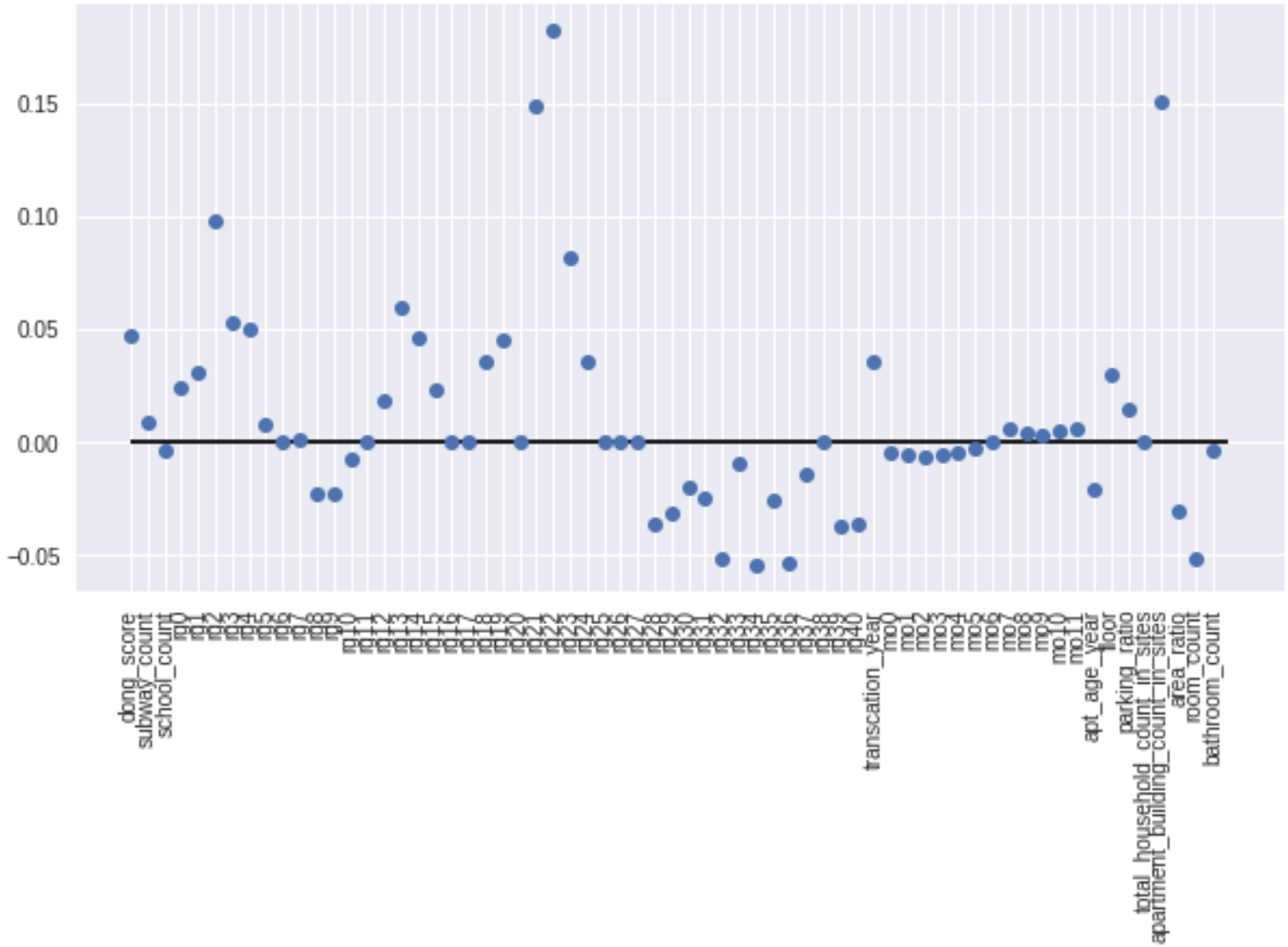
평당 가격

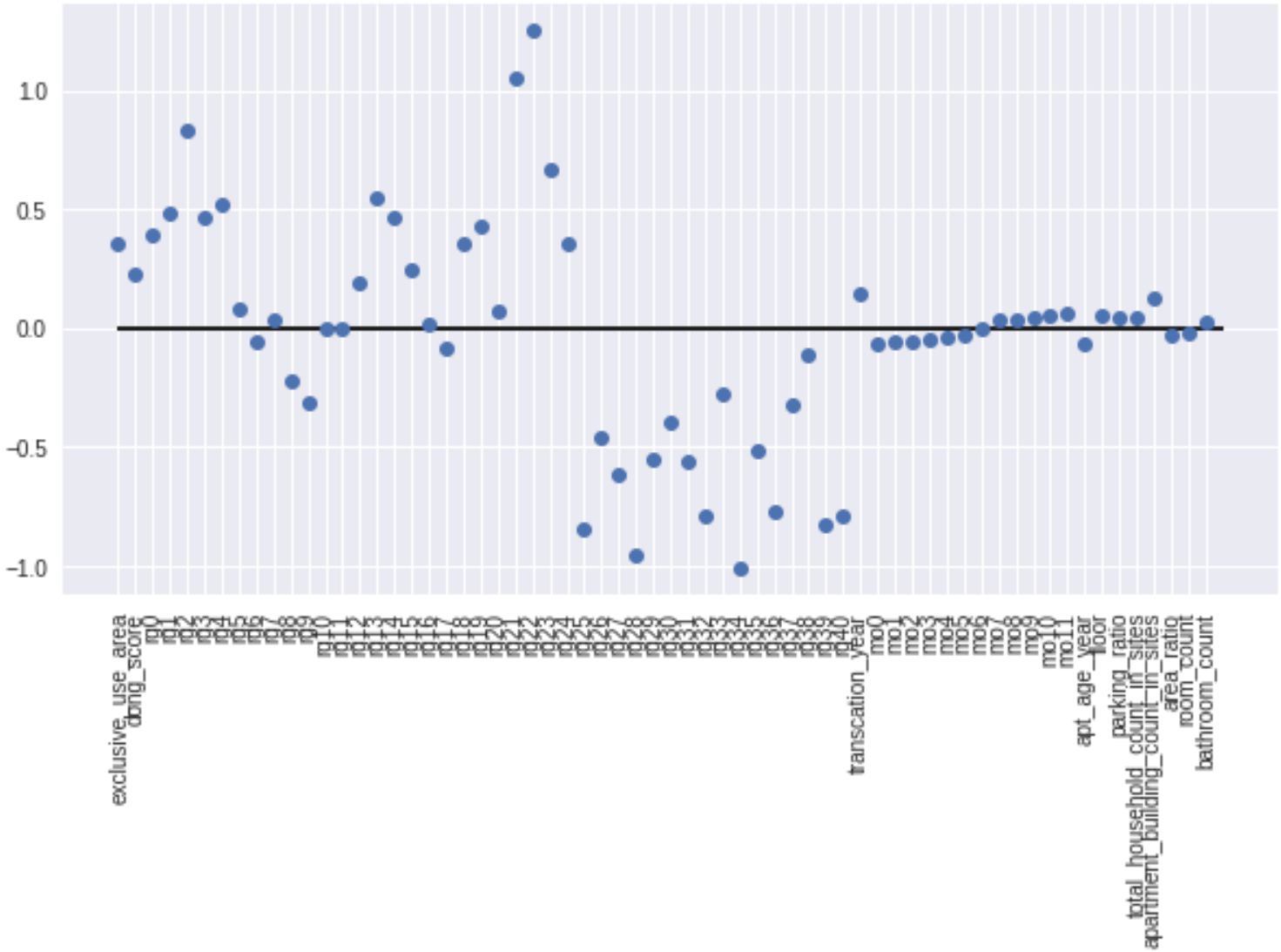

importance
0.00
0.05
0.10
0.15
0.20
feature
hf3
hf2
hf1
hf0
bathroom_count
room_count
area_ratio
apartment_building_count_in_sites
total_household_count_in_sites
parking_ratio
floor
apt_age_year
mo11
mo10
mo9
mo8
mo7
mo6
mo5
mo4
mo3
mo2
mo1
mo0
transcation_year
school_count
subway_count
rg40
rg39
rg38
rg37
rg36
rg35
rg34
rg33
rg32
rg31
rg30
rg29
rg28
rg27
rg26
rg25
rg24
rg23
rg22
rg21
rg20
rg19
rg18
rg17
rg16
rg15
rg14
rg13
rg12
rg11
rg10
rg9
rg8
rg7
rg6
rg5
rg4
rg3
rg2
rg1
rg0
city

Dong score 추가

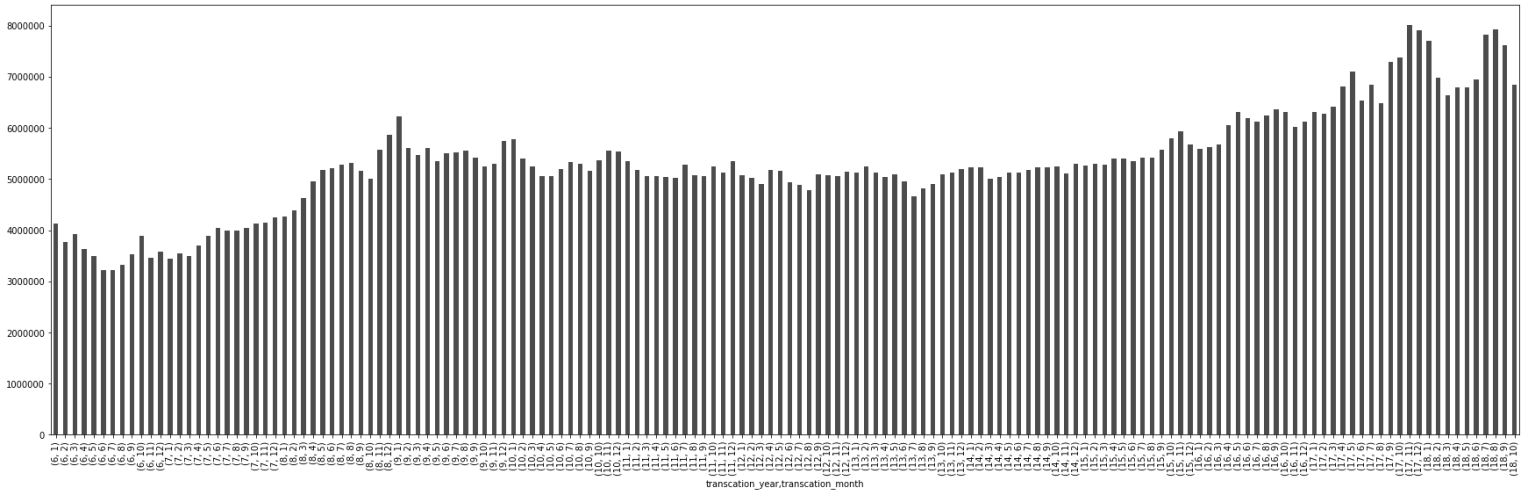



front_door_structure2
heat_fuel2
heat_type2
area_ratio
apartment_building_count_in_sites
total_household_count_in_sites
parking_ratio
floor
exclusive_use_area
apt_age_year
tran_date_time2
address_by_law2
feature
0.00
0.05
0.10
0.15
0.20
0.25
0.30
0.35
importance


0.15
0.10
0.05
0.00
−0.05
dong_score
subway_count
school_count
m0
m1
m2
m3
m4
m5
m6
m7
m8
m9
m10
m11
m12
m13
m14
m15
m16
m17
m18
m19
m20
m21
m22
m23
m24
m25
m26
m27
m28
m29
m30
m31
m32
m33
m34
m35
m36
m37
m38
m39
m40
transcation_year
mo0
mo1
mo2
mo3
mo4
mo5
mo6
mo7
mo8
mo9
mo10
mo11
apt_age_year
floor
parking_ratio
total_household_count_in_sites
apartment_building_count_in_sites
area_ratio
room_count
bathroom_count


1.0
0.5
0.0
−0.5
−1.0
exclusive_use_area
dong_score
rg0
rg1
rg2
rg3
rg4
rg5
rg6
rg7
rg8
rg9
rg10
rg11
rg12
rg13
rg14
rg15
rg16
rg17
rg18
rg19
rg20
rg21
rg22
rg23
rg24
rg25
rg26
rg27
rg28
rg29
rg30
rg31
rg32
rg33
rg34
rg35
rg36
rg37
rg38
rg39
rg40
transcation_year
mo0
mo1
mo2
mo3
mo4
mo5
mo6
mo7
mo8
mo9
mo10
mo11
apt_age_year
floor
parking_ratio
total_household_cdunt_in_sites
apartment_building_count_in_sites
area_ratio
room_count
bathroom_count

3-1. 시간은 매우 중요한 특성



타임 피쳐링

(1) 년도, 월, 일을 별도 컬럼으로 둔다. -> 년도 12개, 월 12개, 일 3개 라벨링
월, 일 시간값 자체를 특성으로 주는 것은 주기성을 확인하는데 효과적이다.

(2) 년, 월, 일을 모두 합친다. -> 12*12*3= 432 라벨링 (성능 증가, 5~10%)
또는 년, 월4분기 -> 12*4 = 48라벨링

(3) 시간에 따른 가격의 변화를 확인하기 위해서는 LAG특성 (성능 증가)
과거의 평균값, 과거의 변화율
특정지역의 전분기, 전년도 평균값, 변화율

Clip slide

Feature Engineering Part 1

Datetime and Coordinate

1. Periodicity

Day number in week, month, season, year
second, minute, hour.

2. Time since
   a. Row-independent moment
      For example: since 00:00:00 UTC, 1 January 1970;
   b. Row-dependent important moment
      Number of days left until next holidays/ time passed after last holiday.

3. Difference between dates

datetime_feature_1 - datetime_feature_2

간단하게 Datetime Feature이 주어졌을 때
각 항목을 나눠서 Feature에 추가

기준점으로부터 기간이 얼마나 지났는지
TimeSeries 대회에서 많이 사용

Datetime끼리 빼서 Feature를 추가하기도 합니다.

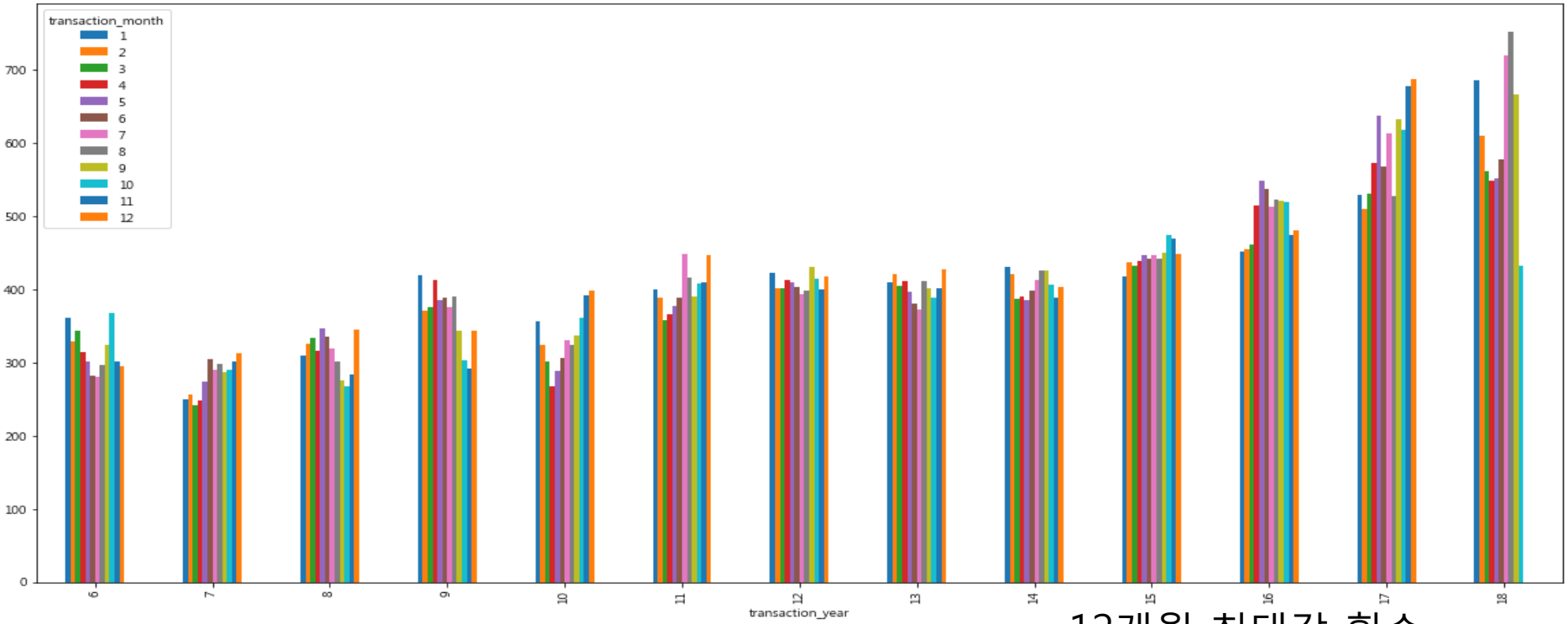
Date	week day	daynumber_since_year_2014	is_holiday	days_till_holidays	*sales*
01.01.14	5	0	True	0	*1213*
02.01.14	6	1	False	3	*938*
03.01.14	0	2	False	2	*2448*
04.01.14	1	3	False	1	*1744*
05.01.14	2	4	True	0	*1732*
06.01.14	3	5	False	9	*1022*

https://www.coursera.org/learn/competitive-data-science

Date	weekday	daynumber_since_year_2014	is_holiday	days_till_holidays	*sales*
01.01.14	5	0	True	0	*1213*
02.01.14	6	1	False	3	*938*
03.01.14	0	2	False	2	*2448*
04.01.14	1	3	False	1	*1744*
05.01.14	2	4	True	0	*1732*
06.01.14	3	5	False	9	*1022*

user_id	registration_date	*last_purchase_date*	*last_call_date*	date_diff	churn
14	10.02.2016	21.04.2016	26.04.2016	5	0
15	10.02.2016	03.06.2016	01.06.2016	-2	1
16	11.02.2016	11.01.2017	11.01.2017	1	1
20	12.02.2016	06.11.2016	08.02.2017	94	0

월별 주기성이 있는가? 없음.
(80~85 한정)



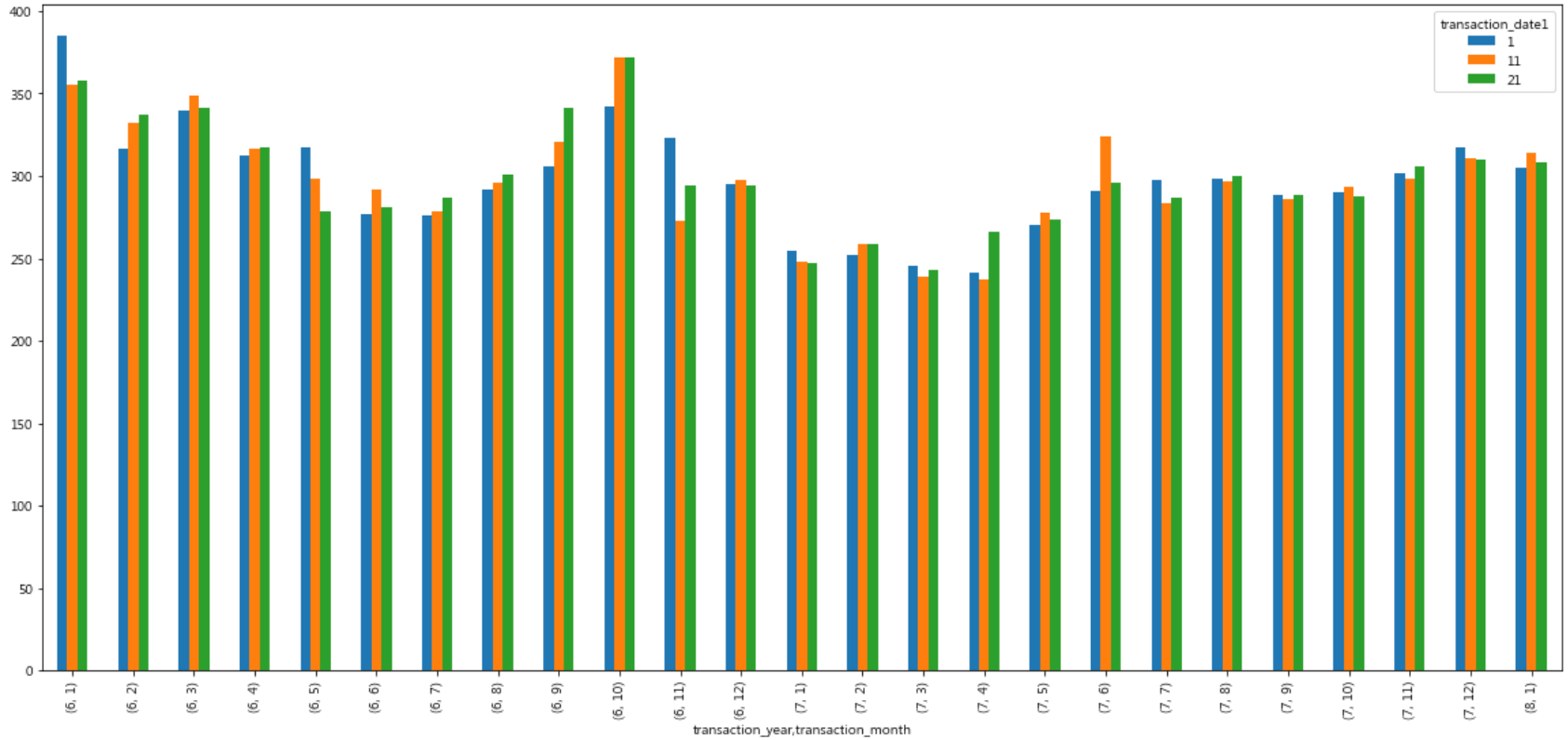
4	5	6	7	8	9	10	11	12
13.000000	13.000000	13.000000	13.000000	13.000000	13.000000	13.000000	12.000000	12.000000
402.195498	408.980250	406.655542	419.707532	414.909906	419.535341	403.017734	402.811302	419.812879
101.008033	108.562971	99.591448	129.673318	130.147177	120.492334	98.601084	115.443423	108.456962
241.863522	261.730044	273.974214	265.695344	277.394286	267.396881	259.107841	280.591372	293.010330
325.631839	336.617902	325.615704	333.877009	323.383446	345.877776	356.259508	304.148912	347.708717
395.692731	390.265324	391.577022	391.795026	393.065966	391.115623	394.620135	399.266546	412.251820
434.714840	441.766721	442.346827	450.056954	438.018380	457.315553	439.435229	426.023665	438.670585
576.354954	628.243574	576.588205	707.643921	751.930721	653.478283	625.181986	694.966251	708.436837

12개월 최대값 횟수

12 4
10 2
5 2
1 2
9 1
8 1
7 1

12 6
8 2
1 2
11 1
10 1
5 1

일별 주기성이 있는가? 없음
(80~85 한정)

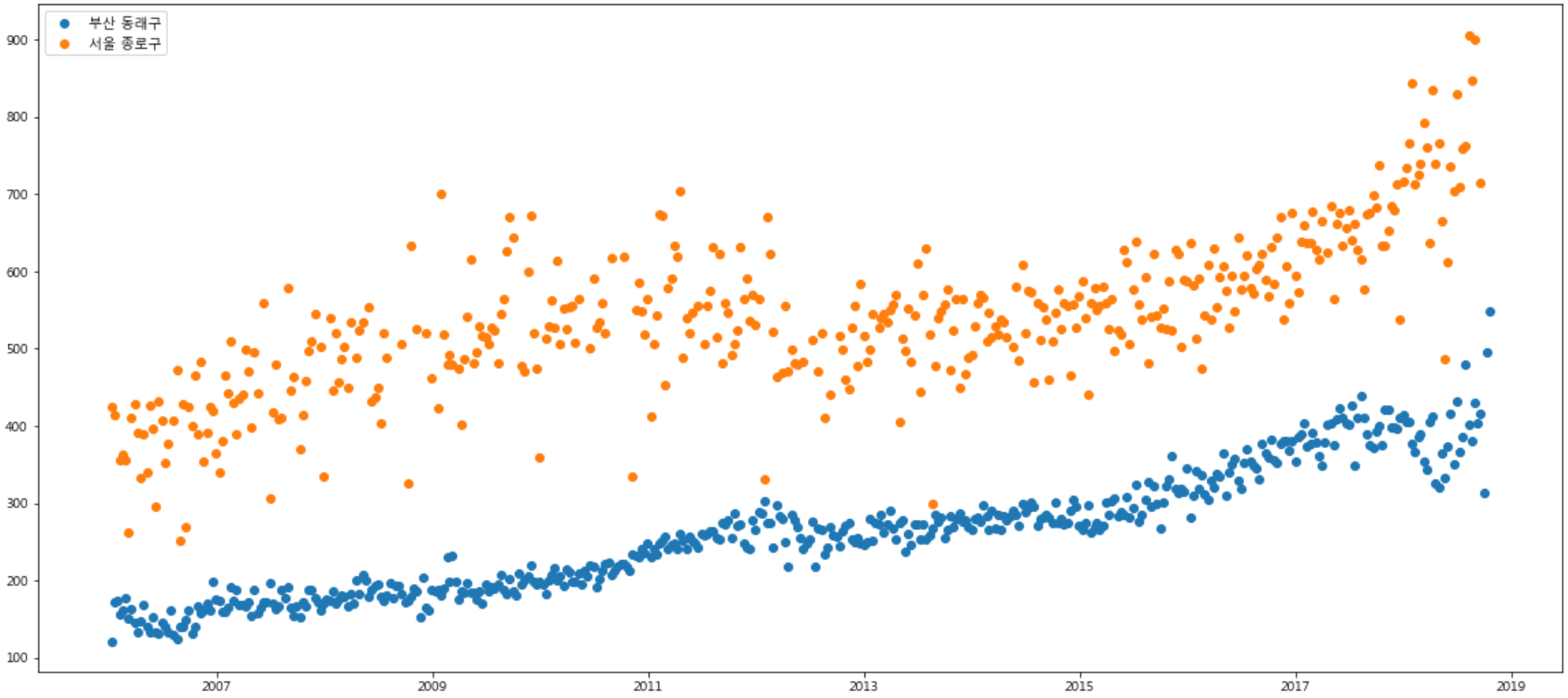


_date1	1	11	21
count	154.000000	154.000000	154.000000
mean	408.988102	409.762186	409.863645
std	103.533128	102.623948	105.966424
min	241.790322	237.662349	236.419958
25%	334.043309	336.989845	333.688443
50%	402.076889	403.086961	396.267199
75%	446.721768	444.225796	450.584054

3분기 최대값 횟수
1~10 59회
11~20 38회
21~30 57회

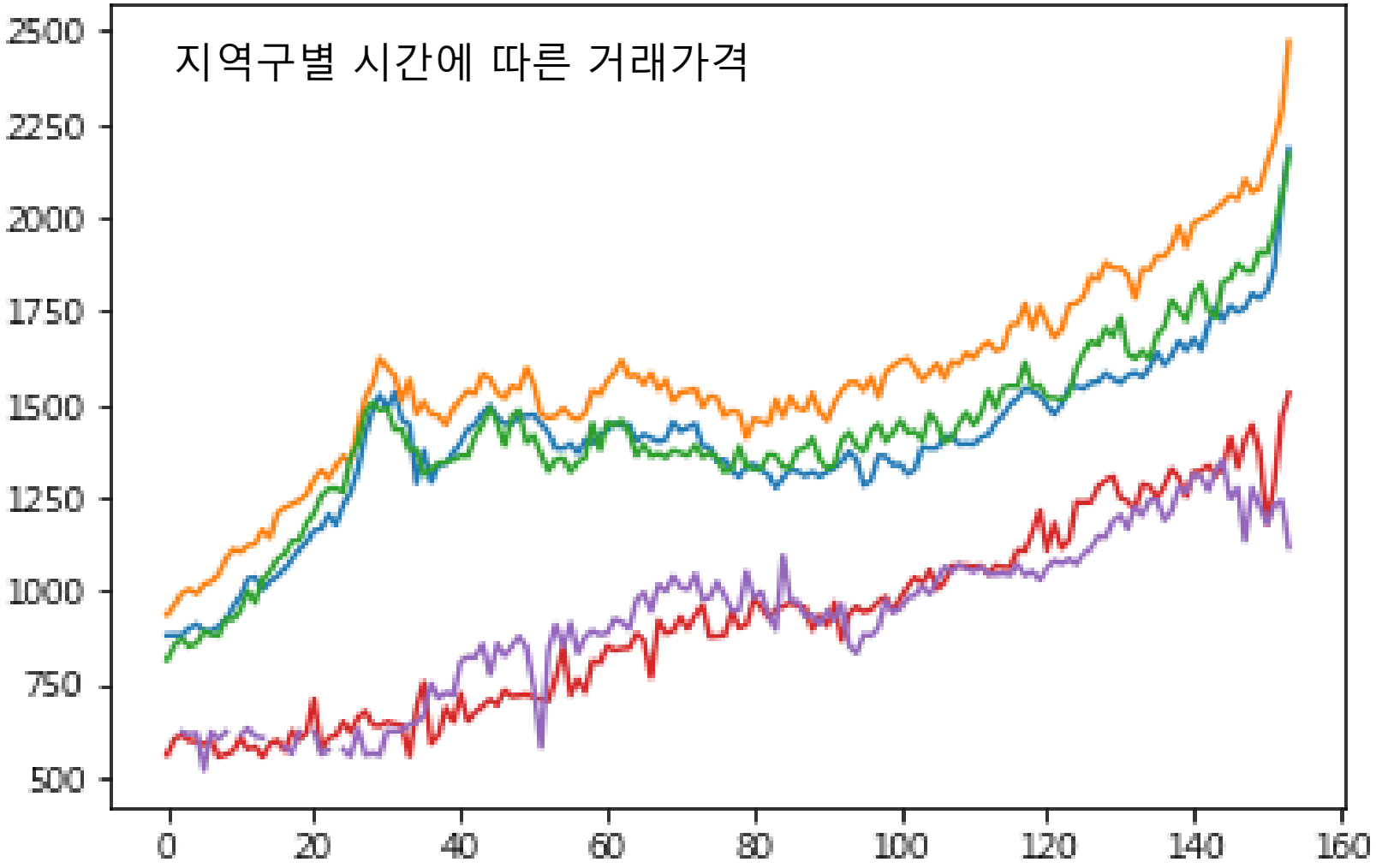
3-2. 지역 또한 매우 중요한 특성

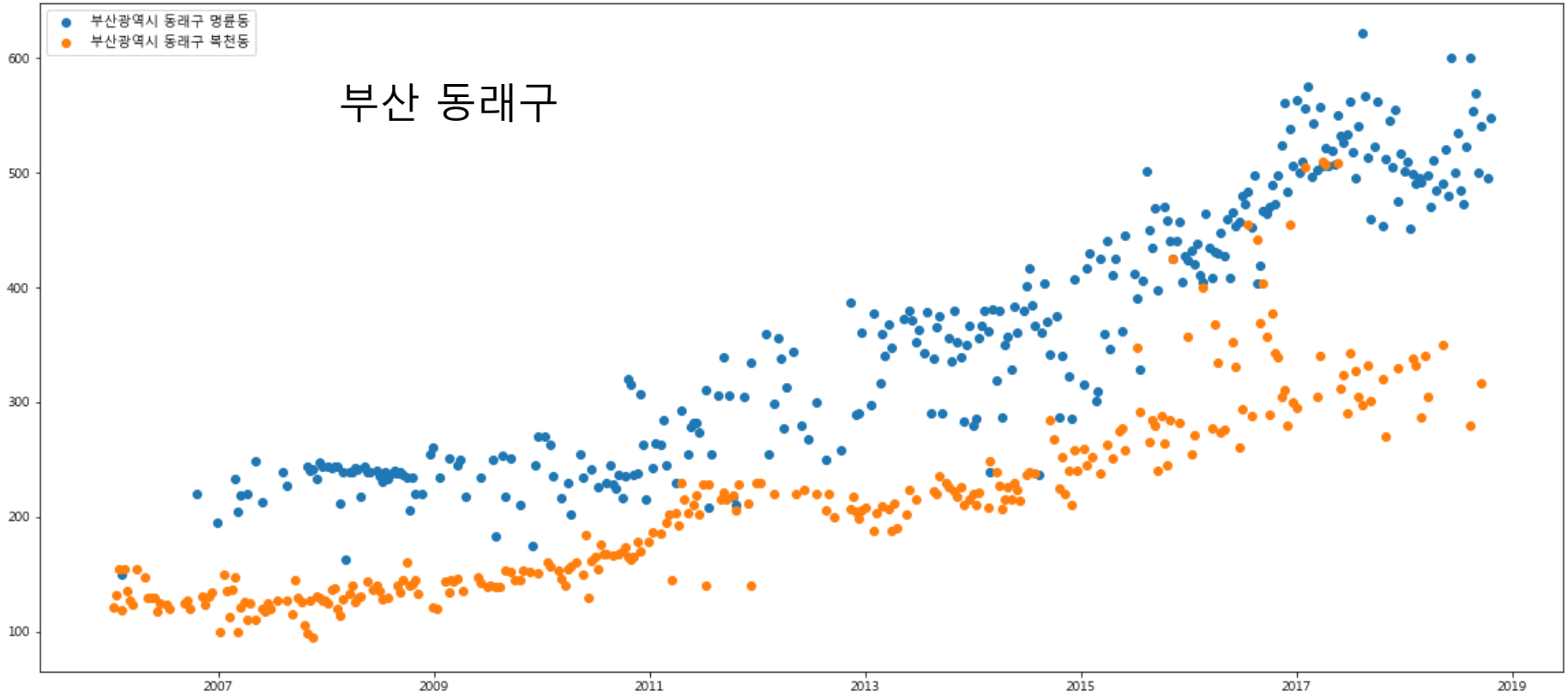
서울, 부산 합쳐서
총 41개의 지역구가 있으며 370개 이상의 법정동이 있음.
지역(위치)간의 가격차이를 구분할 수 있는 특성은?
370개를 모두 인코딩할 수 없다.



동일평형 84~85 -> 지역구 구분 (부산동래구 / 서울종로구)

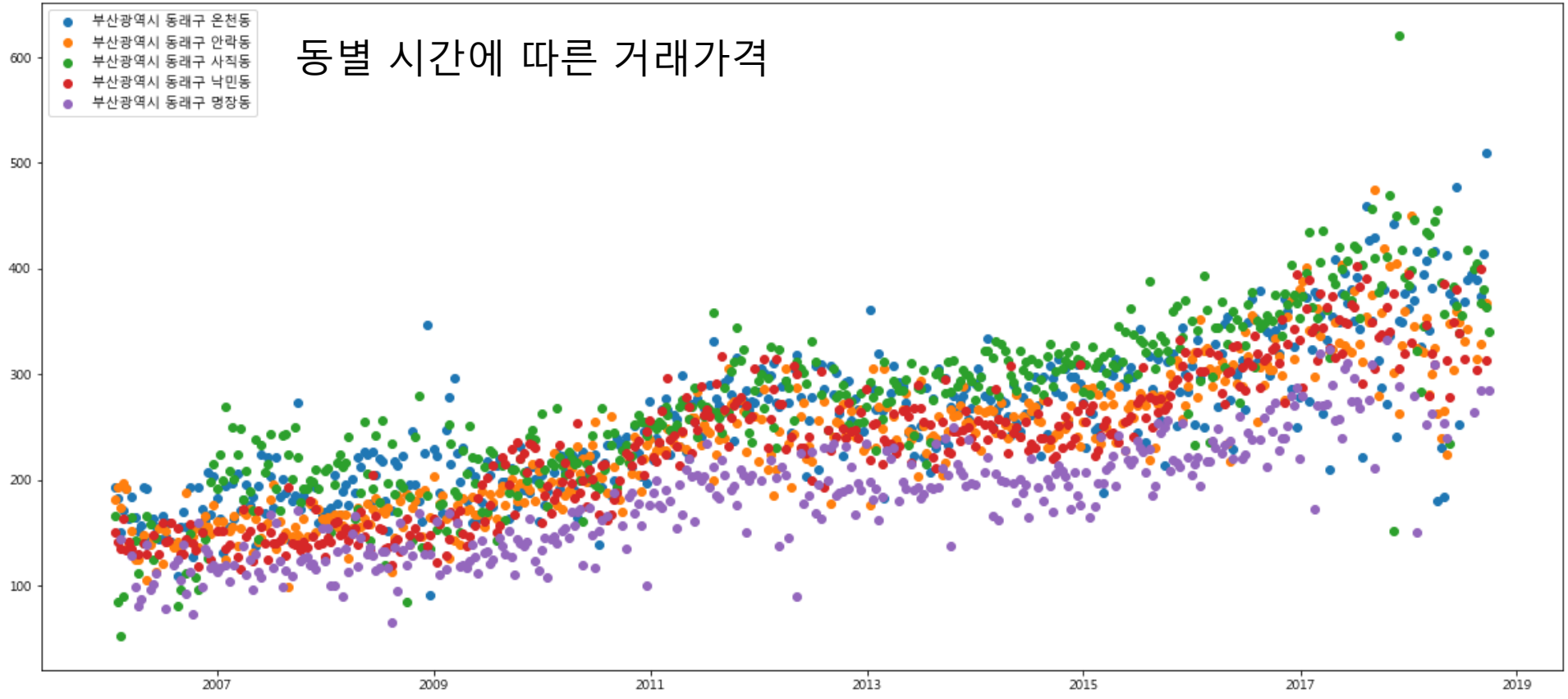
지역구마다 값의 차이가 있음.

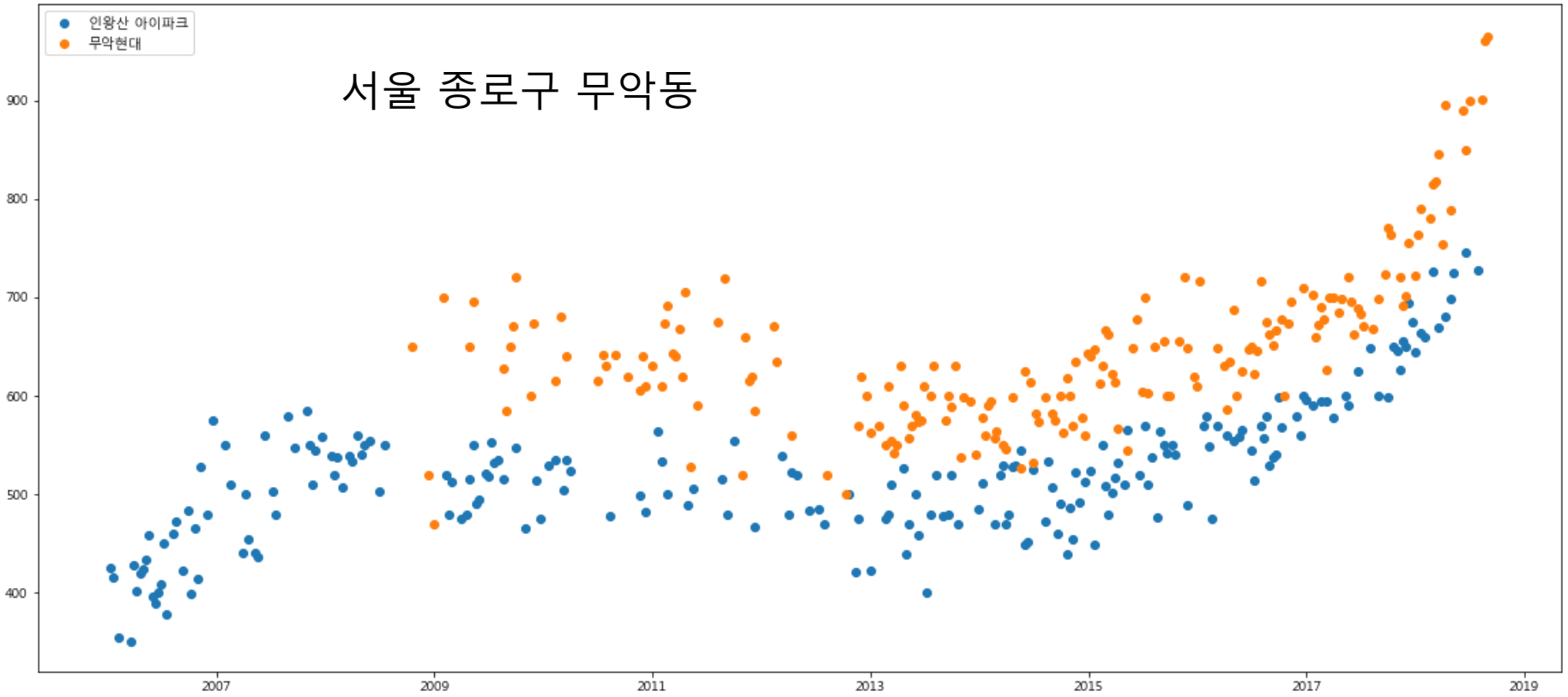

지역구별 시간에 따른 거래가격
2500
2250
2000
1750
1500
1250
1000
750
500
0
20
40
60
80
100
120
140
160



동일평형 84~85 -> 동일지역구 -> 법정동 구분 (명륜동/복천동)

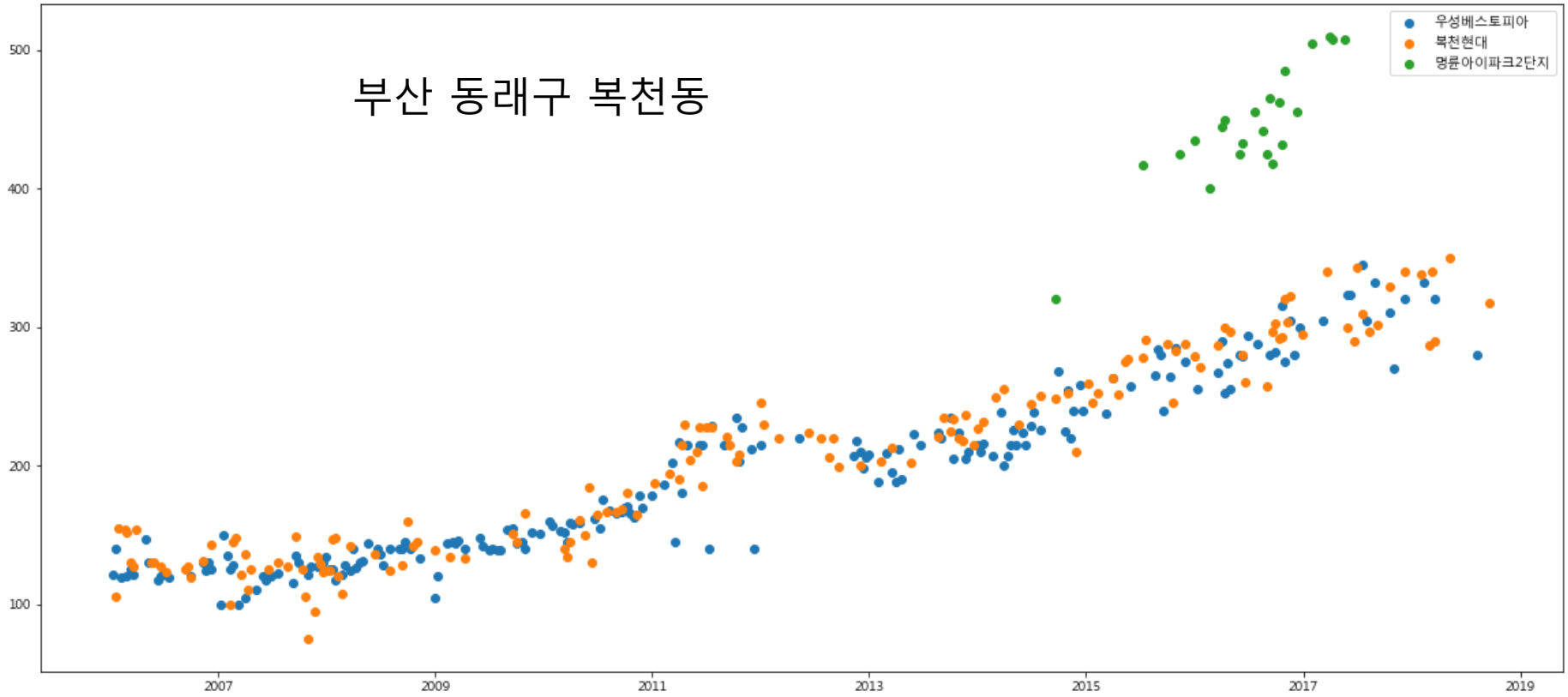
같은 지역구내에서도 법정동마다 값의 차이가 있음.


부산광역시 동래구 온천동
부산광역시 동래구 안락동
부산광역시 동래구 사직동
부산광역시 동래구 낙민동
부산광역시 동래구 명장동
동별 시간에 따른 거래가격
600
500
400
300
200
100
2007
2009
2011
2013
2015
2017
2019



동일평형 84~85 -> 동일법정동 -> 아파트 구분

아파트 신축/구축, 세대수, 층수, 지하철/학교위치 등 세부적인 차이


부산 동래구 복천동
우성베스토피아
복천현대
명륜아이파크2단지
500
400
300
200
100
2007
2009
2011
2013
2015
2017
2019

4. 앙상블의 위력

조금이라도 성능을 높이고 싶을때는 여러 개 모델을 만들어서 평균을 내라.

(신경망 조금씩 다르게 꾸민 다음 평균)
(6천6백, 6천2백, 6천4백, 6천7백 ..... 평균 = 5천7백)

투표기반분류: 여러 분류기의 값을 모아 다수결 투표 또는 평균
배깅: 훈련세트를 무작위로 구성하여 동일한 분류기로 각기 다르게 학습시킴.
부스팅: 앞의 모델을 보완해나가면서 일련의 예측기를 학습시킴.
스태킹: 투표기반분류의 개선판

참고: https://www.slideshare.net/jeanbaptiste.dumont/the-ai-rush-121047435?next_slideshow=1

[참고] 투표기반분류

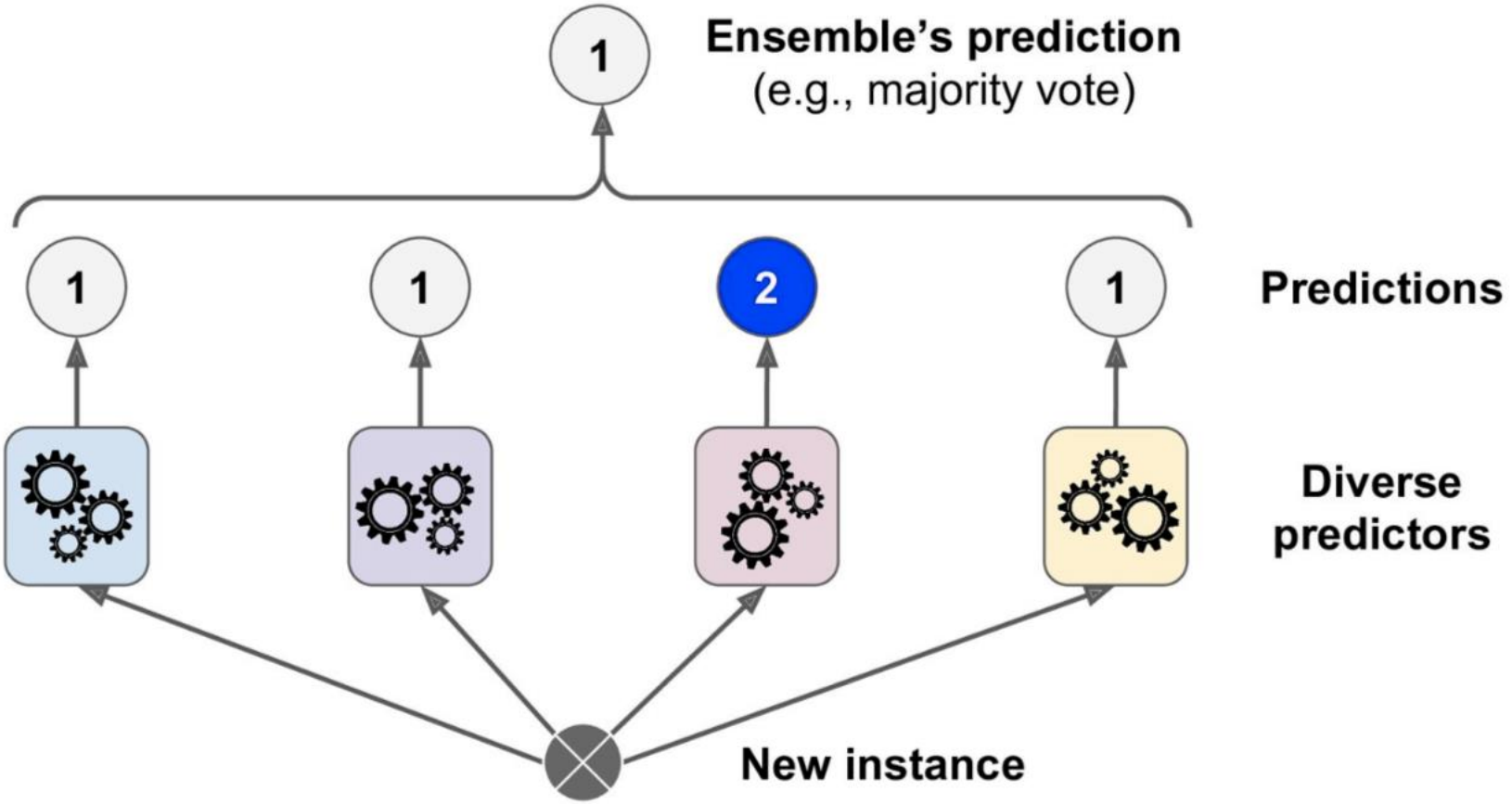
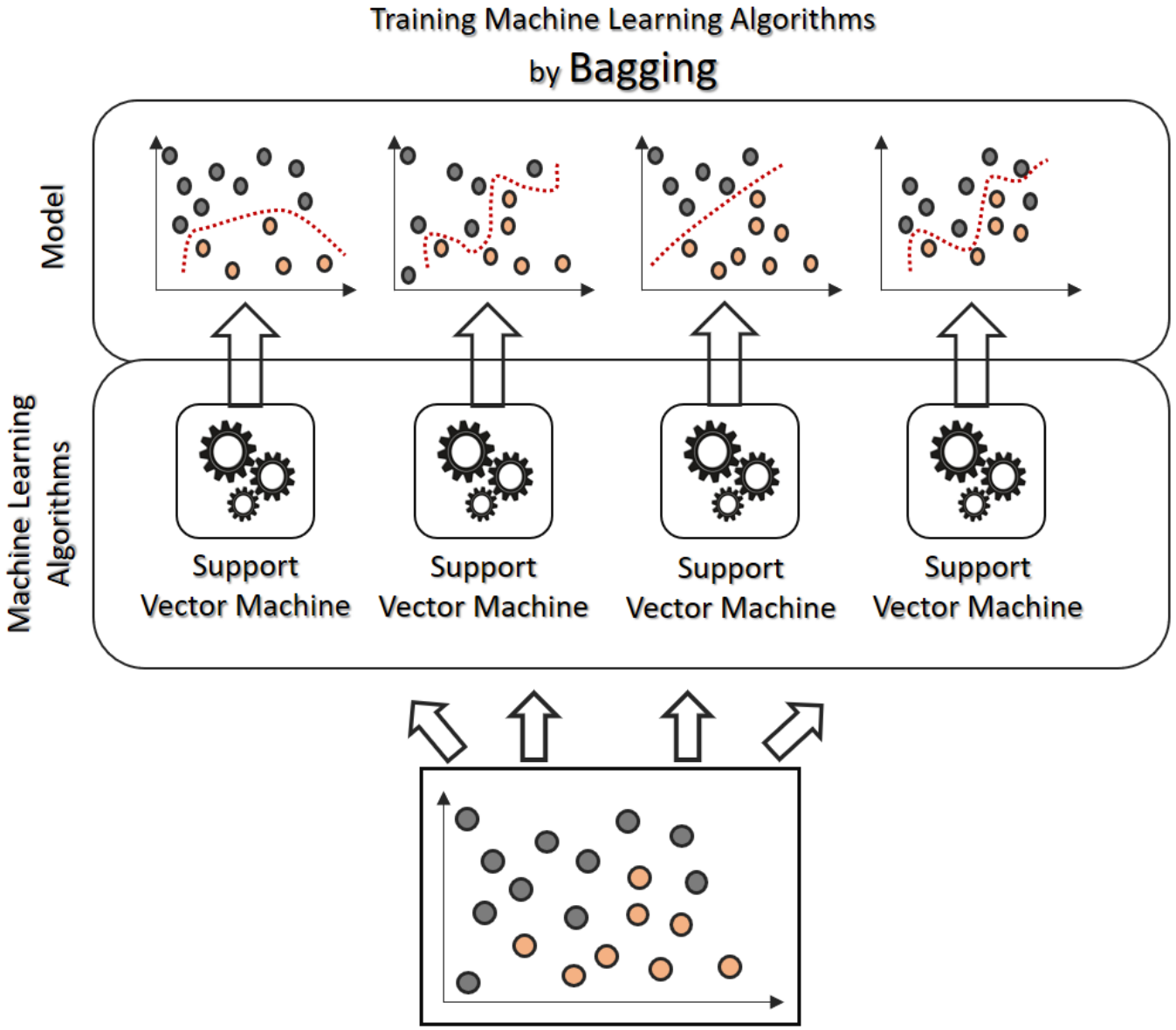
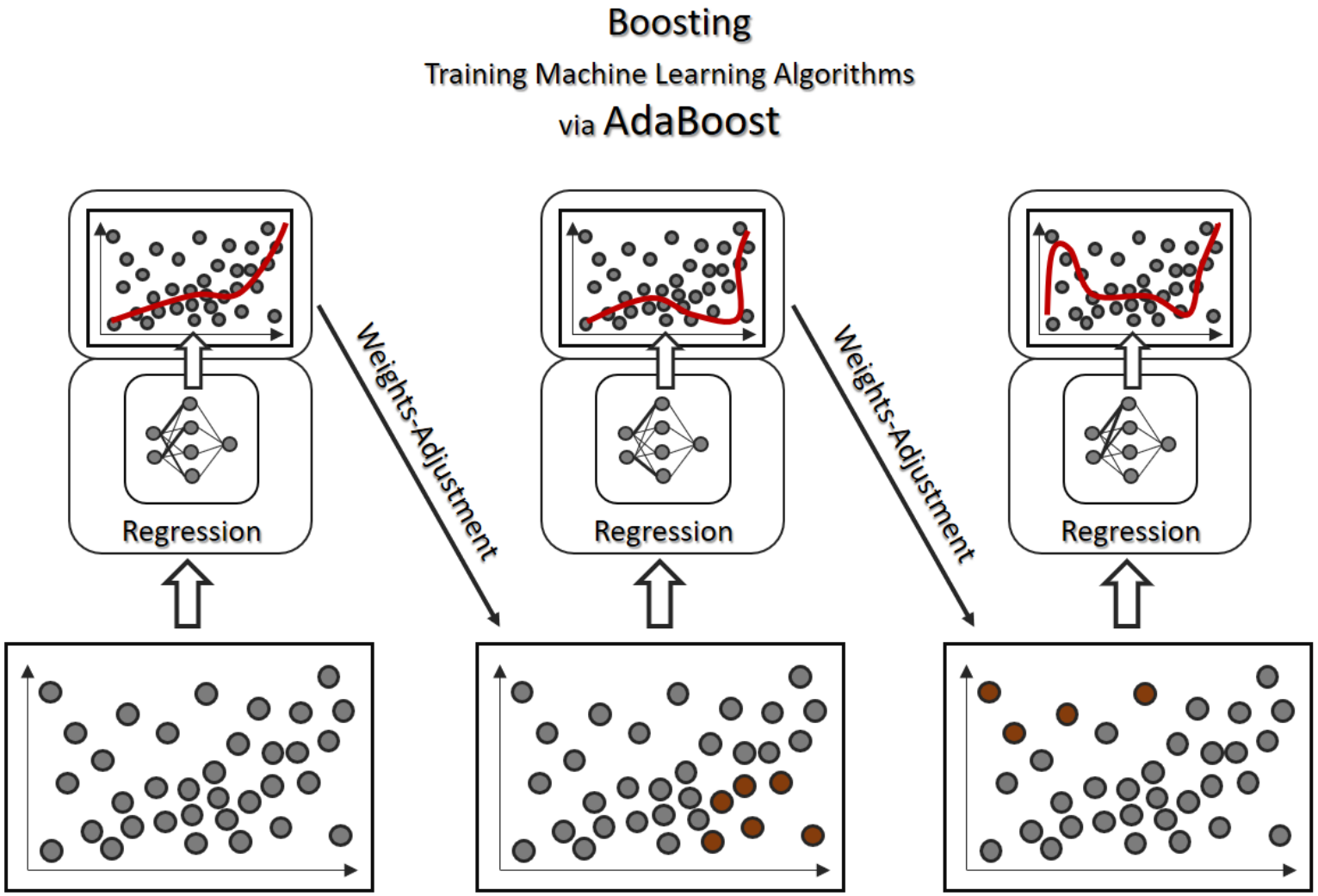


Figure 7-2. Hard voting classifier predictions

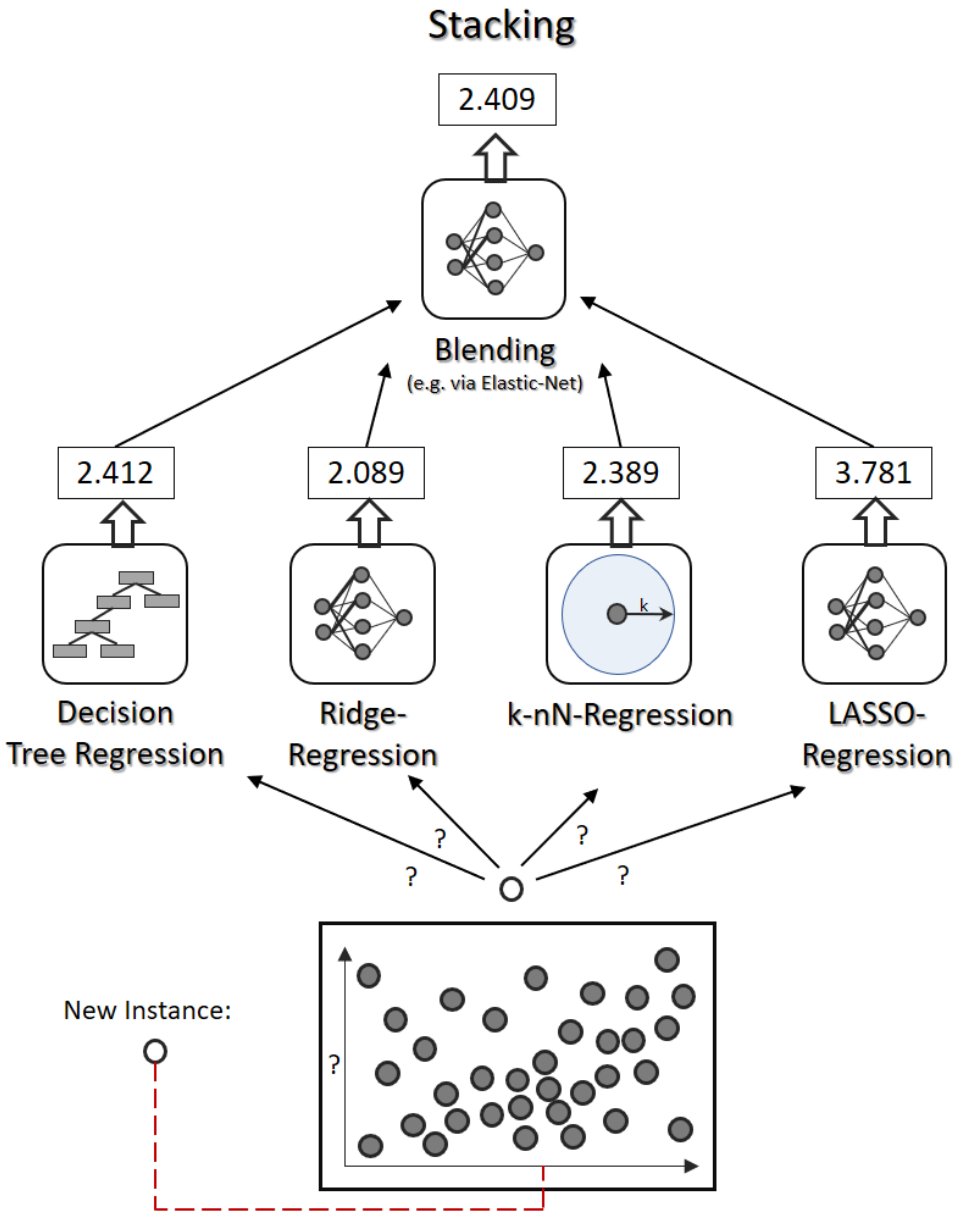
[참고] 배깅


Training Machine Learning Algorithms
by Bagging
Model
Machine Learning Algorithms
Support Vector Machine
Support Vector Machine
Support Vector Machine
Support Vector Machine

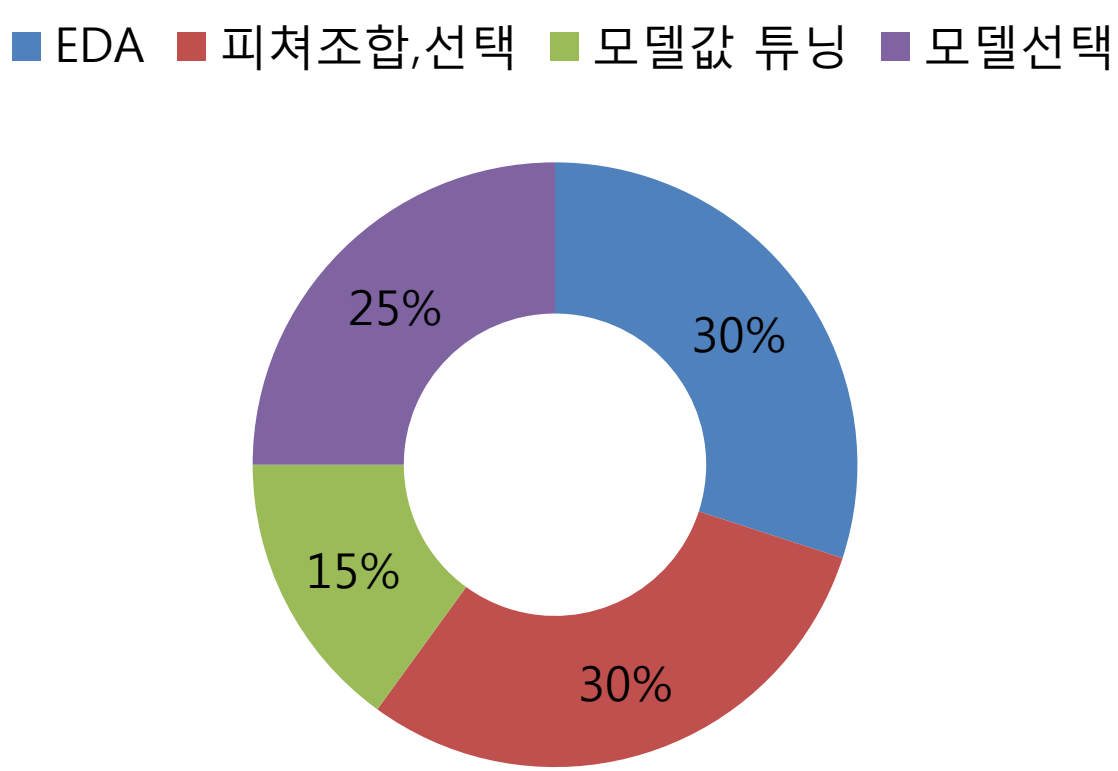
[참고] 부스팅


Boosting
Training Machine Learning Algorithms
via AdaBoost
Regression
Weights-Adjustment
Regression
Weights-Adjustment
Regression

[참고] 스태킹


Stacking
2.409
Blending
(e.g. via Elastic-Net)
2.412
2.089
2.389
3.781
k
Decision
Tree Regression
Ridge-
Regression
k-nN-Regression
LASSO-
Regression
?
?
?
?
New Instance:
?

5. 노가다가 반이다.



여러 피쳐조합과 다양한 모델들을 시험해야함.
모델자체의 파라미터값을 변경해가며 성능을 높임.
이런 노가다작업들을 자동화해야한다.

6. 현재 진행

1. Dacon, 4th competition, 아파트 거래가격 , 다음기회를

2. Kaggle , 타이타닉 생존자, 티셔츠, 공부차원

3. 캐글 커널 공부, 대회 꾸준히 참여

4. 여기서 공부한 내용을 본래의 목적인 주가예측에 적용하자. (main)

7. 결과

DACON Home Competition WinnerCode Community About us 츄

19	quatroappa	58,663,902.50794	2018-12-25 10:31:04
20	SachinKarmani	59,290,055.68309	2019-01-16 03:04:01
21	WJteam	59,407,441.64173	2019-01-24 10:05:42
22	호다닥	60,280,065.44538	2019-01-30 18:11:16
23	hongse	60,459,353.90819	2019-01-22 17:53:15
24	namu2018	60,569,998.92446	2019-01-31 17:53:07
25	claudkim(claudkim)	60,953,046.83016	2019-01-30 18:30:37
26	TAS	61,512,495.30823	2019-01-31 22:03:43
27	Jordana	61,842,849.42295	2019-01-15 18:26:07
28	윤동균	62,411,724.97471	2019-01-31 09:30:54
29	aiodia	62,859,769.29302	2019-01-02 07:47:03
30	김규진	63,515,571.99073	2018-12-10 18:50:12
31	TooDock_Lab	64,369,254.50081	2018-12-14 13:00:14
32	jhp	65,626,040.80520	2018-11-21 15:27:44

31 등 / 143 팀
rmse : 6,400만원
(순위권이 약 4,000 ~ 5,000만원)