공사 원가 예측

경기도시공사

경기주택도시공사

K data
한국데이터산업진흥원

Introduction
Team

경기주택도시공사

김동규
IT경영학과

정수진
컴퓨터공학과

황혜지
통계학과

Contents

Background

Issue

Goal

Preprocessing

Flowchart

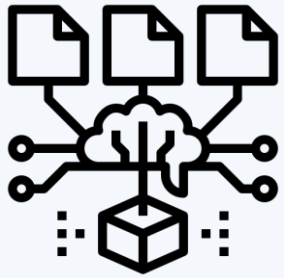
Raw Data

Problem

Levenshtein distance

Clustering

Derived Value



Modeling

Random Forest

Cuckoo Search

LightGBM

Bayesian Optimization

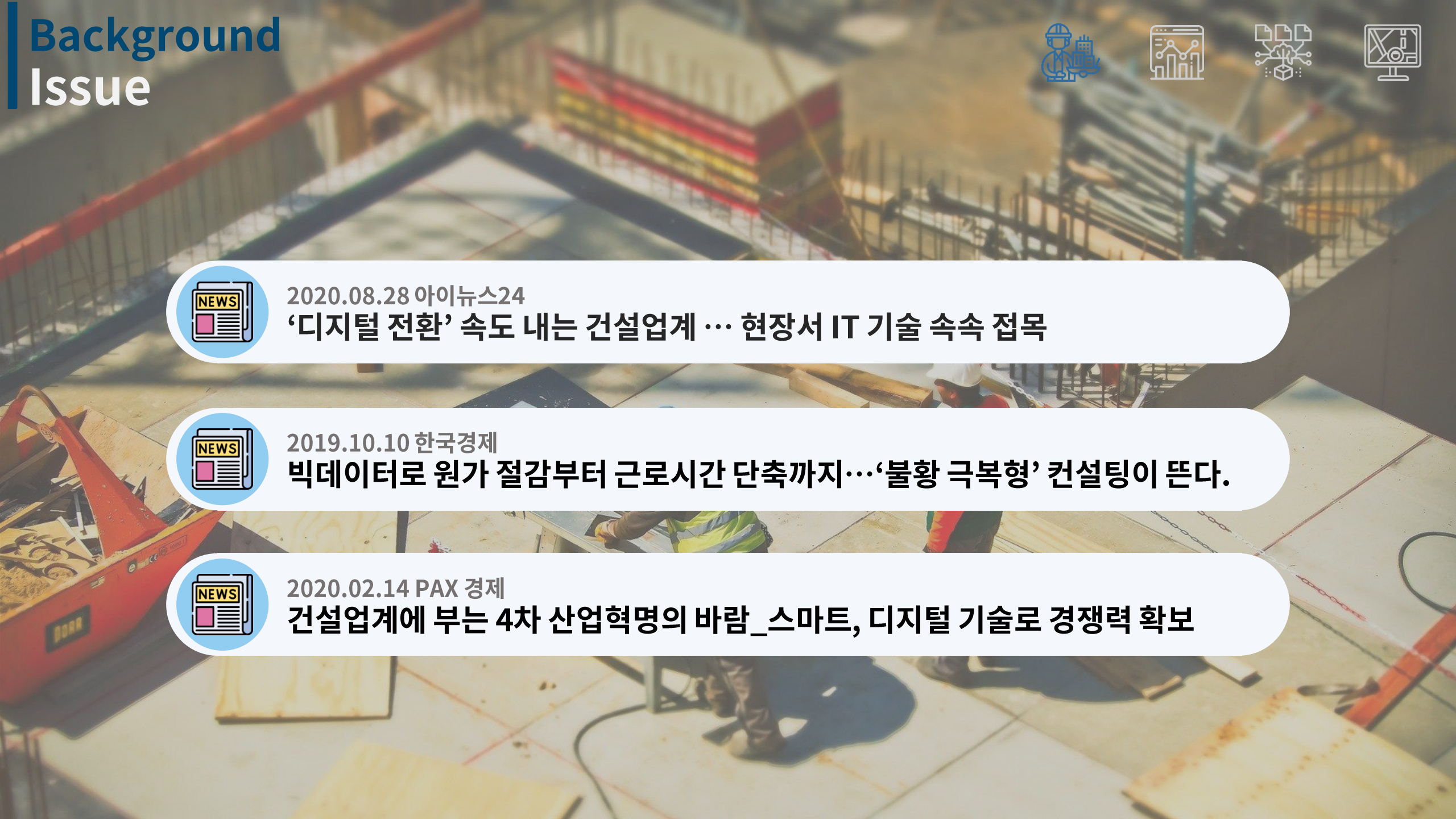
Result

R2 score

simulation

Reference & Tool

Background
Issue



2020.08.28 아이뉴스24
‘디지털 전환’ 속도 내는 건설업계 … 현장서 IT 기술 속속 접목

2019.10.10 한국경제
빅데이터로 원가 절감부터 근로시간 단축까지…‘불황 극복형’ 컨설팅이 뜬다.

2020.02.14 PAX 경제
건설업계에 부는 4차 산업혁명의 바람_스마트, 디지털 기술로 경쟁력 확보

Background
Problem

설계단가

공사 시작 전 예산 산정

도급단가

외주 업체가 산정한 실질적 단가

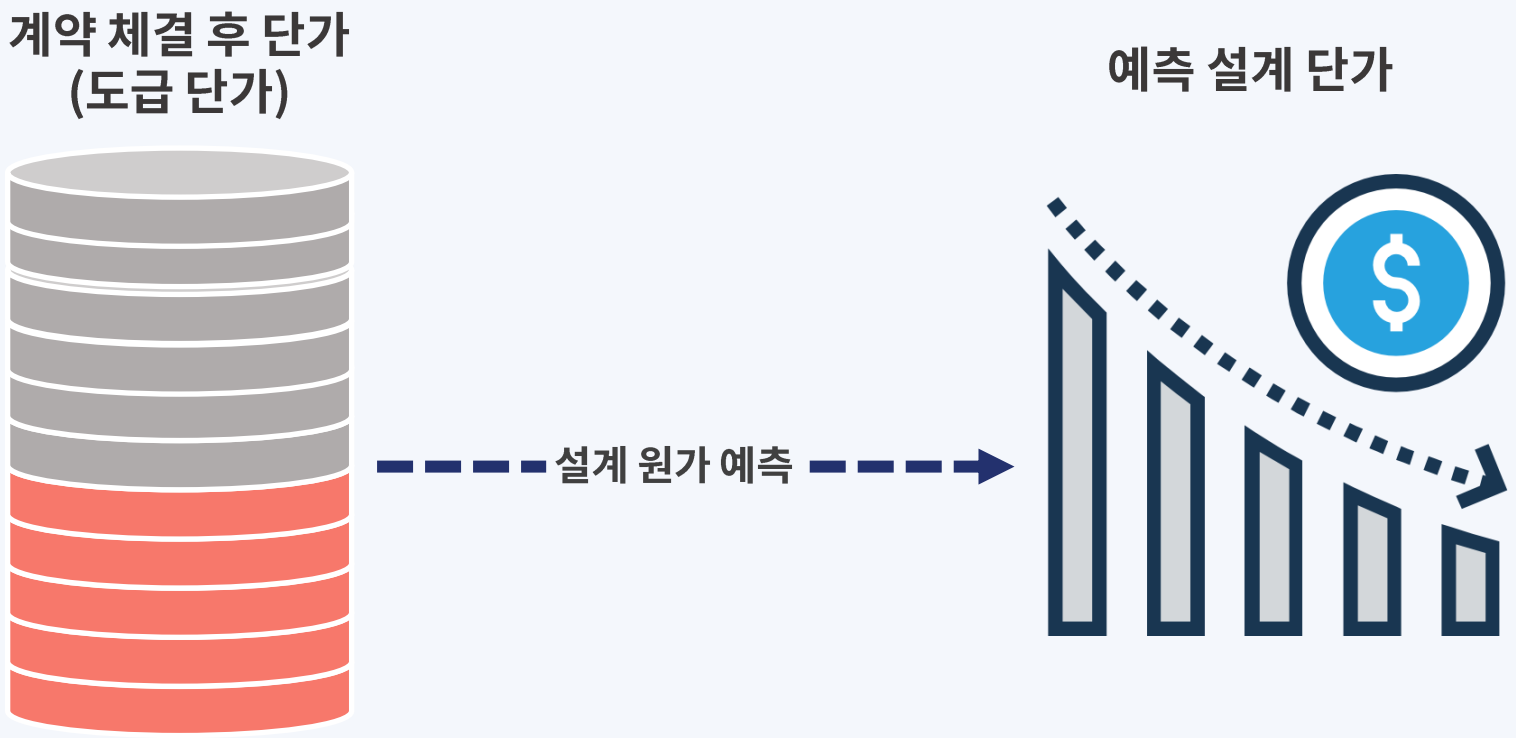
Background
Goal

설계단가가 클수록 도시공사 입장에서 예산 쓰게 되어 손실이 발생

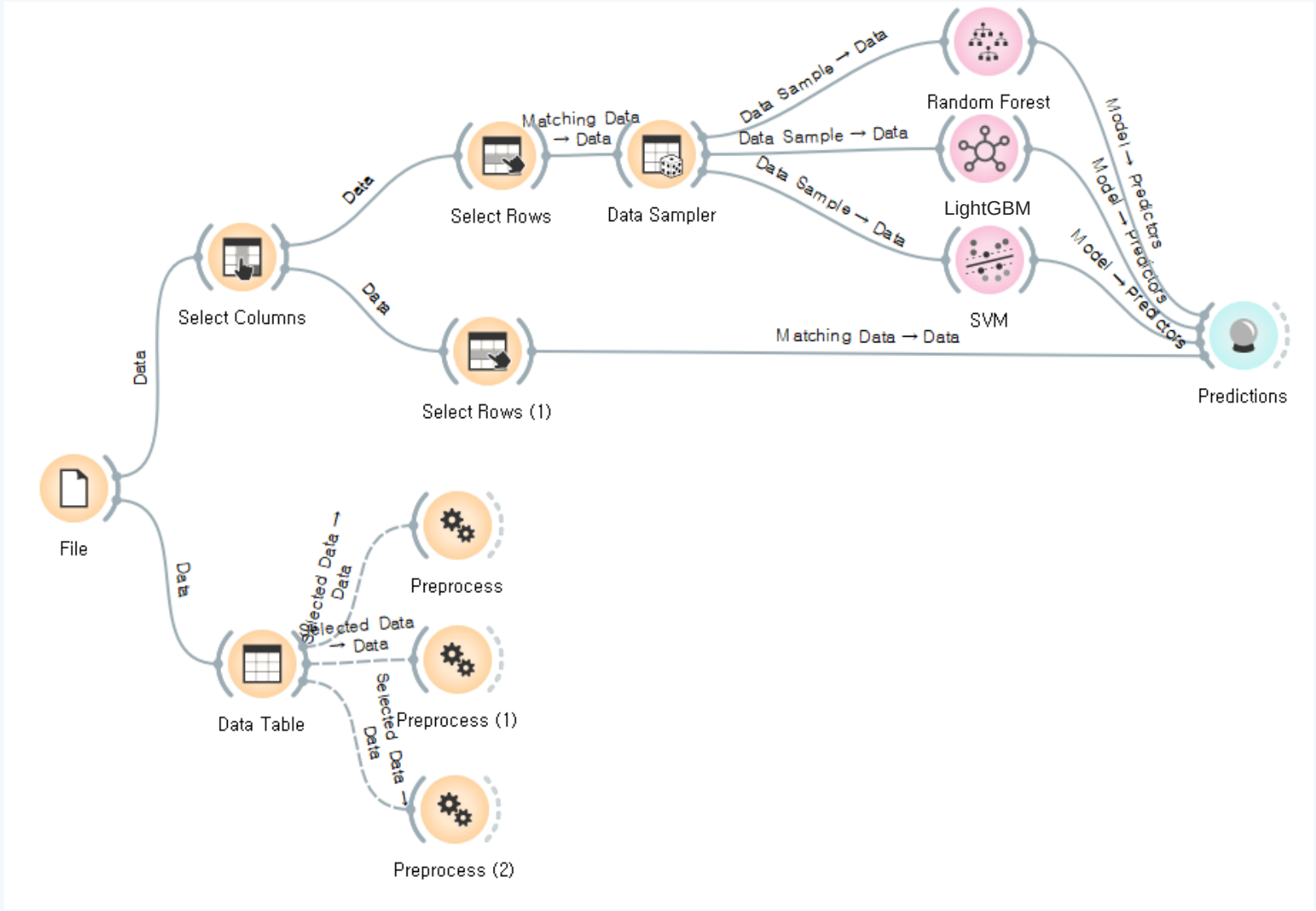
설계 단가를 무작정 낮게 잡을 경우 예산 부족 현상 발생

Background
Goal

도급 단가를 이용한 “설계 단가” 예측을 통한 예산 절감 시스템



Flow


File
Data
Select Columns
Data
Select Rows
Matching Data → Data
Data Sampler
Data Sample → Data
Random Forest
Data Sample → Data
LightGBM
Data Sample → Data
SVM
Model → Predictors
Model → Predictors
Model → Predictors
Predictions
Data
Select Rows (1)
Matching Data → Data
Data
Data Table
Selected Data → Data
Preprocess
Selected Data → Data
Preprocess (1)
Selected Data → Data
Preprocess (2)

Preprocessing
RowData

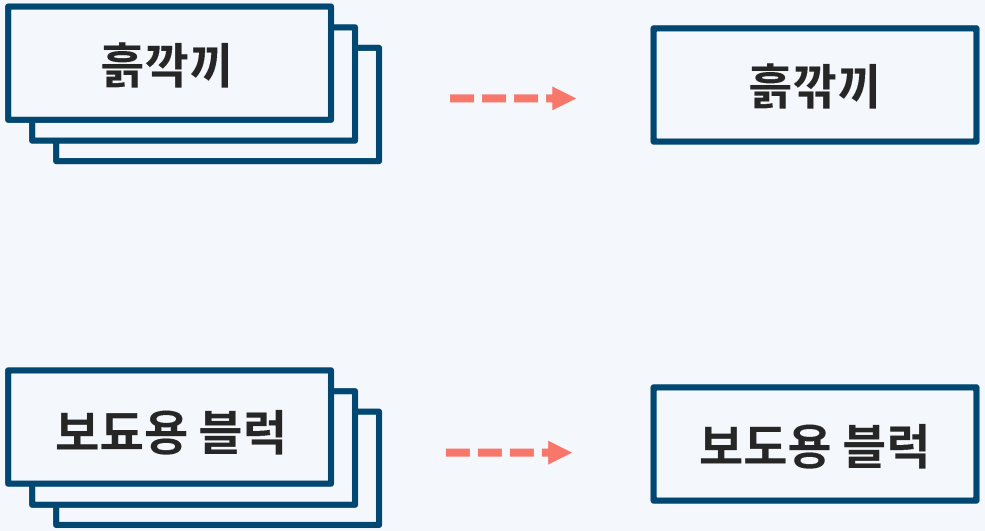
	A	B	C	D	E	F	G	H	I	J	K	L
1	공 종 명	규 격	수량	단위	재 료 비		노 무 비		경 비		합 계	
2					단 가	금 액	단 가	금 액	단 가	금 액	단 가	금 액
3	광교지구 택지개발사업(원가계산)		1	식		40,891,831,357		12,556,737,699		29,229,959,551		82,678,528,607
4	1. 토 공		1	식		6,091,964,729		4,807,955,575		3,585,508,671		14,485,428,975
5	1.01 벌개제근		0					50,239,314				50,239,314
6	1) 벌개제근		2E+05	M2			207	50,239,314			207	50,239,314
7	1.02 표토제거		0			291,060		190,080		219,780		700,920
8	1) 표토제거		2970	M2	98	291,060	64	190,080	74	219,780	236	700,920
9	1.03 지장물 철거		0			72,381,438		258,498,676		172,866,914		503,747,028
10	1) 터파기(토사)	백호우1.0M3	6048	M3	218	1,318,464	1,835	11,098,080	256	1,548,288	2,309	13,964,832
11	2) 되메우기		6048	M3	197	1,191,456	151	913,248	162	979,776	510	3,084,480
12	3) 파이프 철거		0			922,306		37,819,756		1,233,946		39,976,008
13	파이프 철거	D500MM미만	2095	M			9,374	19,638,530			9,374	19,638,530
14	파이프 철거	D500MM이상	742	M	1,243	922,306	24,503	18,181,226	1,663	1,233,946	27,409	20,337,478
15	4) 콘크리트		0			24,653,334		141,003,078		98,804,433		264,460,845
16	기존구조물철거공,콘	T=30cm이상	4882	M3	1,607	7,845,374	22,119	107,984,958	13,609	66,439,138	37,335	182,269,470
17	기존구조물철거공,콘	T=30cm미만	345	M3	582	200,790	17,821	6,148,245	10,237	3,531,765	28,640	9,880,800
18	기존구조물철거공,콘	T=30cm미만	4535	M3	3,662	16,607,170	5,925	26,869,875	6,358	28,833,530	15,945	72,310,575
19	2) 포장		0			44,143,760		67,408,149		69,987,244		181,539,153
20	콘크리트포장 깨기	T=30cm미만	2316	M3	5,219	12,087,204	8,184	18,954,144	8,536	19,769,376	21,939	50,810,724
21	아스팔트포장깨기	T=30cm미만	5160	M3	6,211	32,048,760	9,387	48,436,920	9,732	50,217,120	25,330	130,702,800
22	포장절단	콘크리트	3	M	480	1,440	1,103	3,309	44	132	1,627	4,881
23	포장절단	아스팔트	14	M	454	6,356	984	13,776	44	616	1,482	20,748
24	3) 구조물		0			152,118		256,365		313,227		721,710
25	석축헐기	찰쌓기	81	M2	1,878	152,118	3,165	256,365	3,867	313,227	8,910	721,710
26	1.04 비탈면보호		0			290,699,529		142,578,885		35,226,003		468,504,417
27	1) 비탈면보호공		0			234,109,674		114,244,992		30,302,250		378,656,916
28	식생기반재 취부공법	T=3CM,성토부(토	8223	㎡	10,623	87,352,929	5,184	42,628,032	1,375	11,306,625	17,182	141,287,586
29	식생기반재 취부공법	T=3CM,깍기부(토	13815	㎡	10,623	146,756,745	5,184	71,616,960	1,375	18,995,625	17,182	237,369,330

Preprocessing
Problem

	공사명	분포 유사도	이상치
Problem	오타 처리 데이터 규격	독립변수와 종속변수 간의 선형성 붕괴	Regression 특성상 이상치에 민감
Solve	편집거리 알고리즘 코사인 유사도 Hdbscan 클러스터링	Jenson-Shannon divergence Hierarchical Clustering	Domain 특성 이용 이상치 변수 처리

Preprocessing
Typing Error

잦은 오타 발생
통합된 '공종명'에 오타와 표기법의 차이가
있어 하나로 통합하기 위한 과정을 거침



일관성 없는 명칭 및 규격
도시별로 다르게 작성된 설계, 도급 내역서를
하나의 규격으로 통합

도시1	도시2	도시3
1. 토 공	1.토공	토공
.1.01 구조물 깨기	가. 구조물 깨기	구조물 깨기
..1) 콘크리트깨기	1) 콘크리트깨기	콘크리트깨기
... 콘크리트,철근	가) 콘크리트철근	콘크리트/철근

Preprocessing
Levenshtein distance

동규

MIN distance

딩뇨

하나의 문자열을 다른 문자열로
변환하기 위해 필요한 연산의 최소 횟수

JAMO

한글의 자음·모음을 분리하는 Python
라이브러리를 사용하여 연산의 정확도를 높임

	{ }	ㄷ	ㅗ	ㅇ	ㄱ	ㅠ
{ }	0	1	2	3	4	5
ㄷ	1	0	1	2	3	4
ㅣ	2	1	1	2	3	4
ㅇ	3	2	2	1	2	3
ㄴ	4	3	3	2	2	3
ㅛ	5	4	4	3	3	**3**

동규 vs 딩뇨 → 2

ㄷㅗㅇㄱㅠ vs ㄷㅣㅇㄴㅛ → 3

Preprocessing
Levenshtein distance

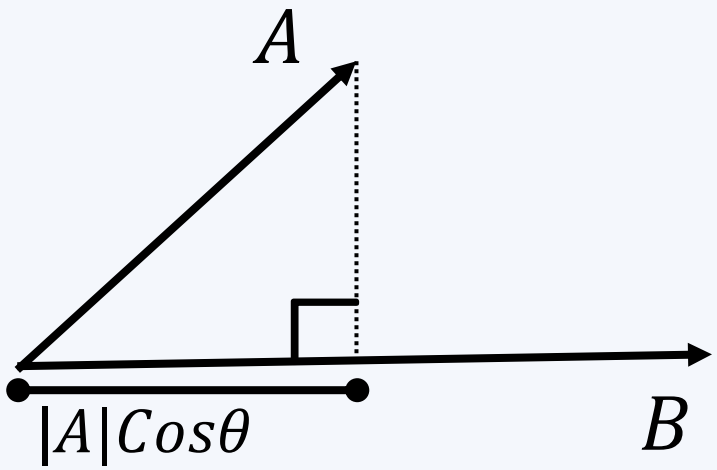
문자열의 편집 거리를 이용하여 편집거리가 3 이하가 되는 것들을 뽑아 공종명에 있는 오타와 표기법을 수정

Count	Original	Distance	Modify
1	보됴용 난간	2	보도용 난간
1	거푸집 유로폼	3	거푸집공 유로폼
1	흙깍기공 리핑암	2	흙깎기공 리핑암
1	강관 말뚝 천공 SDA	3	강관 말뚝 천공
1	재하시험	3	동재하시험

Preprocessing
Clustering

Cosine Similarity

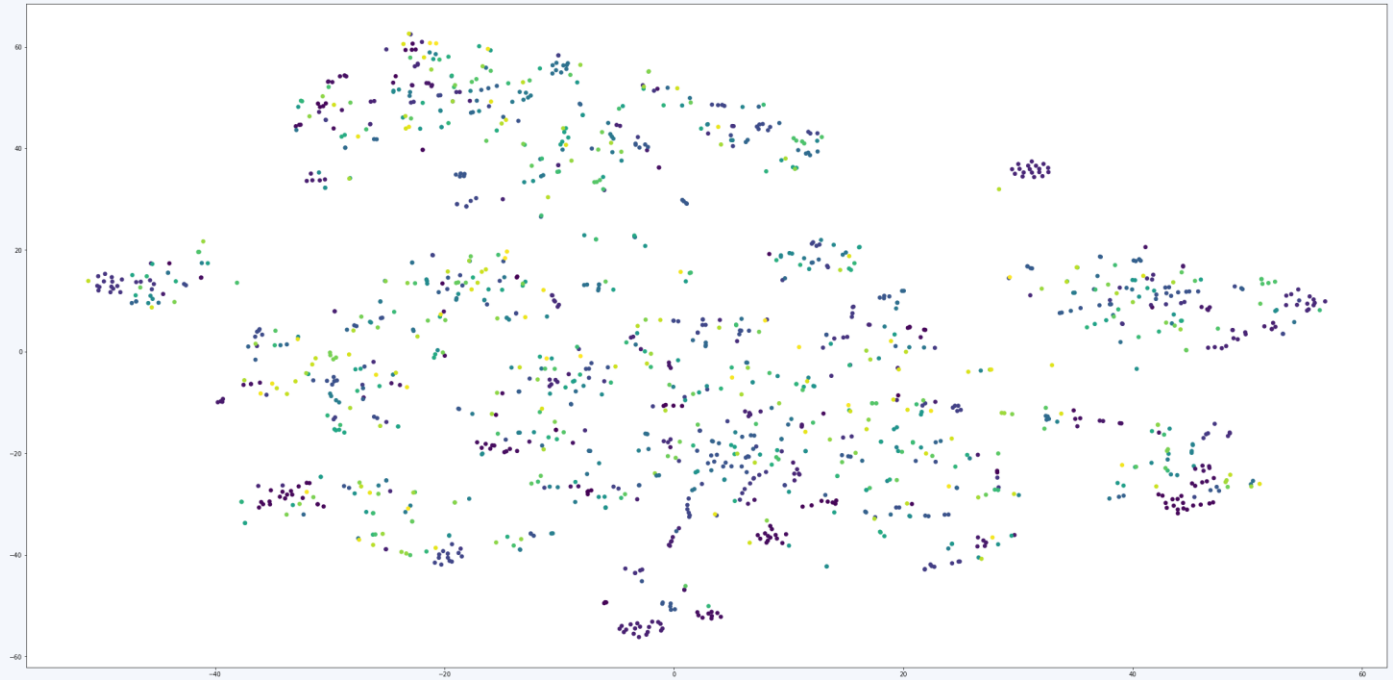
단어간 유사도 계산



HDBScan

밀도기반 클러스터링

비슷한 단어가 많아 경계 값 처리를 위해 사용

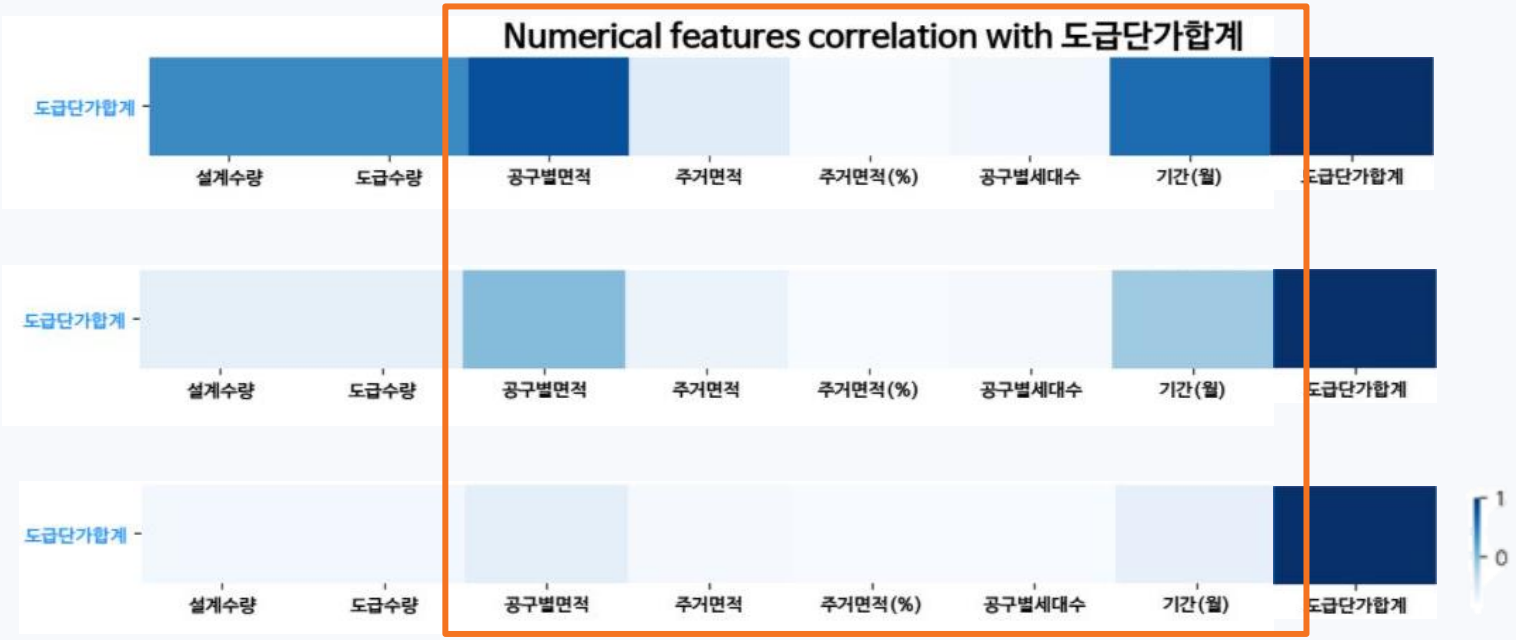


Preprocessing
Correlation Problem

도시

공구

공종명



대규모에서 소규모 단위로 좁혀지면서 외부에서 가져온 변수와 종속변수 간의 선형성이 무너짐

데이터 내부에 관한 고찰

Preprocessing
Jenson-Shannon divergence

$$JSD(P,Q) = JSD(Q,P)$$

KL divergence를 기반
확률 분포 간의 유사도 측정

분포간 거리행렬

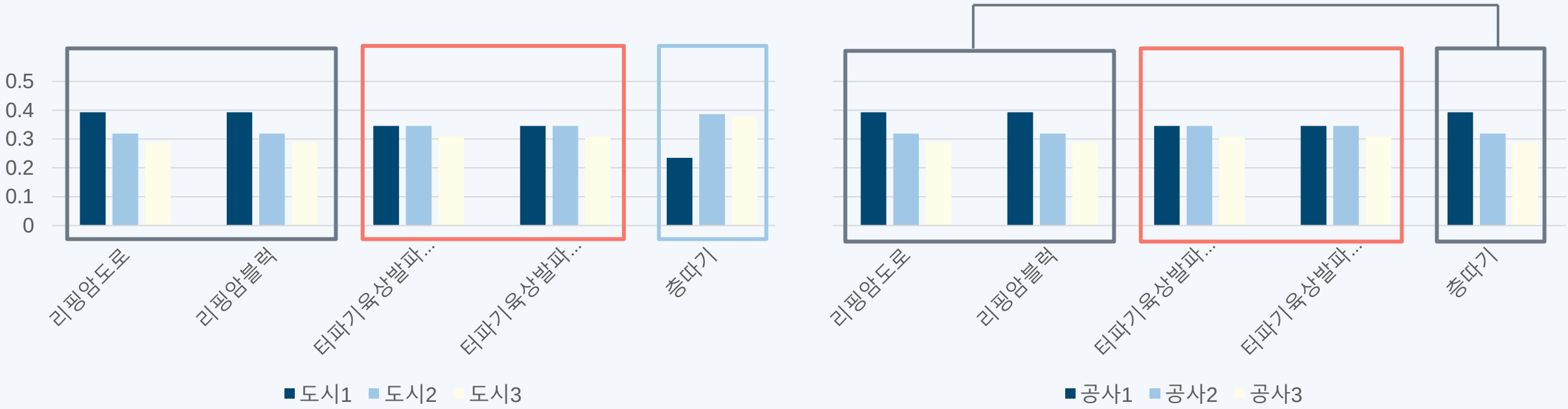
→

거리 측정 가능
대칭(Symmetric)

공종명	2방향 예고표지판	2방향 표지판	2중교통 표지판설치비	90엘보	anchorbolt
2방향예고표지판	0	0	1	1	1
2방향표지판	0	0	1	1	1
2중교통 표지판설치비	1	1	0	0.069877	1
90엘보	1	1	0.069877	0	0.980014
anchorbolt	1	1	1	0.980014	0

Preprocessing
Hierarchical Clustering

X축을 바꿔가며 비슷한 분포의 흐름을 가진 세부 공종명을 묶음



Preprocessing
Derived Value

이상치에 민감한 Regression

공사별로 다른 scale

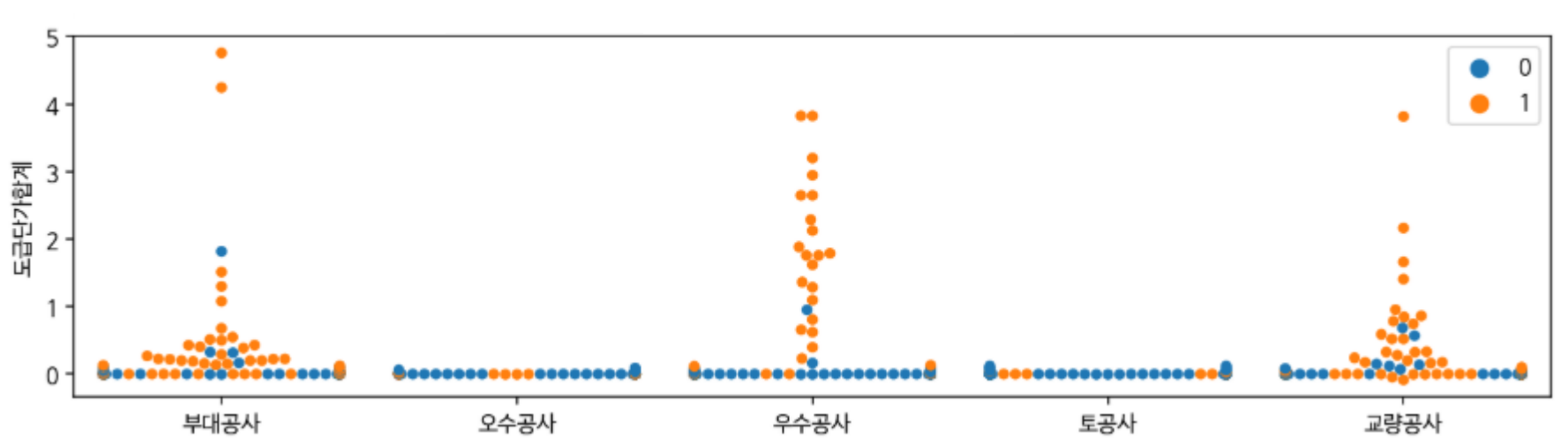
point

이상치 + 설계도급 **단가의 차이가 큰** 공사 찾기

일반적인 outlier 탐색 X

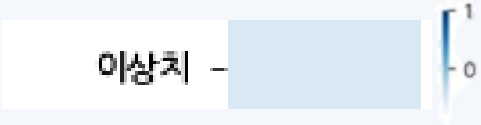
이상치를 판단하는

Binary 변수 생성



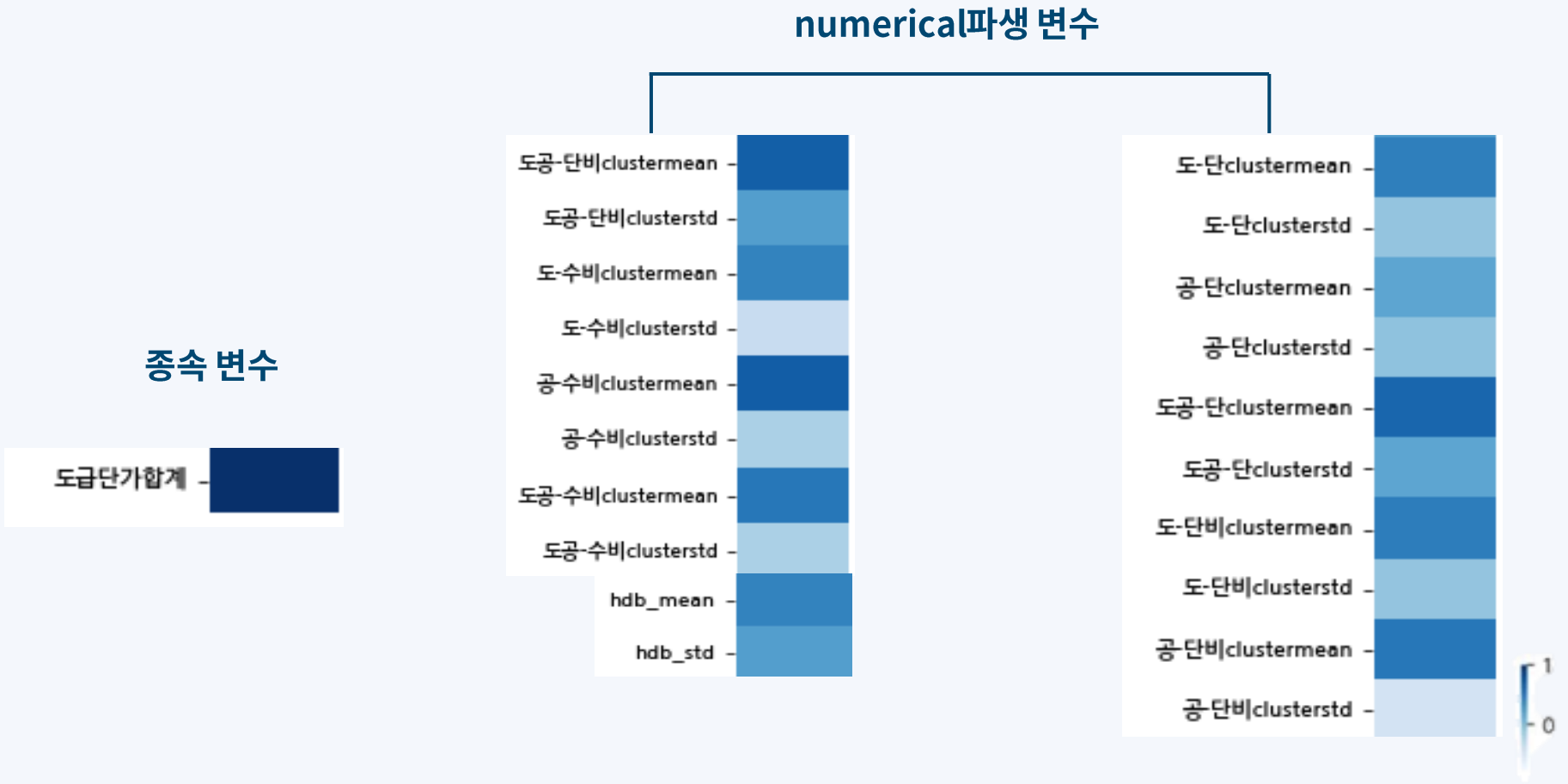
Preprocessing
Derived Value

세가지 접근 방법을 통해 만들어진 변수들

공종명 clustering	단가 clustering	이상치 판단 여부
Hdb_cluster	도-단cluster	
	공-단cluster	
	도공-단cluster	
	도-단비cluster	
	공-단비cluster	
	도공-단비cluster	
	도-수비cluster	
	공-수비cluster	
	도공-수비cluster	

Preprocessing
Derived Value

클러스터링 범주형 변수들의 **대소관계**를 나타낼 수 있는 각 클러스터의 평균값과 표준편차 변수



Modeling

경기주택도시공사

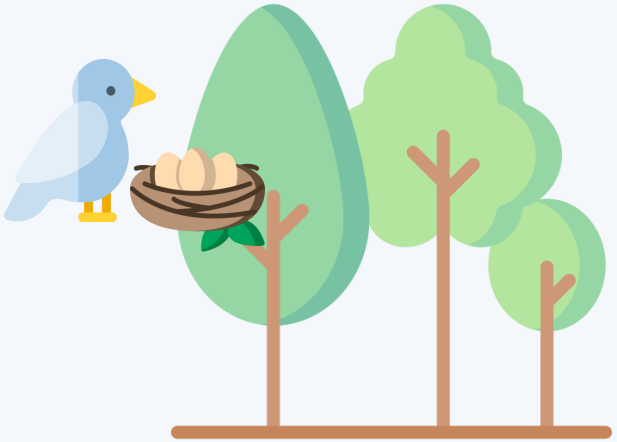
적은 Data 개수

3000개의 Data
미세한 파라미터 조정이 필요
→ Bayesian Optimization, Cuckoo Search

많은 Categorical 변수

일반 선형 회귀모델을 사용하는 것이 어려움
→ LightGBM, RandomForest

Modeling

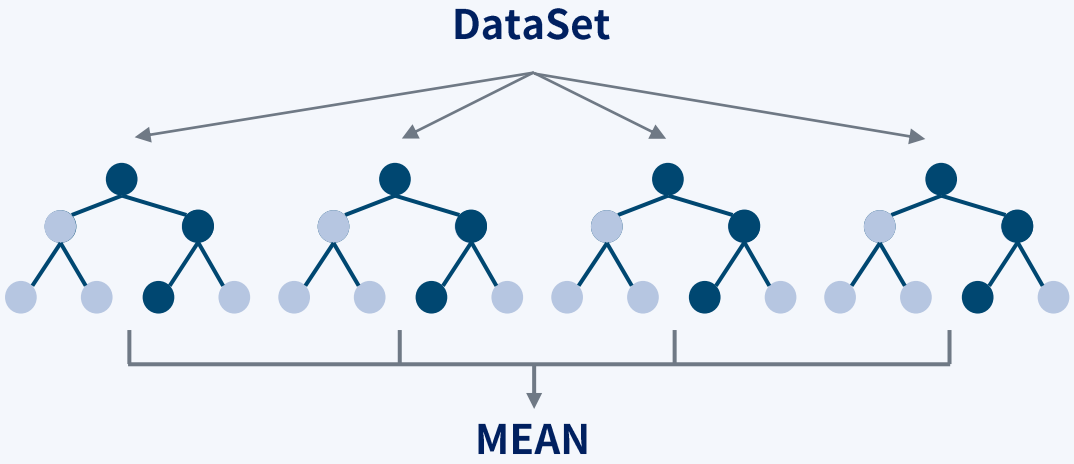


VS

Random Forest
+
Cuckoo Search Algorithm

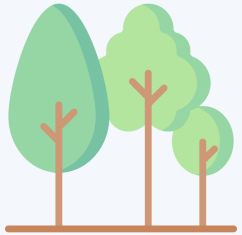
Light GBM
+
Bayesian Optimization

Modeling
Random Forest

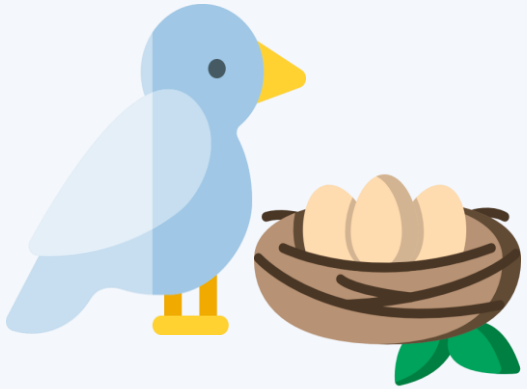


배깅 기반의 트리모델

데이터가 적을 때 개별 Tree의 depth가 깊어지면 Overfitting이 될 수 있음



Modeling
Cuckoo Search



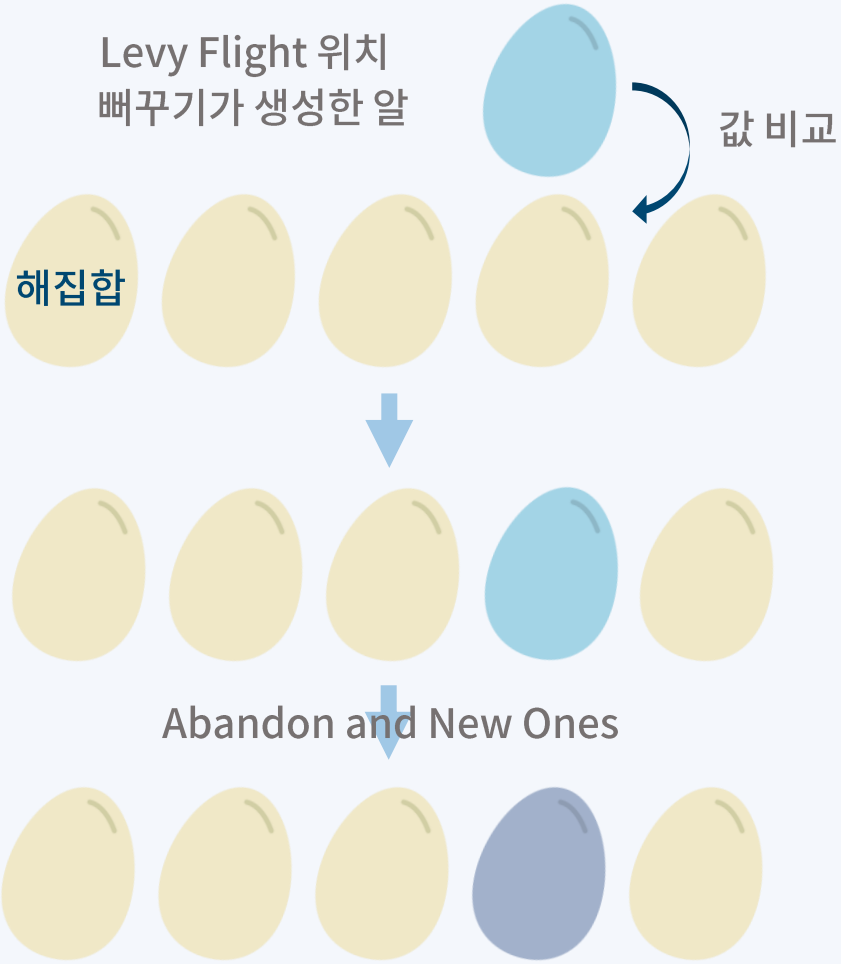
Cuckoo Search

Cuckoo Search 알고리즘은 뻐꾸기의
번식 전략에 기반한 자연에서 영감을 받은 알고리즘

$$x_i^{(t+1)} = x_i^{(t)} + \alpha \oplus Levy(\lambda)$$

새 , 곤충의 비행 패턴

Modeling
Cuckoo Search



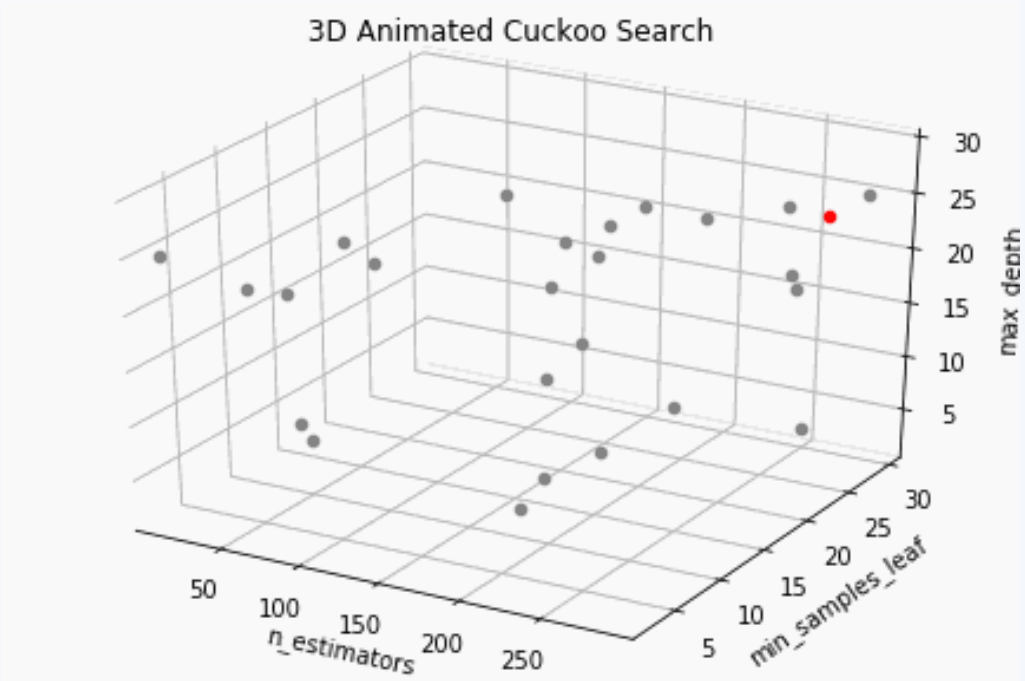
LevyFlight를 이용해 알을 놓을 위치를 정하고 알 생성

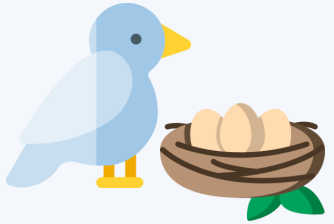
기존의 위치에 있는 값과 비교해 더 우수할 경우 뻐꾸기 알이 자리 차지

일정한 확률로 알 버리는 작업(스위치 파라미터)을 사용
로컬 및 글로벌 랜덤 워크 간의 균형을 유지→“**글로벌 최적화**”

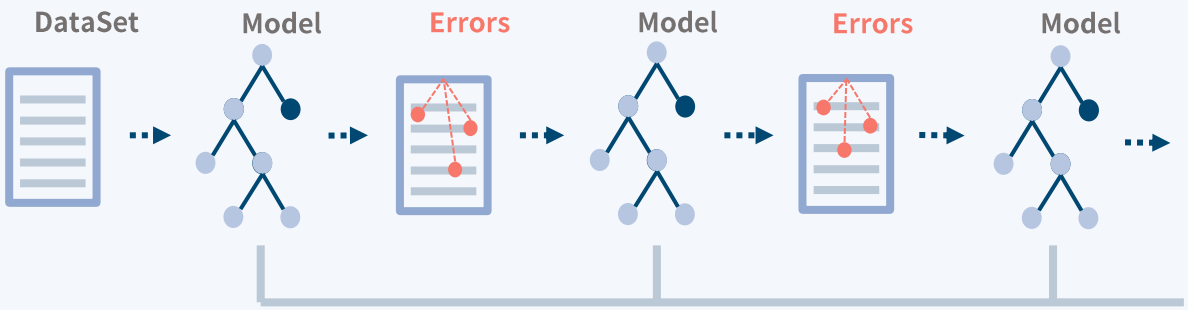
Modeling
Cuckoo Search

begin
Objective function $f(x)$ $x = (x_1, \ldots, x_d)^{\mathrm{T}}$
Generate initial population of
n host nests x_i $(i = 1, 2, \ldots, n)$
while (t < MaxGeneration) or (stop criterion)
Get a cuckoo randomly by Levy flights
evaluate its quality/fitness F_i
if $(F_i > F_j)$
replace j by the new solution;
end
A fraction (P_a) of worse nests
are abandoned and new ones are built;
Keep the best solutions
(or nests with quality solutions);
Rank the solutions and find the current best
end while
Postprocess results and visualization
end



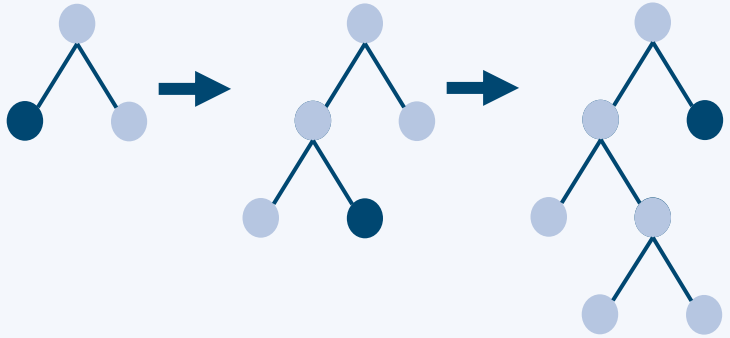


Modeling
Light GBM



Decision Tree를 순차적으로
학습-예측 하며 잔차에 가중치를 부여해 오류 개선

Leaf-wise tree growth



트리의 균형을 맞추지 않고 지속적 분할하며 진행

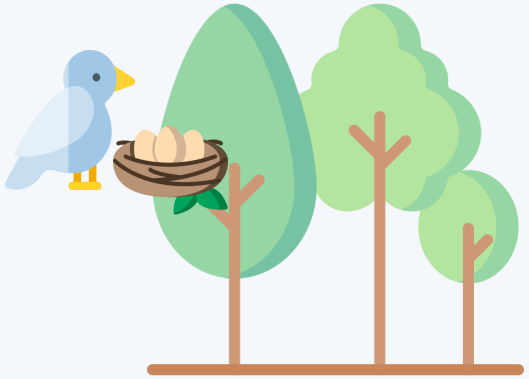
→ Xgboost의 느린 학습 시간 보안
→오버피팅 가능성

Modeling
Bayesian Optimization

입력값의 $f(x)$를 최대로 만드는 최적해를 빠르고 효과적으로 찾는 것이 목적
→ LightGBM의 오버피팅 방지

iter	target	colsam...	learni...	max_depth	min_da...	n_esti...	num_it...	reg_alpha	reg_la...	subsample
1	-1.544	0.08009	0.008336	31.07	2.573	28.77	2.185e+0	40.63	75.78	0.09826
2	-1.449	0.3168	0.00657	29.84	2.437	30.65	1.975e+0	95.03	28.13	0.6237
3	-1.227	0.3895	0.004603	31.77	3.86	31.79	1.763e+0	34.24	66.48	0.0519
4	-1.119	0.2399	0.004871	28.31	3.533	31.42	1.605e+0	10.12	27.1	0.03986
5	-1.221	0.8394	0.006378	31.72	2.991	29.51	1.671e+0	60.18	32.98	0.4671
6	-1.368	0.7277	0.008306	31.94	2.454	30.27	1.528e+0	93.7	27.13	0.5793
7	-1.055	0.3724	0.003367	28.78	3.129	31.55	1.533e+0	8.562	8.307	0.2905
8	-1.781	0.3037	0.00272	29.66	3.031	30.75	1.516e+0	96.79	67.79	0.9401
9	-1.461	0.6951	0.002563	28.91	3.914	30.11	1.889e+0	93.4	23.68	0.2936
10	-1.544	0.7763	0.001898	28.7	2.979	28.63	1.724e+0	66.13	34.8	0.4406
11	-0.9522	0.57	0.007975	28.27	2.324	30.96	1.718e+0	5.026	69.02	0.7911
12	-1.435	0.5844	0.003319	29.23	2.877	30.08	1.977e+0	78.29	89.56	0.7309

Result
R2 score



VS

	RF + CS	LightGBM + BO
Train	0.57	0.79
Test	0.52	0.77

Result
Simultaion

웹 서버 구현 및 모델 예측

원가 예측값

지역, 세대수, 세대면적, 공사기간

지도 및 행정구역 검색 서비스 구현

Reference & Tool

Reference

· KIM, Gwang-Hee; AN, Sung-Hoon; KANG, Kyung-In. Comparison of construction cost estimating models based on regression analysis, neural networks, and case-based reasoning. Building and environment, 2004, 39.10: 1235-1242.

· 남군, et al. 유전 알고리즘–서포트 벡터 회귀를 활용한 공동주택 공사비 예측에 관한 연구. 한국건설관리학회 논문집, 2014, 15.4: 68-76.

· YANG, Xin-She; DEB, Suash. Cuckoo search via Lévy flights. In: 2009 World congress on nature & biologically inspired computing (NaBIC). IEEE, 2009. p. 210-214.

· DEVI, K. Nirmala; BHASKARAN, V. Murali; KUMAR, G. Prem. Cuckoo optimized SVM for stock market prediction. In: 2015 International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS). IEEE, 2015. p. 1-5.

Tool

Q & A