Final Presentation

건설 기계 오일 상태 분류 AI 경진대회
Oil Status Classification of Construction Machinery

T1 나해란 안형주 장영진 장효정

CONTENTS

01. Introduction

Competition

Topic : Oil Status Classification of Construction Machinery

Main task : Binary Classification (Normal, Abnormal)

Host

현대제뉴인

Supervision

DACON

Data

Problem 1 : Limited Features in Test data

Train data

(14095, 54)

Feature : 52

ID	COMPONEN...	AN...	YEAR	SAMP...	ANON...	AG	AL	B	BA	BE	CA	CD	CO	CR	CU
TRAIN_00000	COMPONENT3	1486	2011	7	200	0	3	93	0	0	3059	0.0	0	13	78
TRAIN_00001	COMPONENT2	1350	2021	51	375	0	2	19	0	0	2978	0.0	0	0	31
TRAIN_00002	COMPONENT2	2415	2015	2	200	0	110	1	1	0	17	0.0	0	1	2
TRAIN_00003	COMPONENT3	7389	2010	2	200	0	8	3	0	0	1960	0.0	0	0	1
TRAIN_00004	COMPONENT3	3954	2015	4	200	0	1	157	0	0	71	0.0	0	0	0
TRAIN_00005	COMPONENT3	2061	2008	4	550	0	3	8	0	0	2770	0.0	0	3	179
TRAIN_00006	COMPONENT3	1416	2015	7	616	0	0	21	0	0	130	0.0	0	0	3
TRAIN_00007	COMPONENT3	1170	2009	4	370	0	5	3	3	0	2589	0.0	0	3	6
TRAIN_00008	COMPONENT3	4880	2014	7	200	0	0	1	0	0	11	0.0	0	4	125
TRAIN_00009	COMPONENT1	6748	2015	6	200	0	1	1	0	0	62	0.0	0	0	2

Test data

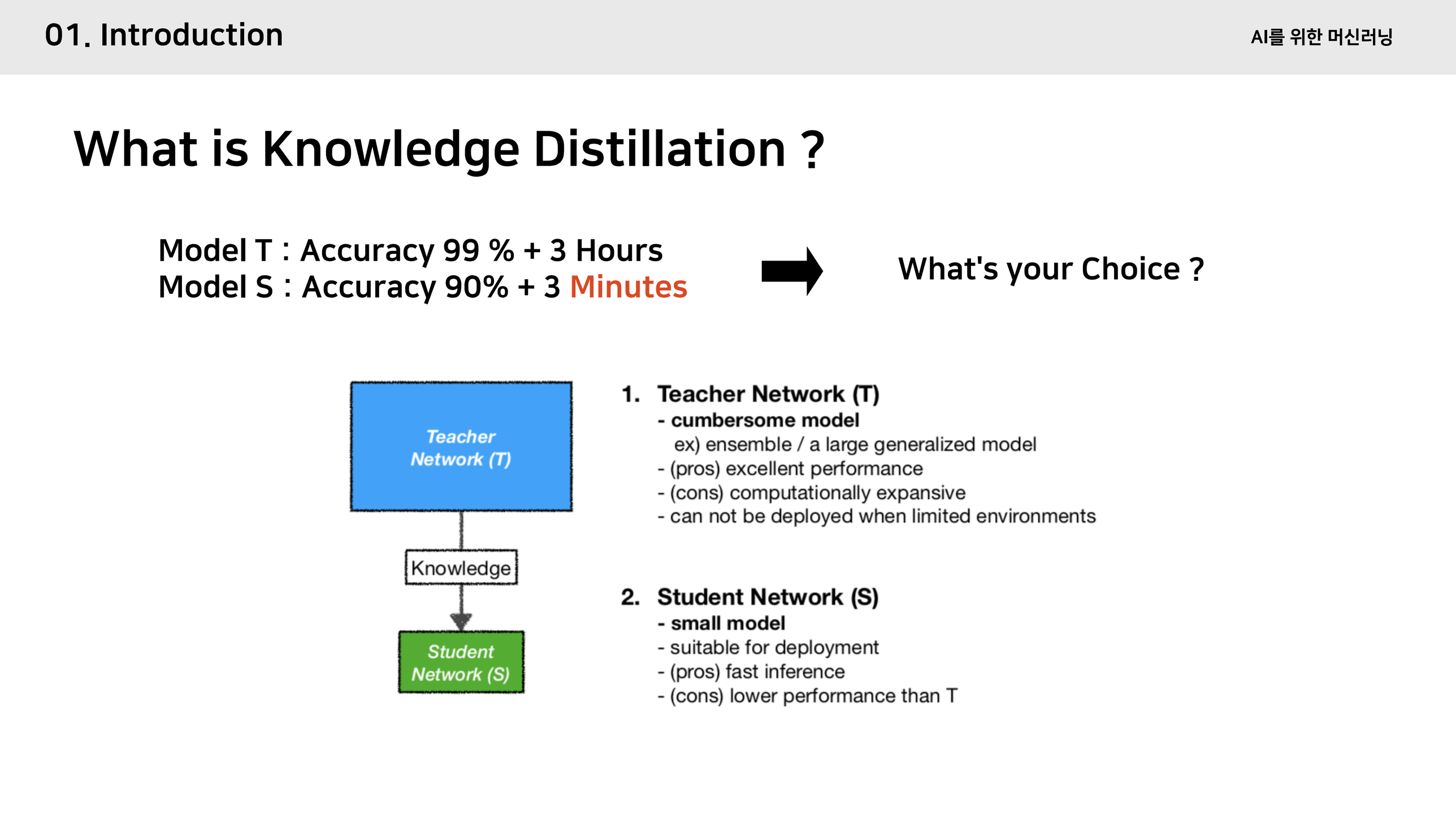
(6041, 20)

Feature : 18

ID	COMPONENT_...	AN...	YEAR	AN...	AG	CO	CR	CU	FE	H2O	MN	MO
TEST_0000	COMPONENT1	2192	2016	200	0	0	0	1	12	0.0	0	0
TEST_0001	COMPONENT3	2794	2011	200	0	0	2	1	278	0.0	3	0
TEST_0002	COMPONENT2	1982	2010	200	0	0	0	16	5	0.0	0	0
TEST_0003	COMPONENT3	1404	2009	200	0	0	3	4	163	0.0	4	3
TEST_0004	COMPONENT2	8225	2013	200	0	0	0	6	13	0.0	0	0
TEST_0005	COMPONENT3	4729	2010	200	0	0	1	0	62	0.0	0	1
TEST_0006	COMPONENT3	1444	2014	200	0	0	0	1	73	0.0	0	3
TEST_0007	COMPONENT4	4912	2021	473	0	0	0	2	8	0.0	0	5
TEST_0008	COMPONENT3	5565	2013	200	0	0	5	205	287	0.0	6	0
TEST_0009	COMPONENT3	1238	2016	200	0	0	1	0	108	0.0	1	0

What is Knowledge Distillation ?

Model T : Accuracy 99 % + 3 Hours
Model S : Accuracy 90% + 3 Minutes

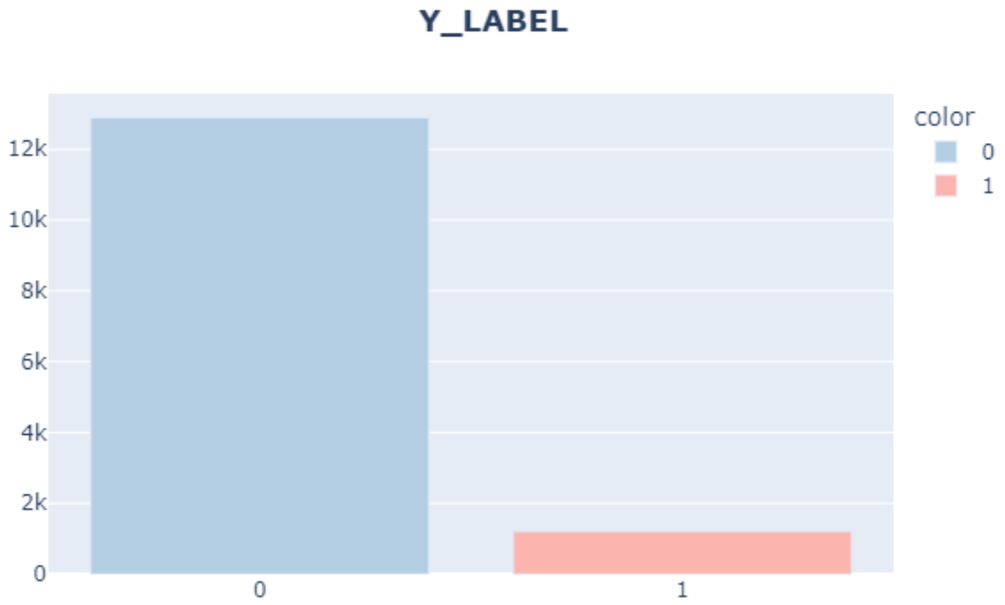


What's your Choice ?

1. **Teacher Network (T)**
   - **cumbersome model**
     ex) ensemble / a large generalized model
   - (pros) excellent performance
   - (cons) computationally expansive
   - can not be deployed when limited environments

2. **Student Network (S)**
   - **small model**
   - suitable for deployment
   - (pros) fast inference
   - (cons) lower performance than T

Problem 2 : Imbalanced Data



Status	Y_LABEL
0 (Normal)	12892
1 (Abnormal)	1203

Only 8 % of Abnormal data

Our Strategy

Problem 1

1. Make a Classification Model
2. Extract Possibilities
3. Set Possibilities as the target data
4. Make a Regression Model using Test data columns

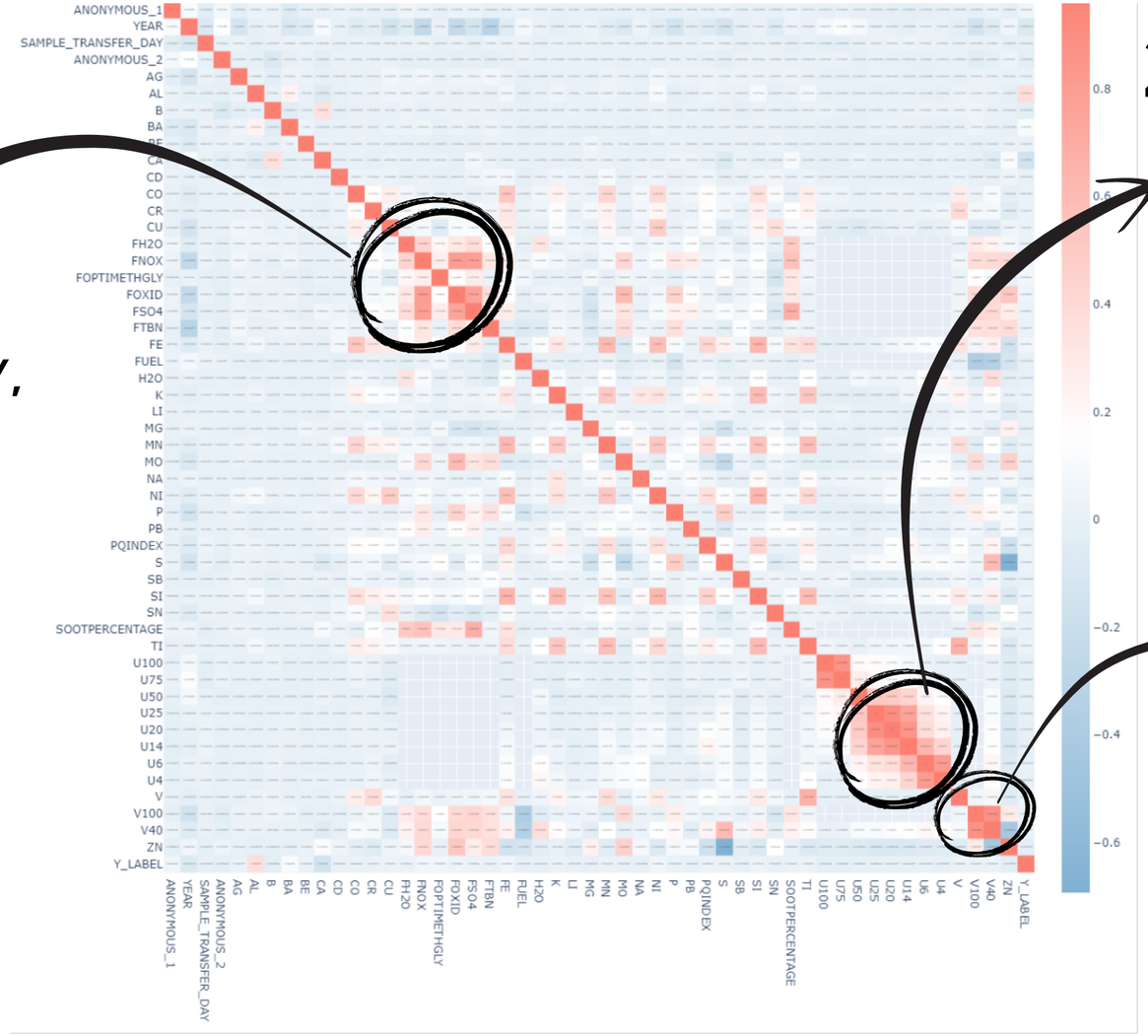
Problem 2

1. Original
2. SMOTE
3. Downsampling

02. Method

EDA

Correlation for all features



1

FH2O, FNOX, FOPTIMETHGLY, FOXID, FSO4, FTBN, FE

2

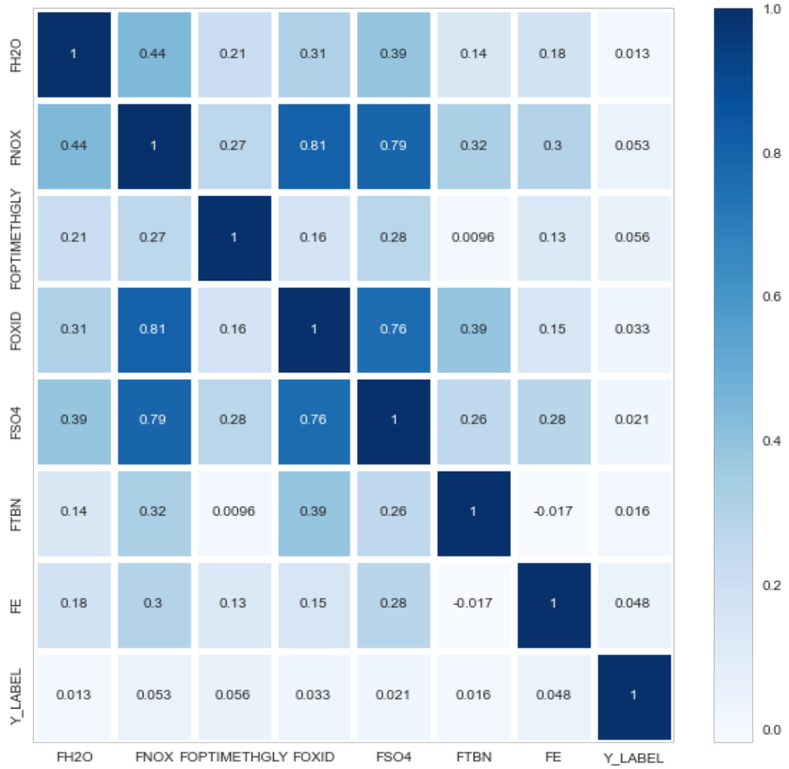
Particle Counts : U100, U75, U50, U25, U20, U14, U6, U4, V

3

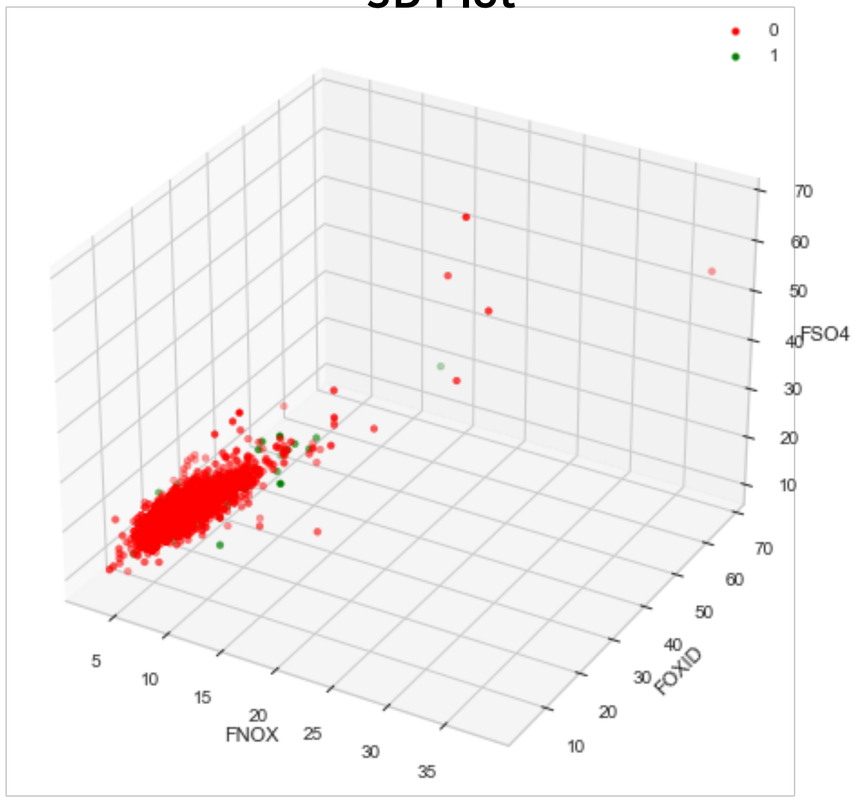
V, V100, V40, ZN

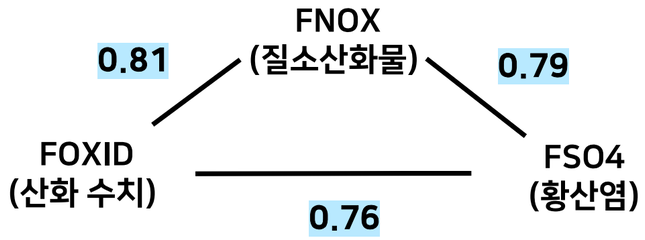
FH2O, FNOX, FOPTIMETHGLY, FOXID, FSO4, FTBN, FE

Correlation Plot



3D Plot



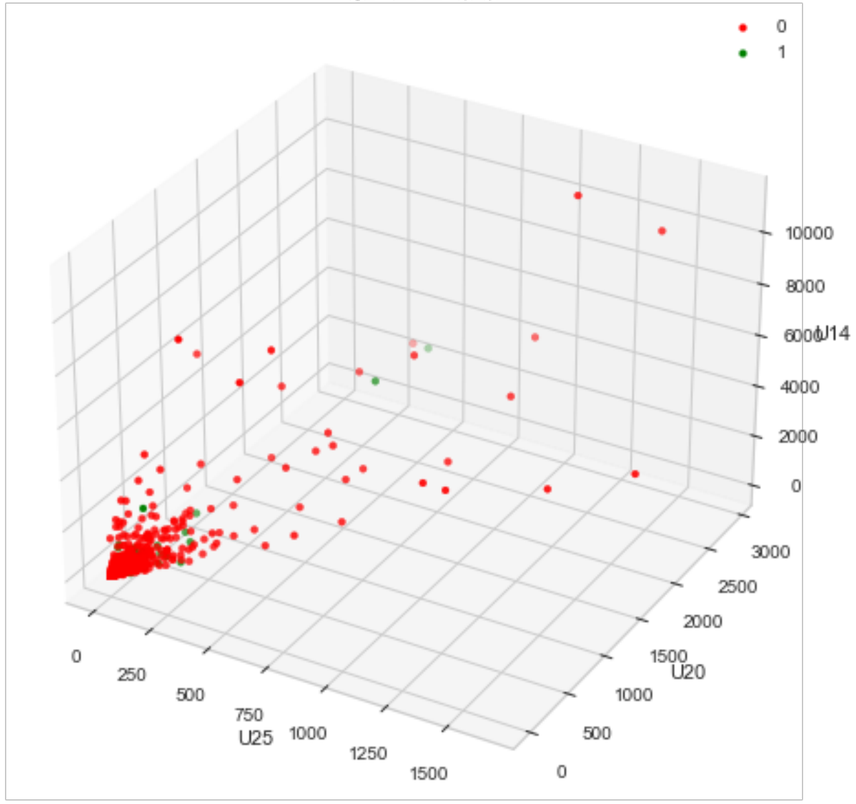


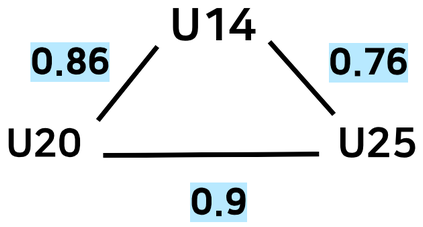
-> FNOX(질소산화물), FOXID(산화 수치), FSO4(황산염) are correlated

Particle Counts : U100, U75, U50, U25, U20, U14, U6, U4, V

Correlation Plot

3D Plot





U75 & U100 : 0.87

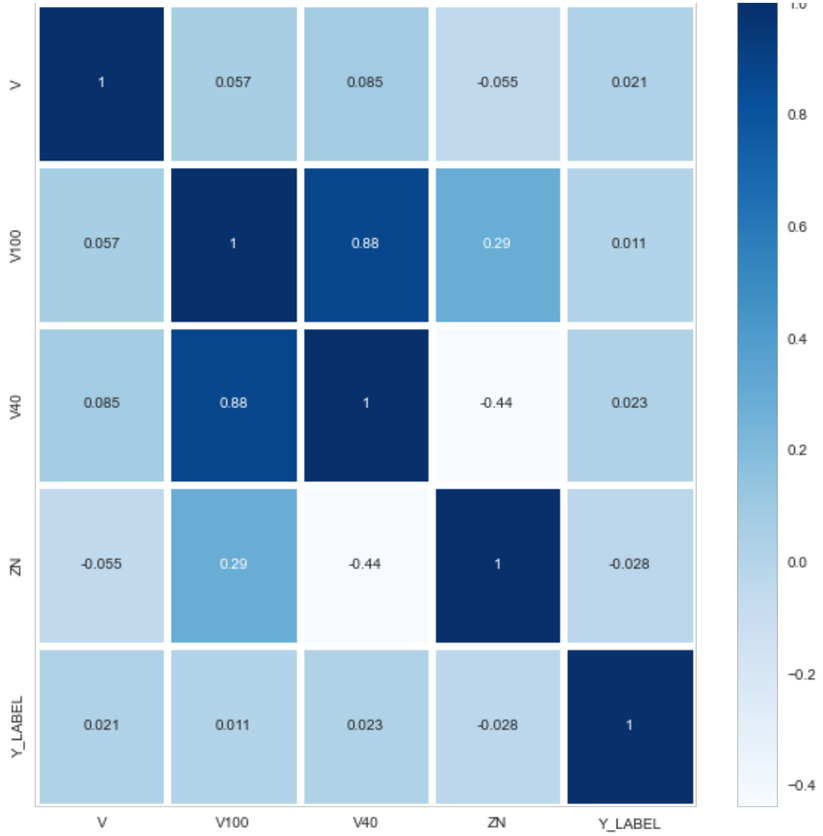
U6 & U4 : 0.82

-> Group as (U14, U20, U25), (U75, U100) (U4, U6)

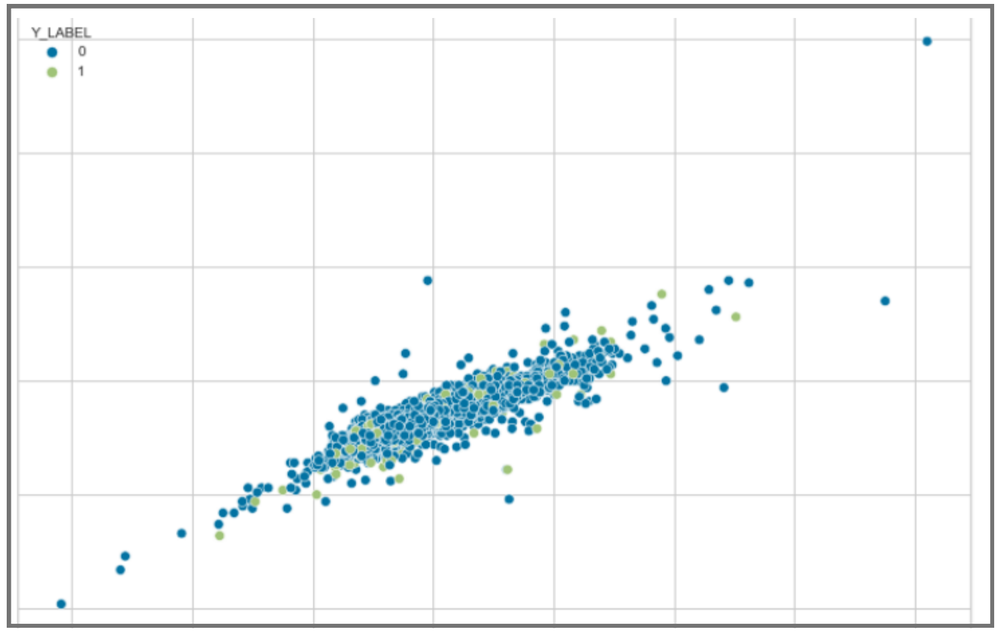
EDA

V, V100, V40, ZN

Correlation Plot



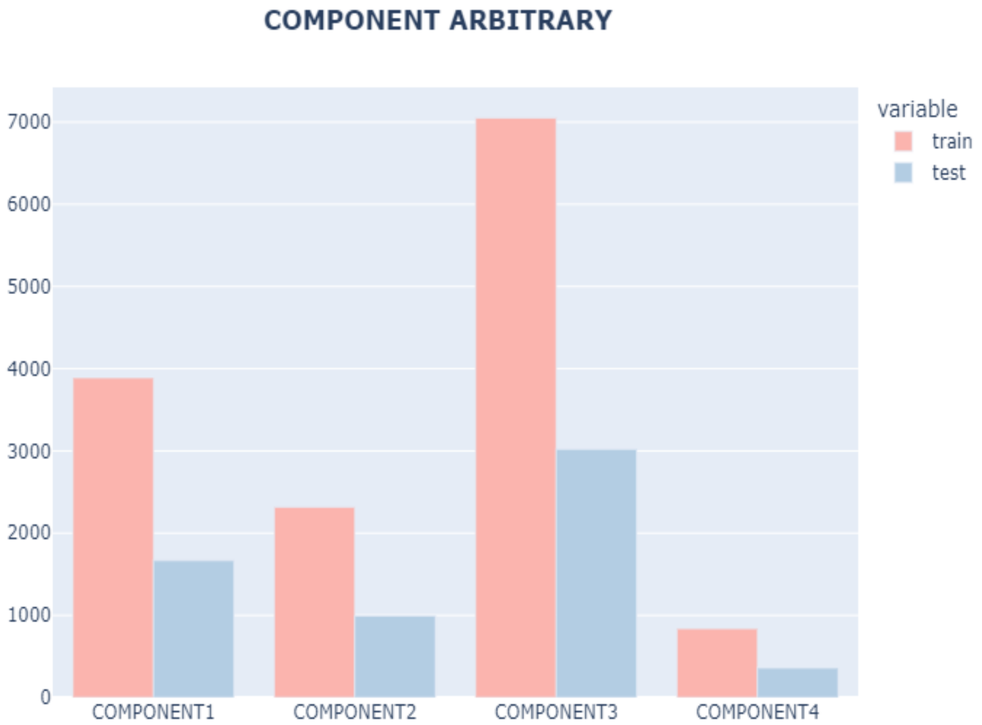
Scatter Plot



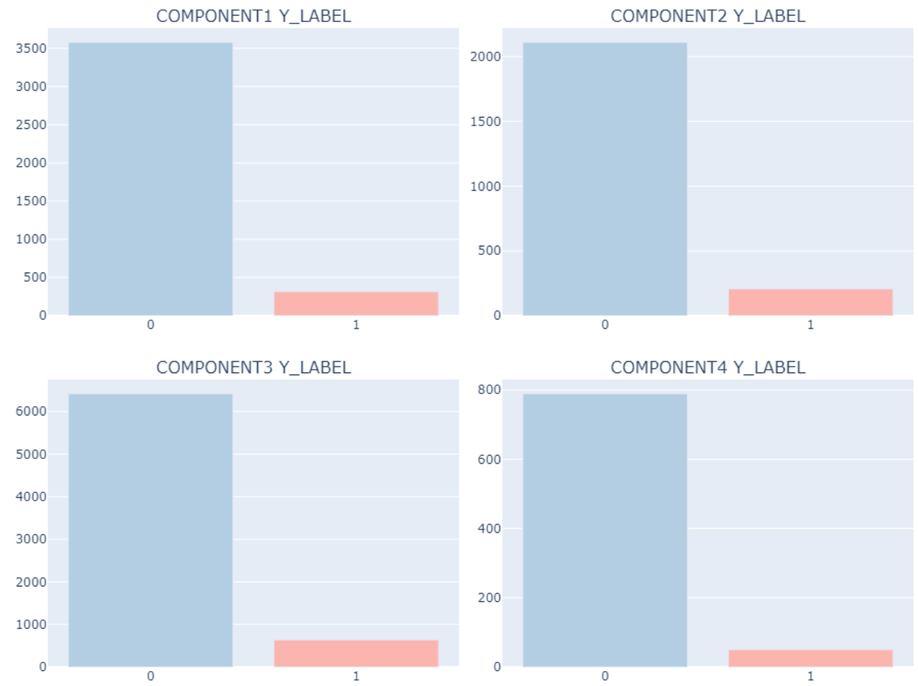
V40 & V100 : 0.88

-> V40 and V100 are correlated

Component Arbitrary

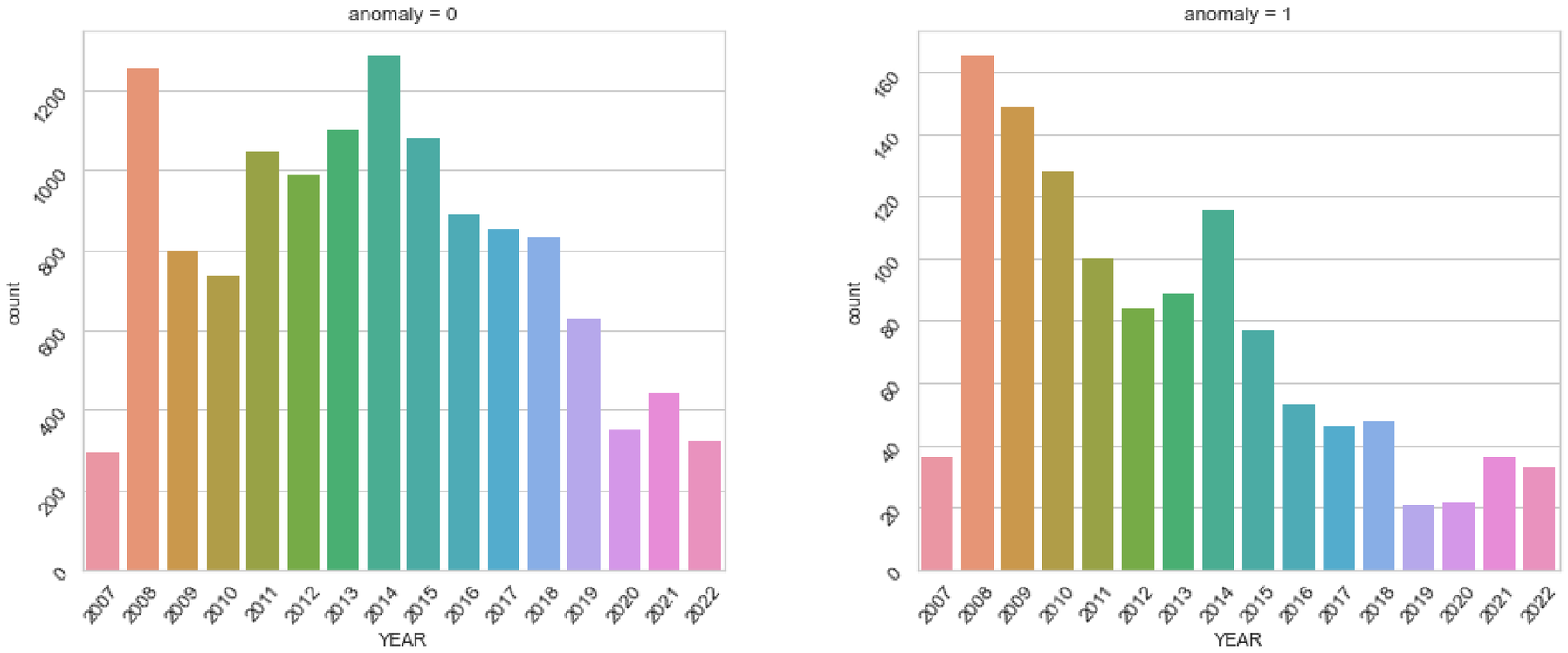


Similar rates in train and test data



Similar rates of abnormal in each Component Arbitrary

YEAR



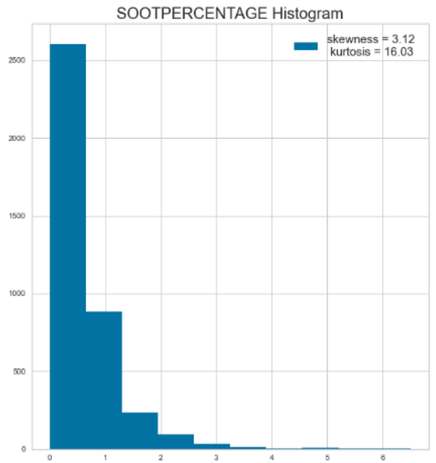
Abnormal is keep decreasing after 2008

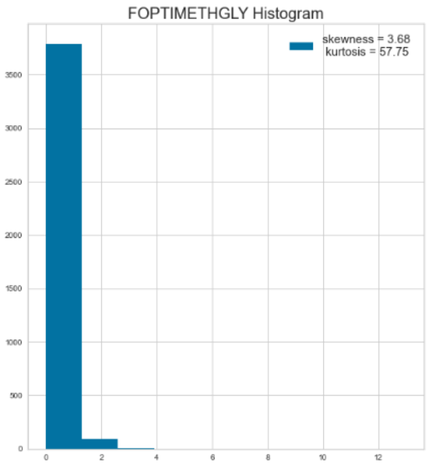
EDA

NaN Columns

	count	rate
U4	11977	0.849734
U6	11977	0.849734
U14	11977	0.849734
U20	11779	0.835686
U50	11779	0.835686
U100	11779	0.835686
U75	11779	0.835686
U25	11779	0.835686
V100	10371	0.735793
FH2O	10205	0.724016
FOXID	10205	0.724016
FUEL	10205	0.724016
FOPTIMETHGLY	10205	0.724016
FSO4	10205	0.724016
FTBN	10205	0.724016
SOOTPERCENTAGE	10205	0.724016
FNOX	10205	0.724016
K	2299	0.163107
CD	1394	0.098900

Example Columns : SOOTPERCENTAGE, FOPTIMETHGLY





Most of the columns with missing values were skewed

Preprocessing

Drop V100 feature

Make a derived variable,
Mean of FNOX, FOXID, FSO4

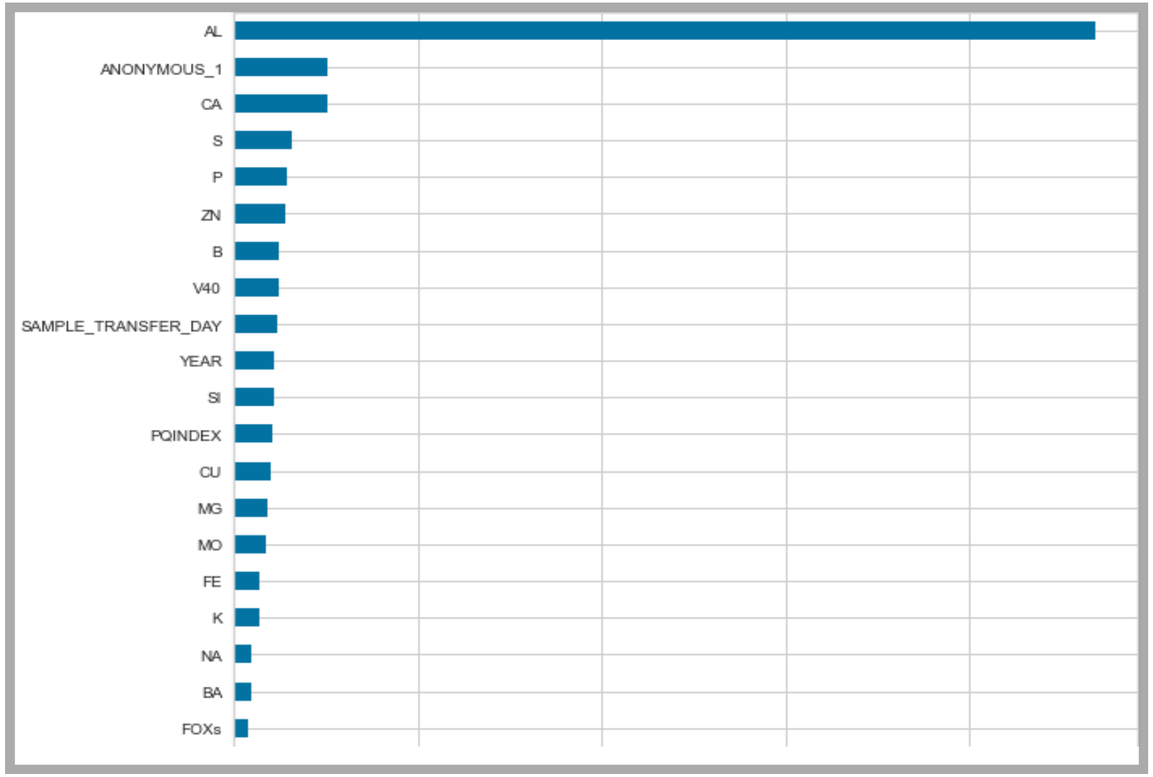
Make 4 groups of Particle Counts
(U4 ~ U14, U14 ~ U50, U50 ~ U75, U75 ~)

Use median for missing imputation

Feature Selection

Select Top 20 Important Features from Tree based models

Ex. Decision Tree



Top 20 columns

AL, ANONYMOUS_1, CA, S, P, ZN, B, V40, SAMPLE_TRANSFER_DAY, YEAR, SI, PQINDEX, CU, MG, MO, FE, K, NA, BA, FOXs

+ Random Forest, XGBoost, LGBM, CatBoost

Completed Final Dataset with 25 features

Feature Selection

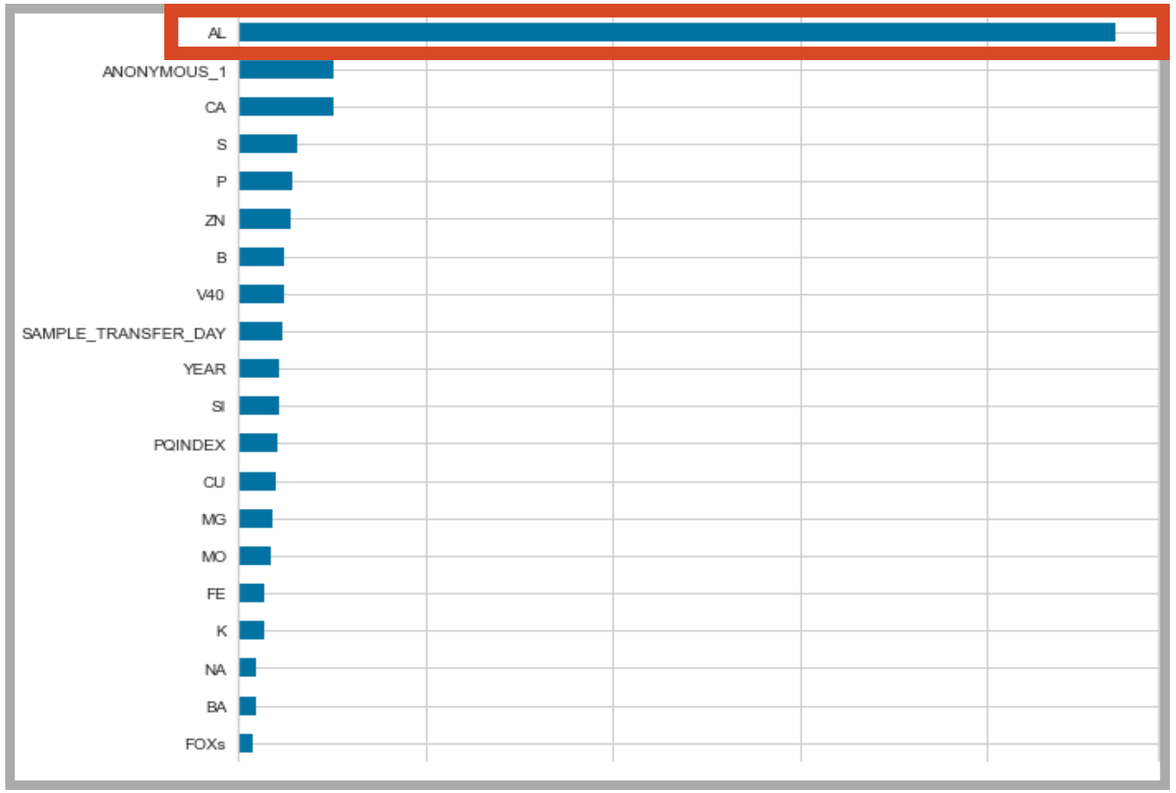
[Final Dataset]

	AL	CA	BA	B	SI	P	S	FH2O	NA	PB	...	FE	ANONYMOUS_2	ZN	V40	PQINDEX	NI	CU	MO	CR	Y_LABEL
0	3	3059	0	93	427	1951	21370	13.0	16	0	...	888	200	75	154.0	8504	6	78	1	13	0
1	2	2978	0	19	0	572	1117	13.0	1	2	...	2	375	652	44.0	19	0	31	0	0	0
2	110	17	1	1	0	328	1334	13.0	2	0	...	4	200	412	72.6	17	0	2	0	1	1
3	8	1960	0	3	1	906	21774	13.0	0	1	...	37	200	7	133.3	44	0	1	0	0	0
4	1	71	0	157	2	309	18470	13.0	2	0	...	71	200	128	133.1	217	0	0	0	0	0
5	3	2770	0	8	32	1129	8685	13.0	4	2	...	550	550	1015	69.7	329	4	179	11	3	0
6	0	130	0	21	1	330	16280	13.0	2	0	...	35	616	24	148.5	26	0	3	0	0	0
7	5	2589	3	3	6	866	17090	13.0	5	2	...	81	370	16	142.9	7735	0	6	1	3	0
8	0	11	0	1	2	451	15960	13.0	2	0	...	166	200	89	140.7	83	4	125	0	4	0
9	1	62	0	1	3	1047	4191	9.0	6	5	...	12	200	1020	98.9	5	0	2	3	0	0

14095 X 26

Feature Engineering

the most important variable in this problem



There is no 'AL' in test data

Top 5 regression models in pycaret

AL
8.113569
10.788605
20.700041
8.416278
15.490443
...
9.730528
10.172429
11.898597
9.938041
5.107710

Make 'AL' variable in the test dataset

03. Experiments

Classification

1. Original

[AutoML]

	Model	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC	TT (Sec)
lightgbm	Light Gradient Boosting Machine	0.9546	0.8918	0.5067	0.9309	0.6557	0.6336	0.6676	0.7400
catboost	CatBoost Classifier	0.9548	0.8926	0.5023	0.9386	0.6543	0.6325	0.6681	4.0540
xgboost	Extreme Gradient Boosting	0.9526	0.8788	0.5201	0.8745	0.6516	0.6279	0.6528	0.3680
rf	Random Forest Classifier	0.9543	0.8778	0.4874	0.9533	0.6450	0.6233	0.6636	0.6300
gbc	Gradient Boosting Classifier	0.9522	0.8896	0.4993	0.8944	0.6407	0.6173	0.6476	1.6980
lr	Logistic Regression	0.9511	0.8640	0.4726	0.9120	0.6223	0.5990	0.6363	0.1400
ada	Ada Boost Classifier	0.9497	0.8570	0.4785	0.8753	0.6181	0.5937	0.6251	0.4360
et	Extra Trees Classifier	0.9512	0.8877	0.4339	0.9872	0.6025	0.5807	0.6370	0.5780
knn	K Neighbors Classifier	0.9475	0.7808	0.4191	0.9264	0.5759	0.5522	0.6025	1.4720
nb	Naive Bayes	0.9239	0.8096	0.5245	0.5588	0.5407	0.4992	0.4998	0.0180
qda	Quadratic Discriminant Analysis	0.9220	0.8140	0.5260	0.5478	0.5360	0.4934	0.4939	0.0220
svm	SVM - Linear Kernel	0.9087	0.0000	0.5408	0.5620	0.5302	0.4839	0.4945	0.0280
dt	Decision Tree Classifier	0.9113	0.7427	0.5395	0.4853	0.5097	0.4613	0.4627	0.1000
lda	Linear Discriminant Analysis	0.9297	0.8334	0.1931	0.9213	0.3175	0.2969	0.4017	0.0440
ridge	Ridge Classifier	0.9237	0.0000	0.1129	0.9545	0.2011	0.1863	0.3123	0.0160
dummy	Dummy Classifier	0.9147	0.5000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0120

One Ordinal Feature (YEAR)

Label Encoding

Numerical Features

Robust Scaling

Top 5 Models

LGBM, CatBoost, XGBoost,
Random Forest, GBC

Classification

2. SMOTE

[AutoML]

	Model	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC	TT (Sec)
lightgbm	Light Gradient Boosting Machine	0.9570	0.8841	0.5206	0.9376	0.6683	0.6474	0.6801	1.1660
xgboost	Extreme Gradient Boosting	0.9562	0.8801	0.5267	0.9096	0.6664	0.6448	0.6728	0.3840
catboost	CatBoost Classifier	0.9563	0.8950	0.5083	0.9400	0.6592	0.6380	0.6729	4.5800
gbc	Gradient Boosting Classifier	0.9548	0.8901	0.5190	0.8952	0.6559	0.6336	0.6612	2.4280
rf	Random Forest Classifier	0.9553	0.8647	0.4855	0.9582	0.6431	0.6220	0.6638	0.7600
ada	Ada Boost Classifier	0.9527	0.8672	0.4946	0.8900	0.6349	0.6119	0.6427	0.6400
lr	Logistic Regression	0.9538	0.8602	0.4793	0.9334	0.6324	0.6104	0.6498	0.2250
svm	SVM - Linear Kernel	0.9488	0.0000	0.4915	0.8465	0.6171	0.5917	0.6195	0.0650
et	Extra Trees Classifier	0.9474	0.8620	0.3743	0.9845	0.5416	0.5197	0.5893	0.7250
dt	Decision Tree Classifier	0.9185	0.7591	0.5678	0.5105	0.5369	0.4925	0.4937	0.1540
nb	Naive Bayes	0.8493	0.7944	0.6132	0.3301	0.4225	0.3510	0.3740	0.0200
lda	Linear Discriminant Analysis	0.9310	0.8333	0.1948	0.8913	0.3184	0.2976	0.3967	0.0730
knn	K Neighbors Classifier	0.9280	0.6793	0.1659	0.8530	0.2754	0.2555	0.3545	1.0940
qda	Quadratic Discriminant Analysis	0.5337	0.7724	0.7686	0.1312	0.2223	0.0931	0.1601	0.0390
ridge	Ridge Classifier	0.9250	0.0000	0.1051	0.9453	0.1875	0.1741	0.2978	0.0220
dummy	Dummy Classifier	0.9168	0.5000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0080

One Ordinal Feature (YEAR)

Label Encoding

Numerical Features

Robust Scaling

Top 5 Models

LGBM, XGBoost, CatBoost,
GBC, Random Forest

Classification

Result Table of Classification

Model	F1 Score (Original)	F1 Score (SMOTE)
LGBM	0.6854	0.6851
XGBoost	0.699	0.6985
CatBoost	0.6889	0.6965
Random Forest	0.6842	0.6843
Gradient Boosting Classifier	0.7008	0.6651
LGBM + XGBoost + CatBoost + Random Forest + GBC	0.6959	0.6935
XGBoost + CatBoost + GBC	0.7026	0.6902
...		
LGBM + CatBoost + GBC	0.6923	0.6849
XGBoost + GBC + Random Forest	0.6992	0.6967
LGBM + XGBoost + GBC	**0.7056**	0.6919

Soft Voting Classifier of LGBM, XGBoost, GBC metric (F1 Score : 0.7056) was the best

Classification

[Dataset for Regression]

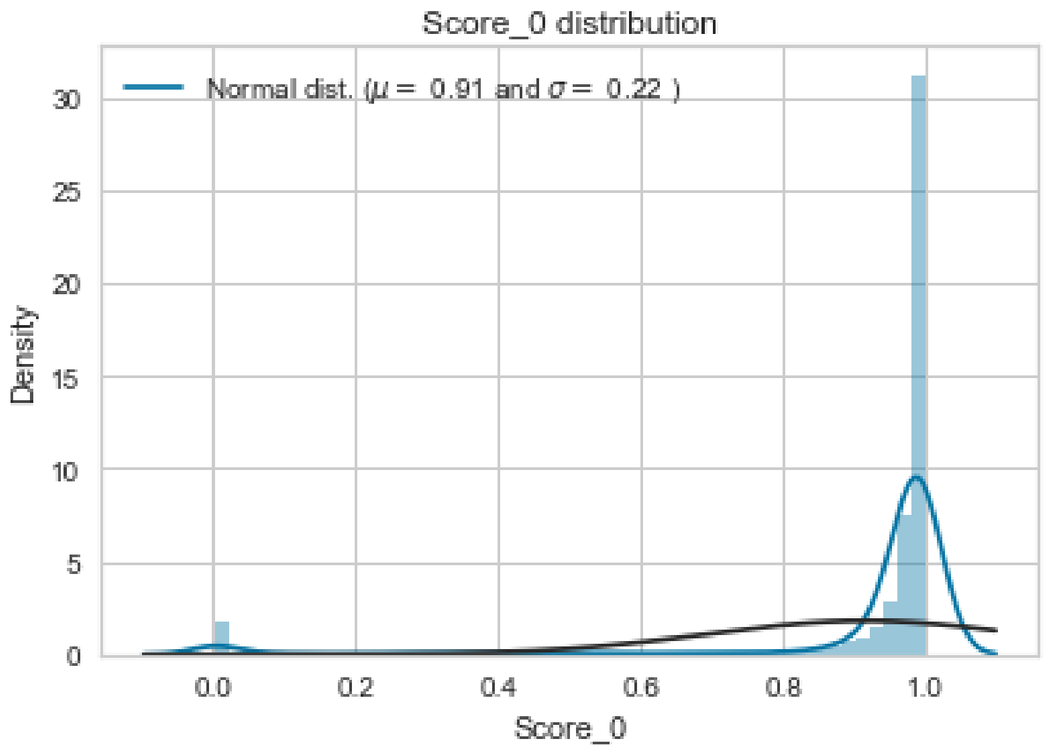
Explanatory Data | Target Data

	ANONYMOUS_1	YEAR	FE	ANONYMOUS_2	ZN	V40	PQINDEX	NI	CU	MO	CR	Y_LABEL	Label	Score_0
0	7.822044	7	4.248495	5.303305	18	4.894101	5.817111	0.000000	0.000000	0.00000	0.693147	0	0	0.9901
1	7.784473	1	5.703782	6.311735	239	4.767289	4.317488	0.693147	2.564949	1.94591	1.098612	0	0	0.9253
2	7.650645	6	1.945910	5.303305	10	4.913390	2.772589	0.000000	1.098612	0.00000	0.000000	0	0	0.9823
3	8.860925	10	3.044522	5.303305	1113	4.085976	2.708050	0.000000	2.079442	0.00000	0.000000	0	0	0.9906
4	8.248267	3	4.836282	5.303305	19	5.182907	6.721426	0.000000	0.693147	0.00000	1.098612	0	0	0.9751
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
10521	11.269822	10	2.890372	5.303305	35	4.905275	2.833213	0.000000	0.693147	0.00000	0.000000	0	0	0.9662
10522	7.625107	6	3.891820	5.303305	51	5.025195	5.241747	0.000000	0.000000	0.00000	0.000000	0	0	0.9984
10523	8.979920	10	4.499810	7.378384	1170	4.177459	4.343805	0.000000	2.564949	0.00000	0.000000	0	0	0.9885
10524	7.640123	3	6.775366	5.303305	20	5.200705	7.449498	1.609438	1.791759	0.00000	2.772589	0	0	0.9794
10525	8.851234	11	5.176150	5.303305	14	4.930148	6.594413	0.000000	0.000000	0.00000	1.098612	0	0	0.9893

14095 X 14

Regression

Problem before Modeling (1)



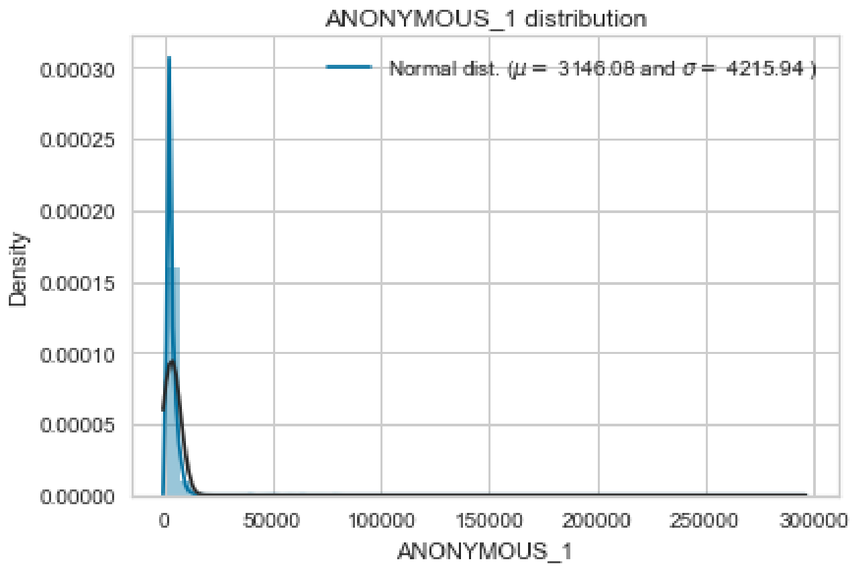
	Score 0
count	14095
mean	0.914513
std	0.220244
min	0.0005
25%	0.9643
50%	0.9868
75%	0.9941
max	0.9996

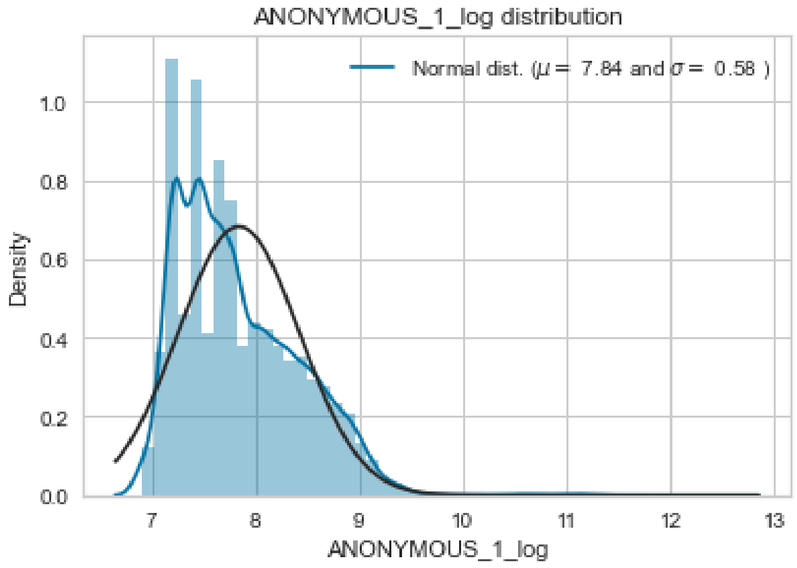
-> There are not many widely known solutions handling Imbalanced Data in Regression

Regression

Problem before Modeling (2)

Log transformation





Skewness	CR	ANONYMOUS 1	FE	CU	NI	V40	PQINDEX	ANONYMOUS 2	MO
Original	54.25	13.59	21.7	21.41	19.9	1.13	7.28	8.64	2.91
Log Transform	1.32	0.92	0.28	0.77	2.91	-0.55	0.8	0.92	1.2

Regression

Experiment 1

Original Data

Experiment 2

Random DownSampling

Experiment 3

Normal and Abnormal ratio 1 : 1

Experiment 4

Set 5 dataset with different ratios

Dataset 1	Dataset 2	Dataset 3	Dataset 4	Dataset 5
1:1	1:2	1:3	1:4	2:1

Experiment 5

Split Abnormal data to 5 sets equally

Abnormal 8.5 % -> 26.2 %

➡ **Final Model**

Regression

< Model Architecture >

Each Dataset have 750 Label 0, 2106 Label 1

Regression

Step 1. Modeling for Each Dataset

For each Dataset, we found the best model and tuned hyperparameter using Optuna

Model 1 (Extra Trees)

max depth : 12
max leaf nodes : 312
n estimators : 222

Model 2 (CatBoost)

l2 leaf reg : 0.0665
max bin : 382
learning rate : 0.0134
n estimators : 426
max depth : 8
min data in leaf : 79

Model 3 (Extra Trees)

max depth : 12
max leaf nodes : 330
n estimators : 492

Model 4 (CatBoost)

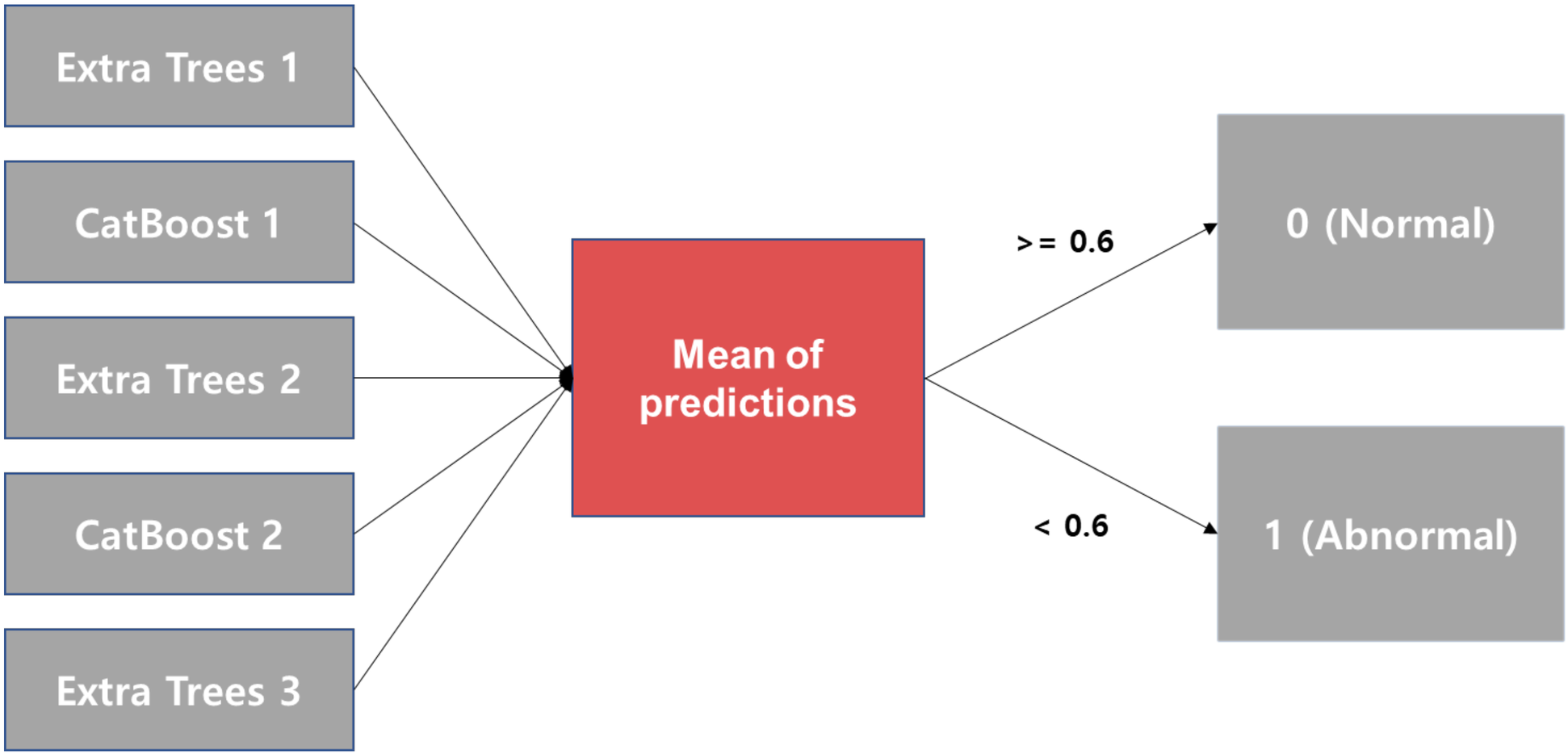
l2 leaf reg : 0.683
max bin : 352
learning rate : 0.011
n estimators : 367
max depth : 7
min data in leaf : 165

Model 5 (Extra Trees)

max depth : 12
max leaf nodes : 192
n estimators : 50

Regression

Step 2. Stacking

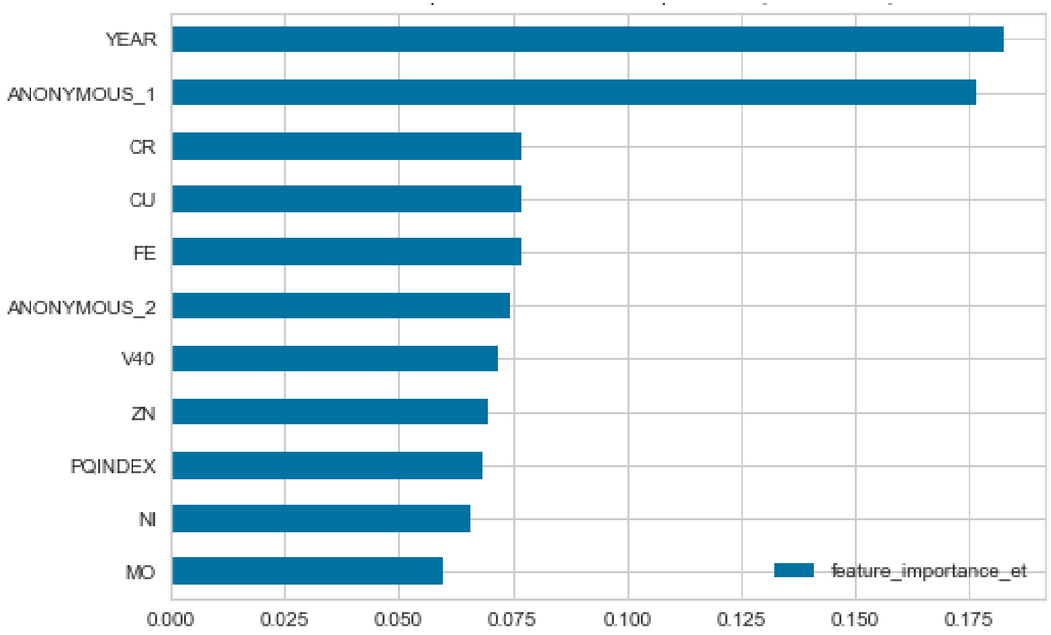


1. Since we used enough complicated models, we used mean for stacking
2. Threshold 0.6 was the best

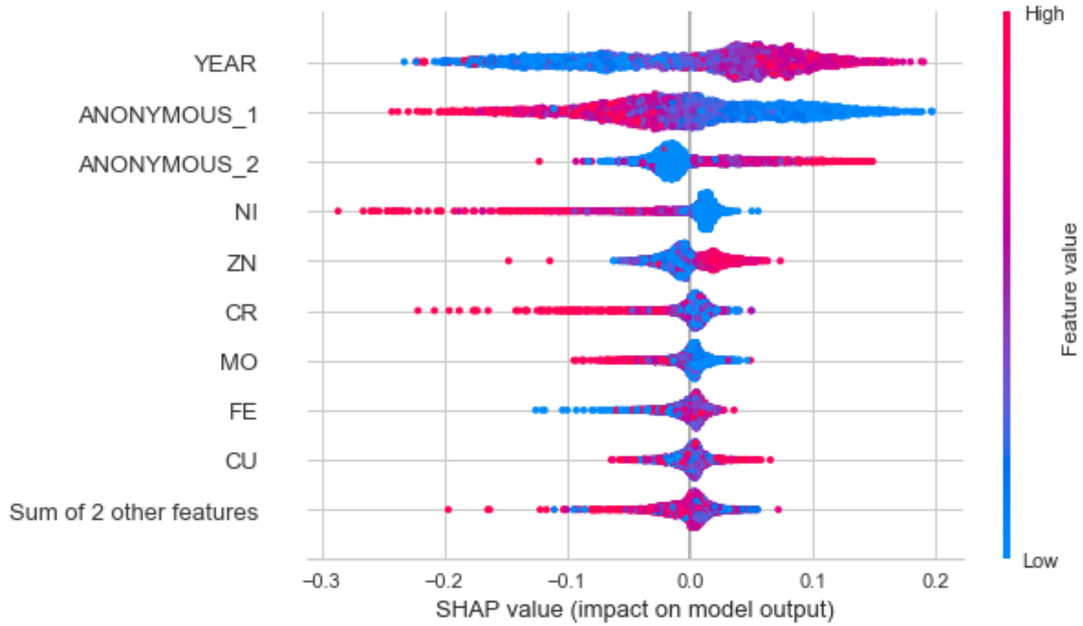
Results

Feature Importance

< Feature Importance plot >



< SHAP value >



1. YEAR & ANONYMOUS 1 were importance features

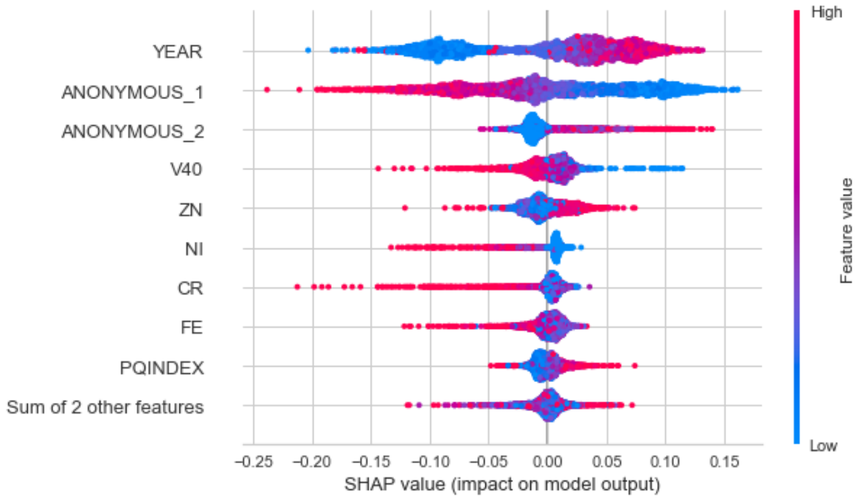
1. As YEAR increases, Score 0 increases -> Recently there are few abnormal

2. As ANONYMOUS 1 decreases, Score 0 increases
-> Less amount of ANONYMOUS 1 makes normal

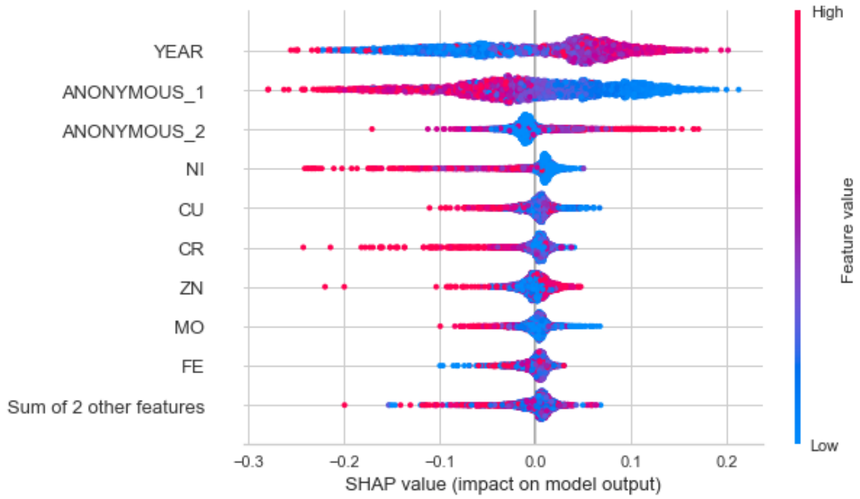
Results

Feature Importance

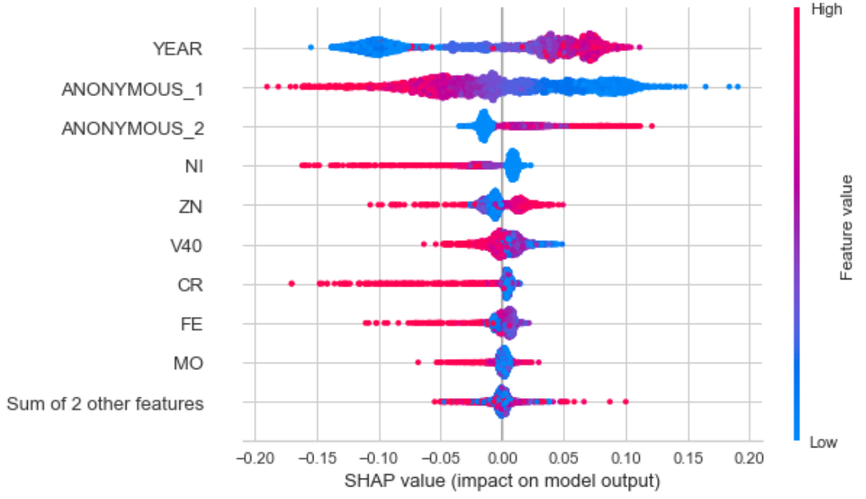
Dataset 2



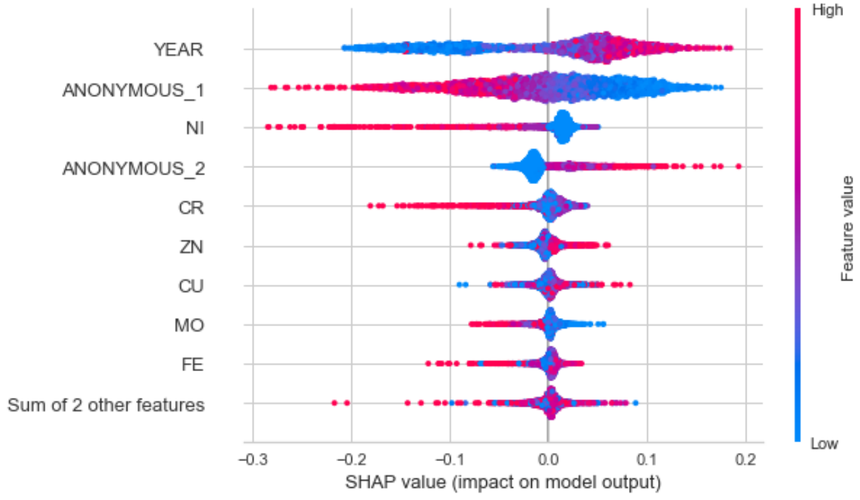
Dataset 3



Dataset 4



Dataset 5



Results

Submission

20	욱근	욱근	0.57507	29	2일 전
21	404 Error		0.57499	36	12일 전
22	알레나		0.5748	10	16일 전
23	RaOOn		0.57461	55	2일 전
24	dlt3		0.57461	10	하루 전
25	sky4268	sk	0.5746	5	하루 전
26	T1	zz 영데	0.57438	34	하루 전
27	out_of_cage	ou ou	0.57423	20	16일 전
28	Oiler_H	싸고 망포	0.57419	20	하루 전

Submission Score : 0.57438

26 place out of 1086 participants, 517 teams

04. Conclusion

Significance

1. Tried many Experiments to handle Imbalanced Data in Regression

2. To prevent information loss, we made our own solution using all data

3. We found important features that influences Abnormal status in Diagnostic Environment

4. Our Model and Strategy can contribute to Construction Machinery Intelligence

Limitations

1. Not enough Domain Knowledge -> Preprocess Missing Value

2. Focused more on Regression, better classification model could make better score

3. Didn't try Deep Learning Model

Thank You
&
Merry Christmas