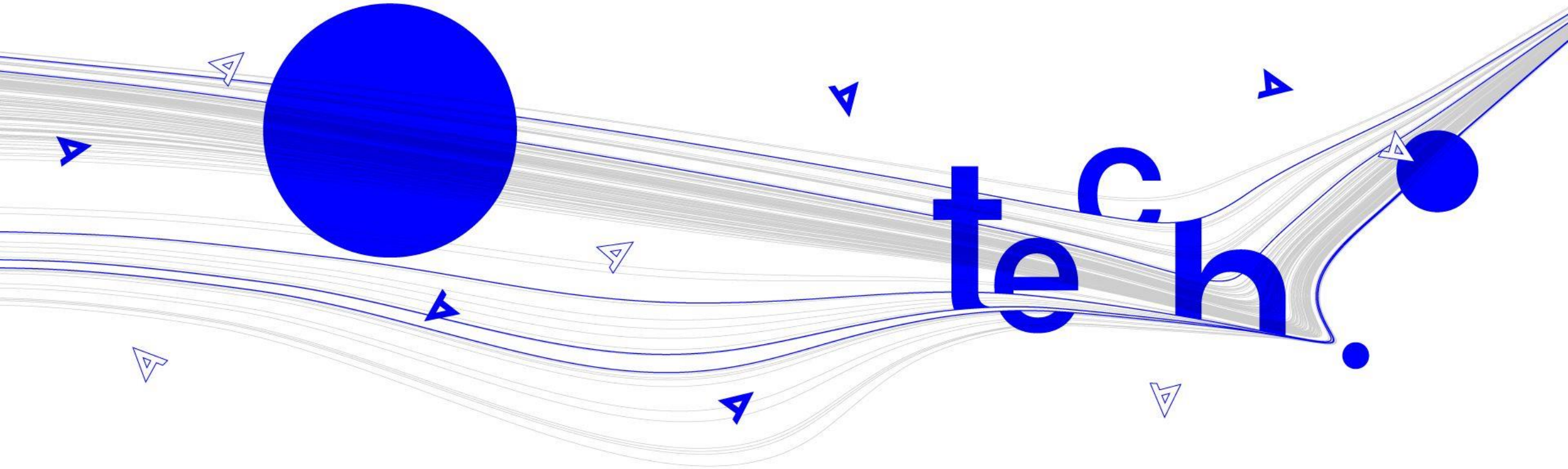


머신러닝을 활용한 중소기업 부도예측 연구

발표자 : 허선우

LABS

LABS

정확한 부도 예측을 위한 머신러닝의 활용성이 높아지고 있는 상황

- 전통적 통계기법은 여러 가정들을 충족해야 된다는 한계점 존재
- 머신러닝과 딥러닝 기법은 비모수적 방법론이므로 여러 가정을 충족할 필요가 없음
- 1980년대 이후 인공신경망, SVM 등 머신러닝 모델이 전통적 통계기법 대비 우수한 예측력을 보임

[전통적 통계기법 및 머신러닝을 활용한 부도예측 연구의 흐름]

연구자	연구 제목	비고
Beaver(1966)	Financial ratios as predictors of failure	단변량 판별 분석 활용 제안
Altman(1968)	Financial ratios, discriminant analysis and the prediction of corporate bankruptcy	다변량 판별 분석 활용 제안
Ohlson(1980)	Financial ratios and the probabilistic prediction of bankruptcy	로지스틱 회귀분석 활용 제안
Zmijewski(1984)	Methodological issues related to the estimation of financial distress prediction models	프로빗 분석 활용 제안
Odom & Sharda(1990)	A neural network model for bankruptcy prediction	인공신경망 활용 제안
Shin et al.(2005)	An application of support vector machines in bankruptcy prediction model	SVM 적용

LABS

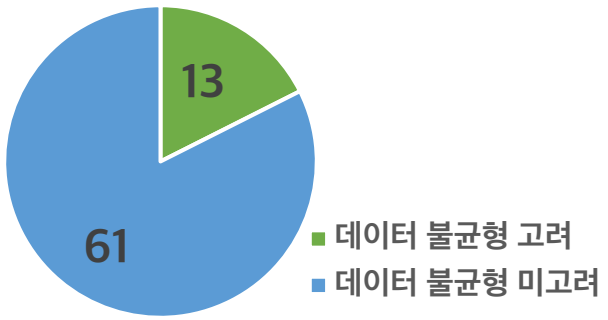
한편, 머신러닝 기법이 사용되면서 데이터 불균형 문제와 모델 투명성 문제가 대두됨

- **데이터 불균형 문제**
  - 기업 부도 데이터는 건전기업과 부도기업의 비율이 1000:1 까지 나는 전형적인 불균형 데이터임
  - 전통적 통계기법의 경우 여러 가정을 적용하는 과정에서 데이터 불균형을 해소

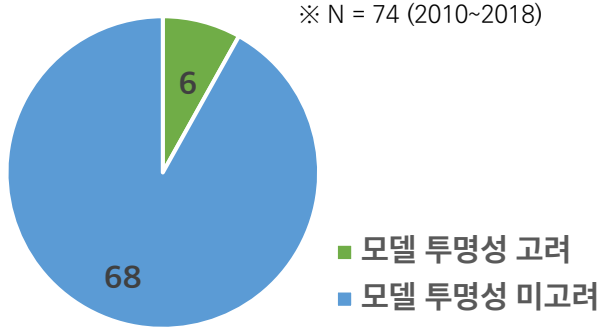
- **모델 투명성 문제**
  - 정확도 향상을 위해 복잡한 모델을 설계한 결과, 모델의 결과 값이 복잡하여 해석할 수 없는 문제가 발생
  - 최근 AI윤리 이슈로 인해 모델의 투명성 확보에 대한 규제강화 및 법제화가 진행 중임

[기업 신용평가/부도예측 연구의 데이터 불균형, 모델 투명성 고려 여부(Dastile et al.,2020)]

※ N = 74 (2010~2018)



데이터 불균형 미고려 비율 82%



모델 투명성 미고려 비율 92%

LABS

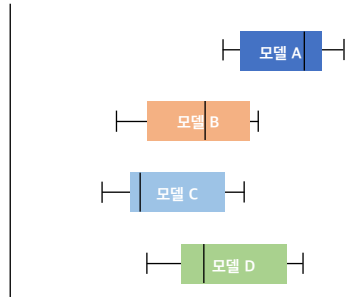
이에 따라, 데이터 불균형 문제와 모델 투명성 문제를 고려하여 연구목표와 평가 지표를 설정하고, 방법론을 구성하였음

연구목표

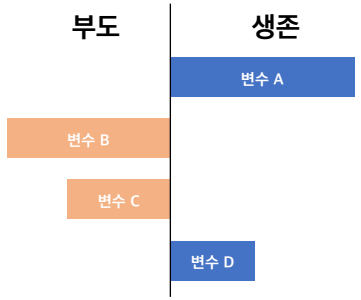
1. 데이터 불균형을 고려하였을 때 어떤 ML모델이 가장 우수한가?
2. 각 재무변수는 예측결과에 어떤 영향을 미쳤는가?

Key Question

[모델 성능 평가]



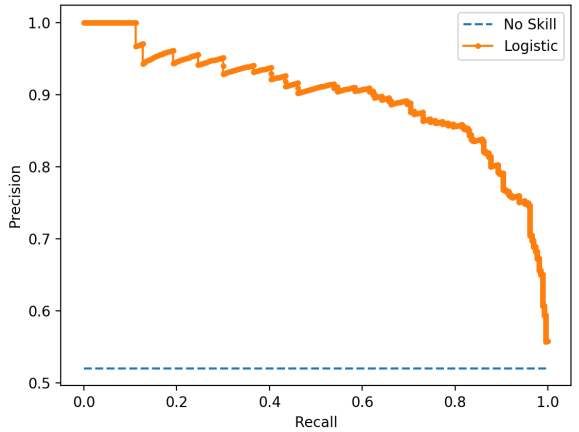
[설명가능성 분석]



평가 지표 설정

- 불균형이 심한 경우 Recall의 증가로 인해 Precision이 낮더라도 F1이 높은 경우가 많음
- Average Precision은 Precision-Recall Curve의 면적임
- Recall과 Precision을 모두 고려할 수 있는 지표

Average Precision



연구 방법론

1. 데이터 수집
2. 특성공학
3. 하이퍼파라미터 튜닝
4. 앙상블 모델 개발
5. XAI

LABS

부도예측 모델링을 위해 국내 중소기업 데이터를 크롤링하여 데이터를 확보하였음

- 공정거래위원회에 등록된 130만 여개의 통신판매사업자 사업자등록번호 확보
- 중소기업현황정보시스템에 사업자등록번호를 검색하여 14만여 건의 재무데이터의 확보
- 국세청에 사업자등록번호를 검색하여 폐업일자 및 폐업여부 확인
- 전체 141,525개 결산재무제표(row)확보, 언더샘플링을 거쳐 9,956건만을 분석에 활용하였음

(index) 사업자등록번호 수집

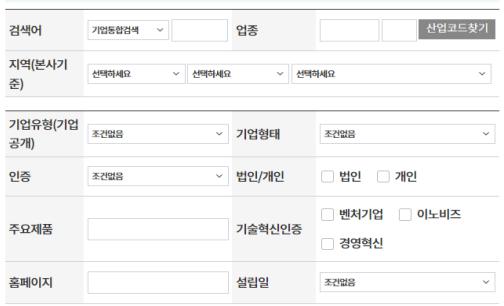
공정거래위원회

- 통신판매사업자 사업자등록번호 130만 건 확보

(X_data) 재무데이터 확보

중소기업현황정보시스템

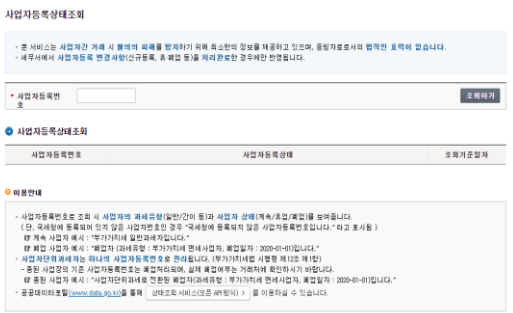
- 14만 건의 결산재무제표 확보
- 기간 : 2018~2021
- 대상기업 : 8만 여개



(y_data) 폐업여부, 폐업일자 확인

국세청

- 8만 여개 기업 대상
- 폐업여부 및 폐업일자 확인



데이터 불균형 처리

언더샘플링

140,569 : 956

9000 : 956

LABS

확보한 데이터를 활용해 Feature Engineering 수행

▪ 크롤링한 데이터에서 활용가능한 6개 변수의 조합을 통해 12개의 추가 파생변수 생성

구분	변수
결산재무제표	총자산
	자본금
	자본총계
	매출액
	영업이익
	당기순이익

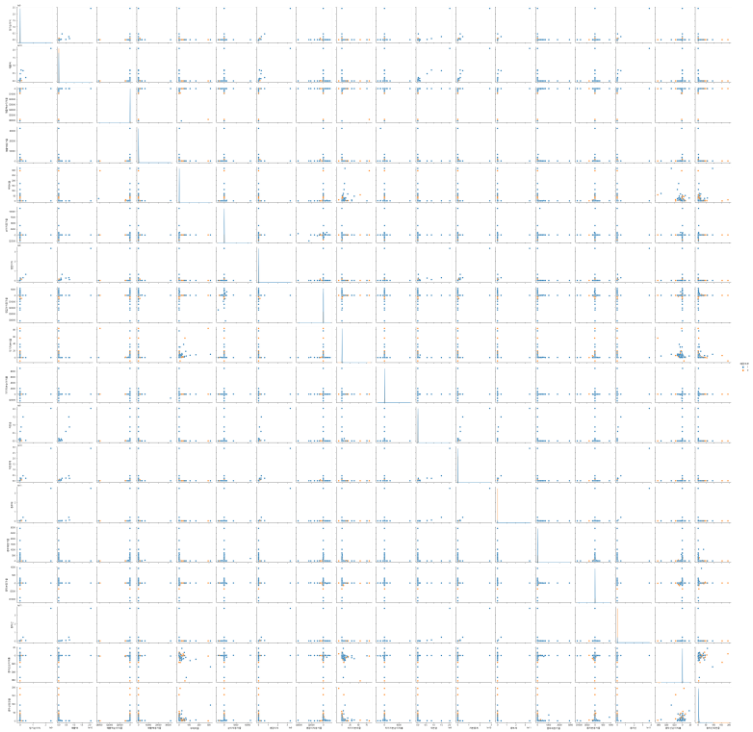
구분	변수	계산식
활동성 지표	총자산회전율	매출액/총자산
수익성 지표	매출액순이익률	당기순이익/매출액
	총자산순이익률	당기순이익/총자산
	자기자본순이익률	당기순이익/자본금
안정성 지표	총부채	총자산 – 자본총계
	부채비율	총부채 / 총자산
	자기자본비율	자본금 / 총자산
성장성 지표	총자본증가율	직전년도 대비 증가율
	총부채증가율	
	매출액증가율	
	영업이익증가율	
	순이익증가율	

LABS

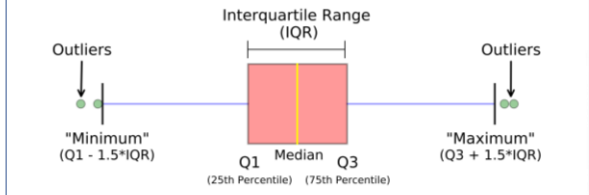
일반화된 순열 중요도 산정을 위해 IQR 방법론을 활용해 이상치를 처리하였음

- Pairplot을 통해 높은 왜도를 확인, 다수의 이상치가 있어 일반화된 순열 중요도 산정이 어렵다고 판단
- X_train을 X_train_new와 X_val로 분할, 이상치 처리 함수를 pipeline에 추가하여 전처리

이상치 처리 전 (X_train)

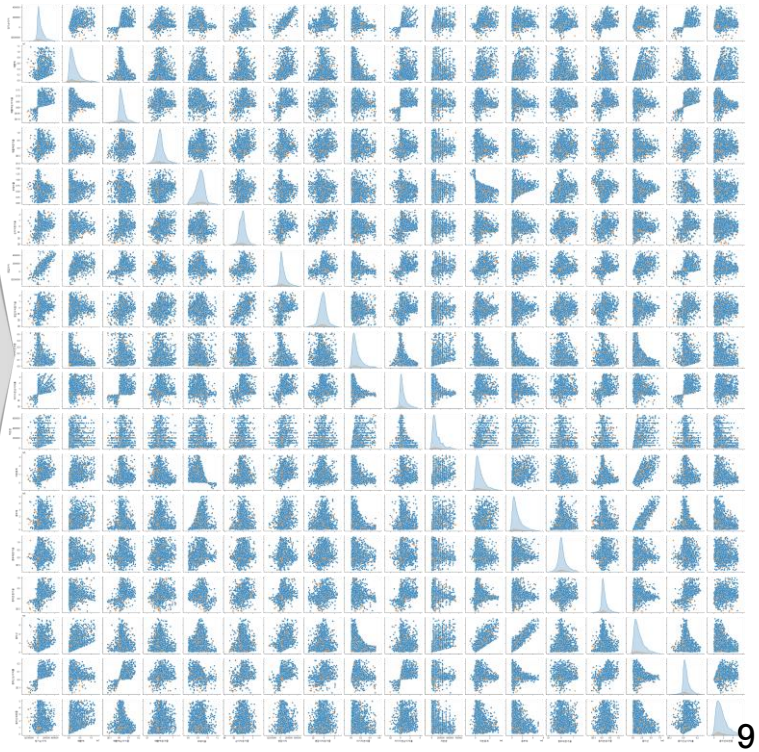


이상치 처리




```
from sklearn.utils.validation import check_array, check_is_fitted
from scipy import sparse
from sklearn.base import BaseEstimator, TransformerMixin
import numpy as np

class drop_outlier(BaseEstimator, TransformerMixin):

    def __init__(self, copy = True, with_IQR = True, with_upper = True, with_lower = True):
        self.with_IQR = with_IQR
        self.with_upper = with_upper
        self.with_lower = with_lower
        self.copy = copy

    def fit(self, X, y=None):

        data_IQR = np.quantile(X, 0.75, axis=0) - np.quantile(X, 0.25, axis=0)
        data_upper = np.quantile(X, 0.75, axis=0) + data_IQR*1.5
        data_lower = np.quantile(X, 0.25, axis=0) - data_IQR*1.5

        self.upper_ = data_upper
        self.lower_ = data_lower

        return self

    def transform(self, X, y=None, copy=None):

        copy = copy if copy is not None else self.copy

        self.upper_ = data_upper
        self.lower_ = data_lower

        if self.with_lower:
            X = X[X > self.lower_]

        if self.with_upper:
            X = X[X < self.upper_]

        return X
```

이상치 처리 후 (X_train_new)



LABS

X_train_new와 X_val 간의 순열중요도를 분석하여 전체 변수 18개 중, 중요 변수 9개를 선택하였음

- 예측모델 : LightGBM
- 반복횟수 : 1000
- Scoring : Average Precision*

Select

Weight	Feature
0.0045 ± 0.0046	총자산
0.0045 ± 0.0052	영업이익증가율
0.0032 ± 0.0047	매출액
0.0022 ± 0.0045	부채비율
0.0016 ± 0.0059	자기자본순이익률
0.0015 ± 0.0036	총부채
0.0015 ± 0.0050	매출액증가율
0.0013 ± 0.0033	자본총계
0.0013 ± 0.0040	총자본증가율

Drop

Weight	Feature
0.0004 ± 0.0037	영업이익
-0.0000 ± 0.0043	총부채증가율
-0.0001 ± 0.0036	총자산순이익률
-0.0003 ± 0.0037	매출액순이익율
-0.0007 ± 0.0039	순이익증가율
-0.0011 ± 0.0057	자기자본비율
-0.0011 ± 0.0031	당기순이익
-0.0013 ± 0.0031	자본금
-0.0014 ± 0.0036	총자산회전율

LABS

이전연구를 바탕으로 6개 모델을 선정하였고, 선정된 모델은 아래 범위에 대해 하이퍼파라미터 튜닝을 수행하였음

- 알고리즘 : 베이지안 최적화
- 반복횟수 : 100
- CV : 3
- Scoring : Average Precision

[2010~2018년 부도예측 연구 모델 (Dastile et al.,2020)]

약자	모델명	선행연구 출현빈도
LR	로지스틱 회귀(Logistic Regression)	38
NB	나이브 베이즈(Naïve Bayes)	7
LDA	선형 판별 분석(Linear Discriminant Analysis)	5
XGB	XGBoost(Extreme Gredient Boosting)	4
EML	극학습기계(Extreme Learning Machines)	2
k-NN	k-최근접 이웃(k-Nearest Neighbor)	10
SVM	서포트 벡터 머신 (Support Vector Machine)	43
ANN	인공신경망(Artificial Neural Network)	31
BA	배깅(Bagging)	13
BO	부스팅(Boosting)	16
RF	랜덤 포레스트(Random Forest)	13
RBM	제한 볼츠만 머신(Restricted Boltzmann Machine)	4
DBN	심층신뢰망(Deep Belief Network)	6
DMLP	심층 레이어 퍼셉트론(Deep Multi-Layer Perceptron)	4
CNN	합성곱신경망(Convolutional Neural Network)	3

[선정 모델 및 하이퍼파라미터 튜닝 범위]

LGBM		RF		XGB	
하이퍼파라미터	범위	하이퍼파라미터	범위	하이퍼파라미터	범위
n_estimators	500	n_estimators	500	n_estimators	500
learning_rate	0 ~ 1	max_depth	Int(2 ~ 20)	scale_pos_weight	0.108
num_leaves	int(20 ~ 50)	max_features	0.5 ~ 1	max_depth	Int(2 ~ 20)
max_depth	Int(2 ~ 20)	-	-	gamma	0.05 ~ 1
-	-	-	-	learning_rate	0.05 ~ 1

MLP		LR		SVM	
하이퍼파라미터	범위	하이퍼파라미터	범위	하이퍼파라미터	범위
solver	'lbfgs'	penalty	[l1, l2]	Kernel	[linear, rbf, sigmoid]
alpha	0.05 ~ 1	solver	[saga, liblinear]	gamma	0.05 ~ 1
hidden_layer_sizes	(32, 128, 64)	C	0.05 ~ 1	C	0.05 ~ 1
activation	[logistic, relu, tanh]	-	-	-	-
Batch_size	[64, 128, 256, 512]	-	-	-	-

LABS

최종적으로 결정된 하이퍼파라미터 튜닝 결과는 다음과 같음

- 알고리즘 : 베이지안 최적화
- 반복횟수 : 100
- CV : 3
- Scoring : Average Precision

LGBM_tuned		RF_tuned		XGB_tuned	
하이퍼파라미터	범위	하이퍼파라미터	범위	하이퍼파라미터	범위
n_estimators	500	n_estimators	500	n_estimators	500
learning_rate	0.5283	max_depth	20	scale_pos_weight	0.108
num_leaves	39	max_features	0.6519	max_depth	18
max_depth	14	-	-	Gamma	0.4546
-	-	-	-	learning_rate	0.1969
Best Average Precision	0.9281	**Best Average Precision**	0.9362	**Best Average Precision**	0.9323

MLP_tuned		LR_tuned		SVM_tuned	
하이퍼파라미터	범위	하이퍼파라미터	범위	하이퍼파라미터	범위
solver	'lbfgs'	penalty	l1	Kernel	rbf
alpha	0.7084	solver	liblinear	gamma	0.9898
hidden_layer_sizes	(32, 128, 64)	C	0.0832	C	0.3169
activation	relu	-	-	-	-
Batch_size	128	-	-	-	-
Best Average Precision	0.9319	**Best Average Precision**	0.9297	**Best Average Precision**	0.9244

LABS

AP를 기준으로 각 모델을 비교한 결과 로지스틱 회귀와 XGBoost가 가장 성능이 뛰어난 것으로 나타남

- **Average Precision(AP)를 기준으로 Baseline과 비교한 결과, 모든 모델이 기준 이상으로 나타남**

Baseline*	
Accuracy	0.9041
Recall	1
Precision	0.9041
F1	0.9496
Average precision	0.9041
ROC_AUC	0.5

LGBM		RF		XGB	
Accuracy	0.8995	Accuracy	0.9026	Accuracy	0.8770
Recall	0.9916	Recall	0.9966	Recall	0.9533
Precision	0.9061	Precision	0.9051	Precision	0.9142
F1	0.9469	F1	0.9487	F1	0.9334
Average precision	0.9321	Average precision	0.9351	Average precision	0.9367
ROC_AUC	0.6155	ROC_AUC	0.6269	ROC_AUC	0.6294

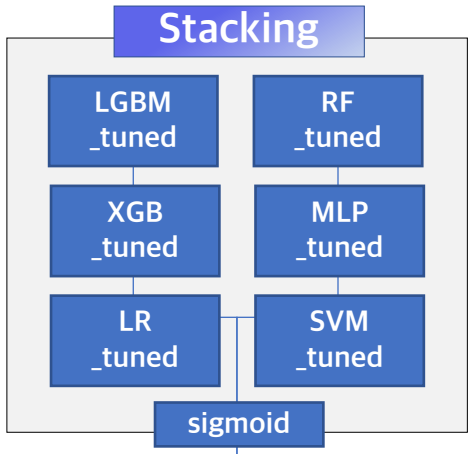
MLP		LR		SVM	
Accuracy	0.9041	Accuracy	0.9041	Accuracy	0.9041
Recall	1	Recall	1	Recall	1
Precision	0.9041	Precision	0.9041	Precision	0.9041
F1	0.9496	F1	0.9496	F1	0.9496
Average precision	0.9321	Average precision	0.9367	Average precision	0.9286
ROC_AUC	0.6184	ROC_AUC	0.6305	ROC_AUC	0.6081

* Baseline의 경우 모든 값을 1로 예측한 상황을 구현
** 모델 성능은 test dataset에 대한 예측값을 비교

LABS

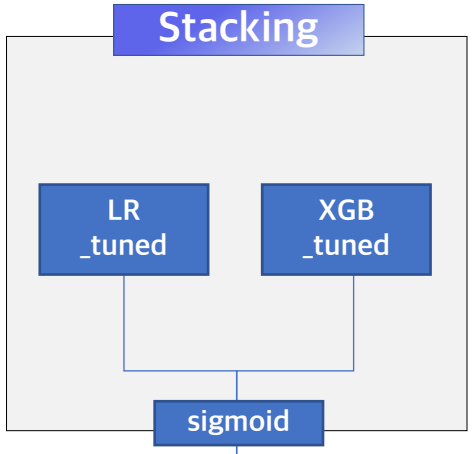
6개 모델을 Stacking 하여 Ensemble 모델을 구현함

- Ensemble_2 모델이 가장 성능이 뛰어난 것으로 나타남



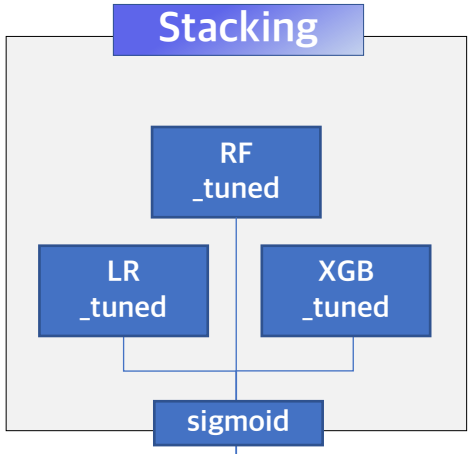
Ensemble_1

Ensemble_1(전체)	
Accuracy	0.9041
Recall	1
Precision	0.9041
F1	0.9496
Average precision	0.9429
ROC_AUC	0.6480



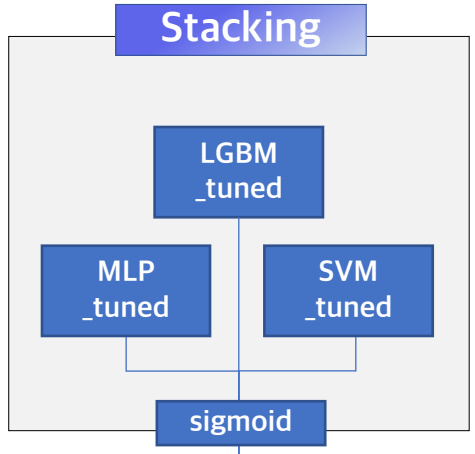
Ensemble_2

Ensemble_2(상위 2개)	
Accuracy	0.9041
Recall	1
Precision	0.9041
F1	0.9496
Average precision	0.9439
ROC_AUC	0.6441



Ensemble_3

Ensemble_2(상위 3개)	
Accuracy	0.9041
Recall	1
Precision	0.9041
F1	0.9496
Average precision	0.9424
ROC_AUC	0.6466



Ensemble_4

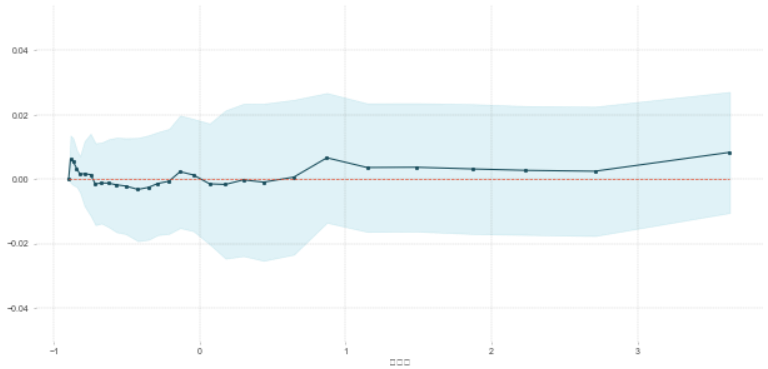
Ensemble_4(하위 3개)	
Accuracy	0.9041
Recall	1
Precision	0.9041
F1	0.9496
Average precision	0.9371
ROC_AUC	0.6426

LABS

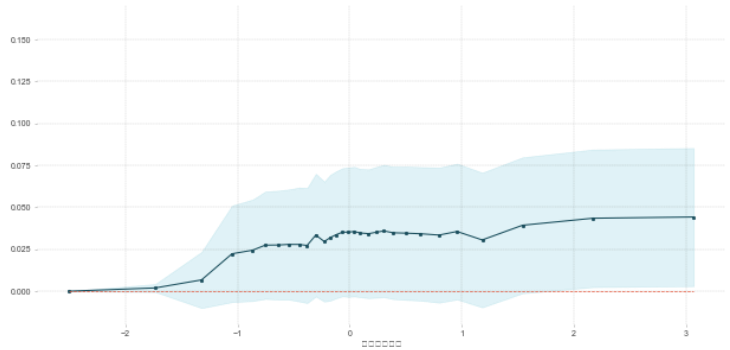
다음은 Ensemble_2의 중요변수 9개에 대해 부분의존도를 분석한 결과임

- 매출액과 매출액 증가율은 높을 수록 생존확률이 증가드는 것으로 나타남
- 영업이익증가율의 경우 높아지더라도 큰 영향이 없는 것으로 나타남

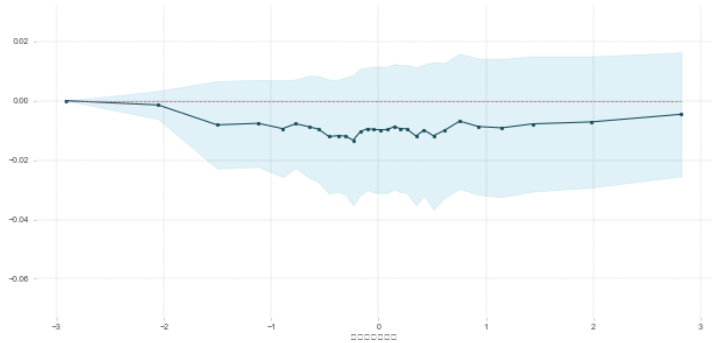
매출액



매출액증가율



영업이익증가율



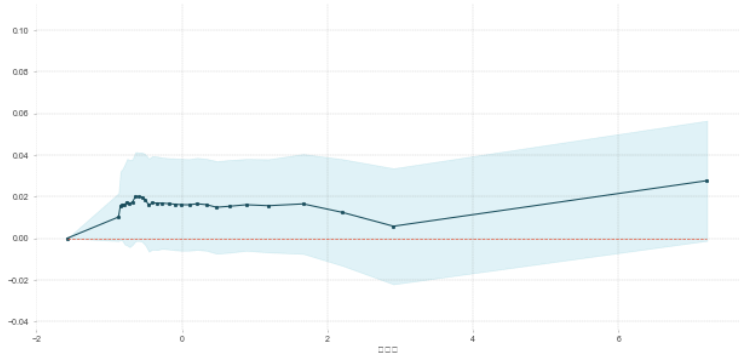
- 모든 값은 표준화 된 값임

LABS

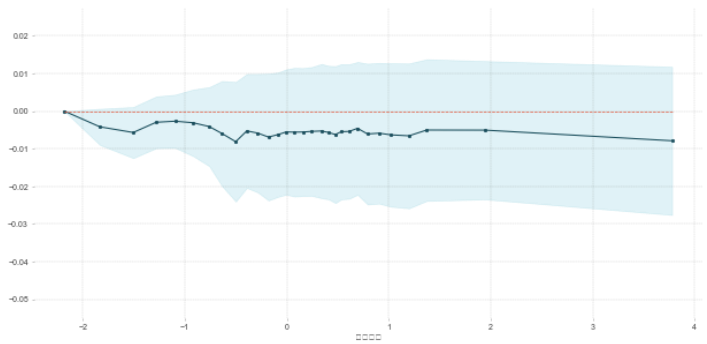
다음은 Ensemble_2의 중요변수 9개에 대해 부분의존도를 분석한 결과임

- 부채가 높은 수록 생존확률이 증가하는 것으로 나타남
- 부채비율이 높을 수록 생존확률이 감소하는 것으로 나타남
- 총자산의 경우 일정 수준 이상인 경우 큰 차이가 없는 것으로 나타남

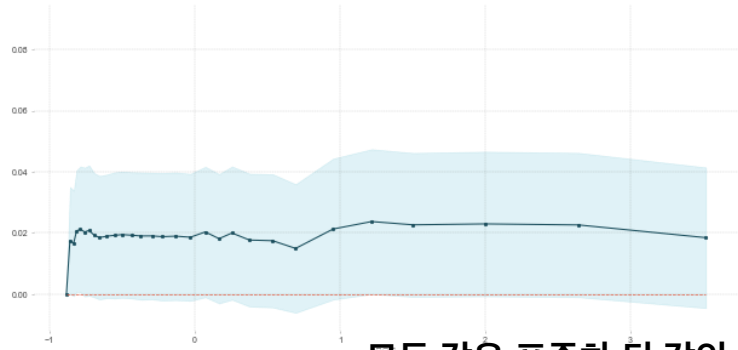
총부채



부채비율(총부채 / 총자산)



총자산(총부채 + 자본총계)



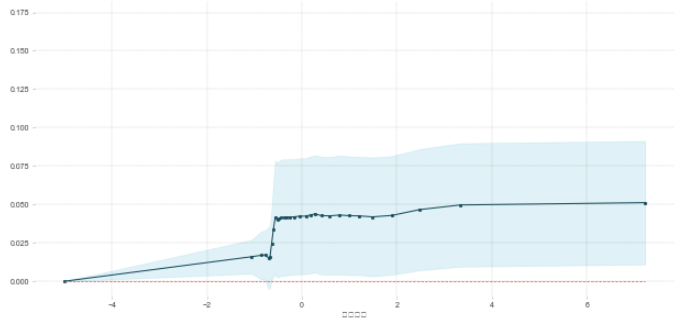
- 모든 값은 표준화 된 값임

LABS

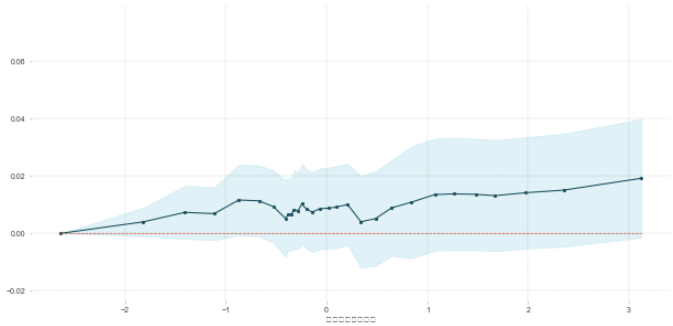
다음은 Ensemble_2의 중요변수 9개에 대해 부분의존도를 분석한 결과임

- 자본총계가 높을 수록 생존확률이 증가하는 것으로 나타남
- 자기자본순이익률은 높을 수록 생존확률이 증가하는 것으로 나타남
- 총자본증가율의 경우 높을 수록 생존확률이 감소하는 것으로 나타남

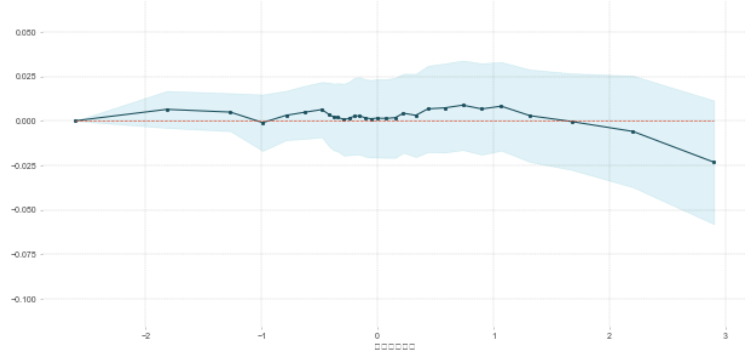
자본총계



자기자본순이익률



총자본증가율



- 모든 값은 표준화 된 값임

III. 종합적 제언

LABS

9개 중요변수에 대한 XAI 해석 결과를, 서로 관련이 높은 결과와 결합하여 재해석하였음

매출 구조

- 매출액과 매출액 증가율은 높을 수록 생존확률이 증가드는 것으로 나타남
- 영업이익증가율의 경우 높아지더라도 큰 영향이 없는 것으로 나타남

자본 구조

- 자본총계가 높을 수록 생존확률이 증가하는 것으로 나타남
- 자기자본순이익률은 높을 수록 생존확률이 증가하는 것으로 나타남
- 총자본증가율의 경우 높을 수록 생존확률이 감소하는 것으로 나타남

부채 및 자산 구조

- 부채가 높은 수록 생존확률이 증가하는 것으로 나타남
- 부채비율이 높을 수록 생존확률이 감소하는 것으로 나타남
- 총자산의 경우 일정 수준 이상인 경우 큰 차이가 없는 것으로 나타남

Insight 1

- 단순히 영업이익이 증가하는 것보다는 자본대비, 투자대비 이익률이 높은 사업을 영위하는 것이 중요함

Insight 2

- 총자본증가율이 높은 경우, 전년 대비 사업이 급격하게 확장된 것으로 볼 수 있음
- 사업역량을 제대로 갖추지 못한 상태로 양적 성장을 기록한 것이므로 외부 환경 변화에 취약한 상태로 볼 수 있음
- 이를 대변하는 지표가 바로 자기자본순이익률이며, 자기자본순이익률은 규모 대비 이익의 효율성을 나타냄
- 결과적으로, 작은 사업이더라도 무리하게 확장하지 않고 사업의 효율성, 내부역량을 다지는 것이 중요함

Insight 3

- 자산에 따른 생존확률 변화가 크지 않다는 것과 부채, 부채비율 증가에 따른 생존확률 변화를 고려했을 때
- 자산 자체보다 자산(부채 + 자본)의 구조가 더 중요함을 시사하며, 단순한 부채증가를 걱정하기보다는 적절한 수준의 부채비율을 유지할 필요가 있음