Marchine Learning - Kaggle Competition

2조 - 최우연, [redacted]

Intro. 머신러닝 분석 과정

Kaggle Competition - bank marketing 참여여부 예측모델 수행 과정

Step1. 데이터 확인

- 독립변수/ 종속변수 확인
- 변수의 특성확인 (연속형/범주형)
- YData Profiling
- 적용가능한 분석모델 확인

Step2. 데이터 전처리

- 파생변수 생성
- 이상치 확인 후 처리
- 표준화, 정규화

Step3. 특성 선택

- 상관관계 분석
- FDR(False Discovery Rate)
- RFECV

Step4. 모델 학습

- 머신러닝 알고리즘 적용
- 하이퍼파라미터 튜닝
- K-Fold 교차검증

Step5. 성능평가

- 모델 성능 비교
- 베스트모델 성능확인

Step6. 예측 및 결과

- 베스트 모델을 통한 예측
- Permutation Importance

Step 1. 데이터 확인

예측에 활용할 데이터 이해 및 탐색

1. 독립변수/종속변수 확인

독립변수 : 행 31647개, 열 15개
종속변수 : label(가입여부 - 가입(1)/미가입(0))

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 31647 entries, 0 to 31646
Data columns (total 18 columns):
 #   Column      Non-Null Count  Dtype
---  ------      --------------  -----
 0   ID          31647 non-null  object
 1   age         31647 non-null  int64
 2   job         31647 non-null  object
 3   marital     31647 non-null  object
 4   education   31647 non-null  object
 5   default     31647 non-null  object
 6   balance     31647 non-null  int64
 7   housing     31647 non-null  object
 8   loan        31647 non-null  object
 9   contact     31647 non-null  object
 10  day         31647 non-null  int64
 11  month       31647 non-null  object
 12  duration    31647 non-null  int64
 13  campaign    31647 non-null  int64
 14  pdays       31647 non-null  int64
 15  previous    31647 non-null  int64
 16  poutcome    31647 non-null  object
 17  label       31647 non-null  int64
dtypes: int64(8), object(10)
memory usage: 4.3+ MB
```

2. 변수의 특성 확인

클래스 불균형 데이터 확인

```
label
0    27945
1     3702
Name: count, dtype: int64
```

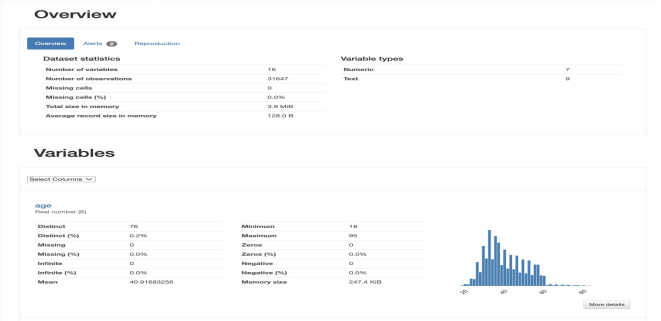
연속형/범주형 데이터 확인

	age	balance	day	duration	campaign	pdays	previous
count	31647.000000	31647.000000	31647.000000	31647.000000	31647.000000	31647.000000	31647.000000
mean	40.916833	1370.050084	15.818277	258.088824	2.752615	40.023604	0.575062
std	10.621773	3122.054996	8.315467	255.737568	3.080952	100.154518	2.433034
min	18.000000	-4057.000000	1.000000	0.000000	1.000000	-1.000000	0.000000
25%	33.000000	73.000000	8.000000	103.000000	1.000000	-1.000000	0.000000
50%	39.000000	451.000000	16.000000	180.000000	2.000000	-1.000000	0.000000
75%	48.000000	1418.500000	21.000000	320.000000	3.000000	-1.000000	0.000000
max	95.000000	102127.000000	31.000000	3881.000000	63.000000	854.000000	275.000000

	ID	job	marital	education	default	housing	loan	contact	month	poutcome
count	31647	31647	31647	31647	31647	31647	31647	31647	31647	31647
unique	31647	12	3	4	2	2	2	3	12	4
top	train00001	blue-collar	married	secondary	no	yes	no	cellular	may	unknown
freq	1	6773	19042	16250	31089	17617	26617	20525	9647	25918

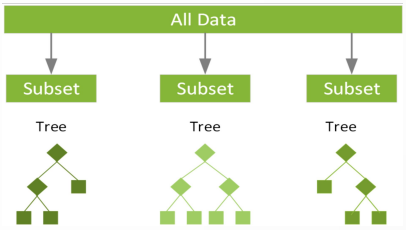
3. YData Profiling

Python라이브러리로 데이터셋에 대한 EDA 수행



4. 적용가능한 모델 선정

범주형과 수치형 데이터를 효과적으로 처리하면서 스케일링 등 전처리 부담이 적은 트리 기반 모델인 Random Forest와 XGBoost를 baseline 모델로 선정

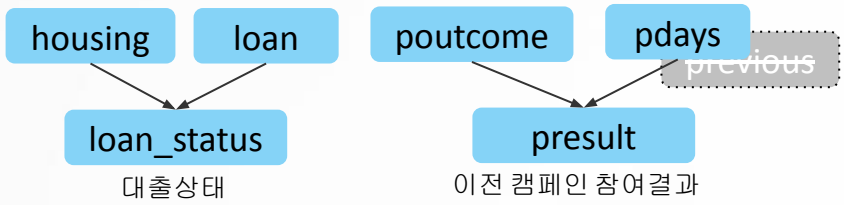


Step 2. 데이터 전처리

예측을 위한 깨끗하고 유의미한 데이터 준비과정

1. 변수 결합으로 파생변수 생성

모델 간소화를 위해 변수결합으로 차원 축소



2. 범주값 재정의로 파생변수 생성

모델의 설명력을 높이기 위한 유사범주 그룹화

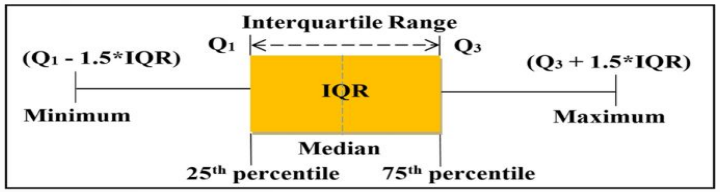
job_group	job
비근로자	'student', 'retired', 'unemployed'
전문직	'management', 'admin.', 'self-employed'
기술직	'technician', 'entrepreneur'
서비스직	'blue-collar', 'services', 'housemaid'

직업군

3. IQR을 이용한 이상치 처리

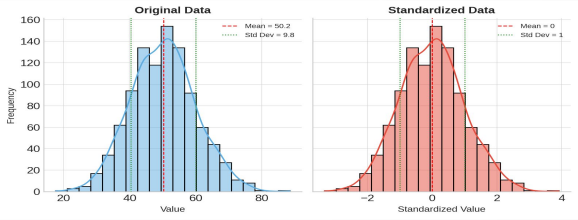
사분위수를 이용한 최소/최대극단값
초과인 값들에 대한 값 대체

- duration(최근 연락시 통화 시간)



4. 표준화, 정규화

연속형 데이터는 StandardScaler로 스케일링
범주형 데이터는 LabelEncoder로 인코딩
-> 머신러닝 모델 학습에 적합한 데이터 형태 변환



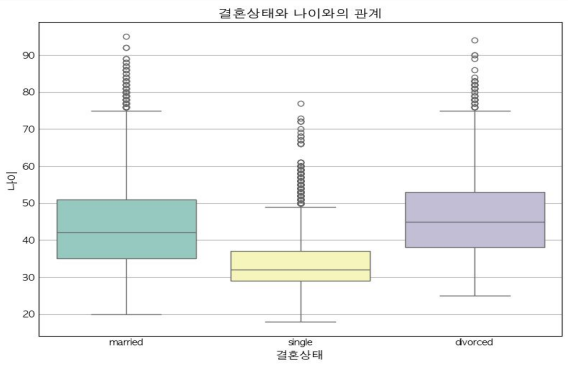
Step 3. 특성 선택(1)

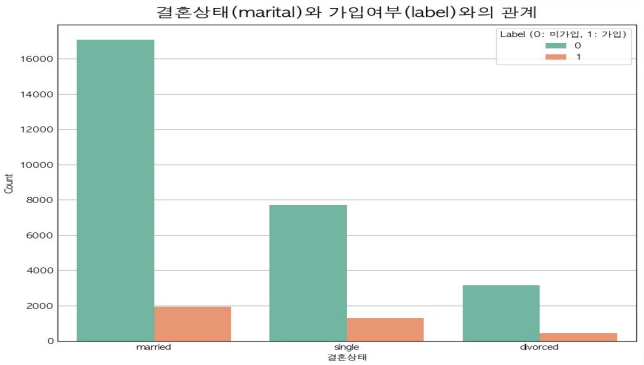
모델의 성능과 효율성 개선을 위한 특성 선택

1. 상관계수를 통한 특성 제외

상관도가 높은 변수 중 하나를 제거 -> 모델이 중복된 정보를 학습하는 것을 방지, 안정성과 예측 정확성 향상

- 결혼상태(marital) 변수는 가입여부(label) 변수와 낮은 상관성을 보이며, 나이(age) 변수와는 상관성이 나타남
  - 탐색 결과, 해당 변수를 제외하는 것이 적절하다고 판단



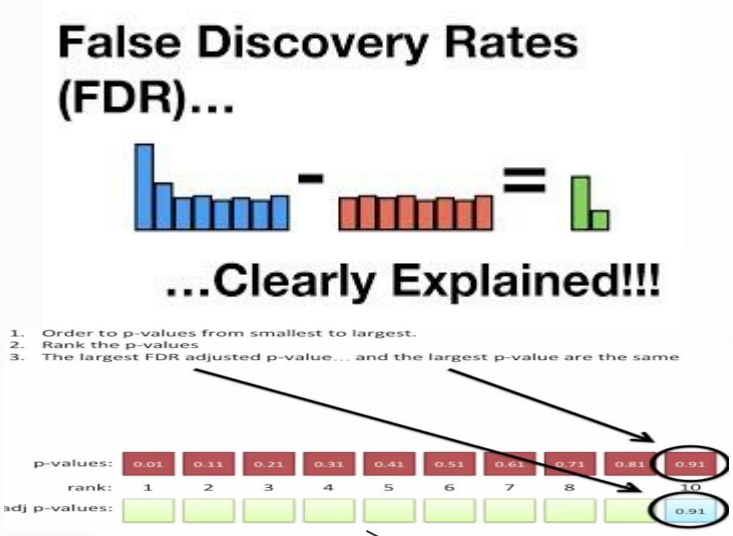


Step 3. 특성 선택(2)

통계적 접근과 모델 기반 접근 방식을 통한 특성 선택

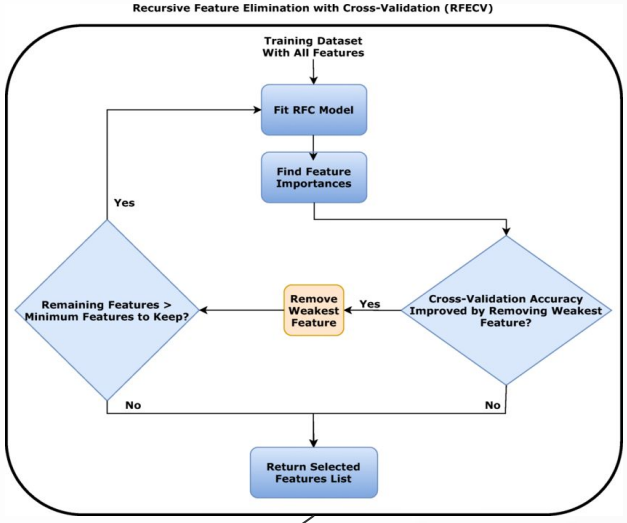
1. False Discovery Rate

연속형 변수들에 대한 잘못된 판단(거짓 발견)의 비율을 통제하고 신뢰성 높은 변수만 선택



2. RFECV

모델의 성능을 잘 내기에 유리한 교차검증을 통한 모델의 정확도를 높일 수 있는 변수들만 선택

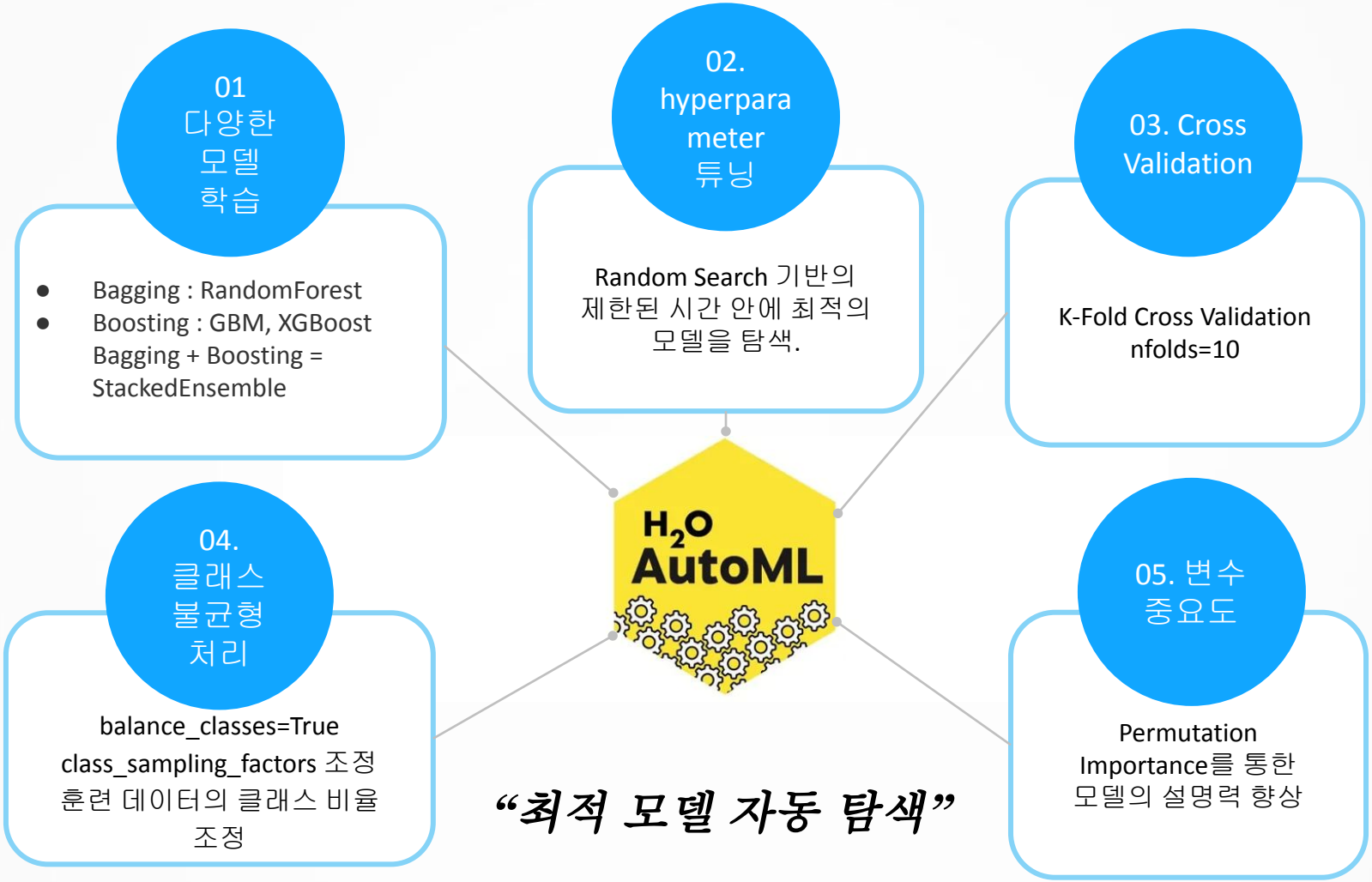


Feature Selection Result

day, month, presult, age, duration, campaign, contact, balance, loan_status

Step 4. 모델 학습

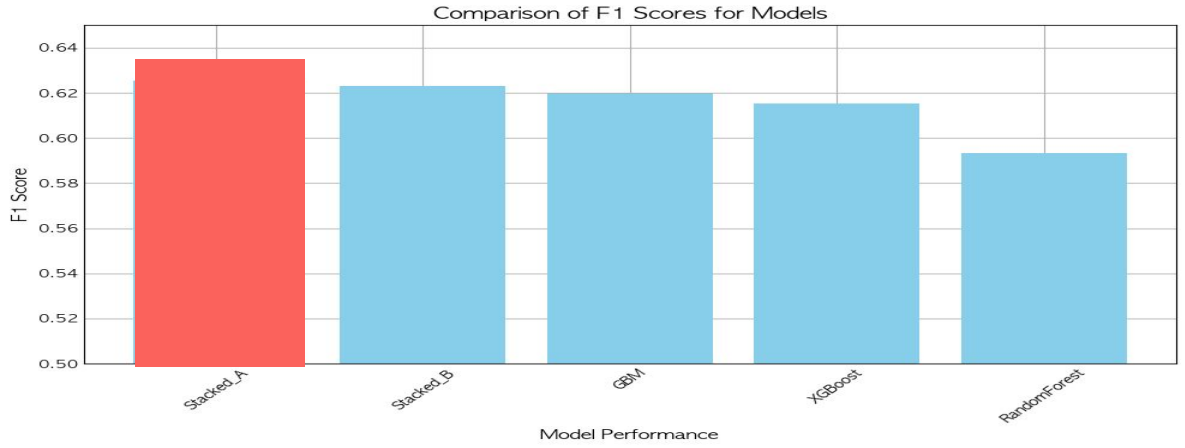
다양한 모델 학습 및 하이퍼파라미터 튜닝을 자동화



Step 5. 성능 평가(1)

다양한 모델 검증을 통한 베스트 모델 선정

1. 머신러닝 알고리즘 별 베스트 모델의 F1 Score 성능비교(by Cross Validation Metrics)



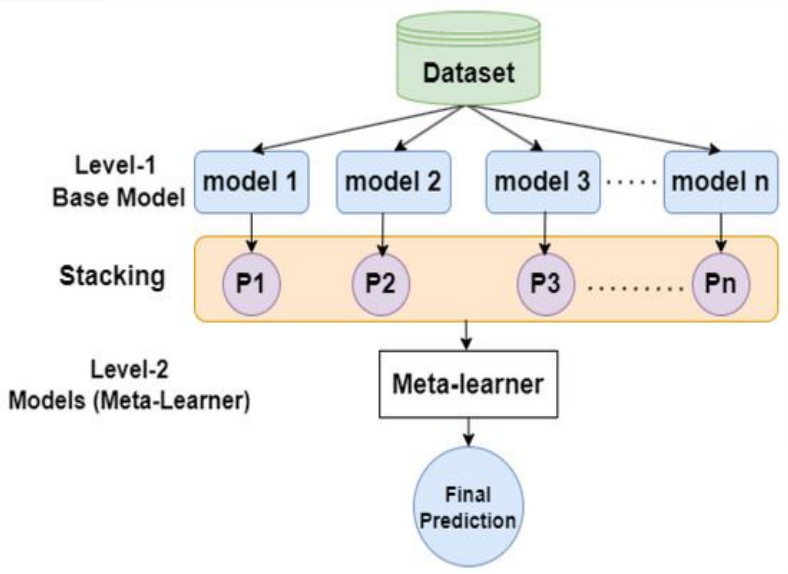
2. 모델별 기타 성능 비교

	aucpr ▲	auc ▲	logloss ▼	mean_per_class_error ▼
StackedEnsemble_AllModels	0.615286	0.933373	0.198316	0.170104
GBM	0.605215	0.928873	0.205849	0.176106
XGBoost	0.592958	0.924955	0.291851	0.177496
RandomForest	0.56601	0.920365	0.250366	0.170555

Step 5. 성능 평가(2)

베스트 모델을 통한 성능 평가

1. StackedEnsemble Model



여러개의 베이스 모델을 결합하여 더 좋은 예측 성능을 내는 방법

1) base models : GBM, RandomForest, XGboost
-> Bagging과 Boosting의 서로 보완적인 특성과 강점조합

2) meta_learner model : GLM
-> GLM은 선형모델로 각 베이스모델의 예측결과를 가중치로 결합
-> 선형조합을 통해 복잡도를 낮추고, 베이스모델의 주요 정보를 간결하게 요약

2. Best Model Summary

Cross-Validation Metrics Summary:

	mean	sd	cv_1_valid	cv_2_valid	cv_3_valid	cv_4_valid	cv_5_valid	cv_6_valid	cv_7_valid	cv_8_valid	cv_9_valid	cv_10_valid
accuracy	0.8970733	0.0059633	0.8922136	0.9087366	0.9007181	0.8971181	0.8967151	0.8911965	0.8957152	0.9026825	0.8974524	0.8881850
aic	1273.7795	45.031708	1277.385	1161.0398	1325.0348	1289.5494	1289.9062	1297.6605	1267.5682	1242.7585	1288.6086	1298.2845
auc	0.9334848	0.0022338	0.9299243	0.9333318	0.9341556	0.9360874	0.9337241	0.933066	0.9358112	0.9355547	0.9335871	0.9296060
err	0.1029267	0.0059633	0.1077864	0.0912634	0.0992819	0.1028819	0.1032849	0.1088035	0.1042848	0.0973175	0.1025476	0.1118150
err_count	325.8	20.670967	335.0	281.0	318.0	332.0	327.0	351.0	331.0	312.0	318.0	353.0
f0point5	0.5756916	0.0169528	0.5652174	0.5742574	0.6038445	0.5839559	0.5808185	0.5622933	0.5739693	0.5740578	0.5949367	0.5435652
f1	0.6312242	0.0109958	0.6306505	0.6228188	0.6450893	0.6414687	0.632171	0.635514	0.6318131	0.6267943	0.6394558	0.606466
f2	0.6991470	0.0163074	0.7132170	0.6803519	0.6923814	0.7115477	0.6934847	0.730659	0.7026225	0.6902002	0.6911765	0.6858296
lift_top_group	7.1165166	0.6784472	7.430328	7.759577	5.867644	7.075092	7.5704155	7.2605095	7.425802	7.9587398	6.2763114	6.5407457
loglikelihood	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
---	---	---	---	---	---	---	---	---	---	---	---	---
mean_per_class_error	0.1651618	0.0099298	0.1557885	0.1724764	0.1734400	0.1588847	0.1703712	0.1433045	0.1633566	0.1685129	0.1741257	0.1713577
mse	0.0627617	0.0024124	0.0638629	0.0572770	0.0646102	0.0622476	0.0639632	0.0618985	0.0627942	0.0607410	0.0653224	0.0648999
null_deviance	2284.2659	95.28581	2252.9453	2060.9067	2402.262	2367.5122	2319.9814	2326.7708	2301.7097	2229.129	2332.442	2249.0
pr_auc	0.6170524	0.0168289	0.6104249	0.6164132	0.6236948	0.6395965	0.628089	0.6171356	0.6341969	0.6025644	0.6177711	0.5806376
precision	0.5439357	0.0208907	0.5286506	0.5458823	0.5791583	0.5510204	0.5509804	0.5221843	0.5409524	0.5435685	0.5685484	0.5084112
r2	0.3917562	0.0142971	0.3853019	0.3849674	0.4049726	0.4102182	0.3930182	0.4002439	0.3959073	0.3816187	0.4006112	0.3607024
recall	0.7536341	0.0273086	0.7814208	0.725	0.7279597	0.7674419	0.7414248	0.8116711	0.7593583	0.740113	0.7305700	0.7513812
residual_deviance	1255.1796	44.956314	1259.385	1143.0398	1307.0348	1271.5494	1271.9062	1277.6605	1249.5682	1222.7585	1270.6086	1278.2845
rmse	0.2504803	0.0048687	0.2527110	0.2393262	0.2541854	0.2494947	0.2529094	0.2487940	0.2505877	0.2464570	0.2555825	0.2547547
specificity	0.9160423	0.0091351	0.9070022	0.9300471	0.9251603	0.9147887	0.9178328	0.9017199	0.9139286	0.9228612	0.9211786	0.9059034

Threshold : 0.2851229

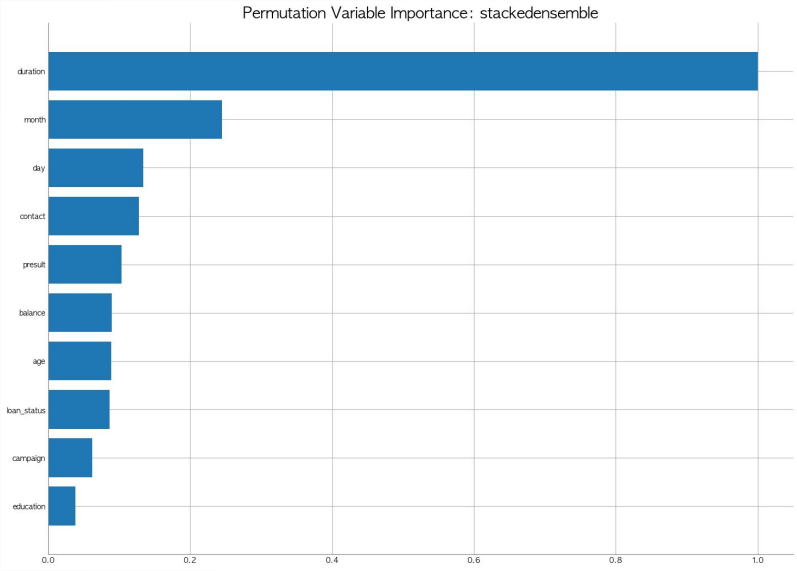
예측값	0(미가입)	1(가입)
실제값		
0(미가입)	25594(TN)	2351(FP)
1(가입)	948(FN)	2754(TP)

F1 = 2 * ((Precision * Recall)/(Precision+Recall))

Step 6. 예측 및 결과

베스트 모델을 통한 예측, 해석, 검증

1. Permutation Importance

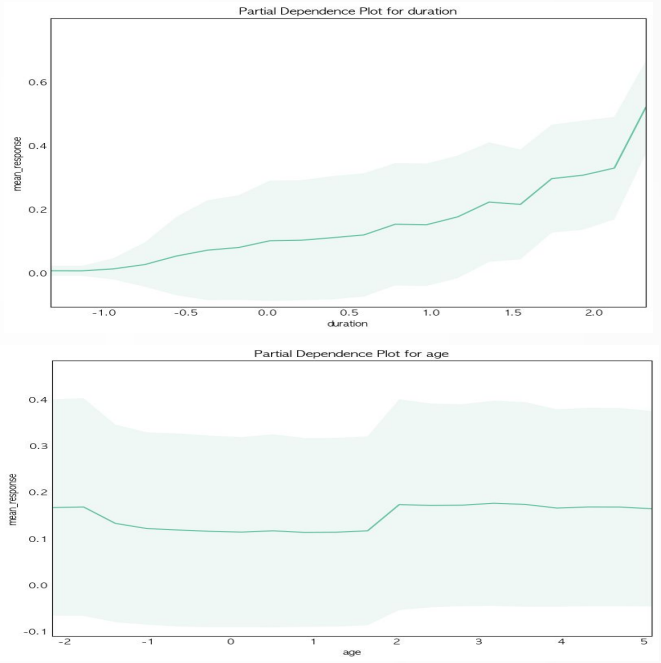


각 변수의 상대적 중요도를 파악하여 모델 해석 가능

모델에서 각 변수가 예측 성능에 얼마나 중요한지를 평가 - 이를 바탕으로 모델을 해석하거나 최적화하는 데 활용

중요변수들 : Duration(최근연락시 통화시간), month(최근 연락을 한 달), presult(이전 캠페인 참여여부) ...

2. PDP Plot



모델의 특정 변수와 타겟간의 직관적인 해석 가능

특정 Feature(특징)가 예측 값에 어떻게 영향을 미치는지 시각화하는 도구로 모델의 설명력을 높임

Thank you