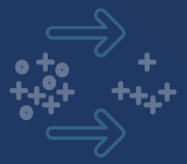
마키아벨리즘 성향 테스트 결과 데이터 기반
국가선거 투표참여 여부 예측

패스트캠퍼스 데이터사이언스 스쿨 14기
머신러닝 프로젝트 3조
박경원 전진경 정성용

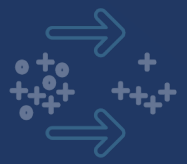






마키아벨리즘 성향 테스트 결과 데이터 기반

국가선거 투표참여 여부 예측

목차




프로젝트 목표

마키아벨리즘 성향 테스트 결과 데이터

Classifier 모델 활용

투표참여 여부 예측

프로젝트 목표

마키아벨리즘 성향 테스트 결과 데이터

Classifier 모델 활용

투표 여부 예측

어떤 성향의 사람이 투표에 참여할까?

데이터 소개

마키아벨리즘이란?

국가의 발전과 인민의 복리증진을 위해서는
어떠한 수단이나 방법도 허용된다는
국가 지상주의적인 정치 이념

출처 - 표준국어대사전

데이터 소개

마키아벨리즘 성향이 **높은** 사람

사람들과 소통할 때
계산적이고 신중하게 접근하는 경향
(낮은 친화력, 높은 성실성)

마키아벨리즘 성향이 **낮은** 사람

사람들과 소통할 때
개인적이며 감정을 이입하여 접근하는 경향
(비교적 수동적, 순응적)

데이터 소개

1. 데이터 내용 (shape = 73489, 105)

: 온라인에서 글로벌하게 진행된 MACH-IV(마키아벨리즘 테스트) 참여자의 테스트 결과 데이터 (출처_심리학 공공데이터 https://openpsychometrics.org/)

2. 데이터 수집기간

: 2017.07.- 2019.03.

3. 데이터 구성 (총 105개 컬럼)

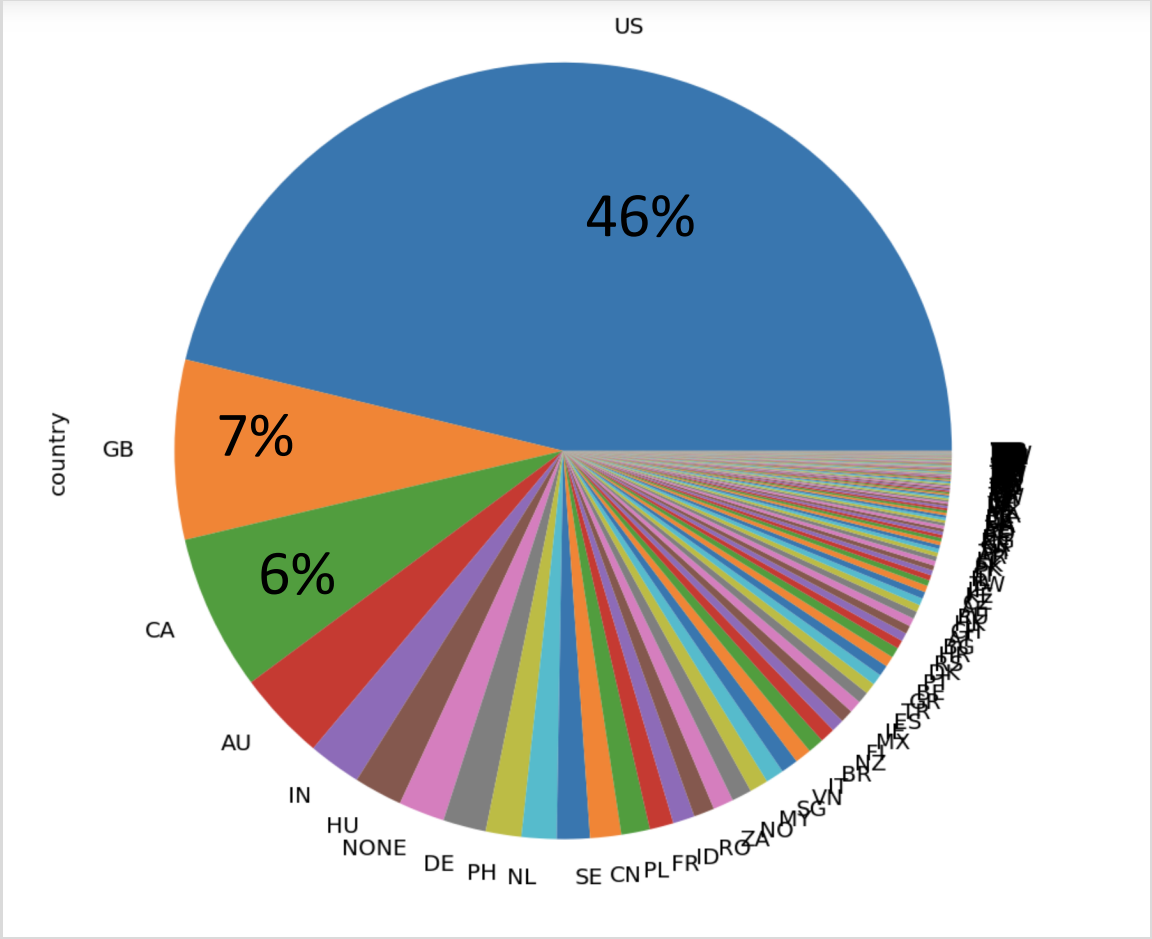
: 마키아벨리즘 테스트 질문지 각각에 대한 답변, 각 질문지의 테스트 내 순서, 참여자의 각 질문지에 대한 답변 시간, 성격 판단 질문지 답변, 답변 신뢰도 측정 질문지 답변, 학력, 고향, 성별, 영어사용여부, 주로 사용하는 손, 종교, 성정체성, 결혼여부, 가족구성원수, 전공, 국가 투표 참여 여부

마키아벨리즘 성향
테스트 결과 데이터 기반
국가선거투표참여 여부 예측

모델링 PROCESS #ver.1

EDA	1차 전처리	AUC 비교	2차 전처리	모델 고도화	모델 앙상블	데이터 예측
데이터 파악	0, null, 이상치 처리	모델 성능의 baseline 확인	Feature 선별	하이퍼 파라미터 튜닝	분류모델 앙상블모델화	최종모델 활용

1) EDA 1
데이터의 편중성 확인

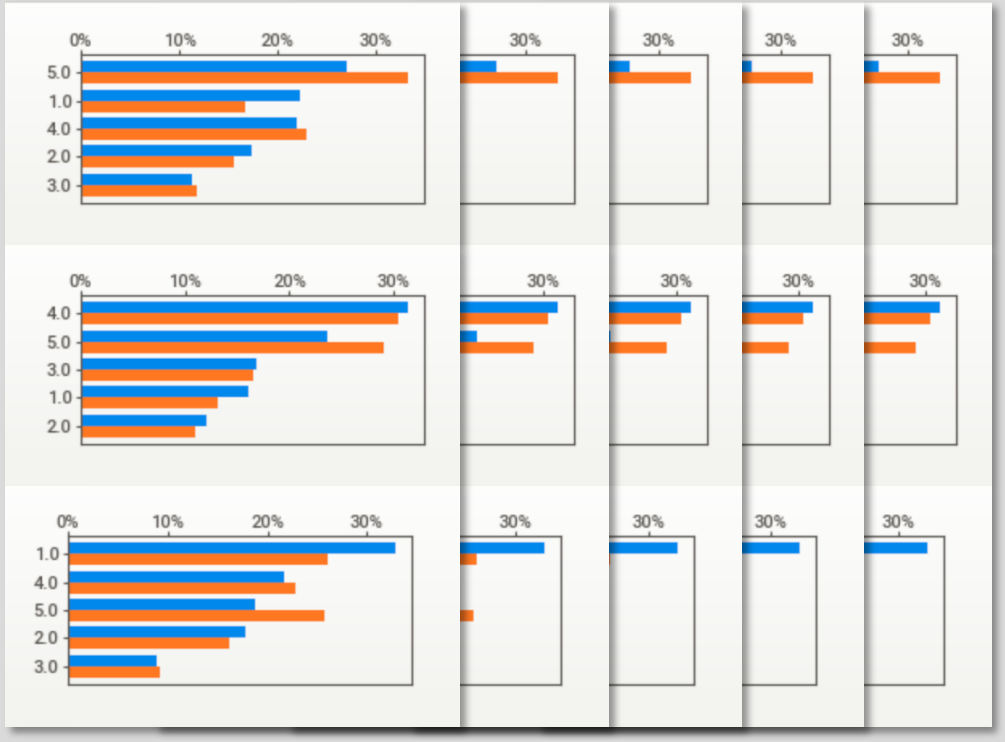


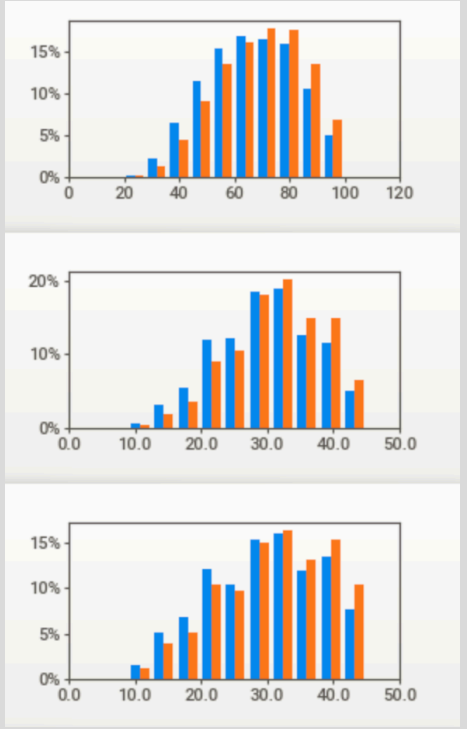
비중 2위 “GB” : 5,011개 (7%)
비중 3위 “CA” : 4,305개 (6%)

총 데이터 수 73,489 중 **“US”국적자 데이터 31,025개 46% 차지**

1) EDA 2

마키아벨리즘 성향 점수화 컬럼 추가 필요성 확인





1) EDA 3

QI 컬럼 데이터 활용 여부 고려

QI - the position of that item in the survey

성향 테스트 결과 데이터의 일관성 유지를 위한
심리검사 방법론적 장치이므로 연관 없는 정보

2) 0, “null” 데이터 처리

“0” 데이터 활용 여부 고려

“0” (o)

“0” (x)

	Model	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC	TT (Sec)
gbc	Gradient Boosting Classifier	0.6972	0.7822	0.7015	0.7066	0.6983	0.3958	0.4000	8.4550
gbc	Gradient Boosting Classifier	0.7008	0.7851	0.7064	0.7118	0.7013	0.4055	0.4113	6.6290

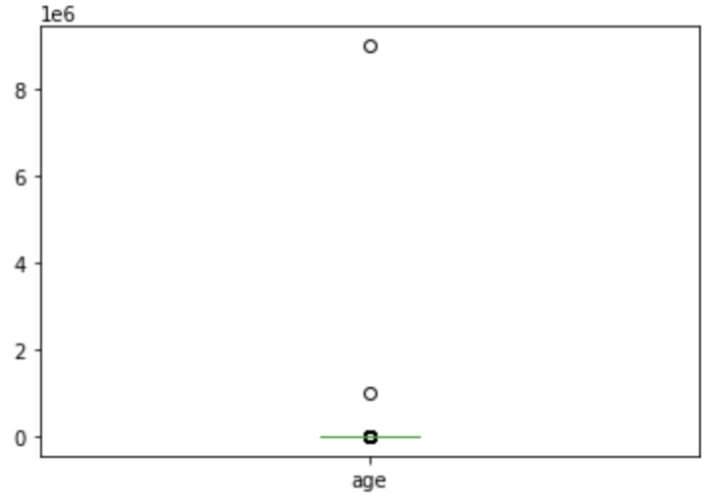
“0” & null 데이터가 모두 제거된 데이터셋의
모델 AUC, Accuracy score가 높음

2) 이상치 처리

"age", "familysize" 컬럼내 이상치 확인

```
df['age'].plot(kind='box')
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x1e7aadfba88>
```



999,999세

```
df_age
```

	age
18	4476
17	3846
20	3610
21	3485
19	3416
...	...
476	1
662	1
87	1
999999	1
410	1

```
df["familysize"].value_counts()
```

```
2             26741
3             16164
1             10157
4              6916
5              2735
6              1156
7               557
⋮                ⋮
2147483647        1
30                1
999               1
2137              1
48                1
18                1
21                1
23                1
900000            1
24                1
96                1
```

형제자매 2,147,483,647명

비정상 데이터가 포함된 age, familysize 컬럼

2) 범주형 데이터 처리

"major" 컬럼 활용 여부 고려

```
major = df['major'].str.lower()
major.str.strip().value_counts()
```

```
psychology                                  3601
business                                    2042
engineering                                 1836
english                                     1577
computer science                            1479
                                            ...
chiropractor                                   1
ancient history and archaeology                1
business & finance                             1
mathematics & economics (2 majors)             1
human development counseling psychology        1
Name: major, Length: 6686, dtype: int64
```

Unique한 데이터가 6,686가지로
범주가 너무 세분화되어
boost 계열 모델 활용 시 오류 발생

3) 모델링 1

1차 전처리만 진행한 데이터셋으로 Auto_ML 실행하여 **모델 성능의 Baseline 확인**

	Model	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC	TT (Sec)
0	Gradient Boosting Classifier	0.6934	0.7781	0.7327	0.6303	0.6775	0.3884	0.3925	16.0459
1	CatBoost Classifier	0.6935	0.7774	0.7201	0.6332	0.6738	0.3870	0.3900	12.5973
2	Light Gradient Boosting Machine	0.6953	0.7750	0.7207	0.6355	0.6752	0.3905	0.3934	0.7697
3	Ada Boost Classifier	0.6880	0.7694	0.7101	0.6285	0.6667	0.3755	0.3781	4.3503
4	Extreme Gradient Boosting	0.6716	0.7541	0.6559	0.6194	0.6370	0.3376	0.3382	28.2983
5	Linear Discriminant Analysis	0.6840	0.7482	0.6233	0.6457	0.6341	0.3562	0.3565	0.9693
6	Extra Trees Classifier	0.6727	0.7458	0.5726	0.6435	0.6059	0.3277	0.3293	1.1899
7	Random Forest Classifier	0.6247	0.6880	0.4404	0.5990	0.5075	0.2157	0.2223	0.1330
8	Decision Tree Classifier	0.6185	0.6130	0.5675	0.5659	0.5666	0.2259	0.2259	1.1074
9	Naive Bayes	0.4454	0.5471	0.9779	0.4409	0.6078	0.0051	0.0187	0.1121
10	K Neighbors Classifier	0.5193	0.5133	0.4266	0.4504	0.4381	0.0187	0.0187	0.4467
11	Quadratic Discriminant Analysis	0.4541	0.5012	0.8912	0.4407	0.5832	0.0024	0.0193	0.3572
12	Logistic Regression	0.5594	0.4741	0.0012	0.1533	0.0023	-0.0019	-0.0174	0.8830
13	SVM - Linear Kernel	0.4920	0.0000	0.4655	0.4291	0.4112	-0.0212	-0.0231	0.2454
14	Ridge Classifier	0.6847	0.0000	0.6222	0.6472	0.6343	0.3575	0.3578	0.1254

TOP3 분류모델 & AUC value

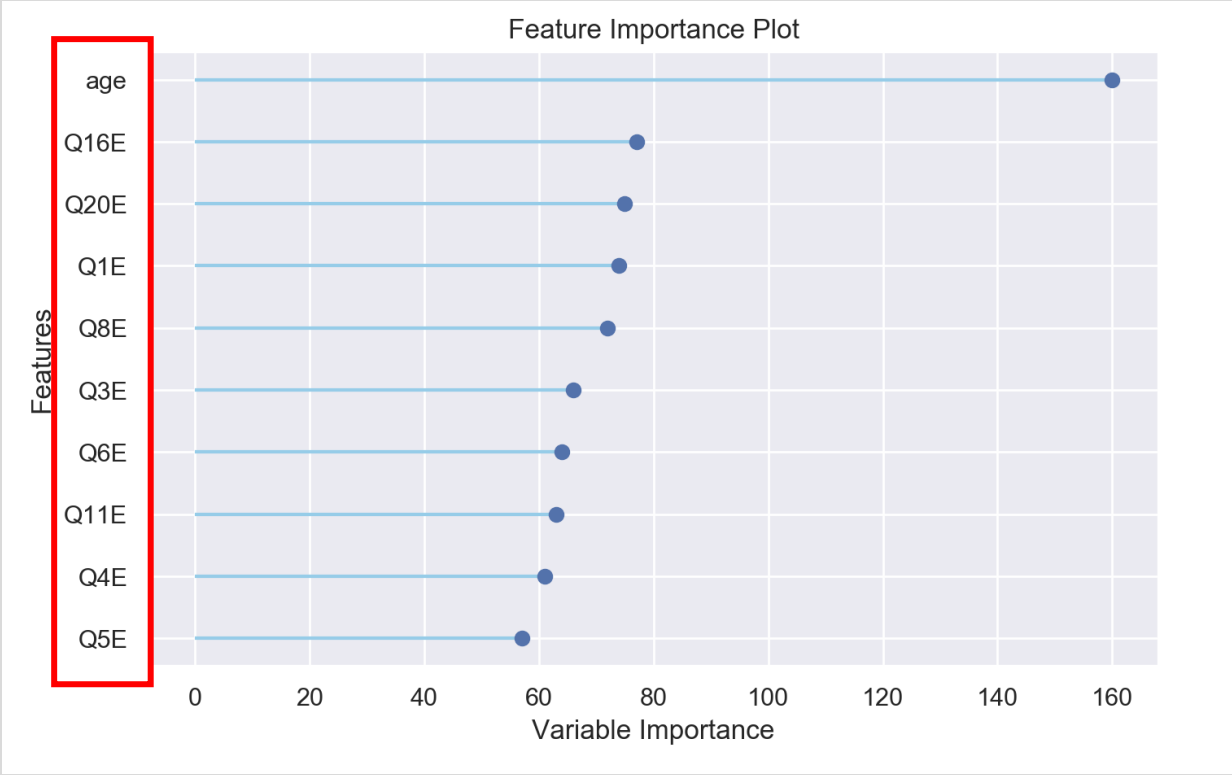
1위 Gradient Boosting Classifier **0.7781**

2위 CatBoost Classifier **0.7774**

3위 LightGBM **0.7750**

3) 모델링 2

1차 전처리만 처리한 데이터셋의 feature importance 확인



Age 컬럼과
QE (테스트 진행 소요시간) 컬럼의
feature importance가 압도적

3) 모델링 3

하이퍼파라미터 튜닝 효과 파악

```
US_gbc = create_model('gbc')
```

	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC
0	0.7321	0.8129	0.7380	0.7491	0.7302	0.4697	0.4830
1	0.7289	0.8080	0.7348	0.7459	0.7270	0.4633	0.4765
2	0.7358	0.80[illegible]	0.74[illegible]	0.74[illegible]	0.7345	0.4763	0.4871
3	0.7389	0.8222	0.7447	0.7551	0.7373	0.4829	0.4956
4	0.7385	0.8033	0.7449	0.7582	0.7362	0.4828	0.4985
5	0.7408	0.80[illegible]	0.74[illegible]	0.7[illegible]83	0.7[illegible]89	0.4869	0.5007
6	0.7315	0.8051	0.7382	0.7521	0.7290	0.4693	0.4857
7	0.7283	0.8004	0.7343	0.7454	0.7264	0.4622	0.4755
8	0.7417	0.8114	0.7479	0.7601	0.7397	0.4888	0.5035
9	0.7385	0.8173	0.7448	0.7578	0.7362	0.4826	0.4981
Mean	0.7355	0.8107	0.7415	0.7532	0.7335	0.4765	0.4904
SD	0.0047	0.0070	0.0047	0.0052	0.0047	0.0092	0.0097

Before

0.8107

-0.0024

```
US_tuned_gbc = tune_model(US_gbc, optimize = 'AUC')
```

	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC
0	0.7225	0.8026	0.7269	0.7333	0.7215	0.4490	0.4569
1	0.7243	0.8048	0.7287	0.7350	0.7234	0.4526	0.4604
2	0.7275	0.8030	0.73[illegible]	0.73[illegible]	0.7270	0.4583	0.4645
3	0.7463	0.8234	0.7514	0.7597	0.7452	0.4969	0.5072
4	0.7260	0.80[illegible]	0.72[illegible]	0.74[illegible]	0.7[illegible]1	0.4572	0.4686
5	0.7362	0.[illegible]	0.[illegible]	0.[illegible]	0.[illegible]34	0.4773	0.4892
6	0.7265	0.8079	0.7328	0.7453	0.7242	0.4590	0.4737
7	0.7256	0.7989	0.7307	0.7389	0.7243	0.4558	0.4658
8	0.7357	0.8084	0.7408	0.7491	0.7345	0.4759	0.4861
9	0.7265	0.8158	0.7320	0.7416	0.7248	0.4581	0.4696
Mean	0.7297	0.8083	0.7348	0.7432	0.7284	0.4640	0.4742
SD	0.0070	0.0072	0.0071	0.0079	0.0070	0.0140	0.0147

After

0.8083

마키아벨리즘 성향
테스트 결과 데이터 기반
국가선거투표참여 여부 예측

모델링 PROCESS #ver.1 중단!!

EDA	1차 전처리	AUC 비교	2차 전처리	모델 고도화	모델 앙상블	데이터 예측
O	O	???	진행 불가			
데이터 파악	0, null, 이상치 처리	모델 성능의 baseline 확인				

Trouble & Solution

Troubles

1. 데이터 편중성
2. 가설을 기각시키는 important features
3. 효과 없는 하이퍼파라미터 튜닝

Trouble & Solution

Solution

1. 데이터 편중성

: 전체 중 46%를 차지하는 미국인 데이터

미국인 데이터만 활용하자!

데이터의 대표성 제고로 분류모델 신뢰도 향상

Trouble & Solution

Solution

2. 가설을 기각시키는 important features

: 마키아벨리즘 성향 결과 데이터보다
분류모델에 더 큰 영향을 미치는 "age", "Q_E" feature들

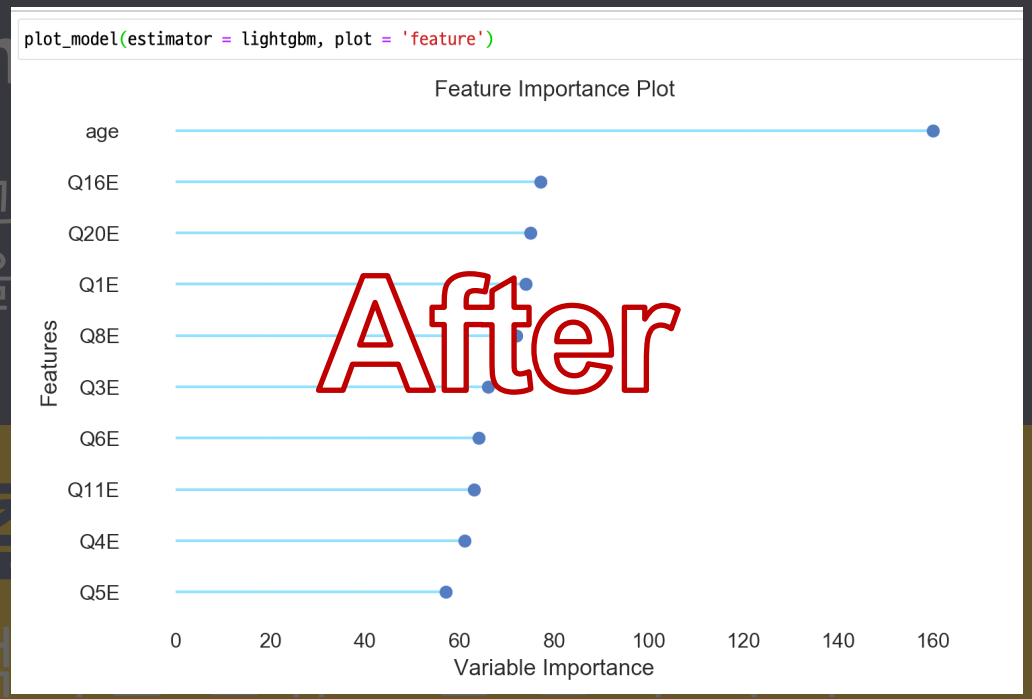
어쩌지?

data scaling 이후에도
동일한 importance 값을 보이는 age, QE 컬럼들

Before

=

After



Trouble & Solution

Solution

2. 가설을 기각시키는 important features

: 마키아벨리즘 성향 결과 데이터보다
분류모델에 더 큰 영향을 미치는 "age", "Q_E" feature들

age, Q_E 컬럼을 제거하자!

가설 검증에 최적화된 분류모델 선택 목적

Trouble & Solution

Solution

3. 효과 없는 하이퍼파라미터 튜닝

: 하이퍼파라미터 튜닝 이후 AUC 값 하락

어쩌지?

게다가, 모델을 생성할 때마다 바뀌는 랜덤값 때문에 하이퍼파라미터 튜닝을 일관성있게 시도할 수 없다

3. 효과

: 하이

Before

```
sifier(ccp_alpha=0.0, criterion='friedman_mse', init=None,
       learning_rate=0.1, loss='deviance', max_depth=3,
       max_features=None, max_leaf_nodes=None,
       min_impurity_decrease=0.0, min_impurity_split=None,
       min_samples_leaf=1, min_samples_split=2,
       min_weight_fraction_leaf=0.0, n_estimators=100,
       n_iter_no_change=None, presort='deprecated',
       random_state=4582, subsample=1.0, tol=0.0001,
       validation_fraction=0.1, verbose=0,
       warm_start=False)
```

After

```
sifier(ccp_alpha=0.0, criterion='friedman_mse', init=None,
       learning_rate=0.1, loss='deviance', max_depth=3,
       max_features=None, max_leaf_nodes=None,
       min_impurity_decrease=0.0, min_impurity_split=None,
       min_samples_leaf=1, min_samples_split=2,
       min_weight_fraction_leaf=0.0, n_estimators=100,
       n_iter_no_change=None, presort='deprecated',
       random_state=920, subsample=1.0, tol=0.0001,
       validation_fraction=0.1, verbose=0,
       warm_start=False)
```

Trouble & Solution

Solution

3. 효과 없는 하이퍼파라미터 튜닝

: 하이퍼파라미터 튜닝 이후 AUC 값 하락

Auto_ML은 Best 모델 선별용으로만 사용

하이퍼파라미터 튜닝은 직접 코딩하여 실행

마키아벨리즘 성향
테스트 결과 데이터 기반
국가선거투표참여 여부 예측

모델링 PROCESS #ver.2

Raw_data, IQR, avg datasets의
모델별 AUC 값 비교

ver.2 추가

0, null, 이상치
처리 완료

EDA	1차 전처리	AUC 비교	2차 전처리	모델 고도화	모델 앙상블	데이터 예측
		모델 성능의 baseline 확인완료	Feature 선별	하이퍼 파라미터 튜닝	분류모델 앙상블모델화	최종모델 활용

familysize 컬럼 이상치 처리 후
IQR*2 기준 & 미국 평균 가족구성원수 기준
dataset 2개 생성

ver.2 추가

1) 추가 전처리

"familysize"의 이상치 제거

IQR*2 기준

```
q1 = np.percentile(samples, 25)
q2 = np.median(samples)
q3 = np.percentile(samples, 75)
upper_fence = q3 + iqr*2.0
lower_fence = q1 - iqr*2.0
print("upper : {}\nlower : {}".format(
```

```
upper : 5.0
lower : 0.0
```

dataset_1

미국인 평균 가족구성원수 기준

3.14명

```
# 가족구성원이 3.14명 이상 데이터 갯수 확인
```

```
len(df_avg[df_avg['familysize'] > 3.14])
```

```
12055
```

```
# 3.14명 이하 데이터 프레임을 ``df_avg``에 포함
```

```
df_withzero_avg = df_withzero_avg[df_withzero_avg["familysize"] <= 3.14 ]
df_avg = df_avg[df_avg['familysize'] <= 3.14]
df_avg
```

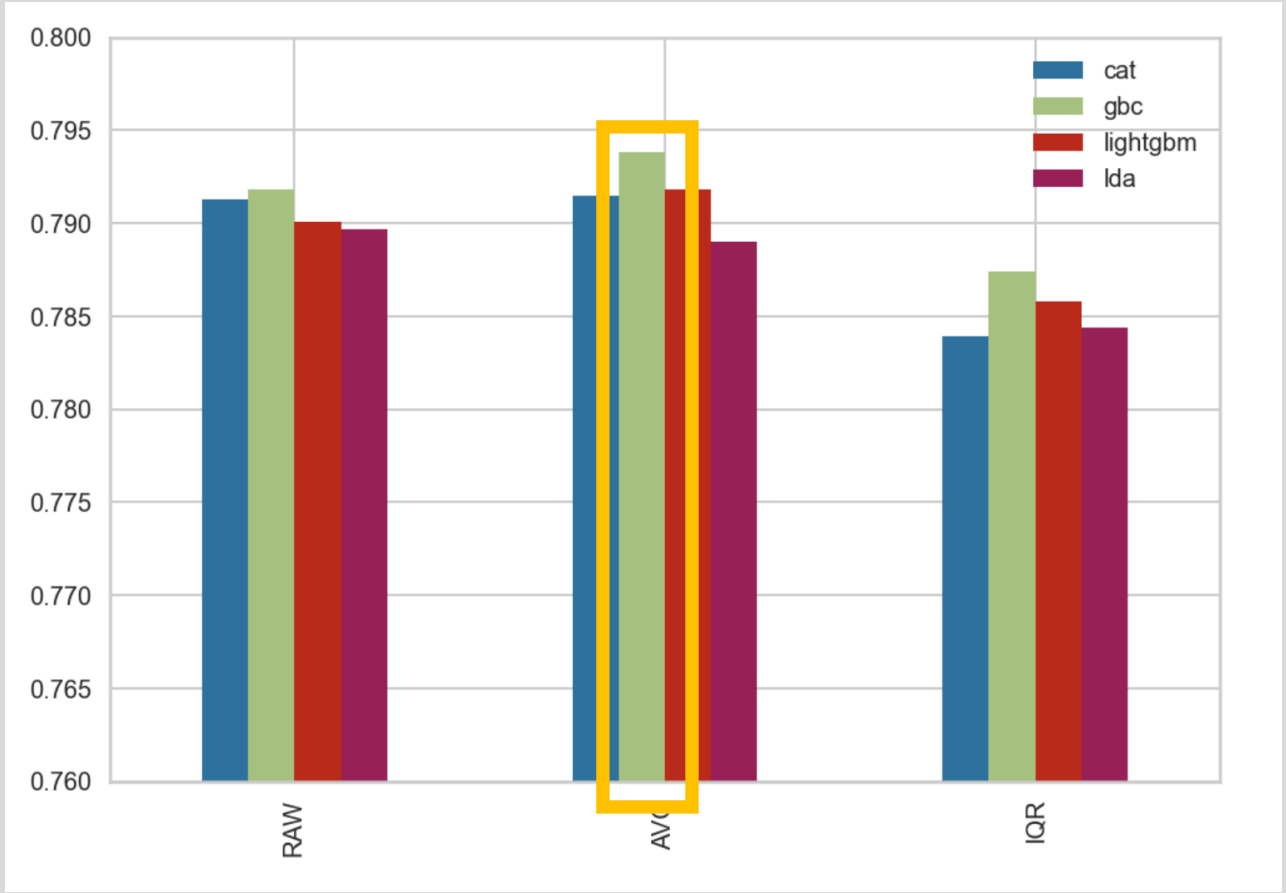
dataset_2

2) 환상의 dataset + model 콤비 찾기

Raw_data, IQR, avg datasets의
모델별 **AUC 값 비교**

	cat	gbc	lightgbm	lda
RAW	0.7913	0.7918	0.7901	0.7897
AVG	0.7915	0.7938	0.7918	0.7890
IQR	0.7839	0.7874	0.7858	0.7844

AVG dataset & gbc 조합 = BEST COMBI



Rawdata AUC vs Combi AUC

2) 환상의 dataset + model 콤비 찾기

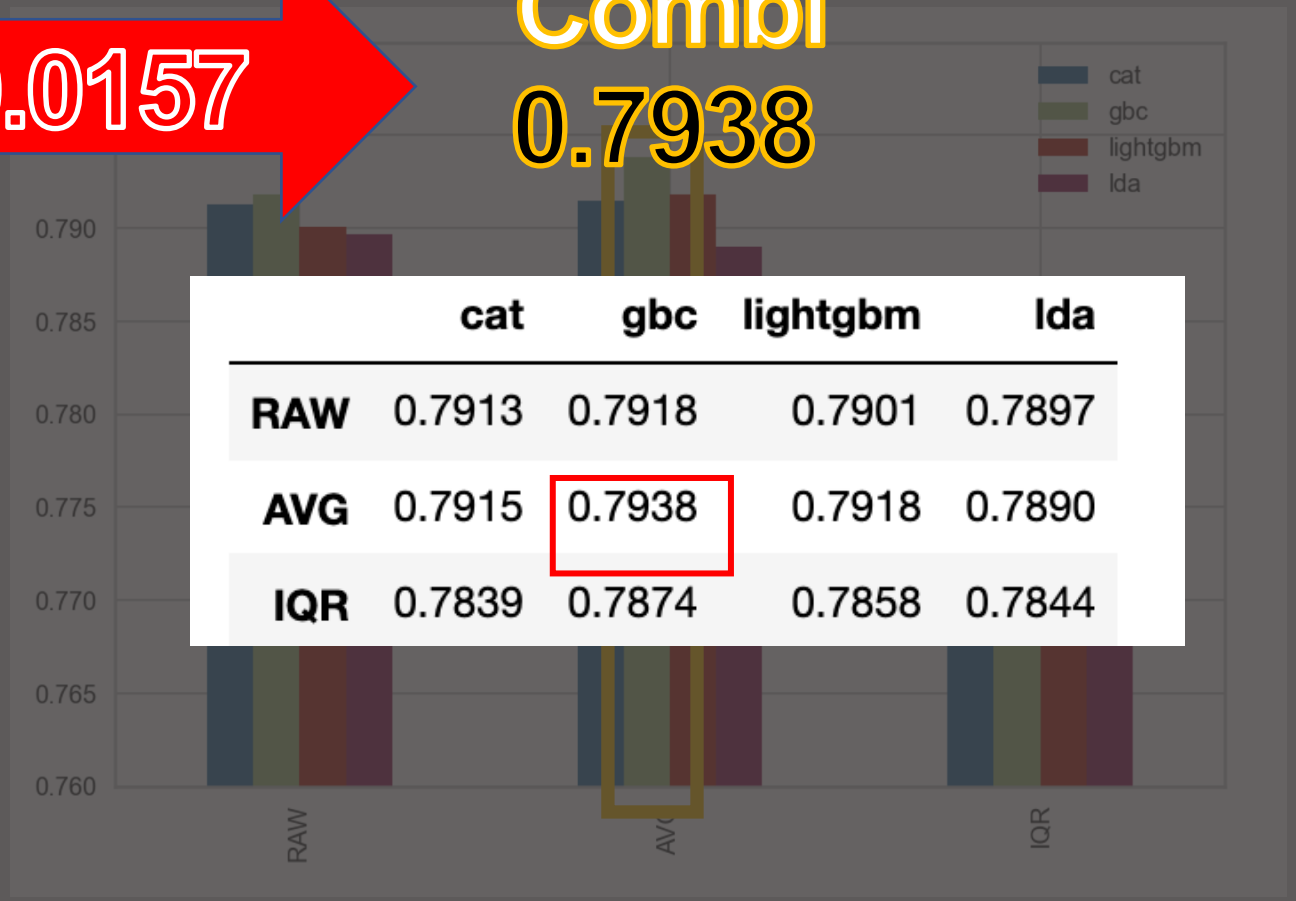
Raw_data, IQR_avg datasets의
모델별 AUC 값 비교

Rawdata
0.7781

+ 0.0157

Combi
0.7938

Model	Accuracy	AUC	Recall
Gradient Boosting Classifier	0.6934	0.7781	0.7327
CatBoost Classifier	0.6935	0.7774	0.7201
Light Gradient Boosting Machine	0.6953	0.7750	0.7207



	cat	gbc	lightgbm	lda
RAW	0.7913	0.7918	0.7901	0.7897
AVG	0.7915	0.7938	0.7918	0.7890
IQR	0.7839	0.7874	0.7858	0.7844

AVG dataset & gbc 조합 = BEST COMBI

to be continued.....

계획 및 과제

1. 계획

① 하이퍼파라미터 튜닝
② 앙상블 모델링 시도 (best model bagging, top3 model voting)

2. 과제

① 모델링 Process의 적합성에 대한 평가
② Solution의 합리성 평가
③ 하이퍼파라미터 튜닝 방법론 연구