2022 여름방학 세미나

방세 2팀
(부제 : 로피탈)
김영호
김예찬
김준서
유종석
윤경선

INDEX

1

데이터 확인 및 EDA

1 데이터 확인 및 EDA

train 데이터 확인

id	target	V1	V2_cat	V3	...	V55_bin	V56_bin	V57_bin
0	0	0	1.0	2	...	0	0	0
1	0	1	2.0	2	...	0	0	0
...	...	...	...	...	...	...	...	...
416646	0	0	1.0	7	...	0	0	0
416647	0	2	1.0	4	...	0	0	1

416648 rows × 58 columns

- Binary 17개, categorical 11개, numeric 27개의 설명변수 확인

- 중복데이터는 없는 것으로 확인

1 데이터 확인 및 EDA

EDA Preview

변수별 시각화

- target
- binary
- categorical
- numeric

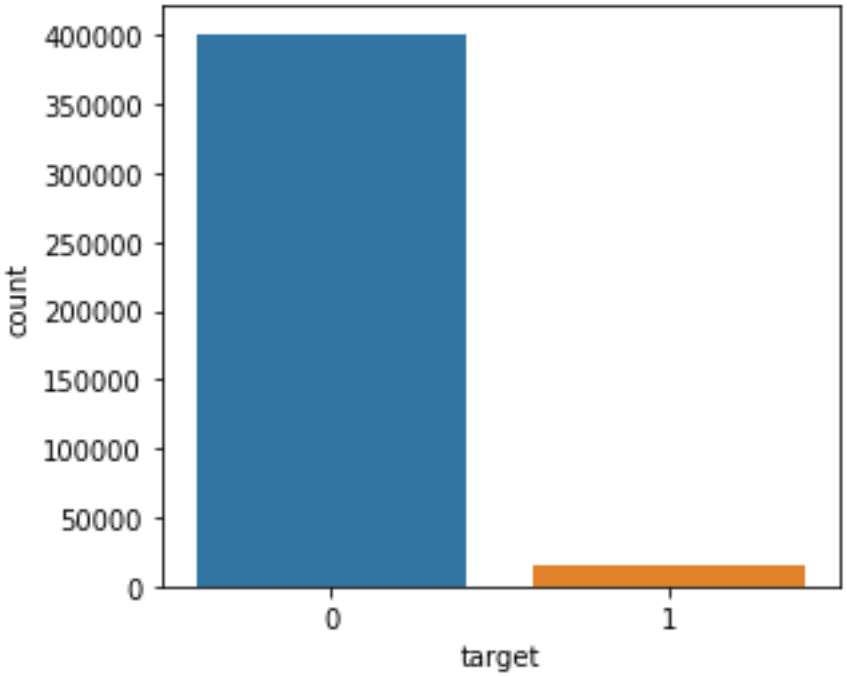
결측치 시각화

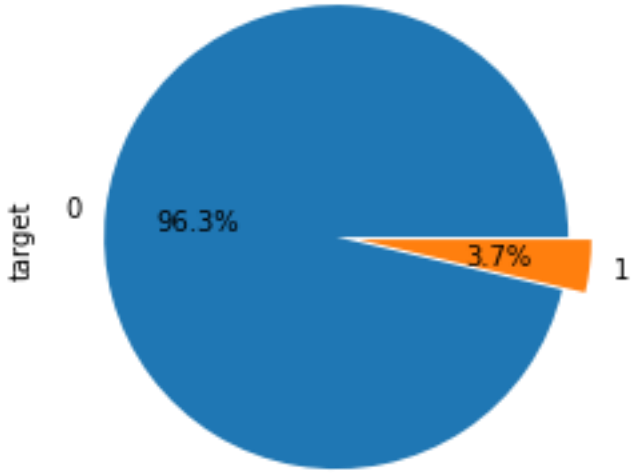
상관관계 시각화

- 범주형 vs 범주형
- 수치형 vs 수치형

1 데이터 확인 및 EDA

변수별 시각화 - target

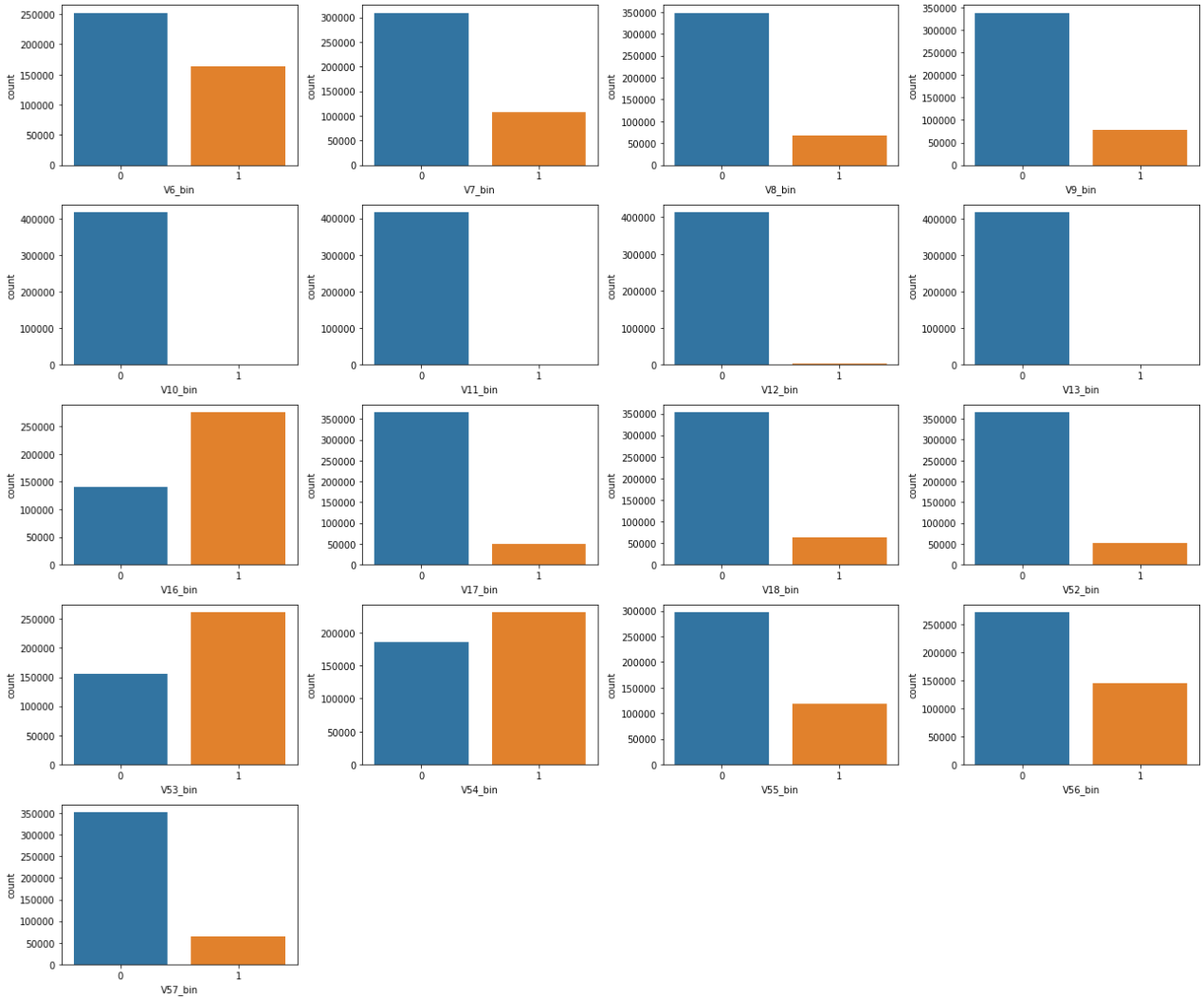




클래스 간 불균형이 매우 심한 것으로 확인

1 데이터 확인 및 EDA

변수별 시각화 – binary



1 데이터 확인 및 EDA

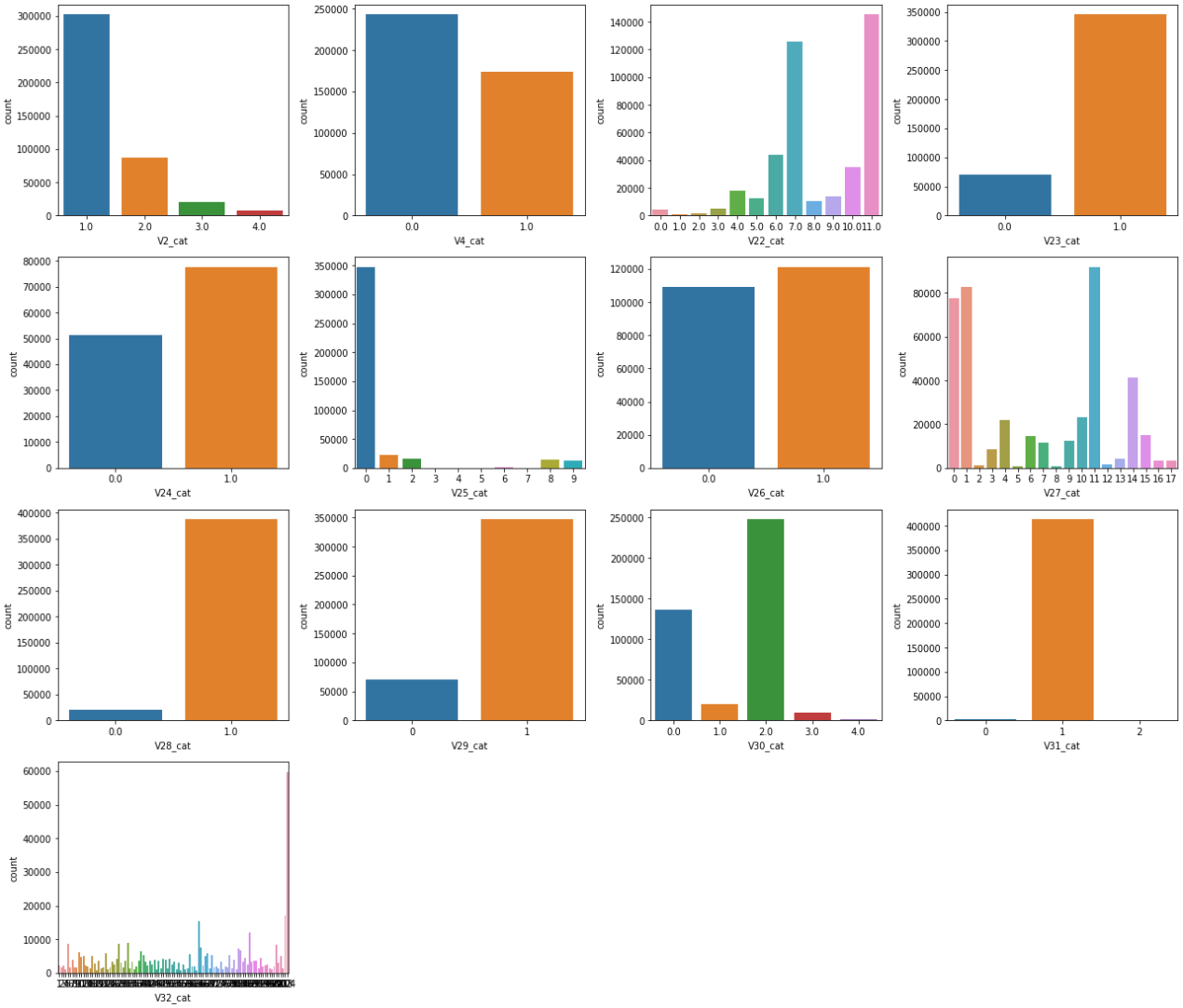
변수별 시각화 – binary

클래스0에 극단적으로 치우친 경우 발견 및

전체적으로 클래스 0에 치우침

1 데이터 확인 및 EDA

변수별 시각화 – categorical

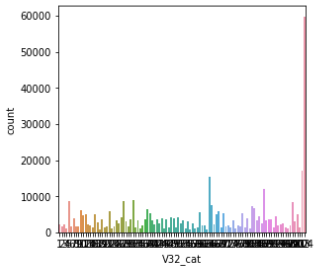


1 데이터 확인 및 EDA

변수별 시각화 – categorical

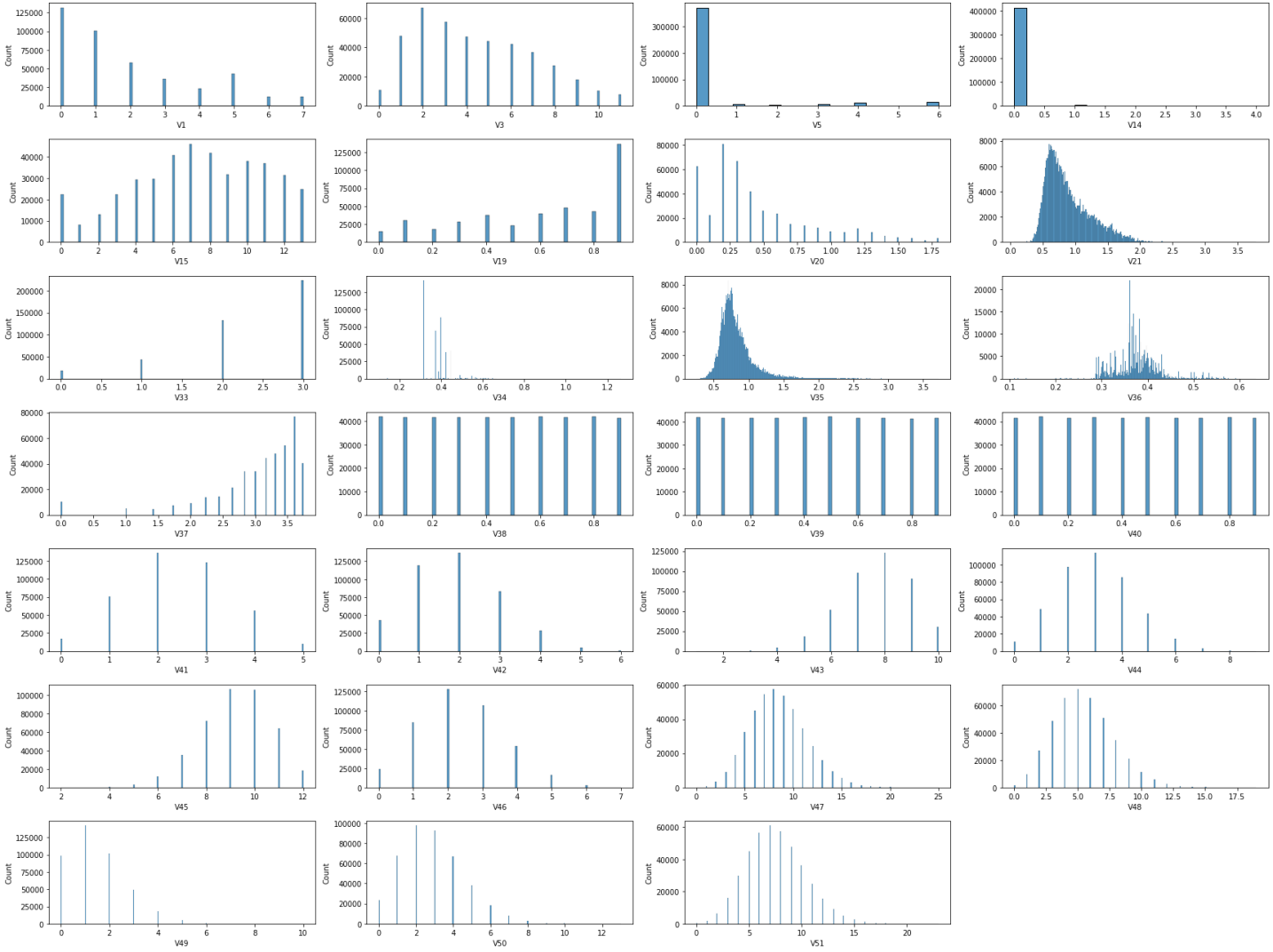
unique값 확인 시 104개의 범주를 갖는 변수 V32_cat 존재

→ 범주의 수를 줄이는 것이 좋지 않을까?



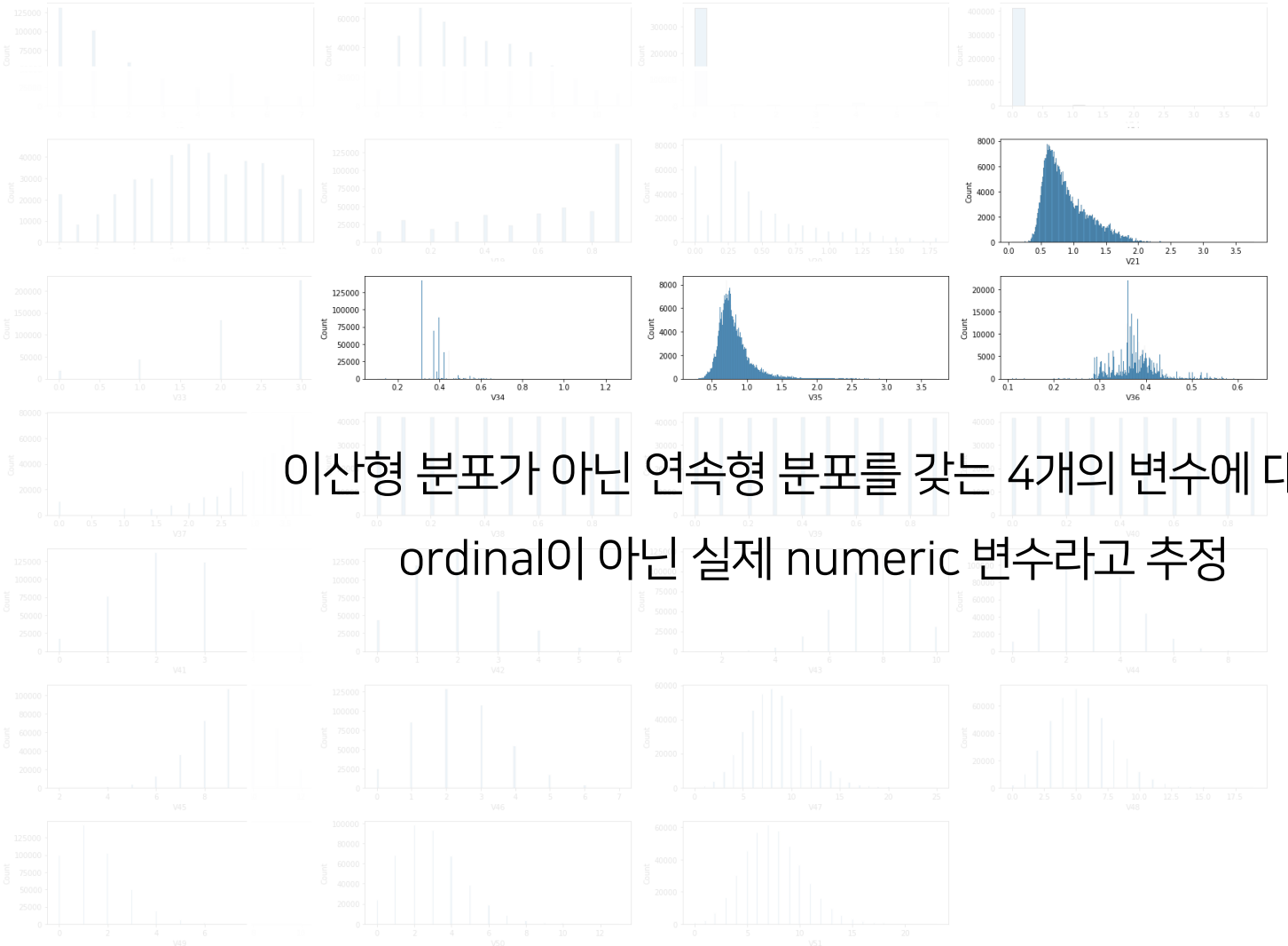
1 데이터 확인 및 EDA

변수별 시각화 – numeric



1 데이터 확인 및 EDA

변수별 시각화 – numeric



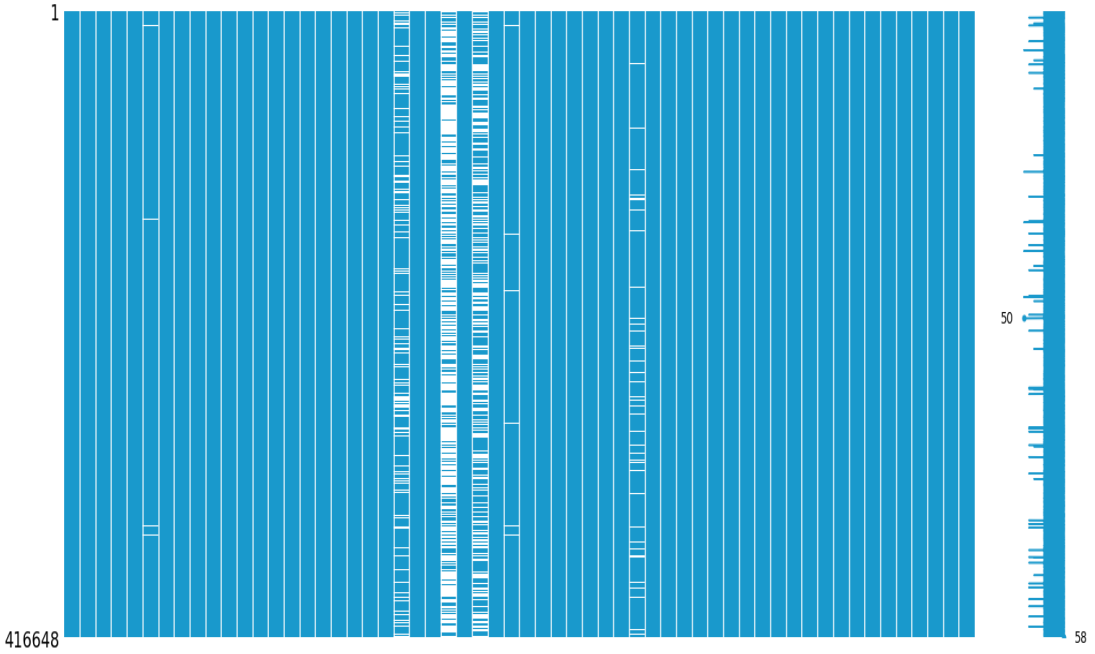
이산형 분포가 아닌 연속형 분포를 갖는 4개의 변수에 대해

ordinal이 아닌 실제 numeric 변수라고 추정

1 데이터 확인 및 EDA

결측치 시각화 및 변수 제거

변수명	변수별 결측치 비율
V24_cat	0.690657
V26_cat	0.447670
V21	0.181112
…	…



결측치 비율이 40%를 넘어가는 두 변수V24_cat과 V26_cat 제거

변수에 대한 정보가 전혀 없는 상황이고, 두 변수를 제거하더라도

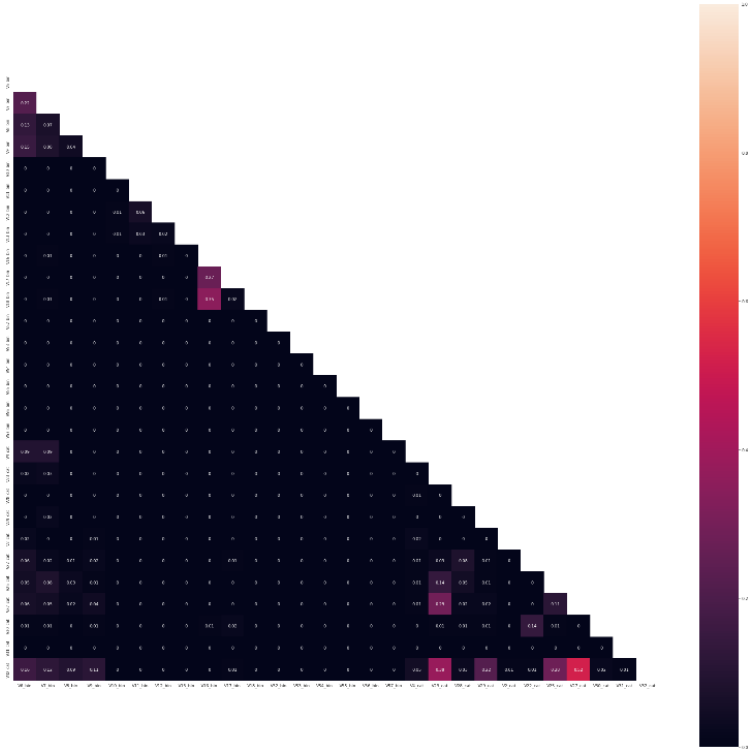
target을 예측하는 데에 이미 충분한 데이터를 갖고 있음

1 데이터 확인 및 EDA

상관관계 시각화

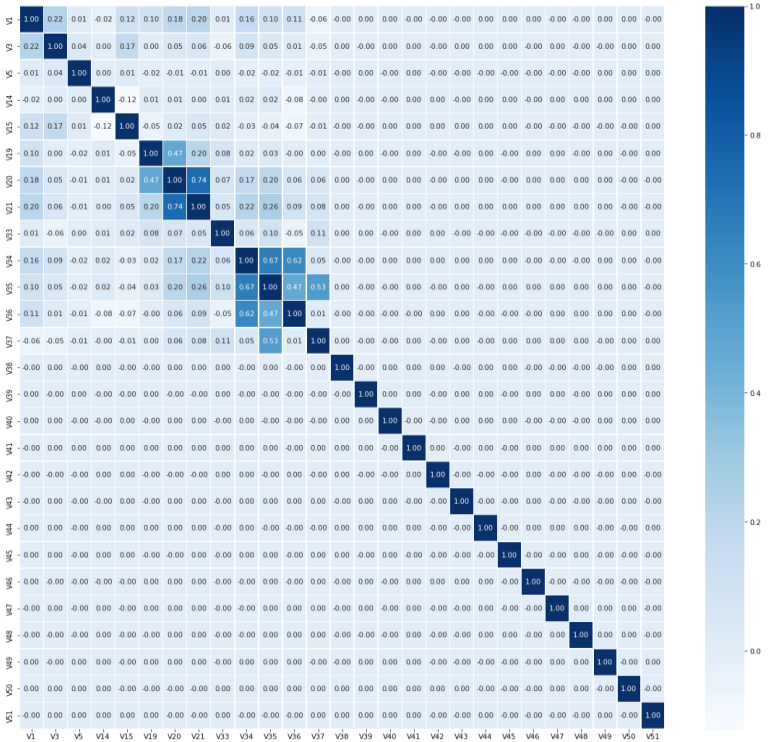
1) 범주형 vs 범주형

: Cramer's V correlation 이용



2) 수치형 vs 수치형

: Pearson correlation 이용



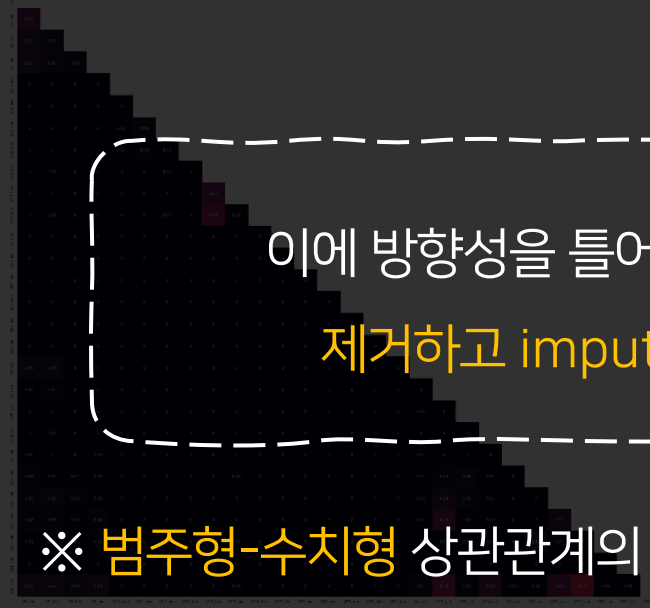
1 데이터 확인 및 EDA

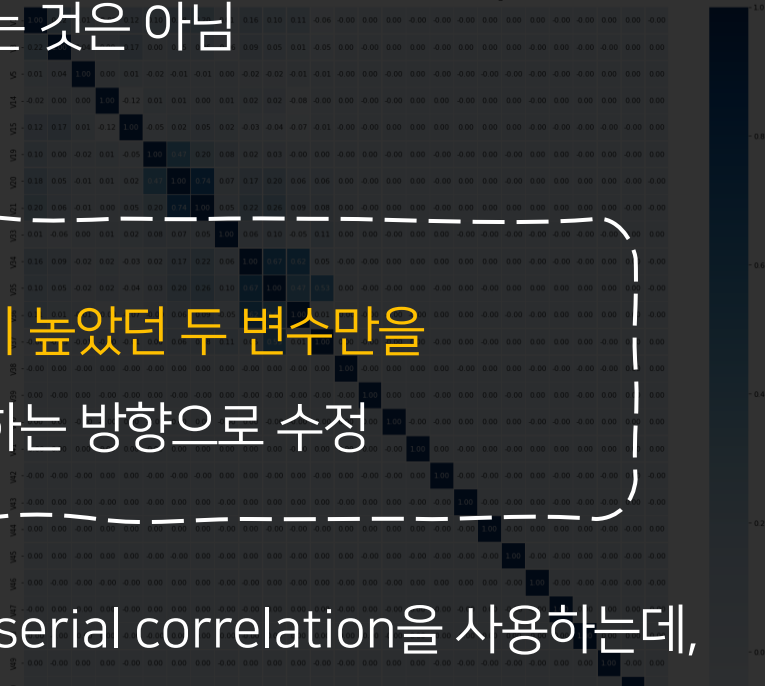
두 설명변수 간의 상관관계가 0.6 혹은 0.7 이상인 변수 제거 및 imputation 진행 후 예측 시 F1 Score가 증가하지 않음

상관관계 시각화

1) 범주형 vs 범주형
: Cramer's V correlation 이용

2) 수치형 vs 수치형
: Pearson correlation 이용





상관관계가 높은 설명변수들을 마냥 제거한다고 해서 모델이 무조건 좋은 예측을 하는 것은 아님

이에 방향성을 틀어, 결측치 비율이 높았던 두 변수만을 제거하고 imputation을 진행하는 방향으로 수정

※ 범주형-수치형 상관관계의 경우 Point-biserial correlation을 사용하는데, 수치형과 범주가 binary인 경우에만 사용하므로 별도의 시각화를 진행하지 않음

2

데이터 전처리

2 데이터 전처리

NA imputation 진행

단일대체법

: 각각의 결측치를 일정한 과정을 통해 생성된 하나의 값으로 대체하는 방법

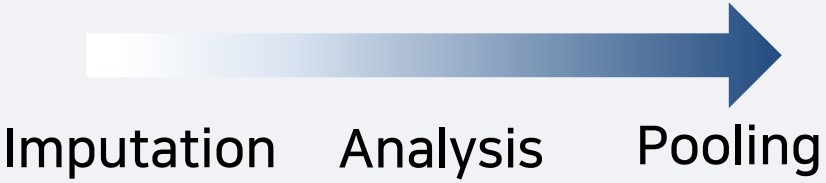
- Mean Imputation
- Median Imputation
- Regression Imputation

다중대체법

: 각각의 결측치를 2개 이상의 서로 다른 값으로 대체하는 방법

KNN

MICE



2 데이터 전처리

NA imputation 진행

단일대체법

: 각각의 결측치를 일정한 과정을 통해 생성된 하나의 값으로 대체하는 방법

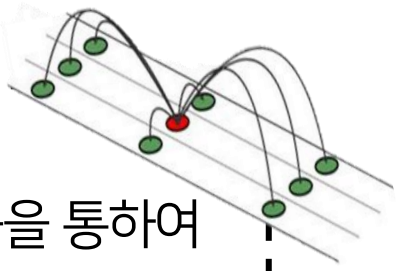
- Mean Imputation
- Median Imputation
- Regression Imputation

간단하고 직관적으로
결측치를 대체할 수 있음
VS
모델의 편향을 높여
모델링의 결과가 안 좋아질 수 있음

2 데이터 전처리

NA imputation 진행

KNN



: K개의 근접한 이웃을 통하여 결측치 대체

수치형 변수와 범주형 변수가 같이 존재하는 경우

↓

범주형 변수

→ one-hot encoding 진행

V32_cat열의 경우, 범주의 수가 104개에 달하기 때문에 차원이 증가!

다중대체법

: 각각의 결측치를 2개 이상의 서로 다른 값으로 대체하는 방법

KNN

MICE

Imputation → Analysis → Pooling

2 데이터 전처리

NA imputation 진행

MICE
(Multivariate Imputation via Chained Equations)

결측치가 여러 변수에 걸쳐 존재할 경우 좋은 성능을 보이는 기법

범주형 변수의 NA값을 처리하기 유리하다고 판단!

Scikit-learn의 IterativeImputer를 사용하여 구현

```
final_train.isnull().values.any() #MICE 후 결측치 유무 확인
final_test.isnull().values.any() #MICE 후 결측치 유무 확인
```

↓

False

MICE를 통해 train, test의 NA값이 모두 대체되었음을 확인

2 전처

NA imputation 진행

MICE

(Multiva on quations)

결측치가 어디 칸수에 걸쳐 존재할 경우 흔한 방법을 보이는 기법

: 범주형 변수의 NA값을 처리할

기 유

final_train.isnull().values.any() #MICE 후 결측치 유무 확인

t_ 후 결측치 유무 확인

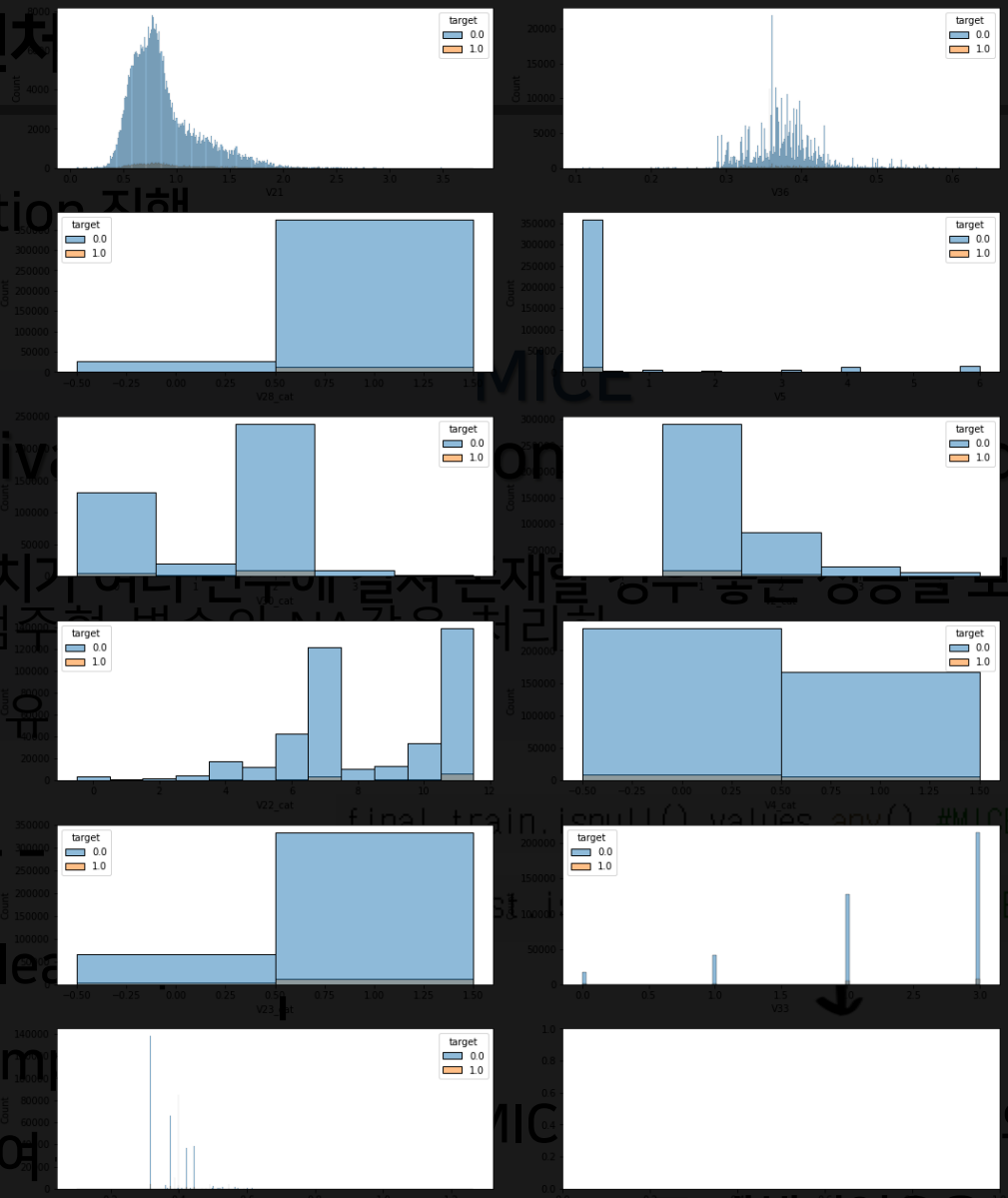
Scikit-lea

IterativeIm

사용하여

IC 의 NA값이

모두 대체되었음을 확인



결측치에 대한 imputation이 잘 이루어졌음을 확인할 수 있음

2 데이터 전처리

numeric Data Scaling 진행

numeric 변수 중
Ordinary일 수 있는 변수 존재

Not Ordinary:
"V21","V34","V35","V36"

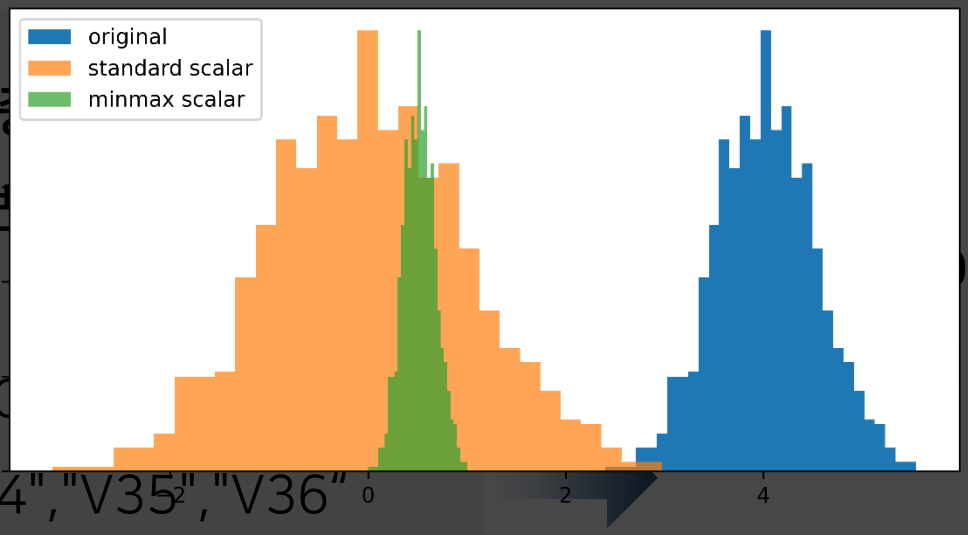
Ordinary:
The other numeric

① Not Ordinary:
Standard Scaling 진행

② Ordinary:
Min-Max Scaling 진행

2 데이터 전처리

Numeric Data Scaling 진행



① Standard Scaling: 각 feature의 평균을 0, 분산을 1로 변경

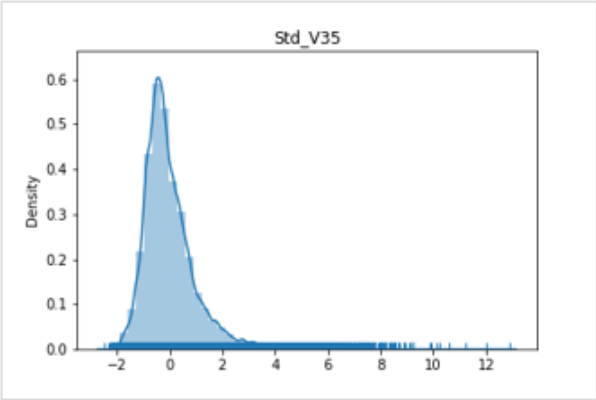
→ 모든 특성이 같은 스케일을 지님

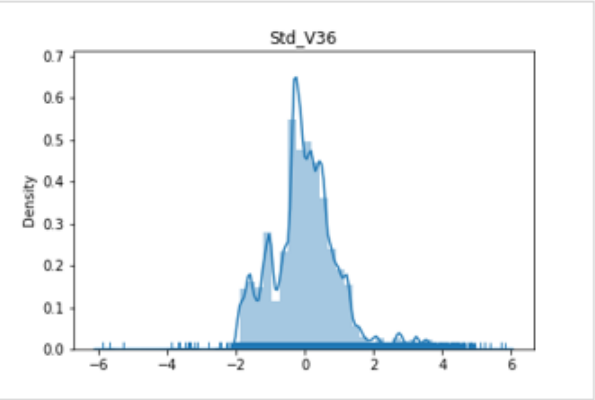
② Min-Max Scaling: 모든 feature가 0과 1 사이에 위치

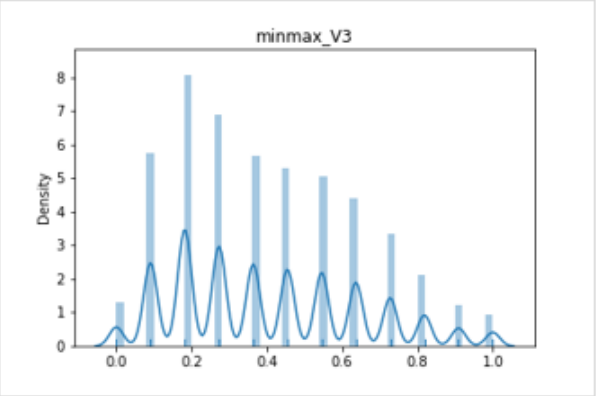
→ Numeric 변수가 Ordinary 형태를 띨 때 자주 사용

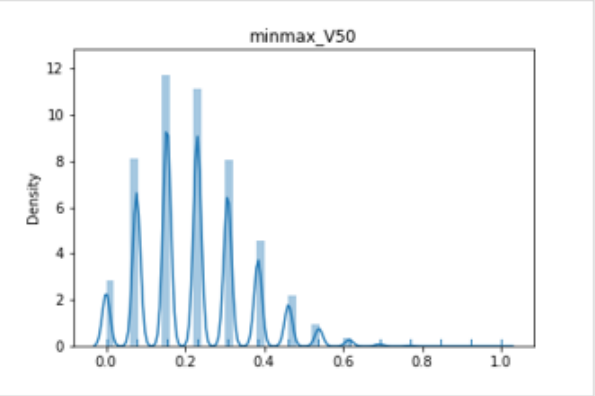
2 데이터 전처리

Numeric Data Scaling 진행









Visualization를 통해서 Scaling은 진행 여부 확인

Categorical Data Encoding 진행

Data Encoding

머신러닝 알고리즘은 문자열 데이터 속성을 입력 받지 않음

: 문자형 카테고리형을 모두 숫자 형으로 표현 → Encoding

상품분류
TV
냉장고
전자레인지
컴퓨터
선풍기
선풍기
믹서
믹서

상품분류_TV	상품분류_냉장고	상품분류_믹서	상품분류_선풍기	상품분류_전자레인지	상품분류_컴퓨터
1	0	0	0	0	0
0	1	0	0	0	0
0	0	0	0	1	0
0	0	0	0	0	1
0	0	0	1	0	0
0	0	0	1	0	0
0	0	1	0	0	0
0	0	1	0	0	0

Categorical Data Encoding 진행

Binary → One-hot Encoding

Not Binary → Binary Encoding

Others → Hash Encoding

: 문자형 카테고리형을 모두 숫자 형으로 표현 → Encoding

너무 많은 차원의 증가 → 예측력 감소

V32_cat 만 Hash Encoding 진행 → Category 축소

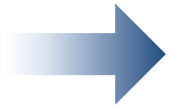
Categorical Data Encoding 진행

Hashing Encoding

Hashing Track을 사용해서 많은 차원의 Dummy Variance를 간편하게 Encoding 하는 기법

: 복잡한 Category 변수를 Encoding 하기에 좋다!

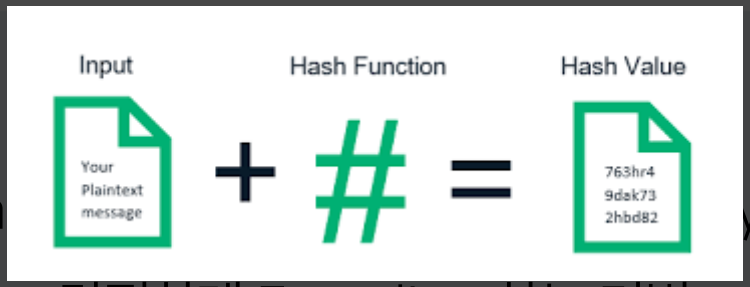
	Category
0	A
1	B
2	C
3	D
4	E
5	F
6	...



	col_0	col_1	col_2	col_3	col_4	col_5	...
0	0	1	0	0	0	1	
1	1	0	0	0	0	0	
2	0	0	0	0	0	0	
3	0	0	0	0	0	0	
4	0	0	0	0	1	0	
5	0	0	0	0	0	0	
...							

2 데이터 전처리

Categorical Data Encoding 진행



Hashing Tra… …y Variance를 간편하게 Encoding 하는 기법

Hash Track이란 어떠한 종류의 값을 입력을 받더라도 '반드시' 어떠한 숫자로 반환하는 방법!!

: 복잡한 Category 변수를 Encoding 하기에 좋다!

즉, Hash를 통해서 반드시 임의의 숫자로 결과를 반환 가능

	Category
0	A
1	B
2	C
3	D
4	E
5	…
6	…

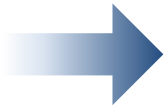
→ Hash Encoding은 Hash의 특징을 통해 아무리 많은 feature의 Category 라도 원하는 개수의 차원으로 표현이 가능

	col_0	col_1	col_2	col_3	col_4	col_5	…
0	0	1	0	0	0	1	
1	0	0	0	0	0	0	
2	0	0	0	0	0	0	
3	0	0	0	0	0	0	
4	0	0	0	0	1	0	
5	0	0	0	0	0	0	
…							

(차원이 많으면 많을수록 정확도는 커지나, 연산량은 증가)

Categorical Data Encoding 진행

Id	V32_cat
1	75
2	28
3	68
⋮	
416645	46
416646	104
416647	64



Id	V32_cat_0	V32_cat_1	V32_cat_2	V32_cat_3	V32_cat_4
1	0	1	0	0	0
2	0	0	1	0	0
3	0	0	0	1	0
⋮	⋮	⋮	⋮	⋮	⋮
416646	0	0	0	0	1
416647	0	0	0	0	1

Hash Encoding을 통해 변수의 개수를 크게 증가시키지 않으면서도
많은 범주를 가지고 있는 열을 수치형으로 처리함

3

모델링

3 모델링

모델링 후보

1. 로지스틱 회귀 모델

① 이진분류를 수행하는데 사용됨

② 로그 오즈를 구한 후 시그모이드 함수 계산

③ 독립 변수의 선형 결합을 이용하여 사건의 발생 가능성을 예측

2. 나이브 베이즈

① 조건부 확률을 계산해 데이터가 클래스에 속할 특징 확률을 계산

② 결측치가 많은 데이터 처리에 특화

③ 예측한 특징이 상호 독립이라 가정하고 계산을 단순화

3 모델링

모델링 후보

3. SVM

① 기계학습 분야로 패턴 인식, 자료 분석을 위한 지도학습 모델

② 분류와 회귀분석을 위해 사용

4. 의사결정나무

① 데이터를 분석하여 이들 사이에 존재하는 패턴을 예측 가능한 규칙들의 조합으로 나타냄

② 전체 자료를 몇 개의 소집단으로 분류하거나 예측

3 모델링

모델링 후보

5. 랜덤포레스트

① 분류, 회귀 분석에 사용되는 앙상블 학습 방법의 일종

② 훈련 과정에서 구성한 다수의 결정 트리로부터 부류(분류) 또는 평균 예측치(회귀 분석)를 출력함으로써 동작

6. LightGBM

① 트리 기반의 부스팅 모델

② Leaf – wise 확장을 하는 특징을 가짐

③ 예측한 특징이 상호 독립이라고 가정하고 계산을 단순화

3 모델링

모델링 결과

1. 로지스틱 회귀

F1_score
0.4907

2. 나이브 베이즈

F1_score
0.5193

3. SVM

Linear : 0.4907
Rbf : 0.4908

4. 의사 결정 트리

F1_score
0.5183

5. 랜덤 포레스트

F1_score
0.4907

6. LightGBM

F1_score
0.5394

4

최종 모델

4 최종 모델

최종 모델링 설명

① MICE Imputation

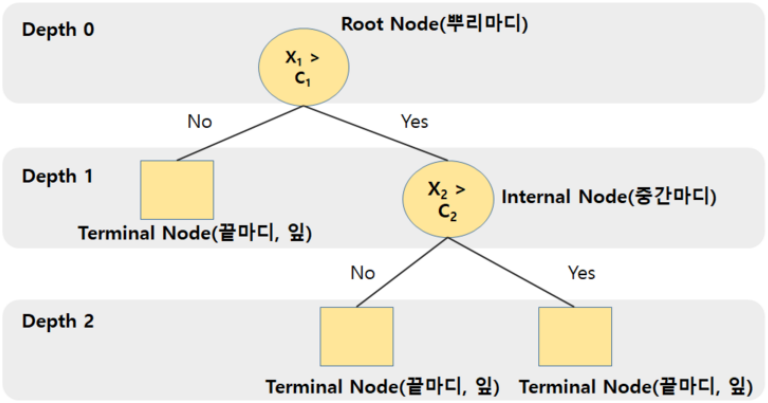
② Hash Encoding

③ LightGBM

리더보드 기준 F1 score : 0.52944

4 최종 모델

트리기반 모델을 사용한 이유



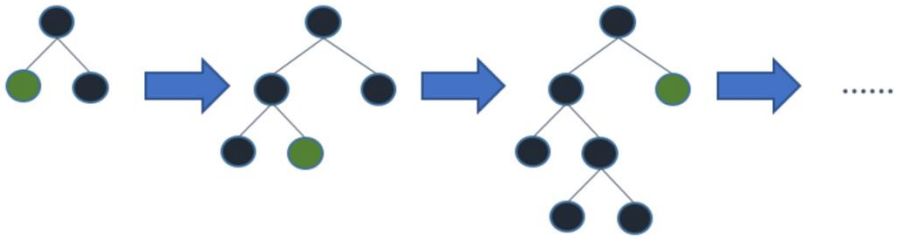
① 트리기반 모델은 전처리 되지 않은 데이터에 강건함

② 변수들이 어떤 의미를 가지는지 파악하지 못한 상태에서 오히려 섣부른 스케일링은 안전하지 않음

③ 숫자의 차이가 모델에 영향을 주지 않기 때문에 트리형 모델을 선택

4 최종 모델

트리기반 모델 중 LightGBM을 사용한 이유



Leaf-wise tree growth

① LightGBM 모델에 'Light'가 붙어있는 만큼 가볍고 속도가 빠름

② 결과의 정확도에 초점을 맞추기 때문에 모델의 성능이 좋음

③ LightGBM은 작은 데이터셋에 사용하면 overfitting 되기 쉽지만
10000개 이상의 충분히 큰 데이터 셋이기 때문에 선택함

4 최종 모델

최종 모델링 설명

```
X_train, X_test, y_train, y_test = train_test_split(
X,y,test_size=0.3,stratify=y,random_state=123)

model = LGBMClassifier(scale_pos_weight=14,
                       boosting='gbdt',
                       learning_rate=0.05,
                       boost_from_average=False,
                       num_iterations=3000,
                       max_depth=10,
                       max_leaves=1023,
                       n_jobs=-1,
                       metric='auc',
                       objective='binary',
                       feature_fraction=0.7)
```

4 최종 모델

최종 모델링 설명

① train_test_split

test_size=0.3
Test size를 0.3으로 설정함으로써 실제 train / test data의 비율을 맞춤
stratify=y
Stratify = y라는 변수를 넣음으로써 Train / Test data의 Target data 비율을 맞춤

최종 모델링 설명

② LGBMClassifier

scale_pos_weight=14
불균형한 데이터에 가중치를 부여하는 것
boosting='gbdt'
경사하강법을 이용하여 가중치 업데이트
boost_from_average=False
불균형한 데이터에 사용했을 때 높은 성능

4 최종 모델

최종 모델링 설명

① MICE Imputation

② Hash Encoding

③ LightGBM

자체 기준
F1_score : 0.5394

리더보드 기준
F1_score : 0.5294

최종적으로 높은 성능 도출함

5

의의 및 한계

5 의의와 한계

이번 방학세미나를 진행하면서!

의의

① 데이터 시각화를 통해 파악한 인사이트를 바탕으로 다양한 전처리 방법을 적용함

② 전처리 과정을 통해 다양한 train set을 만들고, 다양한 트리계열 모델을 적용하면서 최적의 성능을 구현하는 모델을 선택함

한계

범주형 변수 전처리를 위해 encoding,
다양한 type의 변수를 전처리를 위해
scaling을 다양한 방법으로 진행하였지만,
모델의 성능 변화가 크지 않아
이에 강건한 트리기반 모델만을 주로 사용함

5 의의와 한계

소감 한마디

이번 방세 주 많이 힘들었는데 좋은 팀원들 만나서 재밌게 잘 할 수 있었던 것 같았고, 마지막에 리더보드까지 1등 해서 완전 뿌듯한 방세였던 것 같아! 로피탈 화이팅~

일주일이 어떻게 흘러갔는지 모르겠지만 많이 배워갈 수 있었던 시간들이었다~ 로피탈!!

방학 중에 세미나를 하느라 모두들 수고 많았고 정말 모두 열심히 자기 역할 해줘서 좋은 결과 있었던 것 같아. 모르던 부분도 많이 배워갈 수 있었고 좋은 사람 또한 알아가던 기회여서 좋았다. 오로지 피셋 탈출 로피탈 화이팅!

부족한 점 투성이였지만 이번 방세를 통해서 아기새의 첫 비행 연습이랄까요,,,
기존 기수로 한발짝 더 나아갈 수 있는 계기가 되었습니다! 이건 다 멋쟁이 방세2팀 팀원 덕분! 우여곡절이 참 많았지만~ 결과가 잘 나와서 정말 행복하네요!!
일주일간 정말정말 고생많았어!!! 앞으로도~ 가보자고~~~!

5 의의와 한계

소감 한마디2

- 클래스가 극단적으로 불균형한 데이터에 대해 처음 접해볼 수 있는 기회였다.
- imputation과 관련하여 고전적인 방법론에 해당하는 평균, 중앙값, 최빈값을 이용하는 것 뿐만 아니라 MICE를 활용한 imputation에 대해 새롭게 알게 되었다.
- 모델의 특성에 따라 범주형 변수에 대해 때로는 모든 범주형 변수에 대해 인코딩을 진행하지 않아도 된다는 점을 배웠고, 이와 관련하여 다양한 인코딩 방법에 대해 공부할 수 있었다.
- EDA를 통한 데이터와의 소통과 전처리 과정도 중요하지만, 모델링 과정에서의 파라미터 튜닝 과정 또한 예측에 있어 상당한 영향을 미침을 알게 되었다.
- 통데마와 통모머를 수강한 상태에서 방세를 했다면 정말 좋았을 것 같다.
데마는 다음 학기에 수강할 예정이지만,
통모머는 이제 막학기라 들을 수 없다는 사실이 매우 아쉽다.

로피탈의 정리

나 오랜만이쥬~?

6.4.2 정리 함수 $f, g : [a, b] \to R$에서 연속이고, 또한
에서 미분 가능하며, $f(a) = g(a) = 0$이라고 하자. 또, 모든 점
$x \in (a, b)$ 에 대하여 $g(x) \neq 0$ 이고 $g'(x) \neq 0$ 일 때, 다음이 성립한다.

(a) $\lim_{x \to a+} \frac{f'(x)}{g'(x)} = L$ (L은 실수) 이면, $\lim_{x \to a+} \frac{f(x)}{g(x)} = \lim_{x \to a+} \frac{f'(x)}{g'(x)} = L$

(b) $\lim_{x \to a+} \frac{f'(x)}{g'(x)} = \infty \, (or \ -\infty)$ 이면,

$$\lim_{x \to a+} \frac{f(x)}{g(x)} = \lim_{x \to a+} \frac{f'(x)}{g'(x)} = \infty \, (or \ -\infty)$$

나는야 바로바로
오로지 피셋 탈출!

다음 학기도 열심히 달려 나갈 29기 여러분들의 로피탈을 응원합니다^^

THANK YOU

♥ 방세2팀 최고♥