방학세미나

1팀

조웅빈
윤지영
이지윤
박시언
김민서

INDEX

1

INTRO

1 INTRO

분석 흐름

주어진 데이터를 활용하여 성능이 좋은 이진 분류 모델 만들기 위해 …

1 변수 선택

Random Forest, Relief algorithm

2 데이터 불균형 해소를 위한 샘플링

Random Under Sampling, Random Over Sampling, SMOTE

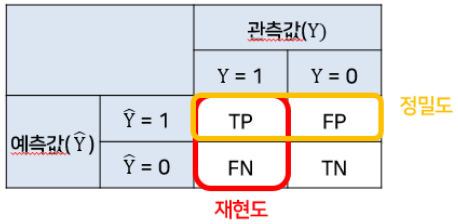
3 모델링

Logistic Regression, LGBM, XGBoost, Deep Learning, Ensemble

F1 Score

정밀도와 재현도의 조화평균

조화평균 : 역수의 산술평균의 역수



$$\text{F1\_Score} = \frac{2}{\frac{1}{\text{Precision}} + \frac{1}{\text{Sensitivity}}} = 2 \times \frac{\text{Precision} \times \text{Sensitivity}}{\text{Precision} + \text{Sensitivity}} = \frac{2\text{TP}}{2\text{FP} + \text{FN} + \text{FP}}$$

왜 F1_Score는 조화 평균을 사용할까?

1. 불균형한 데이터가 주어졌을 때 관측값이 많은 클래스에 패널티를 부여 하기 때문에 관측 값이 많은 클래스에 대한 의존성이 감소해 보다 정확한 성능 파악 가능

2. 정밀도와 민감도를 모두 균형 있게 반영

F1_Score이 **1에 가까울수록 해당 모델의 성능이 우수**하다고 판단!

자세한 내용은 범주팀 3주차 클린업 참고!

2

Preprocessing

2 Preprocessing

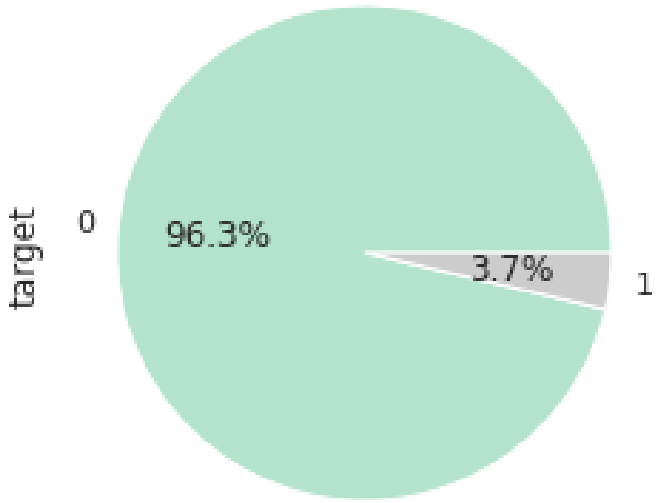
반응 변수 시각화

1. 변수 타입

Target = 0과 1로 구성된 범주형 변수

	Target
0	0
1	0
2	0
3	0
4	0
...	...
416644	0
416645	0
416646	0
416647	0

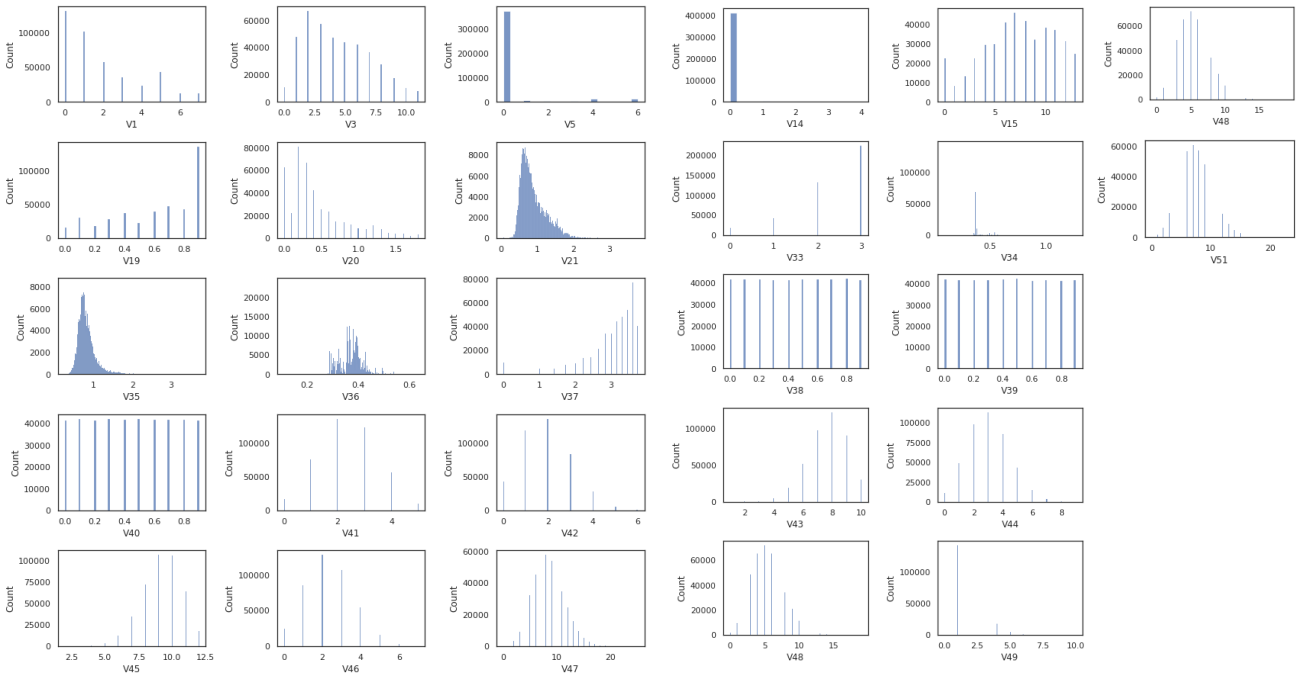
2. 클래스 불균형



2 Preprocessing

설명 변수 시각화

변수 타입: ① 수치형 ② Category ③ Binary



몇몇 변수에서 분포 상 치우침을 확인할 수 있으나,

대부분 정규분포 형태와 흡사하게 분포해있음을 확인

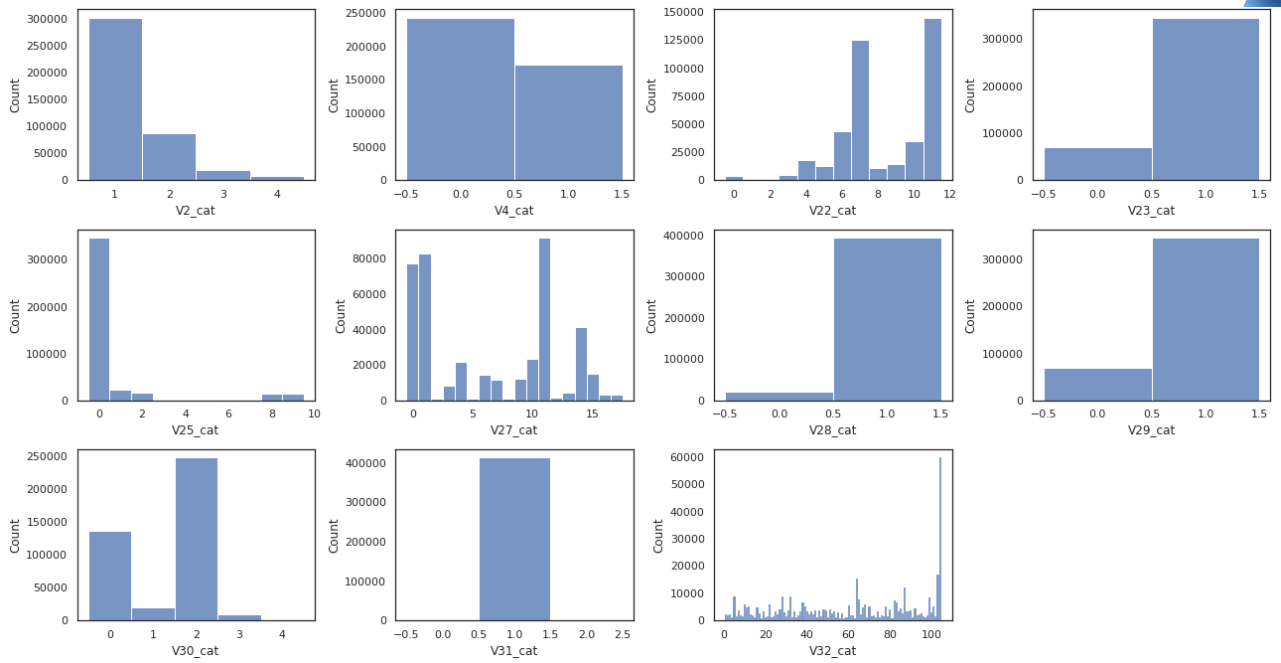
2 Preprocessing

설명 변수 시각화

변수 타입: ① 수치형 ② Category ③ Binary

각 변수 내에서 **클래스 불균형**이 존재한다는 것을 확인

강건한 모델을 사용해서 해결

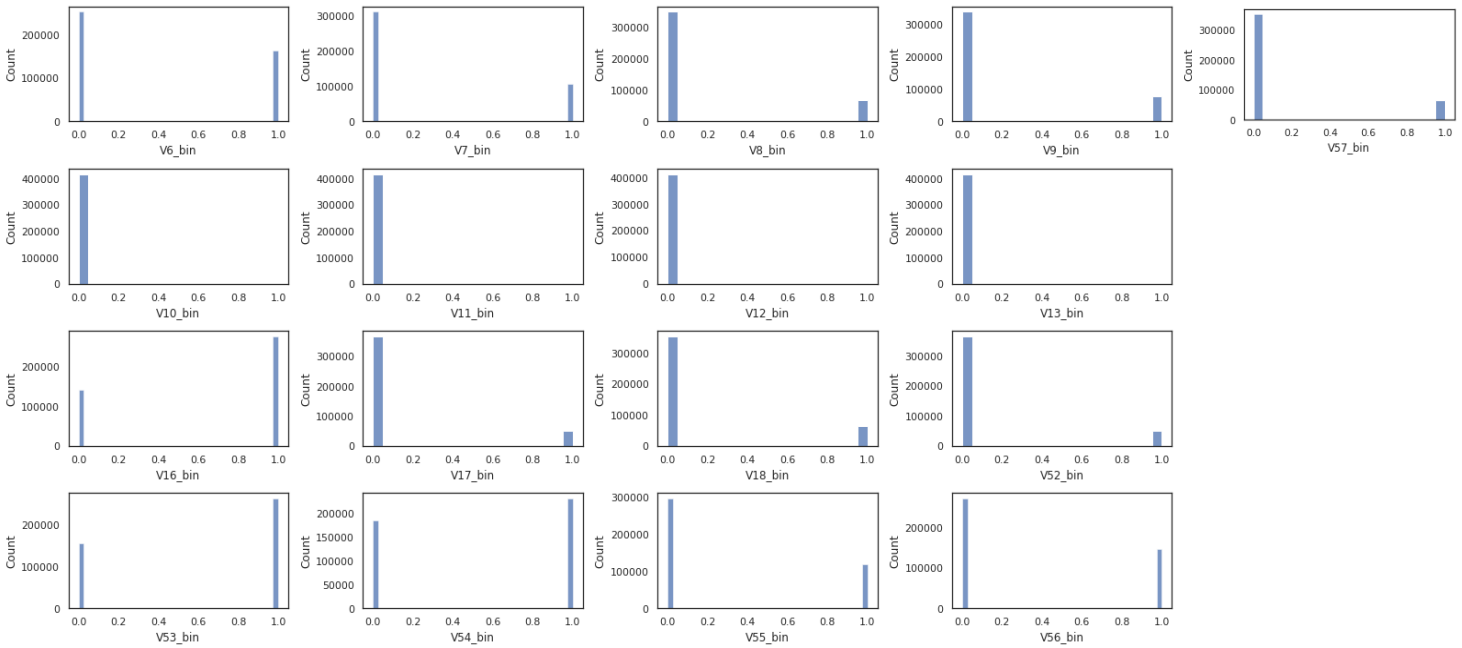


2 Preprocessing

설명 변수 시각화

변수 타입: ① 수치형 ② Category ③ Binary

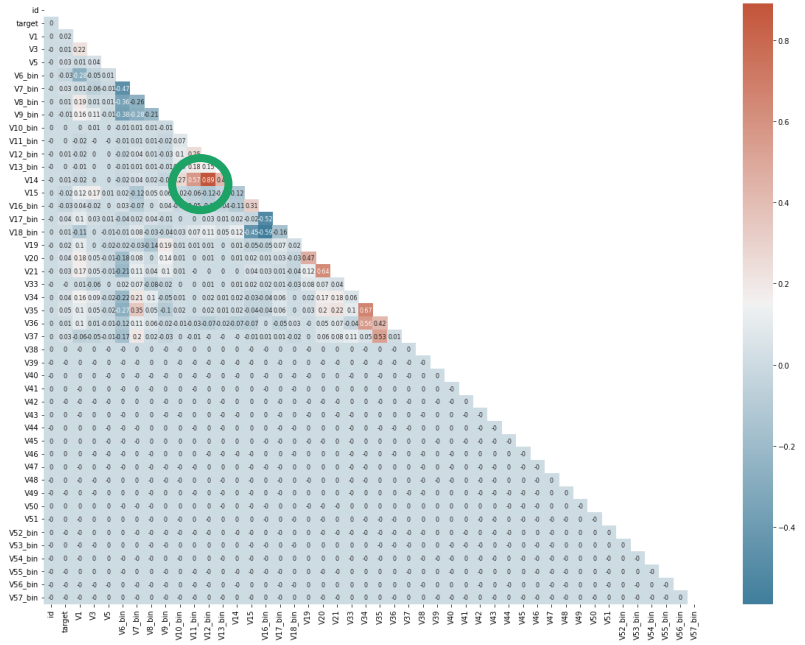
대부분의 변수가 0값이 많음을 확인



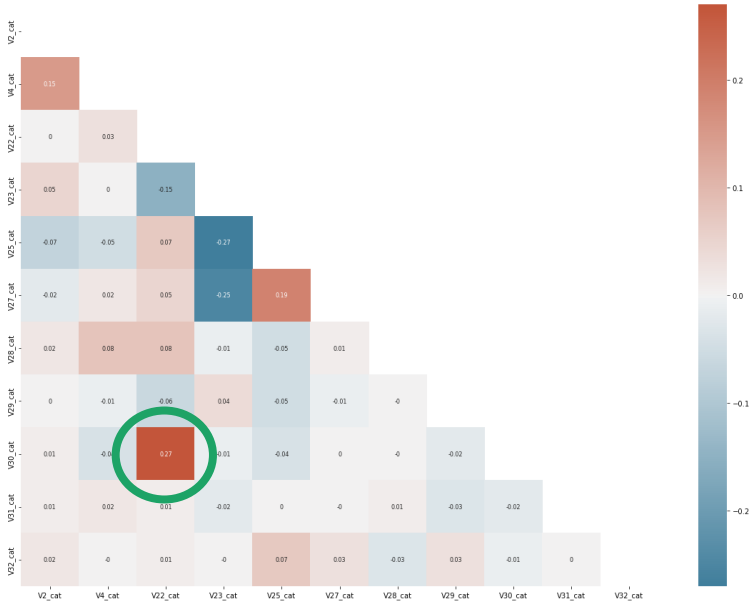
2 Preprocessing

변수간 상관계수

소수의 변수들을 제외하고는 대부분 상관계수가 높지 않음!

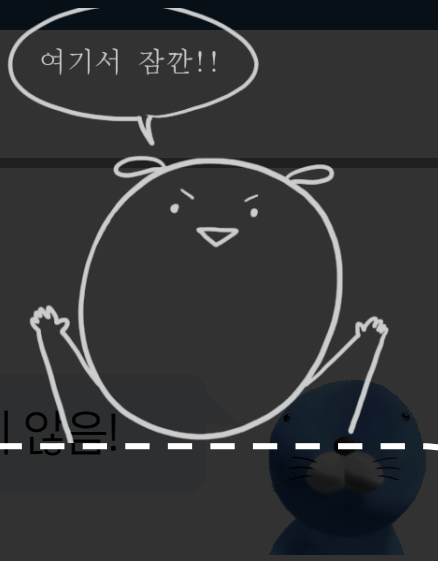


수치형 변수



Cat 변수

2 Preprocessing



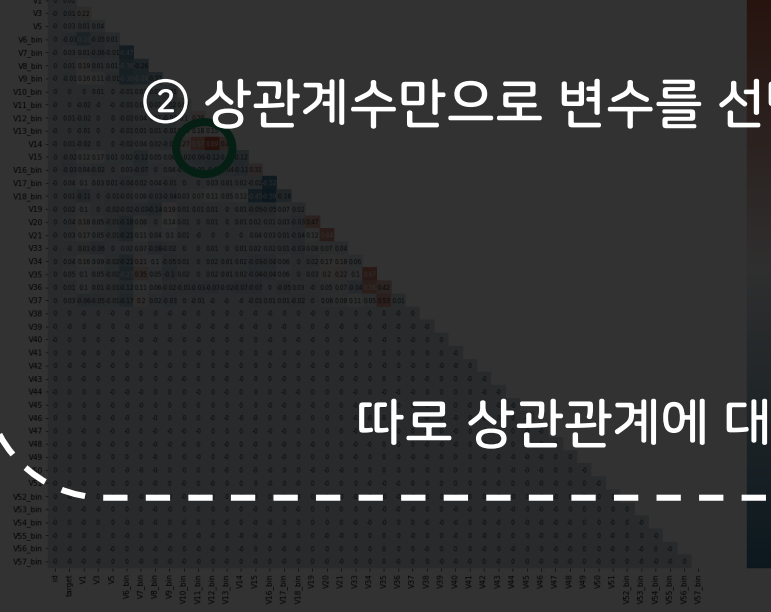
변수간 상관계수

소수의 변수들을 제외하고는 대부분 상관계수가 높지 않음

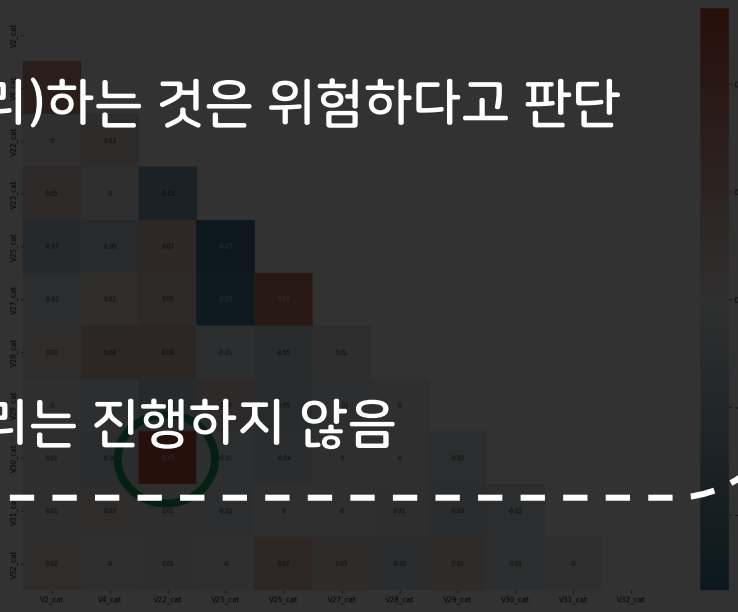
① 이후 변수선택법을 통해, 상관관계는 자동 완화될 것이라 예상

② 상관계수만으로 변수를 선택(처리)하는 것은 위험하다고 판단

따로 상관관계에 대한 처리는 진행하지 않음



수치형 변수

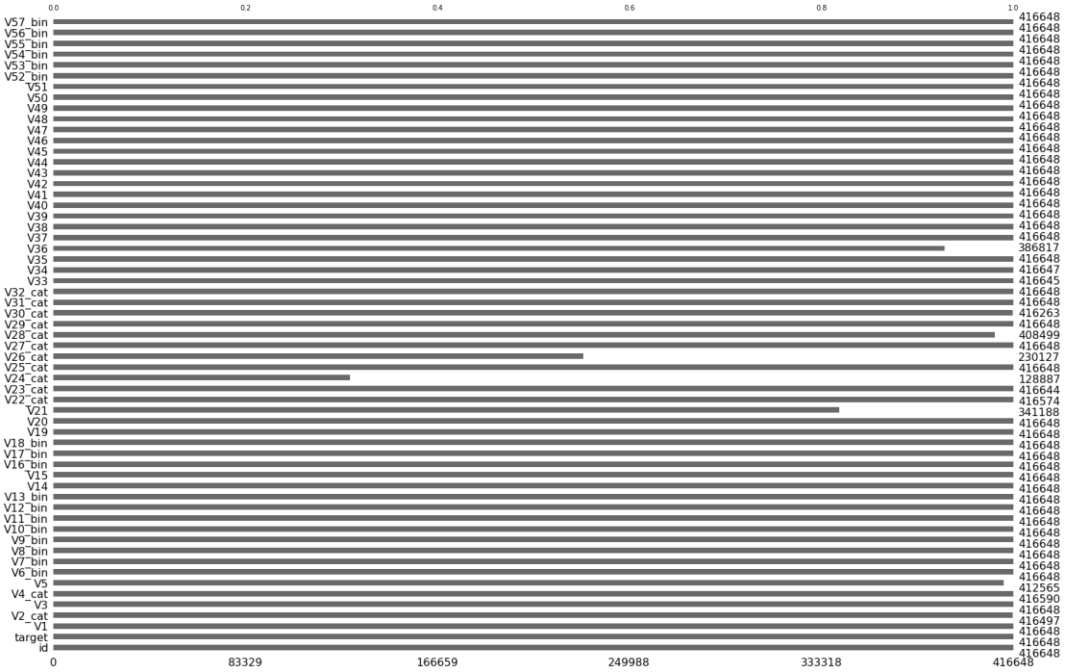


Cat 변수

2 Preprocessing

결측치

변수명	
V24_cat	287761
V26_cat	186521
V21	75460
V36	29831
V28_cat	8149
V5	4083
V30_cat	385
V2_cat	151
V22_cat	74
V23_cat	4
V33	3
V34	1



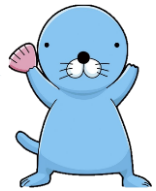
특정 column에 NA가 많다는 것을 확인!

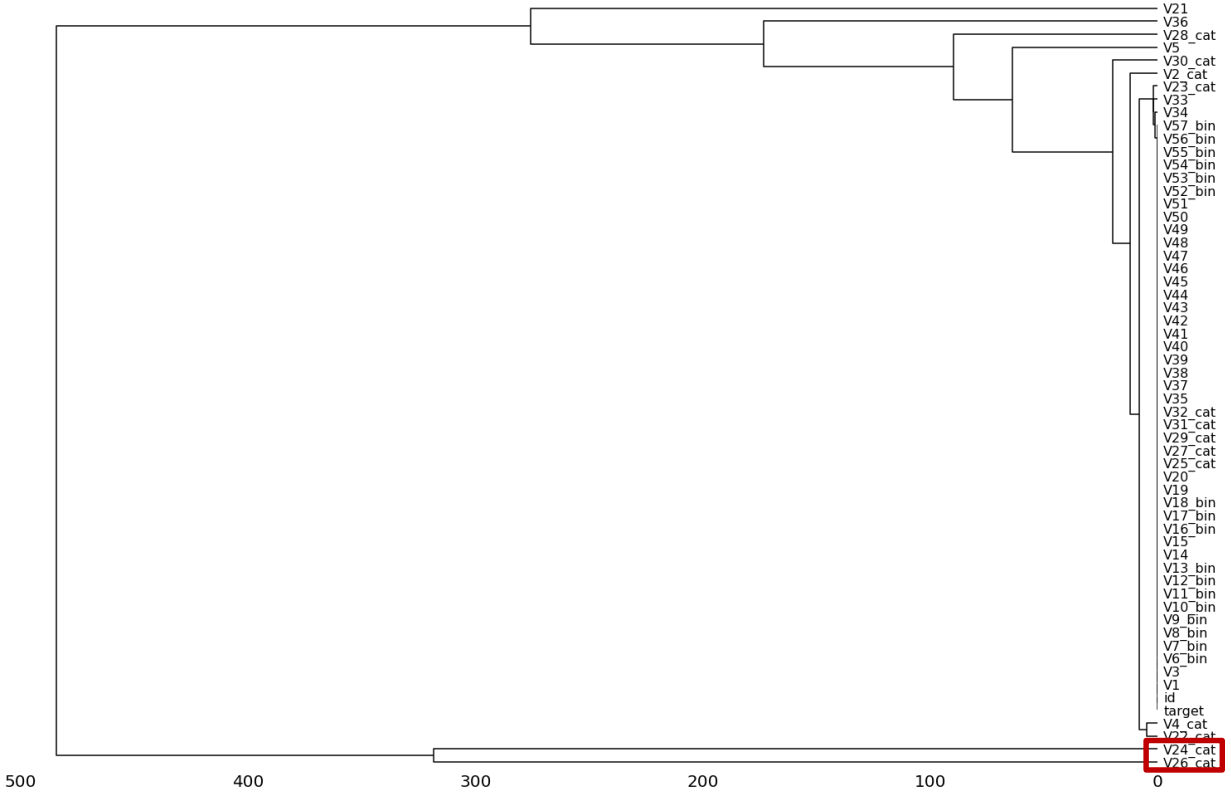
결측치 간의 **관계를 파악**을 위해 시각화 진행

2 Preprocessing

NA Missing에 대한 dendrogram

V24_car과 V26_cat이 **함께** NA 발생하는 경향성 발견





2 Preprocessing

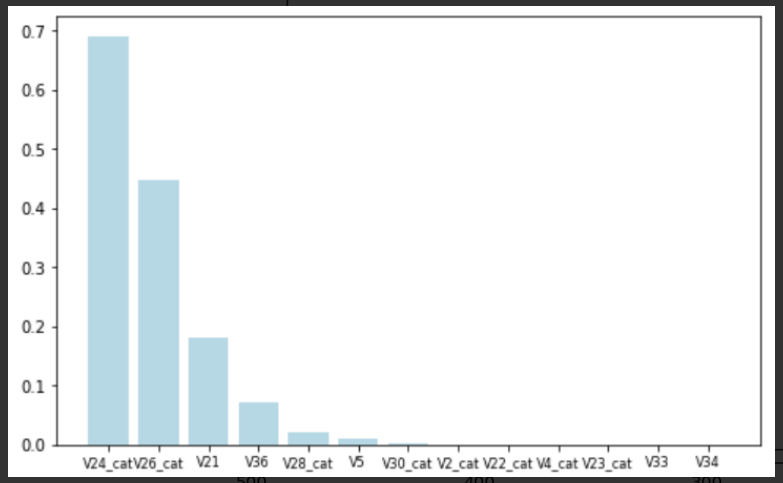
NA Missing에 대한 dendrogram

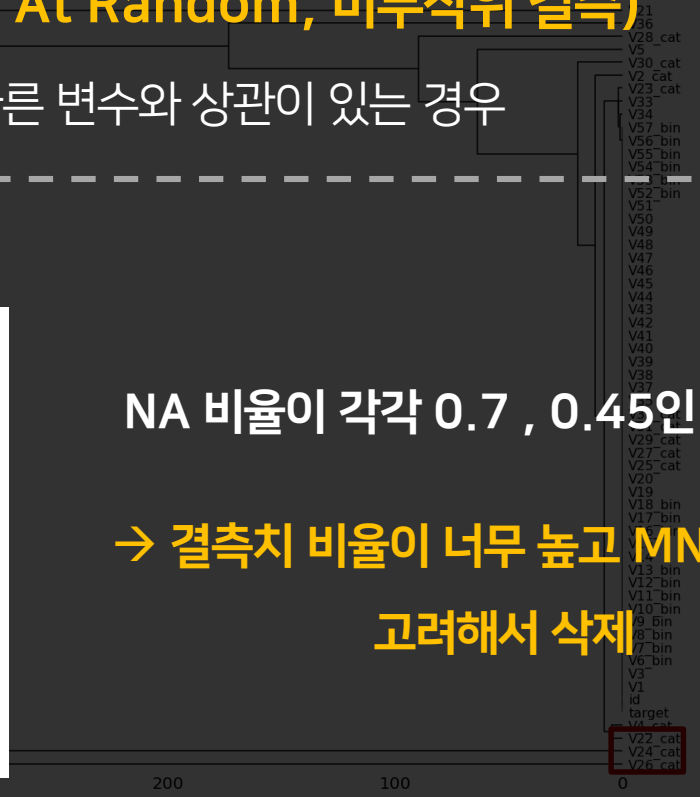
발생한 NA 종류

V24_car과 V26_cat이 동시 NA 발생 연관 경향성 발견

MNAR (Missing Not At Random, 비무작위 결측)

결측 값 발생이 다른 변수와 상관이 있는 경우





NA 비율이 각각 0.7 , 0.45인 것을 확인

→ 결측치 비율이 너무 높고 MNAR인 것을 고려해서 삭제

2 Preprocessing

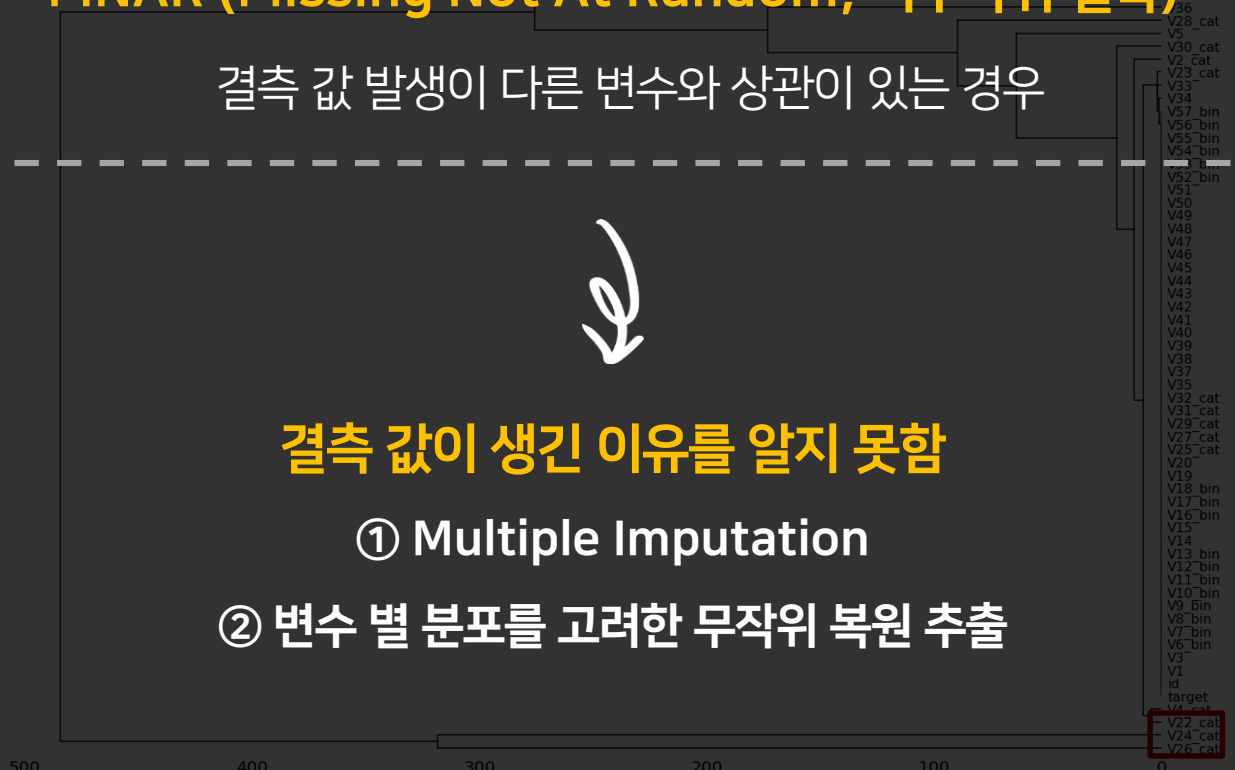
NA Missing에 대한 dendrogram

발생한 NA 종류

V24_car과 V26_cat이 동시 NA 발생 연관 경향성 발견

MNAR (Missing Not At Random, 비무작위 결측)

결측 값 발생이 다른 변수와 상관이 있는 경우



결측 값이 생긴 이유를 알지 못함

① Multiple Imputation

② 변수 별 분포를 고려한 무작위 복원 추출

2 Preprocessing

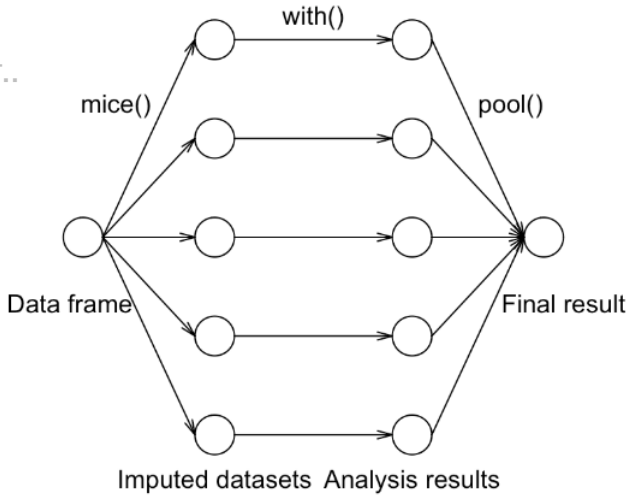
NA Imputation

MICE

누락된 값을 데이터 프레임에 있는 다른 모든 변수를 사용하여

값을 예측하여 데이터셋을 여러 개 만들고 하나로 결과로 통합하는 다중 대치법

코드 열심히 짜준 웅빈 오빠.. 고생했슴돠..



그러나 test 데이터를 사용하여 data leakage 발생

후의 모델링 과정에서 사용하지 못함

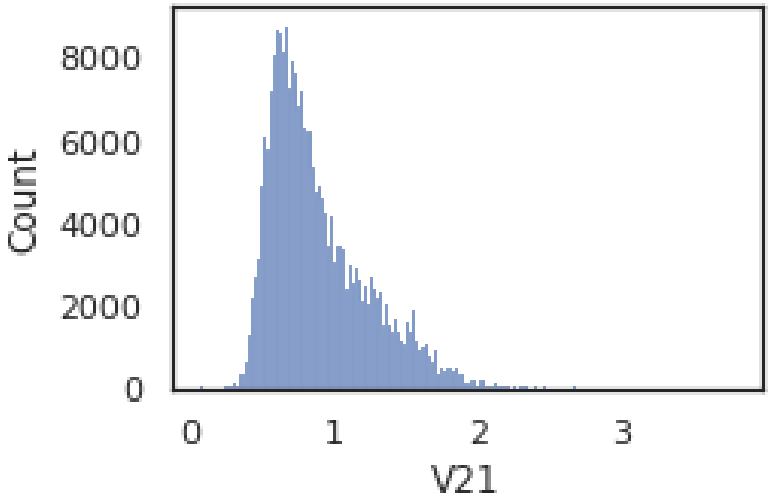
2 Preprocessing

NA Imputation

변수별 분포를 고려한 무작위 복원 추출

진행 방식

1. 변수 별 NA 제외한 값 중 NA 개수만큼
   무작위 복원 추출
2. Sample을 통해 NA값 Imputation



Data leakage를 방지하기 위해
test set에 대해서도 train set의 값 이용

2 Preprocessing

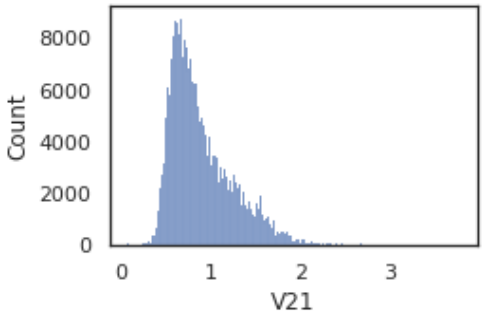
NA Imputation

EX) 변수별 분포를 고려한 무작위 복원 추출

V21 중 NA 제외 value = [1, 1, 2, 2, 2, 2, 3, 3, 4, 5]

V21 NA 개수가 2개라고 가정할 때, 위 값들 중 2개를 **무작위 복원 추출**

	1	2	3	4	5
개수	2	4	2	1	1
확률	0.2	0.4	0.2	0.1	0.1



즉, 해당 변수의 밀도 분포를 고려하여 값을 뽑아

NA Imputation이 가능!

2 Preprocessing

변수 선택의 필요성

416645 x 59

id	target	V1	V2_cat	···	V54_bin	V55_bin	V56_bin	V57_bin
0	1	0	1	···	0	0	0	0
···	···	···	···	···	···	···	···	···
416647	416648	0	1	···	0	0	1	1

변수가 너무 많으면 계산량이 늘어나 학습 시간이 길어져 비효율적임

변수 선택 진행!

2 Preprocessing

Random Forest

Random Forest

다수의 결정 트리들을 학습하는 앙상블 방법으로

다수결의 원칙 / 평균 예측치 등을 통해 최종 결과 도출

대부분 수치형 변수들의 중요도가
상대적으로 높은 것을 확인!

V_30 변수 기점으로 중요도 급감

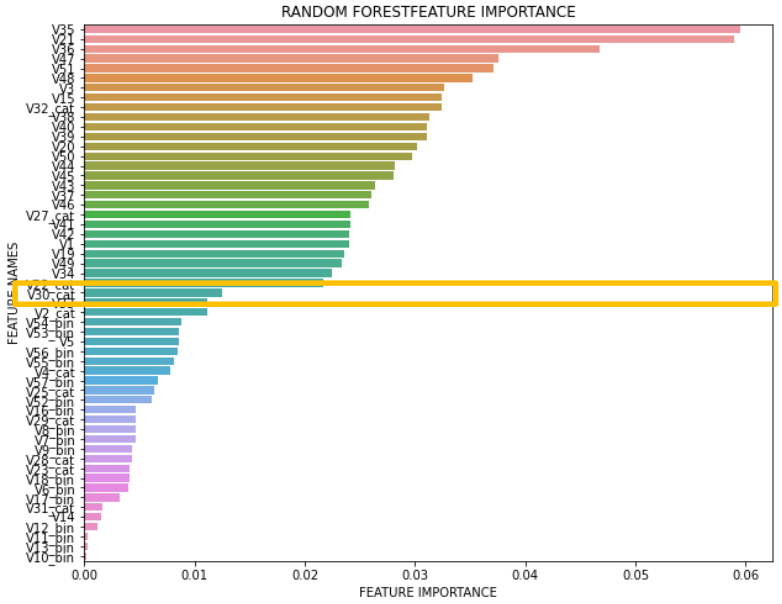
**V30_cat보다 중요도가 높은 변수를
선택하여 진행**

2 Preprocessing

Random Forest

Random Forest

다수의 결정 트리들을 학습하는 앙상블 방법으로
다수결의 원칙 / 평균 예측치 등을 통해 최종 결과 도출



대부분 **수치형 변수**들의 중요도가
상대적으로 높은 것을 확인!

V_30 변수 기점으로 중요도 급감

V30_cat보다 중요도가 높은 변수를 선택하여 진행

Relief Algorithm

Relief Algorithm

이진 분류 문제에서 **변수별 중요도**를 판단할 수 있는 알고리즘으로 독립 변수 별 Y = 0과 1의 분리도(separability)를 산출

(높은 분리도를 가진 X 변수 → Y 변수의 분리에 큰 영향을 미침)

랜포 중요도 결과와 달리 수치, 범주, 이진 변수들의 중요도가 **다양하게** 나타나는 것을 확인

Separabillity가 0 이상인 값들을 기준으로 변수 선택을 진행

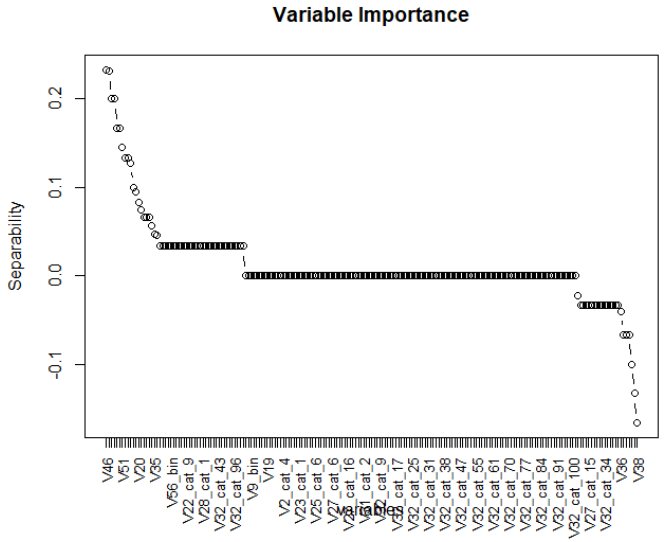
2 Preprocessing

Relief Algorithm

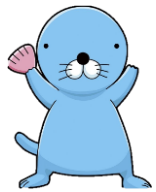
Relief Algorithm

이진 분류 문제에서 **변수별 중요도**를 판단할 수 있는 알고리즘으로 독립 변수 별 Y = 0과 1의 분리도(separability)를 산출

(높은 분리도를 가진 X 변수 → Y 변수의 분리에 큰 영향을 미침)



랜포 중요도 결과와 달리 수치, 범주, 이진 변수들의 중요도가 **다양하게** 나타나는 것을 확인



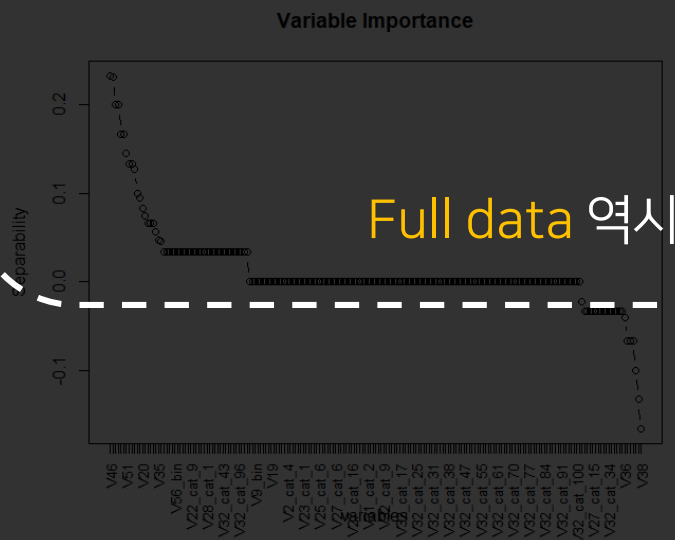
Separabillity가 0 이상인 값들을 기준으로 변수 선택을 진행!

2 Preprocessing

Relief Algorithm

선행 지식이 없는 상태이므로, 변수 중요도만을 보고 함부로

변수 선택을 진행하여 모델링하는 것은 위험하다고 판단



랜포 중요도 결과와 달리 수치, 범주,
이진 변수들의 중요도가 **다양하게**
나타나는 것을 확인

Full data 역시 이후 모델링 과정에서 사용

Separabillity가 0 이상인 값들을
기준으로 변수 선택을 진행!

2 Preprocessing

변수 선택

모델링 결과 Relief Algorithm과 Random Forest 통해
진행한 변수 선택은 Full data보다 비슷하거나
낮은 성능지표를 보여주었기에 **full model로 진행**

학습하는 모델과 파라미터, 적용한 샘플링 방법에 따라 성능이 달라짐
따라서 다양한 조합을 시도했음

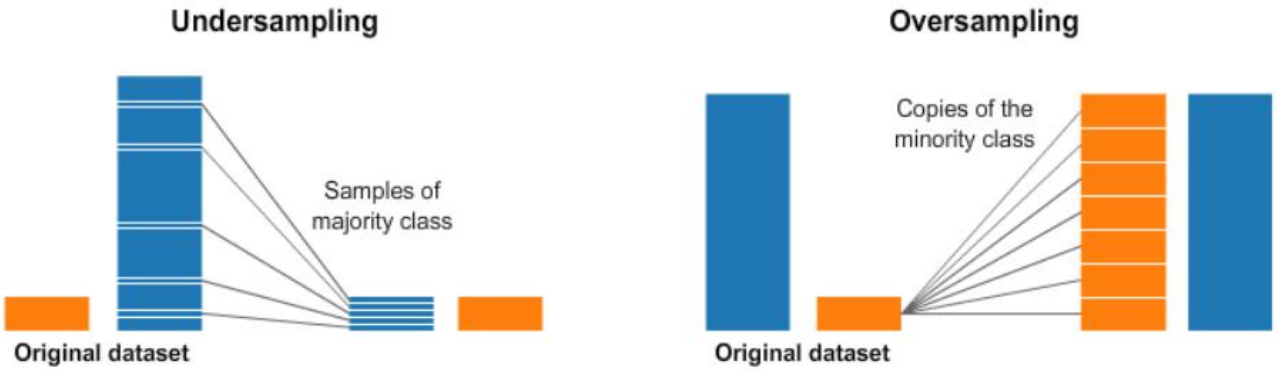
3

SAMPLING

샘플링 (Sampling)

데이터 불균형을 무시하고 모델에 적합하면 **과적합** 될 가능성과 소수의 관측치를 지니는 클래스의 정확도를 **평가지표에 온전하게 반영 못함**

→ **클래스 간 균형**을 맞춰야 함



3 SAMPLING

샘플링 (Sampling) 방법

Under Sampling

다수의 클래스를 소수의 클래스에 맞추어 관측치를 **감소**시키는 방법

→ 정보 누락 문제

Random Under Sampling	Tomek Link Method	CNN

Over Sampling

소수의 클래스의 데이터들을 다수의 클래스의 관측치 수에 맞추어 **증가**시키는 방법

Random Over Sampling	SMOTE	ADASYN

3 SAMPLING

샘플링 (Sampling) 방법

Under Sampling

다수의 클래스를 소수의 클래스에 맞추어 관측치를 **감소**시키는 방법

→ 정보 누락 문제

Random Under Sampling	Tomek Link Method	CNN

Over Sampling

소수의 클래스의 데이터들을 다수의 클래스의 관측치 수에 맞추어 **증가**시키는 방법

Random Over Sampling	SMOTE	ADASYN

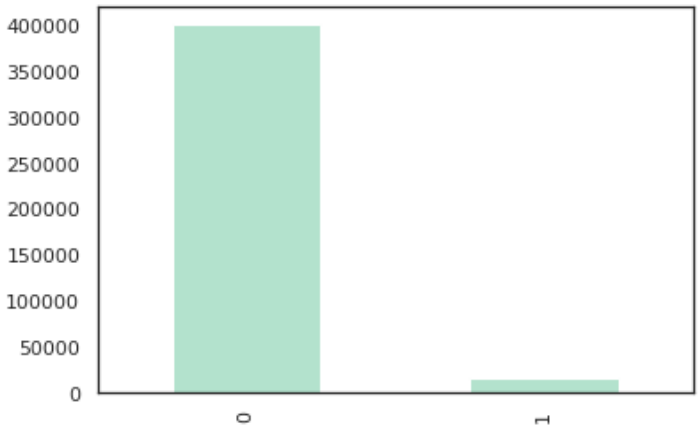
3 SAMPLING

오버 샘플링의 종류

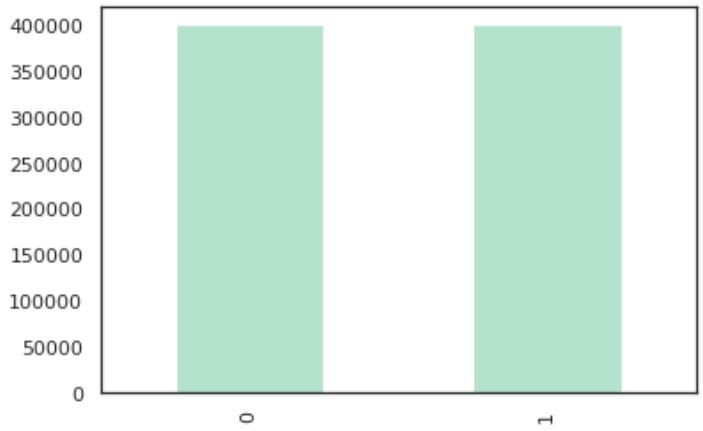
Random Over sampling

무작위로 소수 클래스의 데이터를 복제하여 관측치의 수를 늘리는 것

동일한 데이터의 수가 늘어나면서 과적합 될 확률 높음



416645 rows x 59 columns

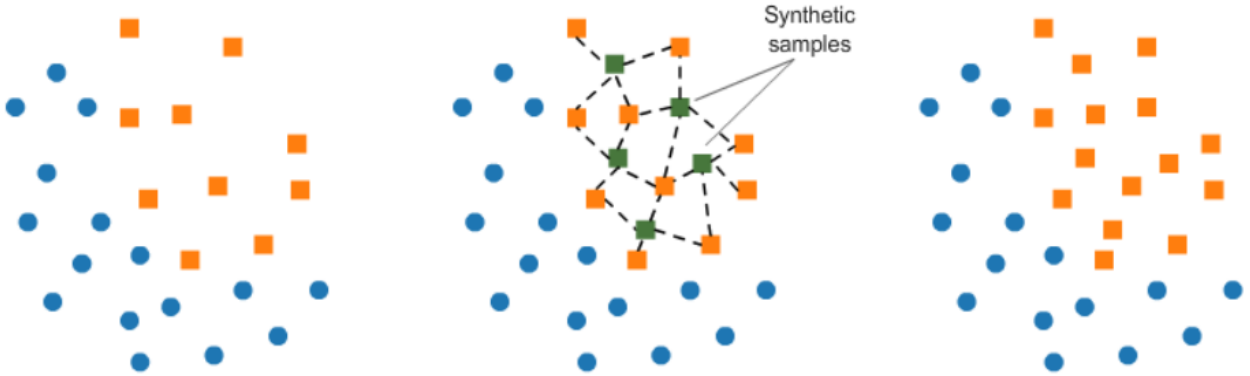


802838 rows x 59 columns

오버 샘플링의 종류

SMOTE (Synthetic Minority Over-sampling Method)

낮은 비율의 클래스에서 임의로 선택한 샘플과 K개의 최근접 이웃 간의 차에 0~1 사이의 임의의 값을 곱하는 방식으로 새로운 데이터 합성

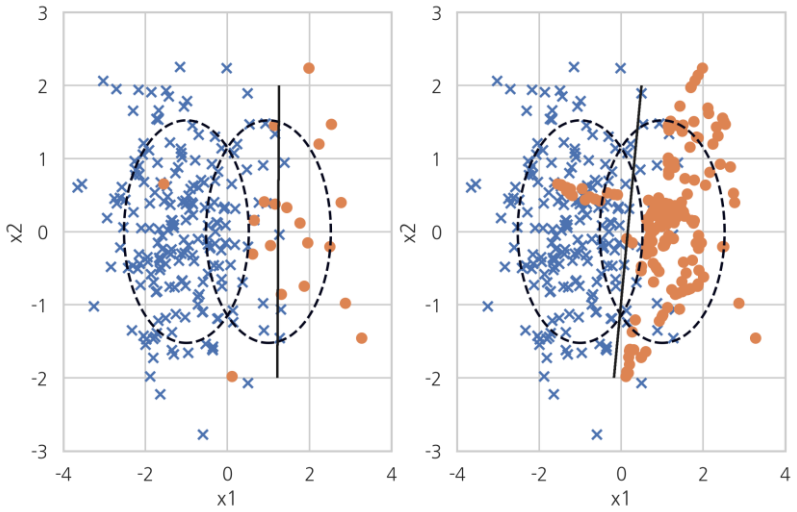


고차원에선 비효율적!

오버 샘플링의 종류

ADASYN (Adaptive Synthetic sampling approach)

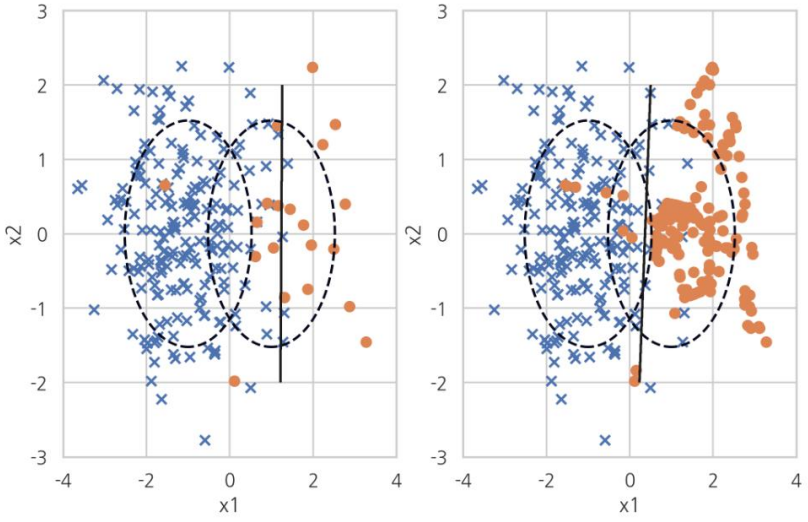
SMOTE의 개선된 버전으로 동일한 과정을 진행한 후 데이터의 임의의 작은 값을 더해줌으로 데이터가 조금 더 분산되게 표현된다



오버+언더 샘플링

SMOTETomek (Hybrid method)

오버 샘플링인 SMOTE와 언더 샘플링인 Tomek를 융합한 방법



4

MODELLING

4 MODELLING

사용한 모델들

시드는 마감일!

SEED: 804

Logistic Regression	나이브 베이즈	Random Forest
ADABOOST	LDA / QDA	MLP
XGBoost	Light GBM (LGBM)	Ensemble model

4 MODELLING

사용한 모델들

시드는 마감일!

SEED: 804

다른 모델들의 성능이
궁금하다면 APPENDIX 참고!

Logistic Regression	나이브 베이즈	Random Forest
ADABOOST	LDA / QDA	MLP
XGBoost	Light GBM (LGBM)	Ensemble model

4 MODELLING

Modelling

SEED:

성능이 좋고 복잡한 모델을 조합을 하면
더 분류를 잘 할 것 같아!

Logis
Regressio

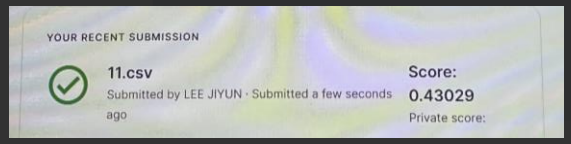
XGBoost

나이브 베이즈

Light GBM
(LGBM)

앙상블 모델

나이브 베이즈



가장 단순한 **logistic regression**
부터 모델링 시작!

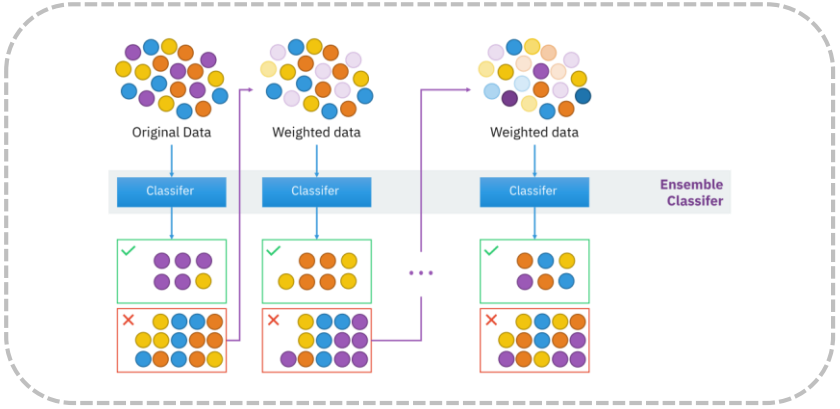
4 MODELLING

Model

Boosting

머신 러닝 앙상블 기법 중 하나로,

약한 학습기를 연속적으로 결합하여 성능이 높은 강한 학습기를 만드는 알고리즘



Boosting

XGBoost

LightGBM

4 MODELLING

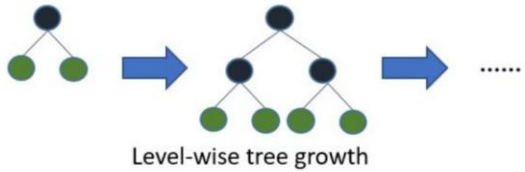
Model

Boosting

XGBoost

- ✓ 트기기반의 부스팅 모델
- ✓ 과적합 규제 기증으로 강한 내구성 보유
- ✓ 분류와 회귀영역에서 뛰어난 예측 성능 발휘

XGBoost:



Lightgbm

(Light Gradient Boosting Machine)

- ✓ XGBoost를 발전시킨 모델로 학습 시간을 단축 시킴
- ✓ Leaf-wise 분할 방식을 사용하여 level wise보다 예측 오류 손실 최소화

LightGBM:

4 MODELLING

Model

Boosting

XGBoost

- ✓ 트기기반의 부스팅 모델
- ✓ 과적합 규제 기증으로 강한 내구성 보유
- ✓ 분류와 회귀영역에서 뛰어난 예측 성능 발휘

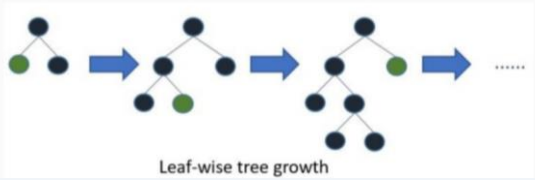
XGBoost:

Lightgbm

(Light Gradient Boosting Machine)

- ✓ XGBoost를 발전시킨 모델로 학습 시간을 단축 시킴
- ✓ Leaf-wise 분할 방식을 사용하여 level wise보다 예측 오류 손실 최소화

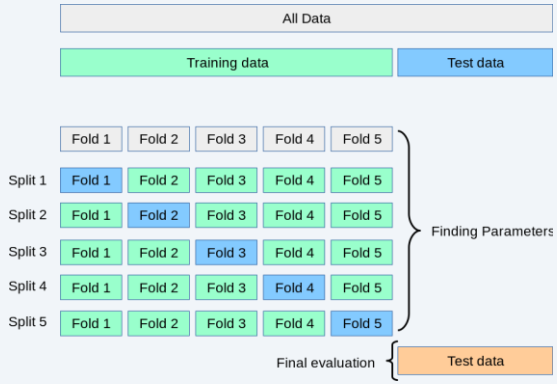
LightGBM:



Cross Validation

K-Fold CV

- ✓ Train set을 train과 K개의 Validation set으로 나누어 모델을 검증하는 방법
- ✓ 장점: 모든 데이터셋을 훈련에 사용할 수 있으며, 평가에도 활용할 수 있음



Grid Search Algorithm

모델에 필요한 hyper parameter의 모든 조합을 고려해, 최고 성능의 hyper parameter를 찾아내는 기법

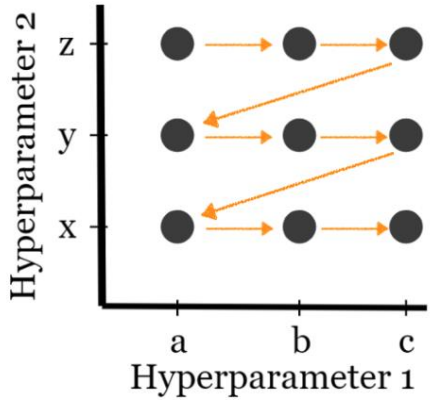
Cross Validation

K-Fold CV

- ✓ Train set을 train과 K개의 Validation set으로 나누어 모델을 검증하는 방법
- ✓ 장점: 모든 데이터셋을 훈련에 사용할 수 있으며, 평가에도 활용할 수 있음

Grid Search Algorithm

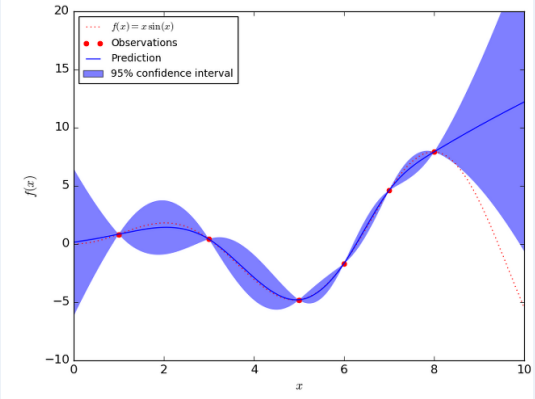
모델에 필요한 hyper parameter의 모든 조합을 고려해, 최고 성능의 hyper parameter를 찾아내는 기법



Cross Validation

Bayesian optimization

- ✓ 순차적으로 목적 함수를 최대(혹은 최소)로 하는 최적해를 찾는 기법
- ✓ 장점: 불필요한 반복 탐색을 줄여 빠르게 최적 파이퍼 파라미터를 찾을 수 있음



다양한 모델에 튜닝을 적용해 fitting한 결과!

4 MODELLING

Modelling

XGBoost

Random Oversampling

Gamma: 8.281

Max_depth: 9

Subsample: 0.9489

Learning_rate: 0.6573

Random_state = 804

F1 score : 0.84913

Kaggle: 0.50556

Lightgbm

Random Oversampling

수치형 변수 표준화

N_estimators : 1000

Num_leaves = 64

N_jobs = -1

Random_state = 804

Is_unbalanced = True

(베이지안 튜닝 도전했으나 성능 더 낮아짐)

F1 score : 0.7514

Kaggle: 0.51430

4 MODELLING

Modelling

왜 튜닝을 했는데 성능이 좋아지지 않았는가에 대한 고찰

Random Oversampling

Gamma: 8.281

Max_depth: 9

Subsample: 0.6

Learning_rate: 0.6573

Random_state = 804

Random Oversampling

수치형 변수 표준화

N_estimators : 1000

Num_leaves = 64

N_jobs = -1

Random_state = 804

Is_unbalanced = True

(베이지안 튜닝 도전했으나 성능 더 낮아짐)

Jupyter에서 적용했을 때는 성능이 향상

하지만 **Kaggle에서는 미비한 차이**를 보임

① 전처리 과정의 문제점?

② 다른 알 수 없는 문제의 존재

F1 score : 0.84913

Kaggle: 0.50556

F1 score : 0.7514

Kaggle: 0.51430

4 MODELLING

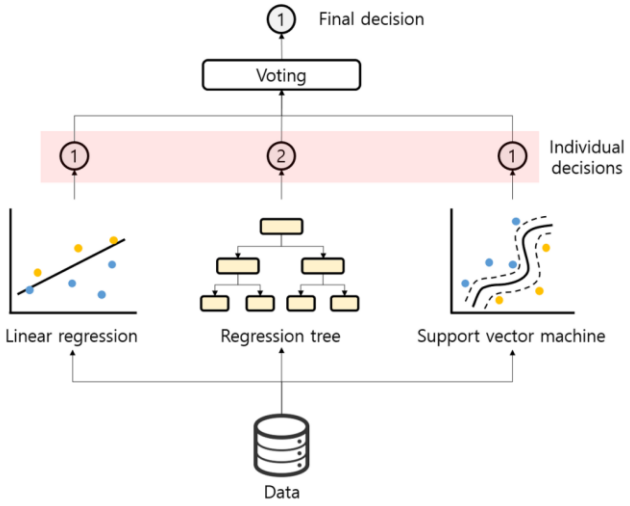
Model

Ensemble model

여러 개의 기본 모델을 활용하여 하나의 새로운 모델을 만들어내는 기법

이미 Ensemble 모델로 설정된 모델을 다시 활용하여 새로운 Ensemble 모델을 만드는 것도 가능

(이른바 Ensemble 모델들의 Ensemble)



4 MODELLING

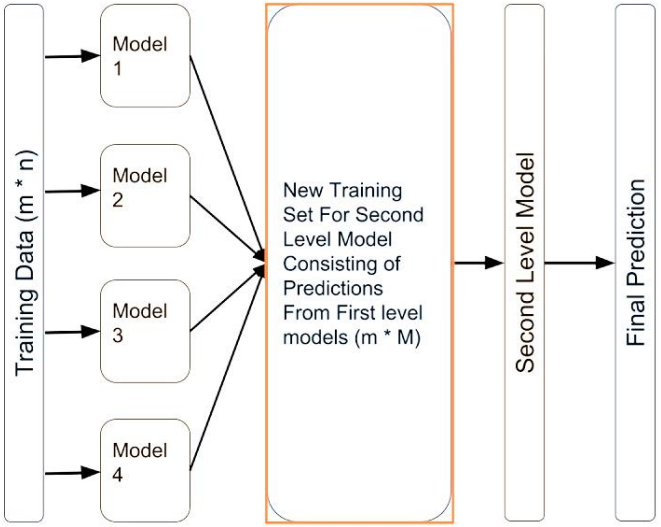
Model

Stacking Ensemble (lgbm/xgboost + logistic)

개별 알고리즘으로 예측한 데이터를 기반으로 스태킹 모델을 통해 다시 학습

Lgbm 모델 2개, xgboost 로 예측한 데이터를 기반으로 로지스틱회귀로 다시 예측을 수행

Lgbm 모델과 xgboost의 조합을 바꿔가면서 예측 수행



결과

다운샘플링 + 앙상블 = 과적합

Smote + 앙상블 = 차이 미비

SMOTETomek + 앙상블 = 차이 미비

랜덤오버샘플링 + 앙상블 = 미비하게 향상

처음에는 결과가 가장 좋을 것이라고 예상했지만 아니었다..

4 MODELLING

Model

Stacking Ensemble (lgbm/xgboost + logistic)

개별 알고리즘으로 예측한 데이터를 기반으로 스태킹 모델을 통해 다시 학습

Lgbm 모델 2개, xgboost 로 예측한 데이터를 기반으로 로지스틱회귀로 다시 예측을 수행

Lgbm 모델과 xgboost의 조합을 바꿔가면서 예측 수행

결과

다운샘플링 + 앙상블 = 과적합

Smote + 앙상블 = 차이 미비

SMOTETomek + 앙상블 = 차이 미비

랜덤오버샘플링 + 앙상블 = 미비하게 향상

처음에는 결과가 가장 좋을 것이라고 예상했지만 아니었다..

4 MODELLING

성능 평가

모델	Sampling 종류	CV 종류 / 튜닝	F1 -score	Kaggle
Logistic Regression	Random over sampling	x	0.5709	0.43029
나이브 베이즈	Random over sampling	x	0.5444	0.4459
Random Forest	Random over sampling	x	과적합	
MLP	Random over sampling	Layer 추가	0.7025	0.50045
앙상블 (Qda, lda, lgbm, adaboost)	Random over sampling	렘 터짐…	0.6374	0.4438

5

RESULT

5 RESULT

최종 모델: LightGBM

Lightgbm

수치형 변수
스케일링 (표준화)

→

GridSearchCV
파라미터 튜닝

→

Is_unbalance 등
파라미터 추가 조정

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
mean_fit_t	std_fit_tim	mean_scor	std_score_	param_ma	param_mir	param_n_e	param_nur	params	split0_test	split1_test	split2_test	mean_test	std_test_s	rank_test_score	
31.84167	1.104935	3.123601	0.031448	2	500	400	16	{'max_dep	0.622661	0.62346	0.620512	0.622211	0.001245	17	
32.86704	1.377578	3.223481	0.057714	2	500	400	64	{'max_dep	0.622661	0.62346	0.620512	0.622211	0.001245	17	
47.58403	1.318849	4.604828	0.140655	2	500	600	16	{'max_dep	0.632915	0.634257	0.631607	0.632927	0.001082	15	
47.67752	2.265569	4.63676	0.094739	2	500	600	64	{'max_dep	0.632915	0.634257	0.631607	0.632927	0.001082	15	
77.92548	0.292901	7.354357	0.072389	2	500	1000	16	{'max_dep	0.646178	0.649937	0.645794	0.647303	0.001869	13	
76.21082	1.051165	8.141988	0.859566	2	500	1000	64	{'max_dep	0.646178	0.649937	0.645794	0.647303	0.001869	13	
45.03498	1.34423	5.412087	0.039635	4	500	400	16	{'max_dep	0.694613	0.698073	0.698296	0.696994	0.001686	11	
44.37609	1.137002	5.388626	0.026373	4	500	400	64	{'max_dep	0.694613	0.698073	0.698296	0.696994	0.001686	11	
65.32625	1.198821	8.746595	1.173791	4	500	600	16	{'max_dep	0.722621	0.726274	0.727447	0.725448	0.002055	8	
65.19857	1.373494	8.293351	0.130491	4	500	600	64	{'max_dep	0.722621	0.726274	0.727447	0.725448	0.002055	8	
106.566	0.791825	15.47098	0.399556	4	500	1000	16	{'max_dep	0.765949	0.772766	0.767793	0.768836	0.002879	5	
106.4722	0.760923	16.57847	0.641976	4	500	1000	64	{'max_dep	0.765949	0.772766	0.767793	0.768836	0.002879	5	
43.1766	1.326401	5.338649	0.922353	6	500	400	16	{'max_dep	0.713938	0.71732	0.712316	0.714525	0.002084	10	
59.76104	1.589773	9.036655	1.093716	6	500	400	64	{'max_dep	0.809924	0.813728	0.808826	0.810826	0.0021	3	
62.47361	1.135189	6.744018	0.090116	6	500	600	16	{'max_dep	0.746245	0.750701	0.747874	0.748273	0.001841	7	
86.13752	0.664184	16.94483	0.195123	6	500	600	64	{'max_dep	0.852454	0.856172	0.85196	0.853529	0.00188	2	
99.80899	0.689454	12.84717	0.117974	6	500	1000	16	{'max_dep	0.795602	0.802554	0.803465	0.800541	0.003512	4	
142.7063	0.566011	28.68112	1.197891	6	500	1000	64	{'max_dep	0.899695	0.90623	0.90374	0.903222	0.002693	1	

5 RESULT

의의

1. 클린업 모든 팀의 내용을 활용할 수 있어서 좋았다
2. 여러가지 이진 분류 모형을 다뤄봤음
3. 다양한 방법을 사용하여 성능 결과가 어떻게 바뀌는지 확인해 봤음

Logistic Regression, LGBM, XGBoost, Deep Learning, Ensemble

5 RESULT

한계

1. 데이터 용량 문제, 테스트셋 이용 규제 문제로 인해 mice 사용 못함
2. 조합에 따라 성능이 다르게 나오는 이유를 해석하기 어려웠음
3. 파라미터 튜닝을 했으나, 과적합 문제가 의심되어 성능이 좋지 않았다
4. 데이터에 대한 선행 지식이 없어 파생변수를 고려하지 못했다

5 RESULT

느낀점

이번 방학 세미나 동안 오로지 데이터만 계속 보고, 모델링만 계속 돌려본 게 처음이라 많이 당황스럽기도 했지만, 좋은 사람들을 팀으로 만나 같이 하면서 또한 많이 즐거웠습니다😊 제가 중간에 알바 간다고 또 시간적으로 참여도 적게 한 것 같아서 미안하기도 하고, 또 다들 너무 잘하는 사람들이어서 고맙기도 하네요..!!
모두 어떤 팀으로 가서 활동할지 아직 모르지만, 그래도 방세 1팀이 최고였다는 사실을 기억해줬으면 좋겠습니다 ㅎㅎ
지윤이, 시언이, 민서, 지영이 다들 정말 정말 수고 많았어..!! 이제 좀 쉬자 다들👍 방세 1팀 최고👍👍

너무나 좋은 팀원들을 만나서 1주 동안 즐겁게 할 수 있었던 것 같아요!! 노력해준 팀원들 너무 고맙고 같이 먹은 저녁들이 다 너무 맛있었습니다…이상할정도로 맛있었던 쭈삼… 다음엔 완전체로 함 갑시다❤

방세를 하면서 1주일이 정말 빠르게 지나간 것 같아요!! 능력자만 모인 1팀으로 활동하면서 정말 많은 것을 배울 수 있었습니다!! 다들 너무 수고 많았고 기회가 된다면 다음에도 같은 팀으로 만나면 좋을거 같아요~~ 방세 1팀 짱짱!! ❤

5 RESULT

느낀점

"튜닝의 끝은 순정"

1주일도 안되는 짧은 기간 동안 모두 수고했습니다!! 진짜 다양한 모델들을 공부해보고 많은 것을 배울 수 있어서 유익했습니다! 다들 수고 많았고 다음 학기에도 열심히 살아봐요… 방세 1팀 짱짱!! ❤

1주일도 안되는 짧은 기간 동안 전처리부터 모델링까지 하게 되어서 정말 빡셌던 일정이었는데 너무너무 수고했어요 다들,, 너무 똑똑하고 열정적인 팀원들을 만난 것 같아서 정말 든든했고, 다양한 모델을 찾아보고 공부해보면서 성장할 수 있었던 것 같아서 좋았어요 방세는 끝났지만 우리 자주 만나고 맛있는 것도 더 먹으러 가요 방세 1팀 짱짱!! ❤

감사합니다

Appendix

Modelling

로지스틱 회귀 모델

✓ 종속변수가 범주형인 데이터를 대상으로 하는 분류 기법

✓ Log-odds를 sigmoid 함수에 넣어서 [0,1] 범위의 확률을 구함

나이브 베이즈

✓ 베이즈 정리에 기반한 통계적 분류 기법으로 조건부 확률에 기반

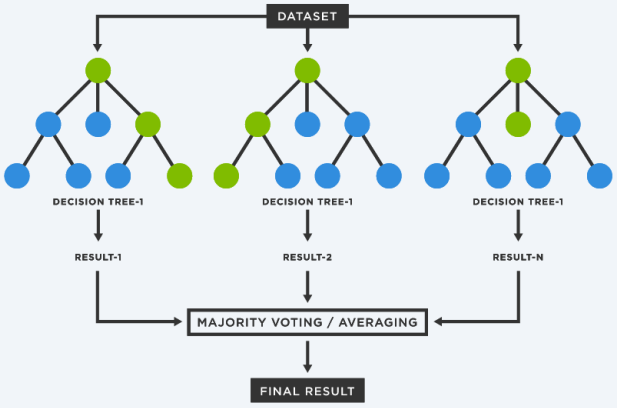
✓ 지도 학습 환경에서 매우 효율적으로 훈련되며 노이즈나 결측치가 많은 데이터 처리에 특화

Modelling

Random Forest

- ✓ 다수의 결정 트리들을 학습하는 앙상블 방법
- ✓ 다수결의 원칙 / 평균 예측치 등을 통해 최종 결과 도출

(각각의 트리들은 동등한 가중치를 가짐)



ADABOOST

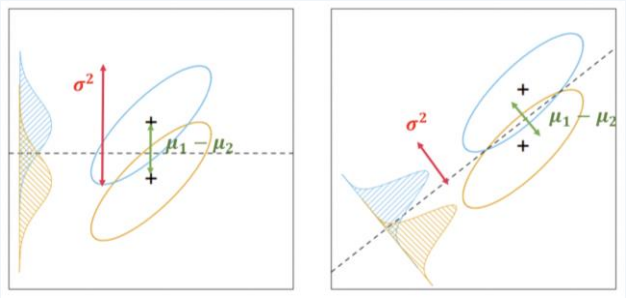
- ✓ 노드 하나에 두개의 leaf를 지닌 트리인 stump 다수를 학습하는 앙상블 방법
- ✓ 특정 stump는 다른 stump보다 가중치가 높거나 낮음
- ✓ 각 Stump의 error는 다음 Stump의 결과에 영향을 줌

Modelling

LDA

✓ 판별분석 과정 중 하나로 독립변수들의 측정값에 따라 데이터가 어느 집단에 속할 것인가에 대해 판별하는 분석방법

✓ 목표: 분산을 최소화 하면서 평균의 차이를 최대화하는 사영(projection) 찾기

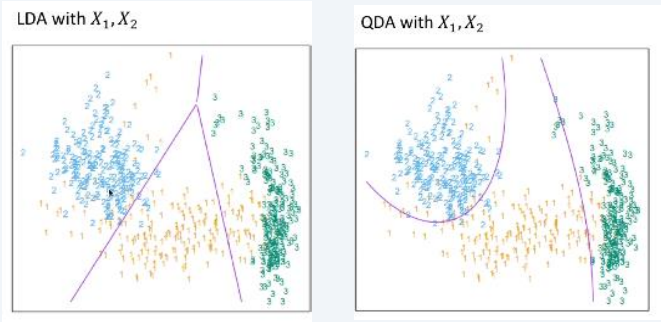


QDA

✓ LDA에서 공통 공분산 구조에 대한 가정을 제외시킨 방법

(즉, 범주별 다른 공분산 구조를 가진다고 가정)

✓ LDA와 달리, Decision boundary가 2차곡선의 형태를 띰



Modelling

MLP

- ✓ 퍼셉트론으로 이루어진 층(layer) 여러 개를 순차적으로 쌓은 다층신경망 구조
- ✓ 인접한 두 층의 뉴런 간에는 fully connected된다는 특징
- ✓ 이미지 분류 문제 등 이진 분류 분석에도 적용 가능