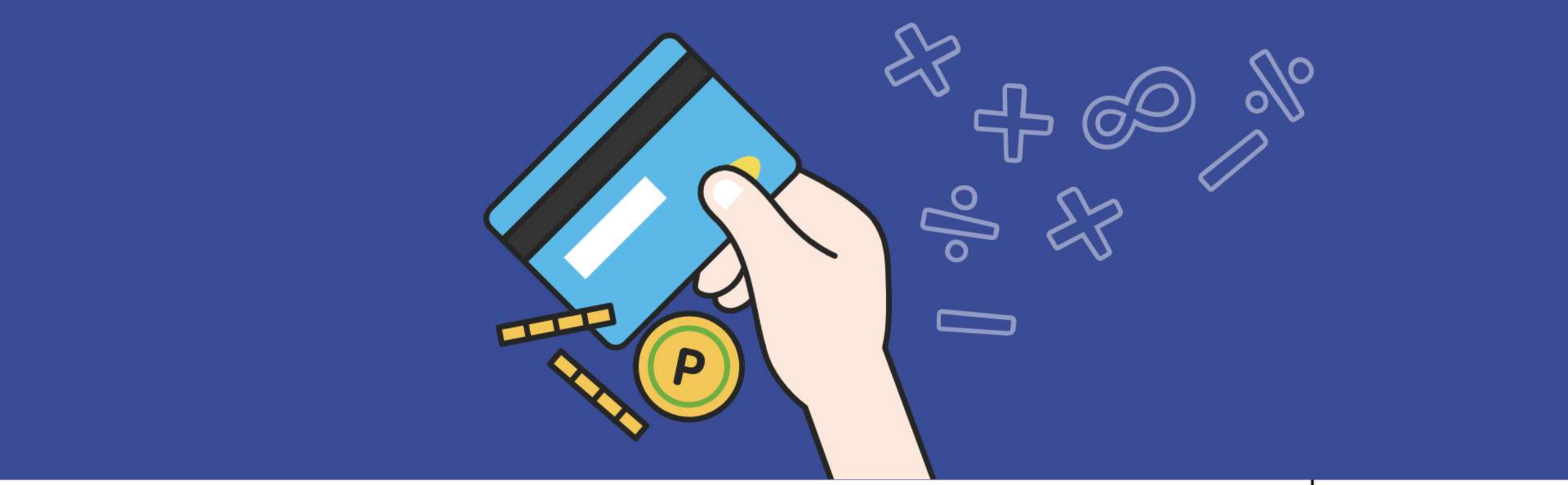


신용카드 사용자 연체 예측 AI 경진대회

Date: 2021.05.23

Dacon

CONTENTS

신용카드 사용자 연체 예측 AI 경진대회

신용카드 사용자 연체 예측
AI 경진대회

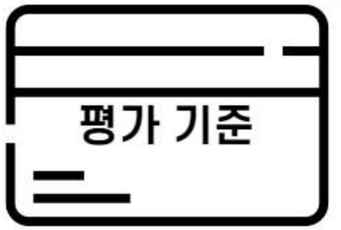
#01

INTRODUCTION

INTRODUCTION

purpose of competition

신용카드 사용자 데이터를 보고 사용자의
대금 연체 정도를 예측하는 알고리즘 개발



심사 기준

Log loss.
모델의 출력 값과
정답의 오차를 정의
하는 평가 지표

주의 사항

test set은 “*모른다고 가정*”
하고 모델을 학습.
즉, 모델 학습은 오직 train s
et으로만 훈련되어져야 한
다.

INTRODUCTION

DATA SET

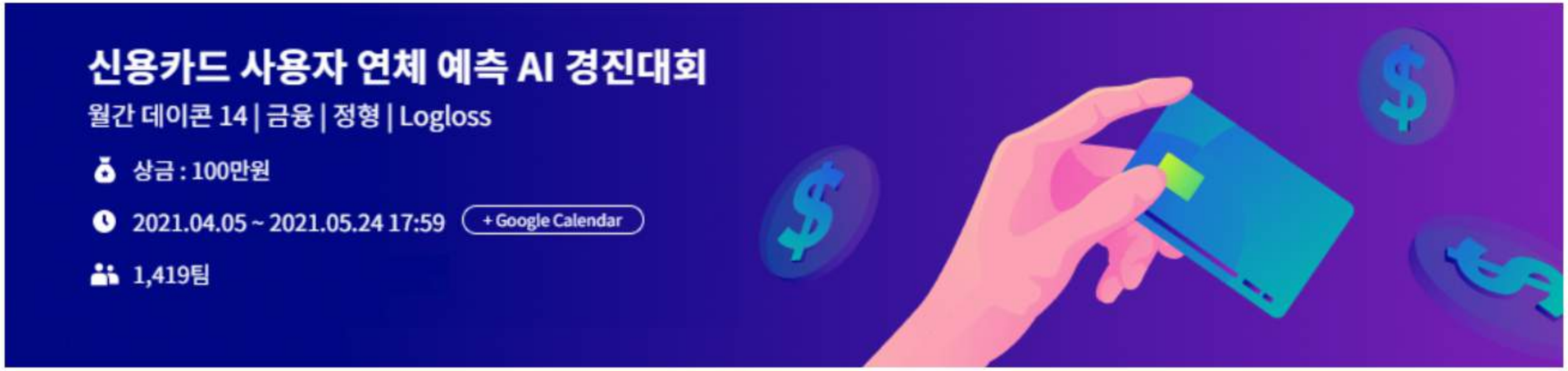
	ROWS	COLS	SIZE(mb)
train	26,457	20	3.3
test	10,000	19	1.3
sample_submission	10,000	4	0.12

이 대회에서 사용되는 Dataset은 신용카드 대금 연체를 기준으로 가공한 feature이므로
대회에 사용된 데이터 셋은 일반 통계자료와 상이하다.

사용된 데이터: Xiong Xuetang (https://mp.weixin.qq.com/s/upjzuPg5AMIDsGxlpqnoCg)

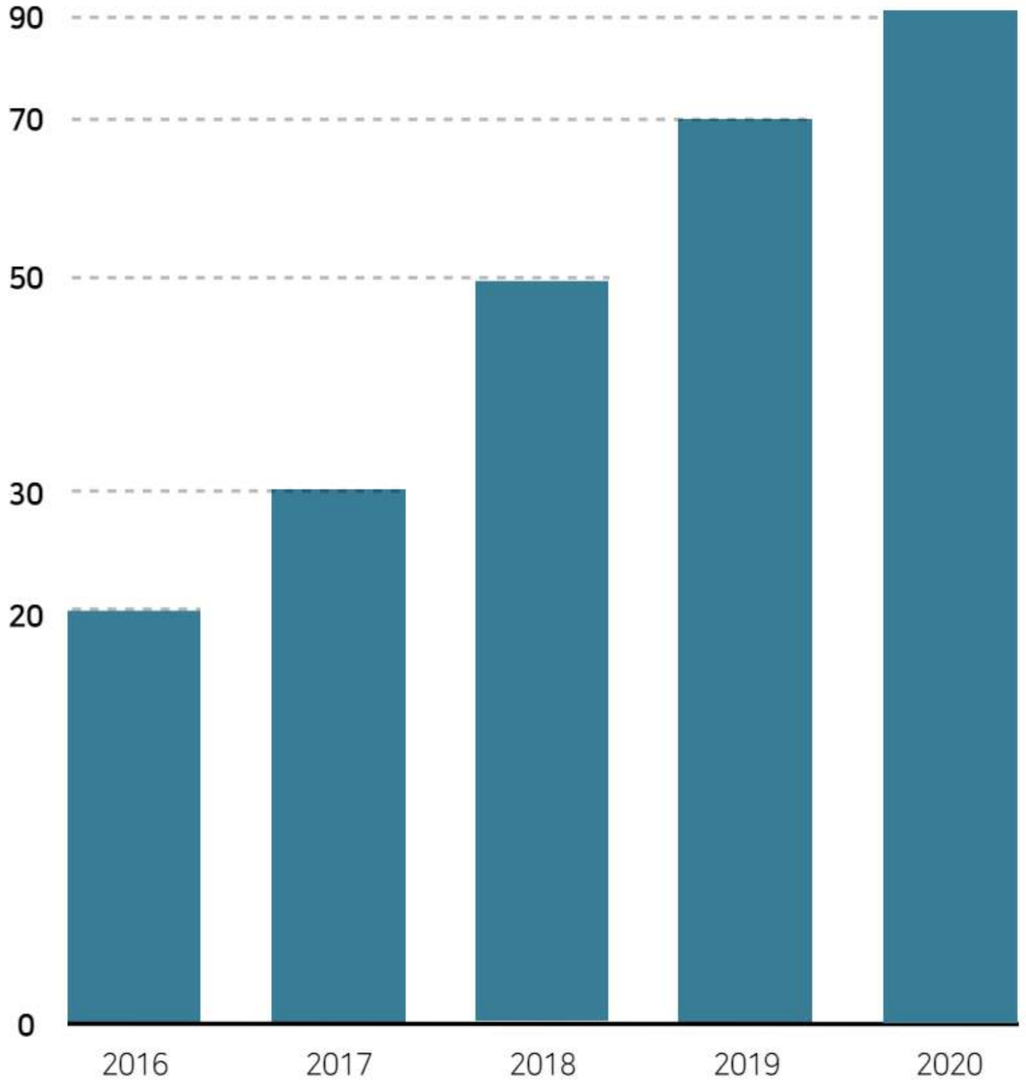
INTRODUCTION

Overview Of Competition



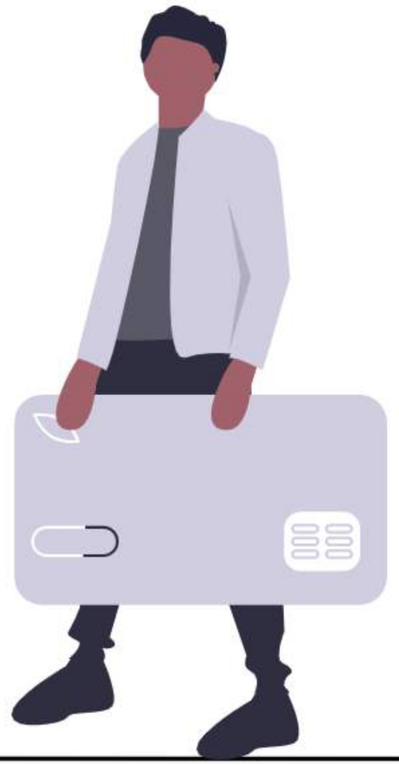
Competition Timeline April 5th , 2021 ~ May 24th, 2021

Duration Of Participation 25 days (April 30th , 2021 ~ May 24th, 2021)



#02

EXPLORATORY DATA ANALYSIS (EDA)



EDA

변수 정의

변수	설명	형태	Data Type
credit	신용도	종속변수	flaot
gender	성별	F(female) or M(male)	object
car	차량 소유 여부	N or Y	object
reality	부동산 소유 여부	N or Y	object
child_num	자녀 수	0,1,2,...,19 / 0,1 (자녀유무)	integer
income_total	연간 소득	Continuous	flaot
income_type	소득 분류	'Commercial associate', 'Working', 'State servant', 'Pensioner', 'Student'	object
edu_type	교육 수준	'Higher education' ,'Secondary / secondary special', 'Incomplete higher', 'Lower secondary', 'Academic degree'	object
family_type	결혼 여부	'Married', 'Civil marriage', 'Separated', 'Single / not married', 'Widow'	object

변수들은 가상데이터다

EDA

변수 정의

변수	설명	형태	Data Type
house_type	생활 방식	'Municipal apartment', 'House / apartment', 'With parents', 'Co-op apartment', 'Rented apartment', 'Office apartment'	object
DAYS_BIRTH	출생일	데이터 수집 당시(0)부터 역으로 셈	integer
DAYS_EMPLOYED	업무 시작일	데이터 수집 당시(0)부터 역으로 셈	integer
FLAG_MOBIL	핸드폰 소유 여부	1	integer
work_phone	업무용 전화 소유 여부	0 or 1	integer
phone	가정용 전화 소유 여부	0 or 1	integer
email	이메일 소유 여부	0 or 1	integer
occyp_type	직업 유형	범주형	object
family_size	가족 규모	1,2,....20	float
begin_month	신용카드 발급 월	데이터 수집 당시(0)부터 역으로 셈	float

변수들은 가상데이터다

EDA

시각화

빅데이터 시각화

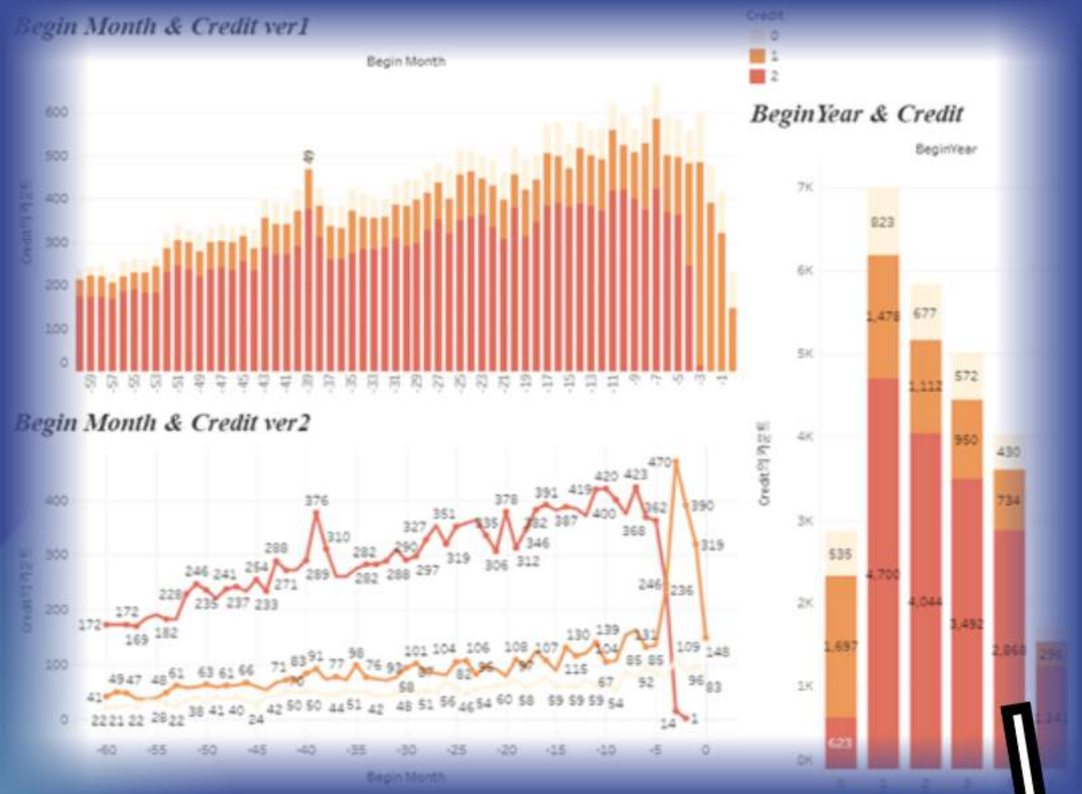
Tableau 사용

인텔리전스에 중점을 둔 대화 형 데이터 시각화 소프트웨어를 사용

시각화 진행

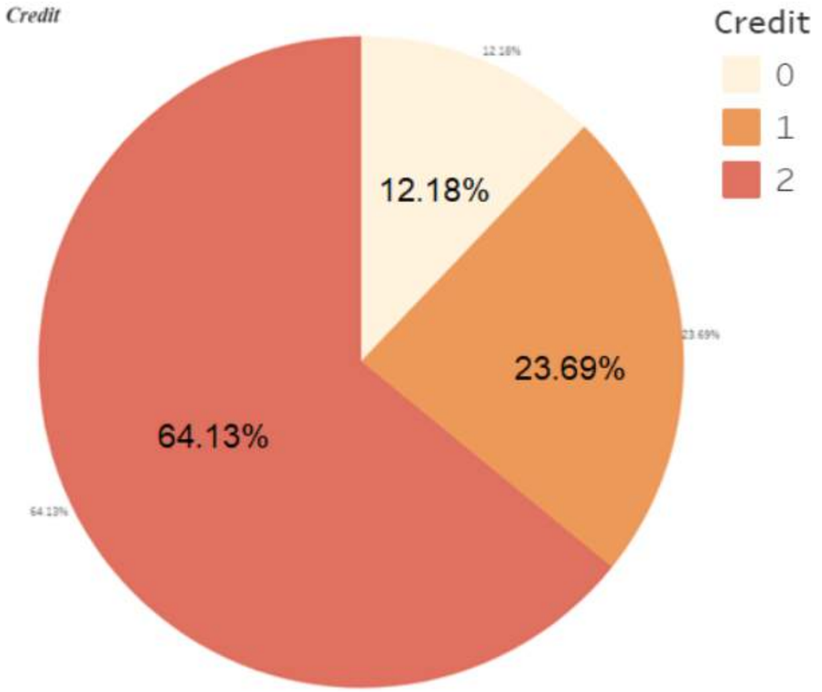
Tableau를 사용하여 시각화를 진행

Tableau 사용



EDA

종속변수 - Credit



Tableau를 사용하여 시각화를 진행

종속변수 - Credit

숫자가 작을수록 더 높은 신용도를 가진 사용자를 의미함

Credit이 2의 분포가 가장 높은 것을 확인할 수 있다

Tableau 이용한 시각화

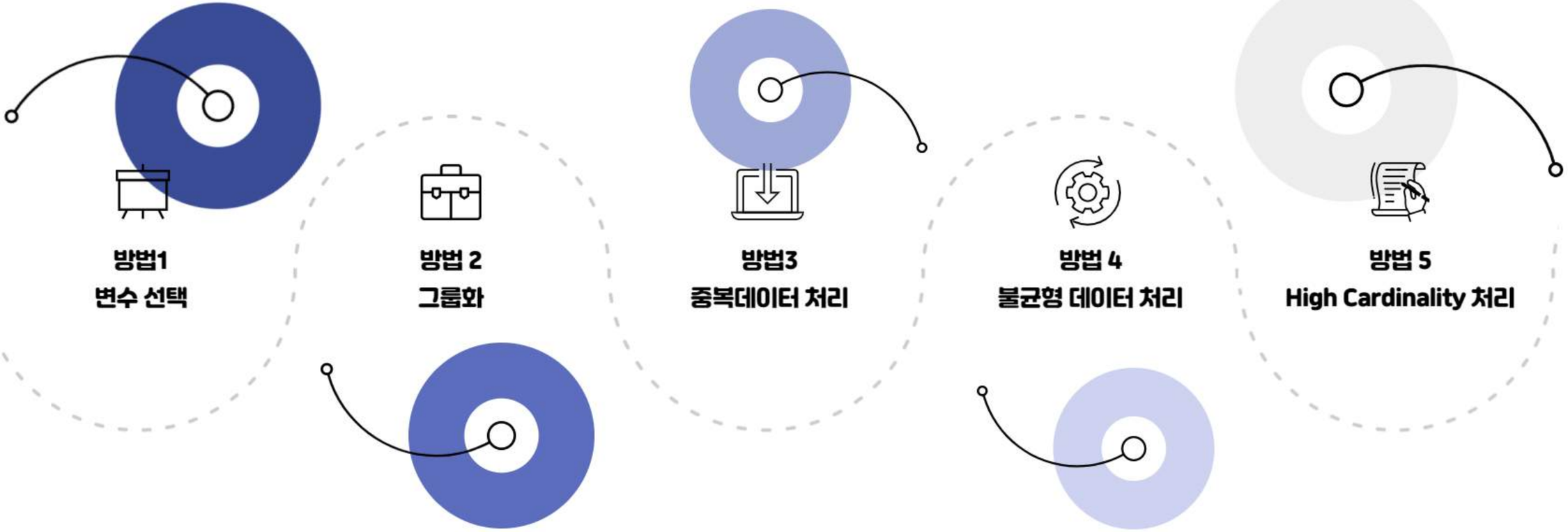
#03

FEATURE ENGINEERING

데이터 가공 및 전처리

FEATURE ENGINEERING

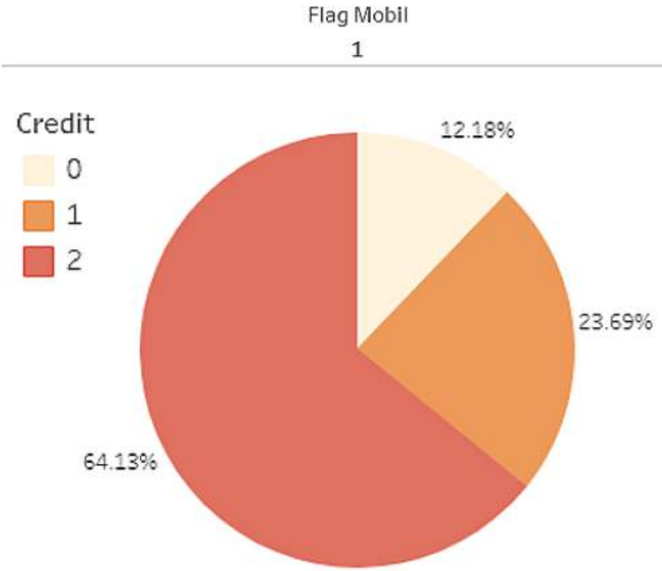
Logloss를 줄이기 위한 방법론



FEATURE ENGINEERING

방법 1) 변수 삭제

Flag Mobile & Credit



'Flag Mobil' feature의 경우 모두 1로 이루어져있다.

➡ 분석하는데 있어서 의미가 없다고 판단!

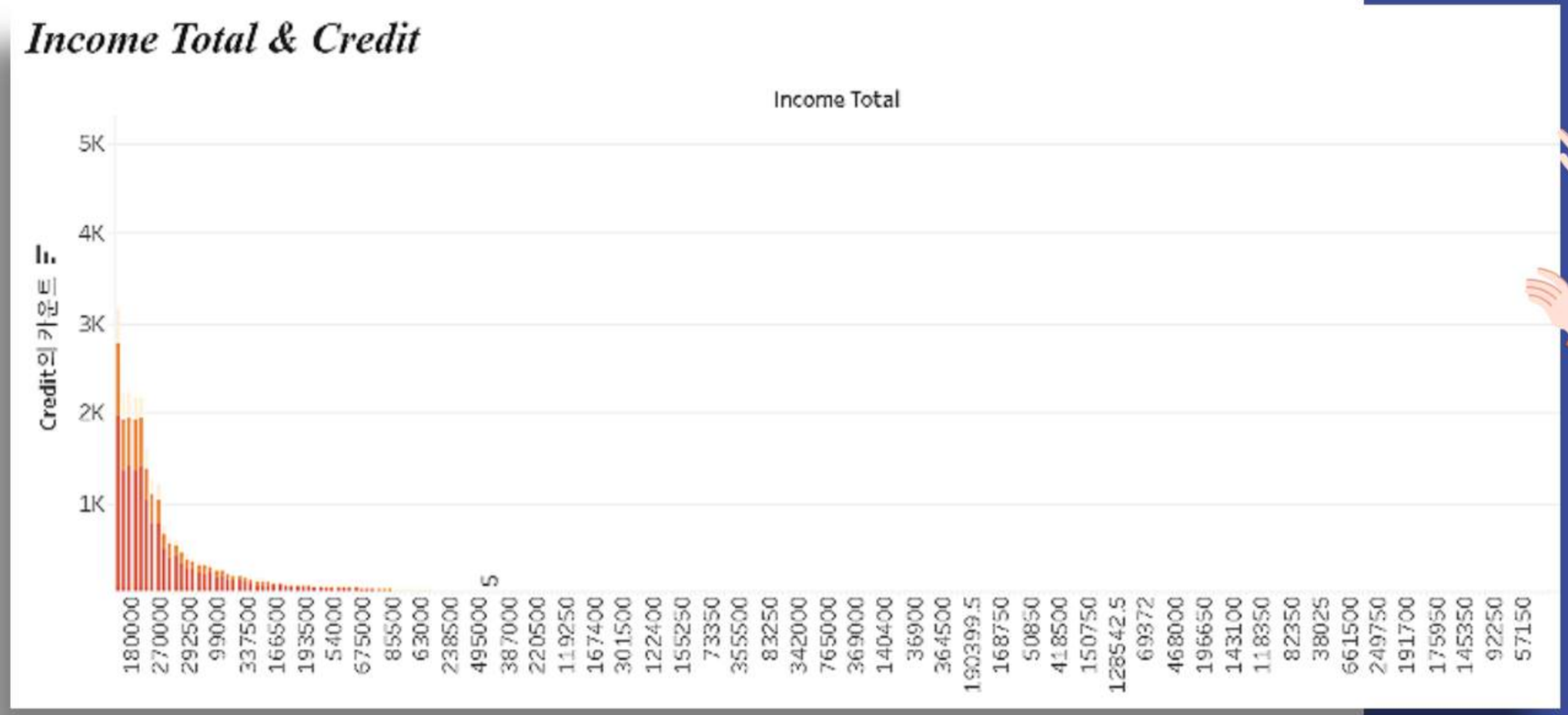
변수 삭제

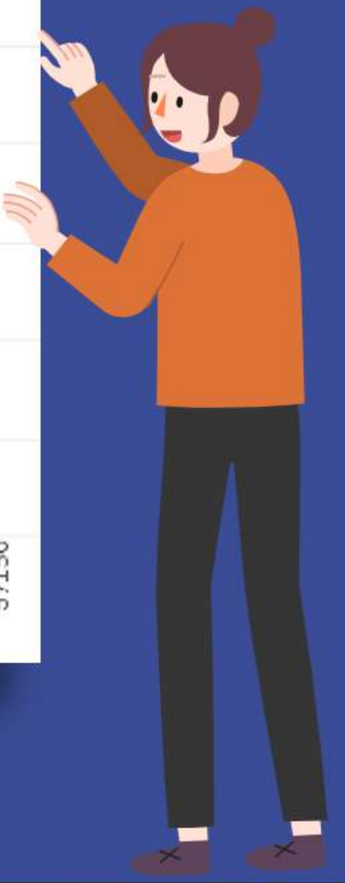
Drop한 변수	Log Loss
Index	0.731
Index, Flag Mobil	0.736

모두 1로 이루어진 변수인 'Flag Mobil'을 Drop한 뒤의 "Log Loss"를 비교

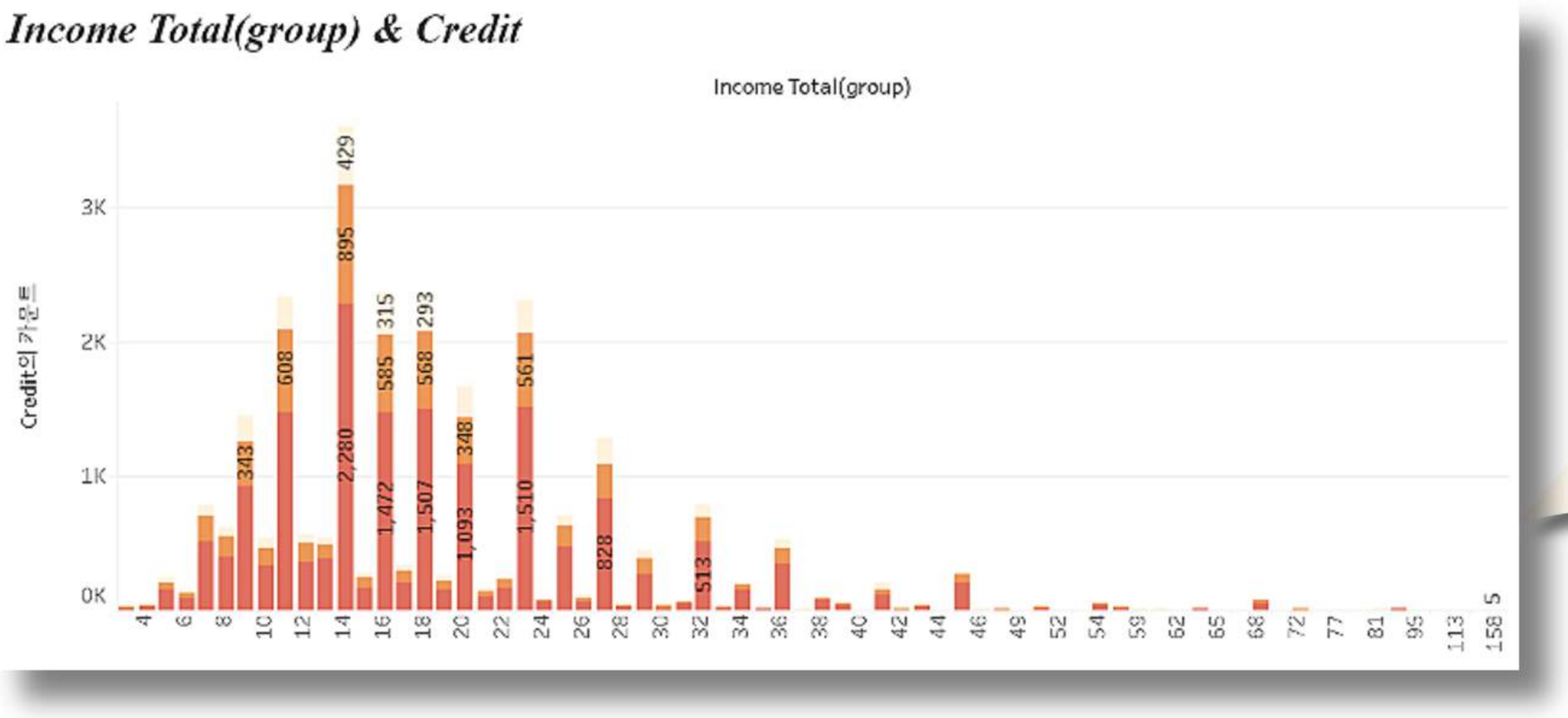
➡ 오히려 더 높게 측정

방법 2) 그룹화 - Income Total





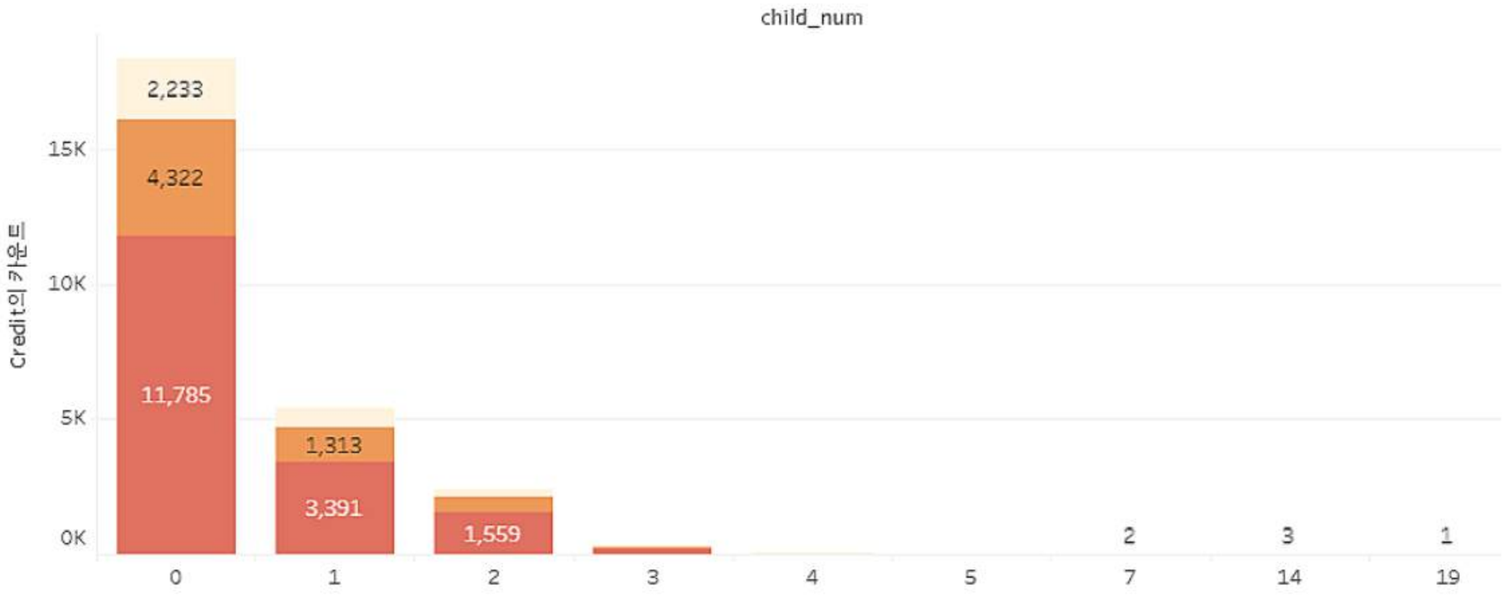
방법 2) 그룹화 - Income Total



'Income_total'(연간소득) 을
10,000 단위로 그룹화

방법 2) 그룹화 - Child Num

Child Num & Credit

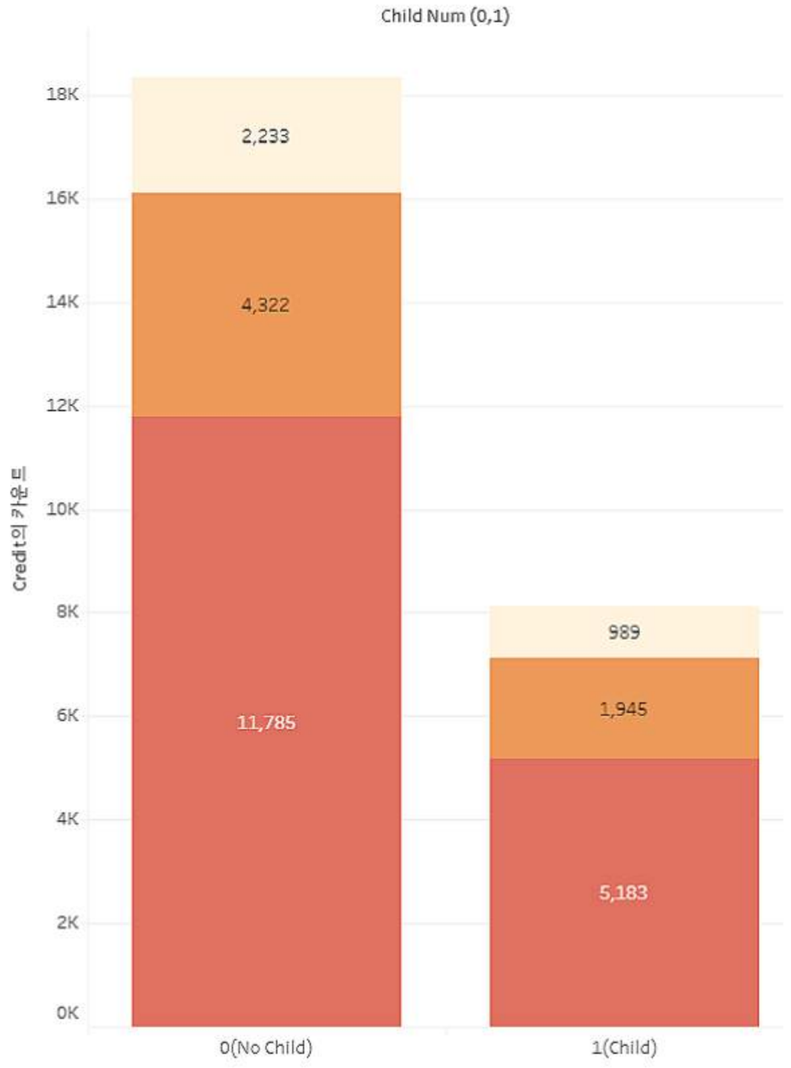


- 'child_num'(자녀 수)이 1 이상일 경우

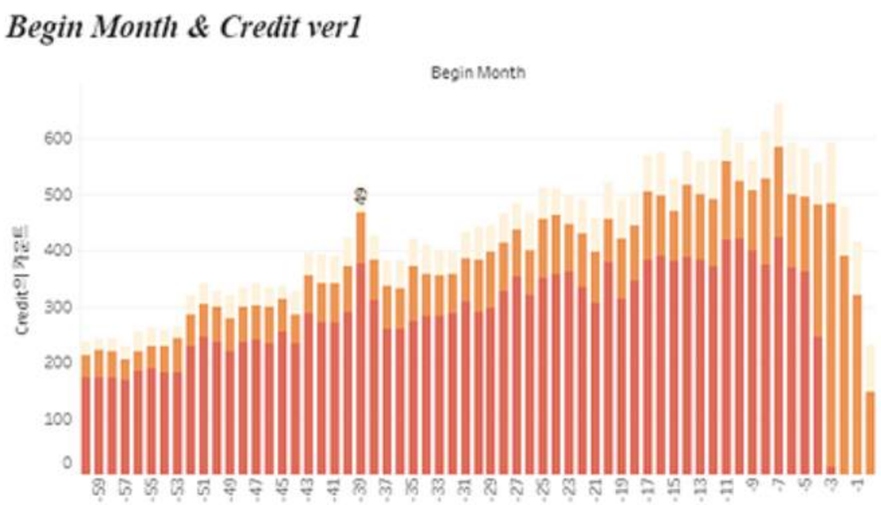
➡ '자녀 있음' 으로 그룹화

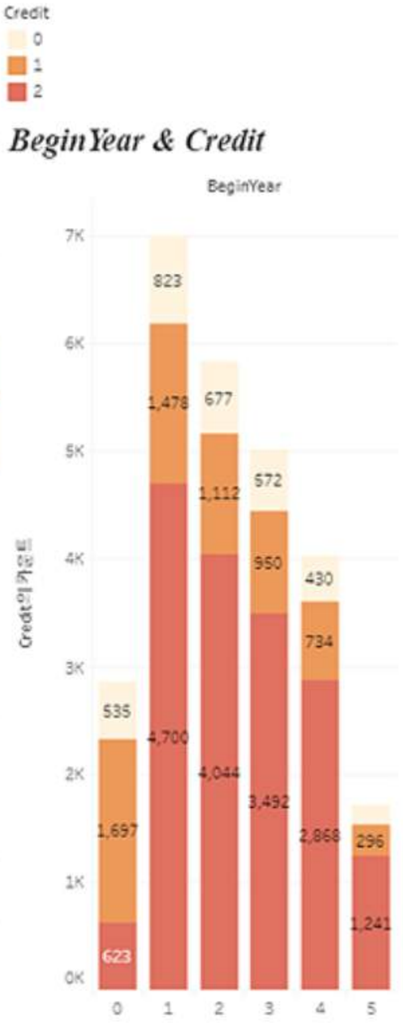
- child_num을 자녀 유무(0,1)로 변수를 재정의하였다.

Child Num(0,1) & Credit (2)



방법 2) 그룹화 - Begin Month



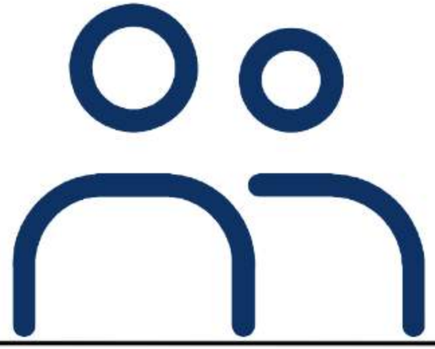


- 'begin_month'가 0개월~ 3개월까지의 신용도

➡ 낮은 신용도 분포

- 신용카드를 발급했을 때 신용도가 소폭 하락하는 경향을 보임

➡ 따라서 begin_month(카드 발급월)를 12개월 단위로 묶어서 'begin_year'로 재정의

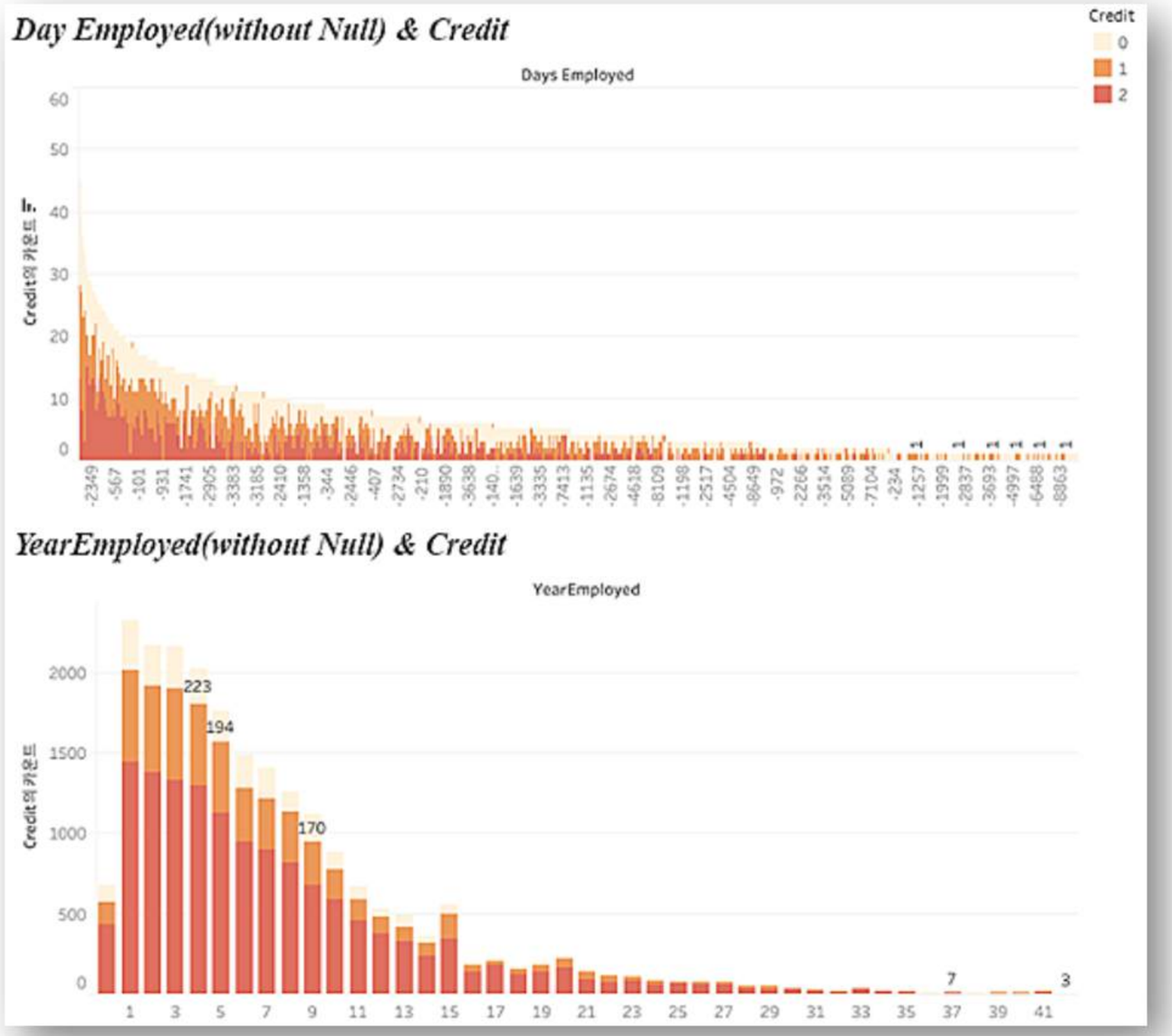


방법 2) 그룹화 - Day Employed

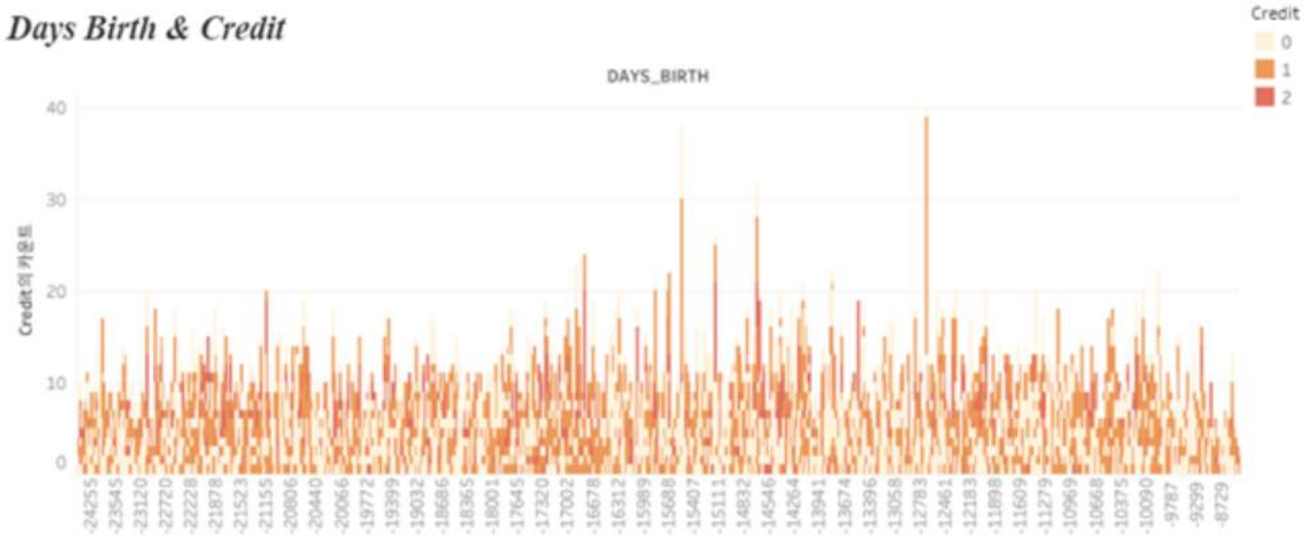
- 'day_employed'의 분포에서의 "365243"

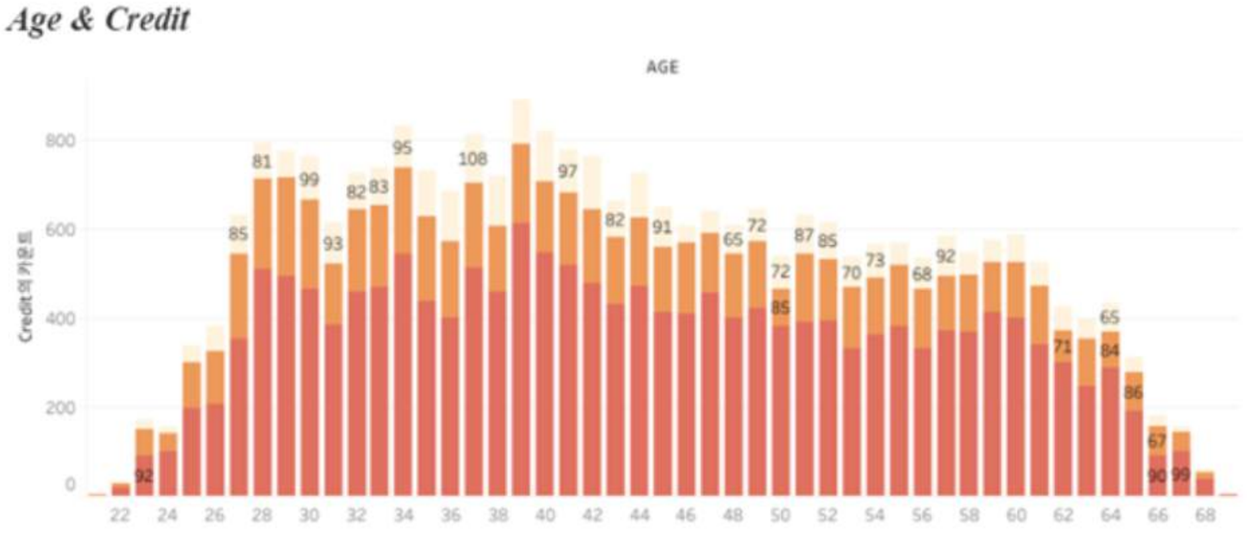
  ➡ NULL 값

- 'day_employed'를 365일로 나누어 범주화 하였다.



방법 2) 그룹화 - Day Birth





'day_birth'를 365일로 나누어 'Age'로 재정의하였다.

변수 제거 & 그룹화 결론

시각화 결과를 보고 EDA 과정에서 변수를 제거 or 그룹화

↓

과적합은 줄어들지만 정확도(Accuracy)까지 줄어들어 Log loss 는 커졌다.

“

사용하는 데이터셋이 프로그램으로 가공된 데이터이기 때문에 특정 경향을 띄고 있다.

”

↓

따라서 EDA를 기반으로한 판단대로 변수를 제거하거나 그룹화하게되면

그 경향에서도 멀어지기 때문에 Log loss 또한 안좋게 나온 것으로 보인다.

방법 3) 중복 데이터

중복 데이터가 분석에 끼치는 영향은?

- 중복 데이터가 있을 경우, 분석에 안좋은 영향 끼칠 가능성 UP

→ 이러한 영향을 최소화하기 위해서 "중복 데이터를 제거" 하였다.

```
1 overlap_begin.groupby('credit')['credit'].value_counts()
```

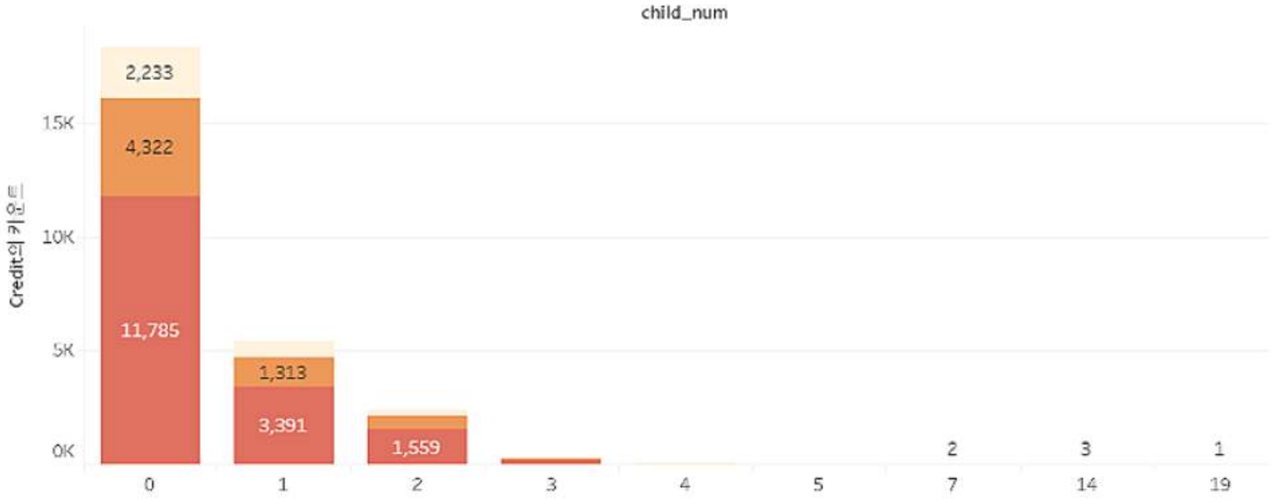
```
credit  credit
0.0     0.0      2049
1.0     1.0      4390
2.0     2.0     13936
Name: credit, dtype: int64
```

	Log loss
중복 데이터 제거 전	0.723
중복 데이터 제거 후	0.735

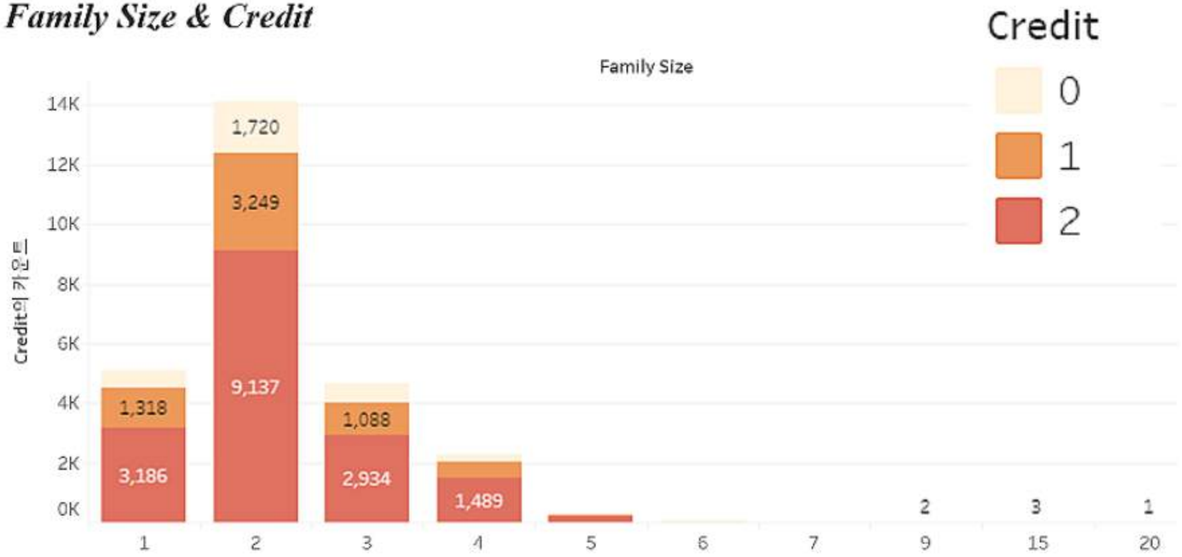
중복 데이터 제거 전의 Log loss가 더 좋게 나옴!

방법 4) 불균형 데이터

Child Num & Credit



Family Size & Credit



- 불균형 데이터인 변수들: reality, income_total, income_type, edu_type, family_type, house_type, family_size 등

➡ 대부분 Feature 들이 불균형한 데이터 분포를 띄는 것을 확인할 수 있다.

방법 4) 불균형 데이터

불균형 데이터로 인한 발생 문제

① 과적합 문제가 발생

② 정확도는 높아질 수 있지만 분포가 작은 값에 대한 정밀도와 클래스의 재현율이 낮아지는 문제가 발생할 수 있다.

데이터셋의 불균형 문제를 해결하기 위해 Over Sampling과 Under Sampling의 단점을 보완한 Combine Sampling 기법을 채택하였다!

	Log loss
Before 불균형 데이터 처리	0.7936
After SMOTE	0.7128
After SMOTE + ENN	0.5615
After SMOTE + TOMEK	0.7043

중복 데이터 & 불균형 데이터 처리 결론

중복 데이터 처리

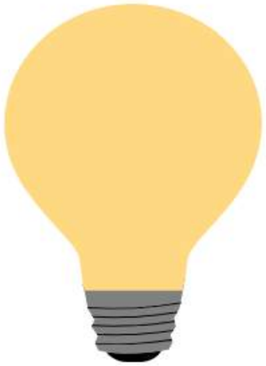
정확도 및 Logloss가 큰 폭으로 안좋아짐

→ Test 데이터 셋에도 중복 데이터가 많기 때문!

불균형 데이터 처리

정확도는 높아지지만 변수선택, 그룹화의 결론과 같이 Test 데이터 셋의 경향에서 벗어남

→ 크게 과적합되어 결과적으로 Logloss가 커짐



여기서 주목해야할 점

- 중복데이터 처리를 하면 Logloss가 안좋아지지만, 불균형 데이터 처리와 함께 할 경우 큰 폭으로 Log loss가 좋아진다!

- 만약 가상의 데이터 셋이 아닌 실제 데이터 셋이라면 중복 데이터 삭제 후 불균형 데이터 처리하는 것이 굉장히 좋은 Skill이 될 것이라 생각한다.

방법 5) High Cardinality 처리

Cardinality

① 전체 행에 대한 특정 컬럼의 중복 수치를 나타내는 지표
② 중복도가 낮을 수록 Cardinality가 높으며 중복도가 높을 수록 Cardinality가 낮다.
③ 여러 컬럼을 동시에 인덱싱할 때 Cardinality가 높은 컬럼(중복이 적은 컬럼)을 우선순위를 두는 것이 인덱싱 전략에 유리하다.

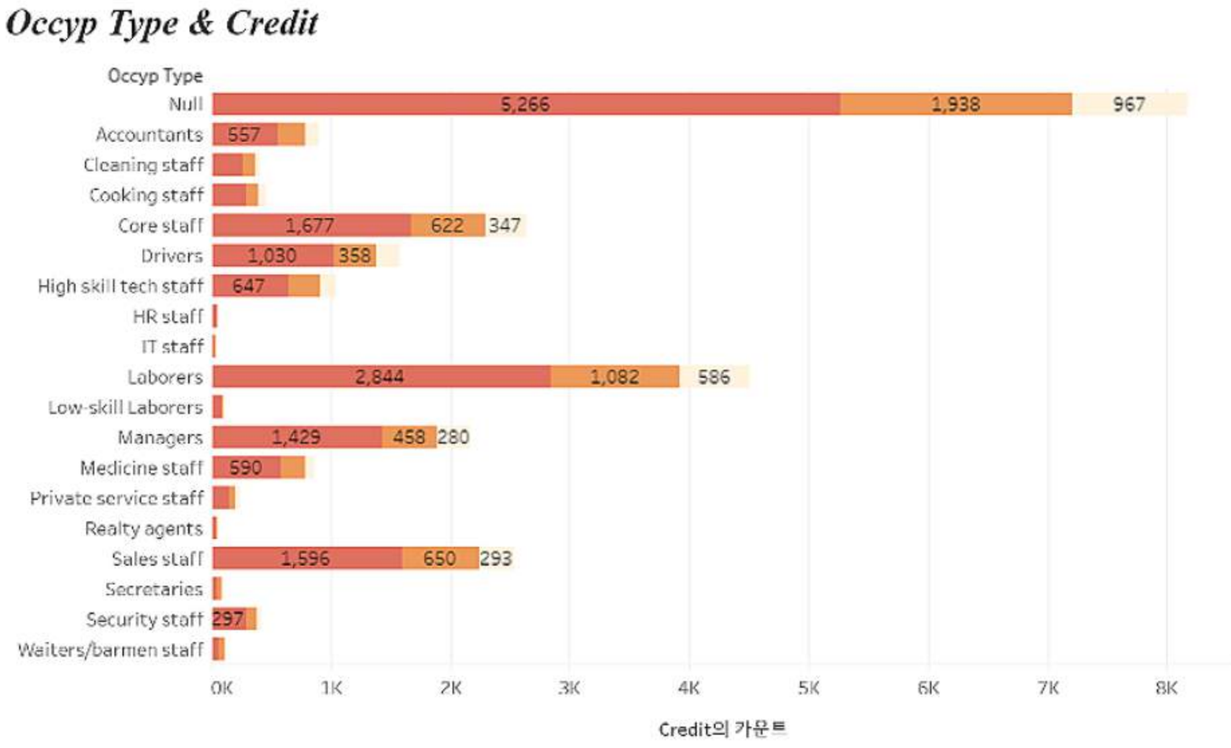
High Cardinality 처리하는 Encoding 방식

- Encoding 방법

- One-hot, Label, Binary, BaseN, Hashmap Encoding

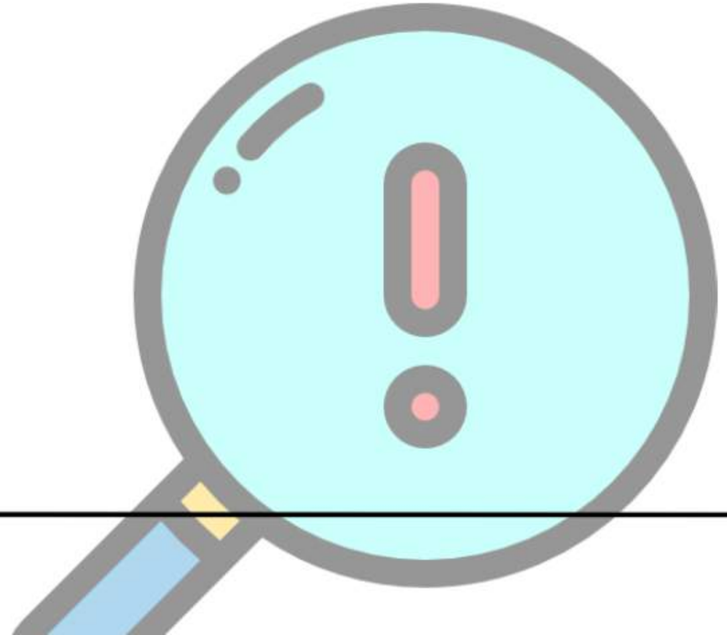
- 데이터 타입에 따른 Encoding 방법

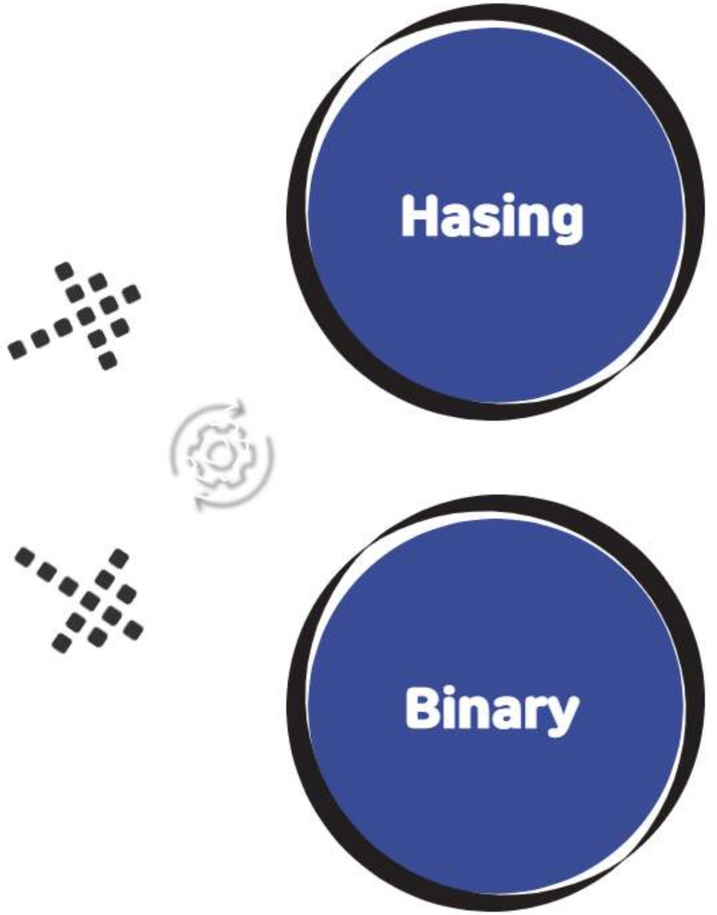
- Binary Encoder: 서열척도이며 High Cardinality인 경우
- Hashing Encoder:
  명목척도이며 High Cardinality인 경우



방법 5) High Cardinality에 따른 Encoding 방법

	High Cardinality Feature
명목형	family_type, house_type, edu_type, occup_type
순서형	Days_employ, begin_month





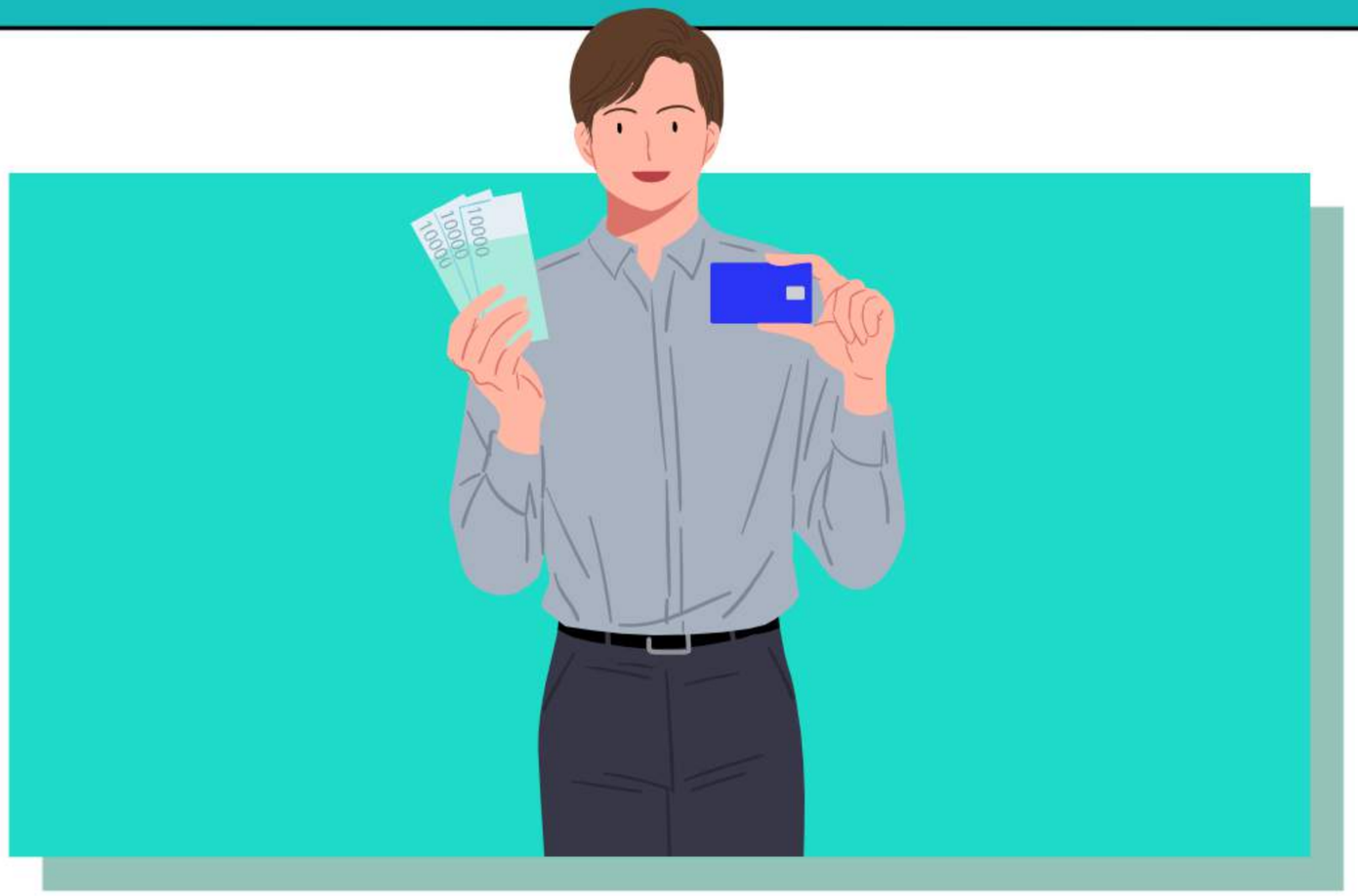
High Cardinality 처리 결론

High cardinality 사용하여 인코딩을 하였을때 logloss는 향상되지 않았다.

✓ **High cardinality를 사용하면 과적합이 큰 폭으로 줄어듬**

#04
MODELING

모델링 학습 및 튜닝



MODELING

Optuna

- 하이퍼파라미터 튜닝에 쓰고 있는 최신 Automl 기법이다.

- 빠르게 튜닝이 가능하다는 장점이 있다.

- 하이퍼파라미터 튜닝 방식을 지정할수 있다.
  → 직관적인 api인 튜닝된 lightgbm도 제공해준다.

- 다른 라이브러리들에 비해 직관적인 장점이 있어
  코딩하기 용이하다.

	LOG LOSS
기본 LightGBM	0.7281
튜닝된 LightGBM	**0.7107**

MODELING

LightGBM

XGBoost의 효율성 문제를 보완하여 나온 알고리즘이며 Gradient Boosting 프레임워크 제공

특 징

① XGBoost보다 더빠른 학습과 예측수행시간

② 메모리사용량이 적음

③ 대용량 데이터에 대한 뛰어난 예측 성능

④ 병렬 컴퓨터 기능을 제공

⑤ GPU까지 지원

⑥ 10,000건 이상의 데이터셋에 적합

Parameter

application

➡ 가장 중요한 파라미터로

모델의 어플리케이션을 정함

EX) regression: 회귀분석 /

binary: 이진 분류 /

multiclass: 다중 분류

```
lgbm_model=LGBMClassifier(objective='multiclass')
```

MODELING

최종 결론

- 우리가 사용하는 data set의 경향을 해치는 것은 좋지않을것이라고 판단.
- train set과같은 평균과 분산의 데이터셋을 만들어 합쳐 좀더 강력한 모델을 만들면 logloss가 상향될 것이라 판단.

그 가정대로 모델링을 한 결과, logloss가 상향되었다.

#05

REFERENCES

REFERENCES

[1] Hale, J. (2020, October 5). Smarter Ways to Encode Categorical Data for Machine Learning. Medium. (https://towardsdatascience.com/smarter-ways-to-encode-categorical-data-for-machine-learning-part-1-of-3-6dca2f71b159)

[2] 하이퍼파라미터 튜닝을 쉽고 빠르게 하는 방법. (n.d.). DACON. Retrieved May 27, 2021, from https://dacon.io/competitions/official/235713/codeshare/2704?page=2&dtype=recent

THANK YOU

신용카드 사용자 연체 예측 AI 경진대회