2022 빅콘테스트

고객 데이터 분석을 통한 대출상품 신청여부 예측 및 고객 군집화 마케팅

행복한 짱구네

팀장
김진호 jinho5913@naver.com

팀원
나요셉 skdytpq98@naver.com
신예주 yejushin2000@gmail.com
윤경서 rudtj0107@naver.com
이상우 woo990410@gmail.com

목차

01.

분석 주제 및 배경

| 핀다(주)에서 제공한 고객 데이터를 활용한 상품신청여부 예측 및 군집화

앱 사용성 데이터를 통한 대출신청 예측분석

- 다른 여러 복합적인 서비스
- 앱 사용이 비교적 복잡
- 덜 직관적인 서비스 제공

타사 서비스

VS

본문 내용을 입력해주세요.

finda

- 대출에 포커스
- 앱 사용이 비교적 쉽고 편함
- 직관적인 서비스 제공
- 대출환승 서비스

핀다에서는 장기 렌트/리스 서비스, 대출 관련 보험 등 조금 더 다양한 분야의 서비스를 제공하고자 함

-> 효과적인 마케팅을 위해서는 각각의 서비스를 고객의 니즈에 맞게 추천

-> 고객의 특성을 살려 비슷한 고객들의 군집화를 통해 분석을 실시

02.

데이터 소개

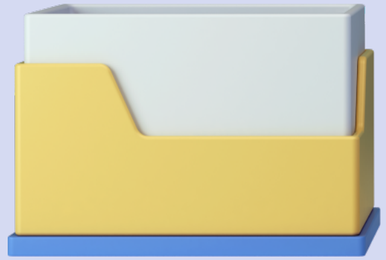
| 핀다(주)에서 제공한 고객 데이터(USER / LOAN / LOG)

User Spec

user의 신용정보를 포함한
데이터로 신청여부를 예측하는데
유의미한 feature가 많이
포함되어 있음

Loan Result

사용자가 신청한
대출별/금융사별 승인결과를
포함한 데이터로 예측레이블을
포함하고 있음



Log Data

finda App의 로그 정보로
후에 유저별 로그 시퀀스를
뽑아내는 과정을 거침

03.

데이터 전처리

| 데이터에 포함되어 있는 결측값, 이상치 등을 처리하는 방법

User_spec 전처리

Data Leakage 문제가 없는 열 처리

- 유저생년월일 & 성별
  - 동일한 유저번호에 값이 존재하면 동일한 값으로 / 단일값은 '기입안함'으로 처리
- 한도조회당시유저신용점수
  - '0'으로 처리_결측값들은 대체로 추천된 상품을 신청하지 않는 것으로 판단되어 값의 의미를 살리기 위해
- 근로형태 & 고용형태 & 주거소유형태 & 대출목적 & 개인회생자여부 & 개인회생자납입완료여부
  - '기입안함' or '누락'으로 처리_단순유저의 행동적 판단요인이 작용했다고 판단
- 입사연월
  - 결측치는 '기입안함'으로 처리 / 날짜 형식이 연-월-일인 경우와 연-월인 경우의 통일성을 위해 연-월인 경우 '01'을 붙임

03.

데이터 전처리

| 데이터에 포함되어 있는 결측값, 이상치 등을 처리하는 방법

Data Leakage 문제가 있는 열 처리

- 생성일시 기준 월 데이터
  - 3, 4, 5월 데이터는 train / 6월 데이터는 test로 split
- 연소득
  - 동일한 유저번호에 값이 존재하면 평균으로 / 단일값은 train, test별 전체 평균으로 처리_추천상품에 대한 만족도가 낮아 입력하지 않은 단순 행동적 요인 이라고 판단
- 대출희망금액
  - 연소득과 같은 방법 및 근거로 처리 / 값이 0이거나 너무 큰 경우 이상치로 탐지되었지만 군집분석을 위해 그대로 둠
- 기대출수 & 기대출금액
  - 동일한 유저번호에 값이 존재하면 평균으로 / 단일값은 '0'으로 처리_신규유저로 판단 / 너무 많거나 큰 경우 사업자금이 목적인 것으로 여겨 이상치로 판단X

03.

데이터 전처리

| 데이터에 포함되어 있는 결측값, 이상치 등을 처리하는 방법

Loan_result 전처리

- 승인금리
  - 금융사 별 상품번호 평균으로 대체_금융사에서 값을 보내주지 않은 경우로 결측치인 행동을 drop해도 되지만, 대출상품을 신청한 경우에 있어서 중요한 역할을 한다고 판단되었기 때문
- 승인한도
  - 승인금리와 같은 방식으로 처리 / 0인 값을 가지는 이상치를 발견했으나 그 이유가 있을것이라고 판단되어 값 보존

04.

Feature Engineering

| 새롭게 생성한 FEATURE에 대한 설명

상품코드

'특정 금융사의 상품이 추천되었을 때
상대적으로 다른 상품들 보다
신청할 확률이 높은 경우가 있다고 판단'
금융사번호 + 상품번호

유저타입

'대출조회가 많을수록 더 많은 상품에 노출되고 그
에 따른 신청수가 많을 것으로 판단'
유저번호별 신청서번호의 고유값들을 count하여
대출조회횟수 생성
-> (대출조회횟수 전체 평균이 2.8회) 3회까지를
소극적 이용자 / 나머지를 적극적 이용자로 정의

상품매력도

'유저에게 여러 상품이 추천되었을 때,
승인한도가 대출희망금액을 만족하면
승인금리가 낮을수록 신청할 가능성이 높다고 판단'
승인한도가 대출금액 이상인 경우
승인금리 값을 그대로 사용 / 미만인 경우
그 차이에 따라 차등 패널티 부여

연이자부담지수

'연소득 대비 이자가 차지하는
퍼센트에 따라 신청여부가 나뉜다고 판단'
->연소득이 0인 경우 값이 무한대
-> 근로형태가 OTHERINCOME인 경우
연이자부담지수의 1분위수로 대체_주식이나
코인과 같은 비노동적 소득이라고 판단(대출상환
능력 충분) / 나머지는 연이자부담지수 3분위수
로 대체(대출상환 능력 부족)

3-5월 평균 순위변동폭

'시계열적인 요소의 일정함을 가정했을 때,
각 상품별 트렌드 추세를 feature로 사용할 시
테스트 데이터 추론에 좋은 영향을 미칠 것이라 판단'
상품코드의 월별 추천건수를 내림차순으로 정렬
-> 3-5월 상품순위를 매긴 후,
순위 변동 폭을 이용해 월 평균을 내어 산출

6월 예상순위

'테스트 데이터 예측성능 향상을 목표로 생성
6월의 순위를 feature로 만들면 신청여부를
결정할 때 좋은 기준점으로 작용할 것으로 판단'
3-5월평균순위변동폭을 만드는 과정에서
5월 순위에 3-5월평균순위변동폭 합산

04.

Feature Engineering

| 새롭게 생성한 FEATURE에 대한 설명

기출등급

'기대출수와 기대출금액을 하나로 묶어서 나타낼 수 있는 feature를 생성하고자 함'

기대출수와 기대출금액을 총 5개의 구간으로 나눈 후, 둘의 합을 다시 5개의 구간으로

나이

'사회생활을 얼마나 했는지의 정도를 입사연월이 아닌 나이를 활용해 대략 나타내보고자 함'

유저생년월일을 이용해 각 유저의 나이
-> 연령대별로 feature 생성
-> 연령대별 대략적인 회사 연차를 나타내는 feature 생성

신청서조회비율

'유저별로 대출상품조회의 수가 대출상품 신청으로 귀결될 확률이 높다고 판단'
(같은 조건의 상품이 추천될 경우 대출상품조회를 많이 한 유저일수록 상품신청확률이 높아진다는 근거) 유저번호별 신청서번호를 count한 후 train, test 별 전체 합계로 나눠서 산출

신용점수증감율

'유저별로 신용점수가 일정하지 않은 경우 신용점수가 높아지거나 낮아지는 시점에 신용거래로 대출상품신청 확률이 변화할 것이라 판단'

유저번호 별 신용점수의
(최대값-최소값)에 로그를 씌워 산출

앱버전_세부내역

'앱버전이 업데이트됨에 따라 새로운 핵심 서비스가 등장하기 때문에 이를 반영하고자 생성'

아이폰 앱스토어 핀다의 버전관리에서 현재버전 및 과거버전까지의 정보를 각각 추가

04.

Feature Engineering

| 새롭게 생성한 FEATURE에 대한 설명

log 데이터로 만든 sequence feature

1. 유저별 신청서 list, log 데이터 안의 유저별 시퀀스 확보

2. 로그 시퀀스와 신청서 번호로 dictionary 생성(log의 time stamp에 따라 맵핑)
: 이 때 로그 시퀀스 길이를 제한하고, 명칭별로 토큰화하는 과정을 거침

3. 해당 로그 시퀀스와 매핑된 신청서 번호가 대출 조회까지 간 신청서인지 판단하기 위해
실제 대출 조회까지 간 데이터를 positive로 두고 GRU 이진분류 모델을 개발 → 92.89%의 정확도

4. Linear layer를 추가하여 각 시퀀스당 10차원의 vector를 추출하여 feature로 사용

04.

UnderSampling

| UnderSampling의 필요성 및 적용 방법

Loan 데이터●의 불균형이 매우 심해
UnderSampling 필요성을 느낌

Loan 데이터 ●
하나의 신청서에 여러 조회 결과가 출력
(신청 여부와 조회상품은 조회결과 별로 상이)

랜덤 UnderSampling 문제점

1. 신청서의 대출 신청 여부가 유저 정보와 관련이 있으므로 랜덤하게 삭제될 경우 유저의 특성/패턴을 파악할 수 없다.
2. label을 고려하지 않을 경우 유저 정보가 편향될 수 있다.

각 유저별
대출조회신청서의
수를 구하여 값이 0인
데이터 개수 파악

>

0으로 labeling 된
신청서가
5개 이상이면 85%
2~4개면 50%
랜덤으로 제거

>

위 과정을 거쳐
걸러진 데이터는 삭제
이후, 남아있는 불균형의
문제는 랜덤
UnderSampling

05.

Modeling

| Model 선정이유 및 성능 / 모델 해석

05.

Modeling

I Model 선정이유 및 성능 / 모델 해석

CAT

CAT 선정이유

연관성

트리 기반 모델로써 각 Tree가 독립적이지 않으며 서로 영향을 주고받는 모델
생성된 Feature 대부분은 서로 완전한 독립성을 띈다고 말할 수 없기 때문에
Feature 간의 상호 작용을 Model 내재적으로 학습하는 것이
유리할 것이라고 가정

범주형 feature

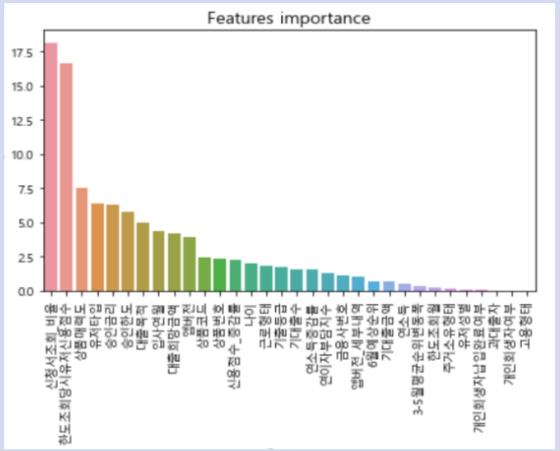
다른 부스팅 기반 모델에 비해 범주형 데이터 처리에 유리한 모델
Encoding과정에서 Data Leakage 최소화를 위해 시계열적인 특성을 판단하여, Encoding하고자하는 데이터의 이전 데이터만을 수집한 뒤 진행

시계열 데이터

시계열 데이터를 효율적으로 처리하는 모델
신청서 각각의 데이터에 경우 시계열적 요소를 무시할 수 없기 때문에 미래 데이터를 참조하여 예측하지 않는 Catboost가 적절한 모델이라 판단

05. Modeling

I Model 선정이유 및 성능 / 모델 해석



신청서조회_비율
한도조회당시유저신용점수
상품매력도
유저타입
승인금리
승인한도
대출목적
입사연월
대출희망금액
앱버전

최대 성능
0.8679

> 유저타입이라는 범주형 변수가
> 상대적으로 높은 중요도를 보임
> 이는 catboost모델의 범주형 변수
> 처리 방법인 Response encoding과
> Categorical Feature Combination
> 의 영향으로 판단

상위 10개 feature

CAT

02

모델 해석

03

Response encoding - Target encoding과 비슷한 범주형 인코딩 방법이지만 시간 순서에 따라 학습데이터 중 일부를 적용하는 방식으로 Target value의 누출을 방지
Categorical Feature Combination - Label 값 기준으로 split할 때 대체가능한 두 변수를 합쳐주어 one-hot-encoding시 spares한 피처생성을 방지

01

모델 파라미터

cat모델 자체인코딩사용
task_type = “GPU”
one_hot_max_size = 4
early_stopping
use_best_model = True

05.

Modeling

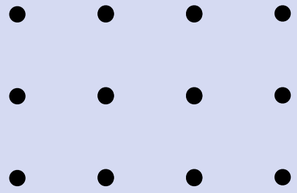
I Model 선정이유 및 성능 / 모델 해석

LGBM

LGBM 선정이유

학습속도

GPU 활용이 가능하며 가볍고
학습속도가 매우 빠른 모델
현재 보유한 데이터의 크기가 매우 크기
때문에 Feature Selection과 데이터
샘플링의 여러 방법론에 대한 반복적인
실험을 위해 기본 Base Model로 지정



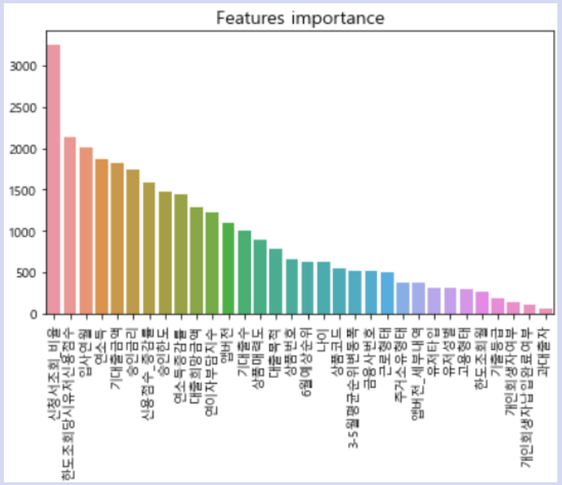
Ordered Vector

Catboost 모델과 마찬가지로 범주형
변수에 대한 Encoding 진행 시,
One-Hot Vector를 생성하는 것이
아닌 Ordered Vector를 생성

05.

Modeling

I Model 선정이유 및 성능 / 모델 해석



신청서조회_비율
한도조회당시유저신용점수
입사연월
연소득
기대출금액
승인금리
신용점수_증감률
승인한도
연소득증감률
대출희망금액

> 상대적으로 범주형 변수들이 아닌 연속형 변수들이 importance 상위권
> 이는 train데이터에 overfitting 방지를 위해 hyperparameter tuning을 최대한 적게한 결과로 판단

최대 성능
0.8610

LGBM

01 모델 파라미터

n_estimators = 400
bagging_fraction=0.3
bagging_freq =100
cat_smooth=0
early stopping

02 상위 10개 feature

03 모델 해석

실제로 n_estimator 파라미터만 변경하여 진행
다른 learning_rate, max_depth 등 train데이터에 overfitting시킬 우려가 있는 파라미터들은 defalut 값으로 적용시켜 모델의 일반화에 중점

05.

Modeling

I Model 선정이유 및 성능 / 모델 해석

RANDOM FOREST

RANDOM FOREST 선정이유

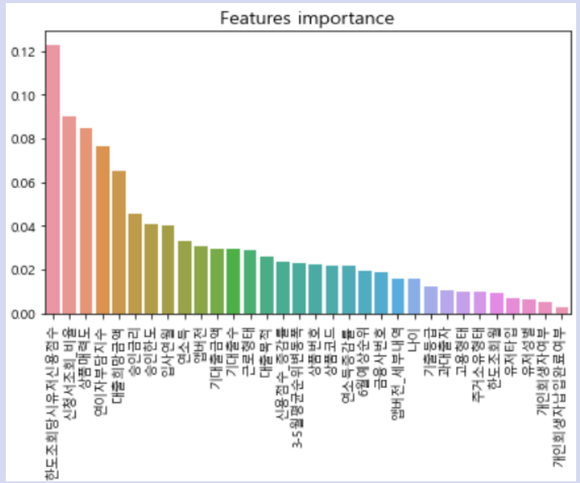
오버피팅 방지

overfitting 문제에 덜 노출된다는
장점이 있는 모델
데이터셋의 특성상 대체적으로
hight cardinality문제가 우려되어
최대한 overfitting을 방지하는데
적합한 모델이라 판단하여 사용

05.

Modeling

I Model 선정이유 및 성능 / 모델 해석



한도조회당시유저신용점수
신청서조회_비율
상품매력도
연이자부담지수
대출희망금액
승인금리
승인한도
입사연월
연소득
앱버전

최대 성능
0.8625

> 상대적으로 범주형 변수들이
> importance 상위권에 존재X
> 이는 catboost와 비교해 범주형 변수들을
> 모델 내부적으로 다루는 과정이 없고 파라
> 미터 튜닝또한 진행하지 않아 범주형 변수
> 들에 대한 중요도가 감소한것으로 판단

RF

01 모델 파라미터

따로 설정X

02 상위 10개 feature

03 모델 해석

05.

Modeling

I Model 선정이유 및 성능 / 모델 해석

DNN

DNN 선정이유

딥러닝 모델

딥러닝 신경망 구조를 활용하여 모델 예측에 비선형성을 가한 알고리즘
햔제 시용하고자 하는 모델은 대부분 선형성을 띄고 있기때문에 앙상블 효과 극대화 및 오버피팅 규제를 위해 비선형성 모델을 추가하고자 함

05.

Modeling

I Model 선정이유 및 성능 / 모델 해석

모델 통합 해석

각 모델의 feature importance를 기준으로 해석해보았을 때,
dnn을 제외한 세개의 모델의 경우 트리계열의 모델이기에 공통적으로
중요한 feature들을 선별해볼 수 있었음
→ 한도조회당시유저신용점수, 신청서조회_비율, 상품매력도,
승인금리, 승인한도 크게 5가지
이를 통해,
유저의 신용정보와 밀접한 관련있는 부분(한도조회당시유저신용점수),
유저의 핀다 앱 사용과 관련된 부분(신청서조회_비율),
추천된 상품이 유저에게 적합한 지와 관련된 부분(상품매력도,승인금리,승인한도)이
유저의 상품신청여부를 예측하는데 중요한 역할을 하고있다고 판단됨.

06.

Clustering

I Clustring 과정에 사용한 기법 및 선정 이유

사용 기법

- K-Means
  - n개의 중심점을 찍은 후 중심점에서 각 점간의 거리의 합이 가장 최소화가 되는 중심점 n의 위치를 찾고, 이 중심점에서 가까운 점들을 중심점을 기준으로 묶는 클러스터링 기법

선정 이유

- 일반적으로 군집화에서 가장 많이 사용하는 알고리즘
- 본 데이터의 shape이 크기 때문에 다른 클러스터링 알고리즘보다 쉽고 간결한 K-means 사용

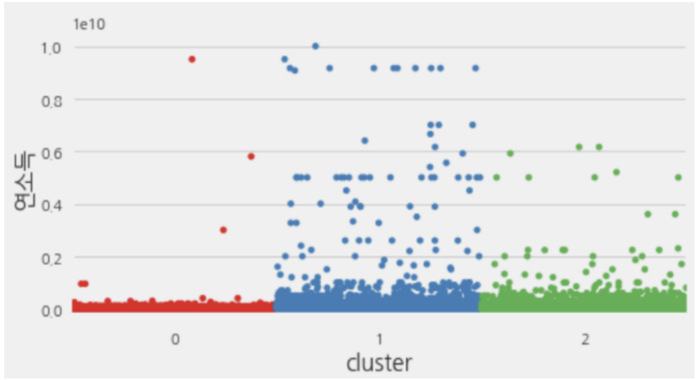
06.

Clustering

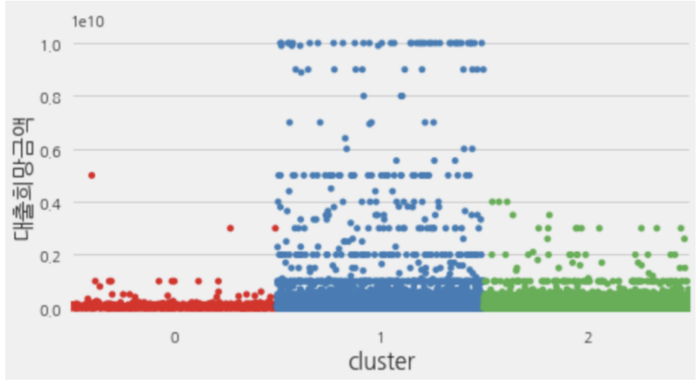
Clustering 진행 후 각 Cluster별 EDA를 통한 결과 해석

연소득 - 희망 대출금액

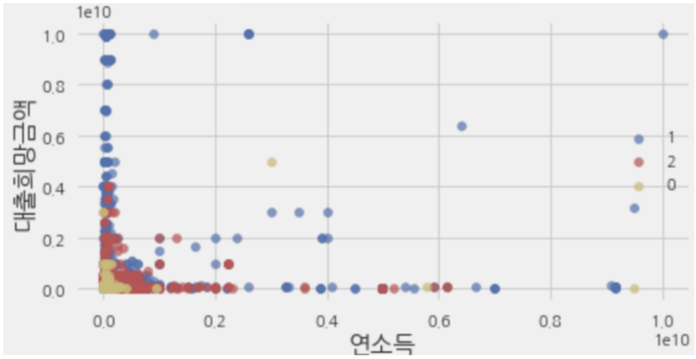
연소득



대출희망금액



연소득 - 대출희망금액



연소득별 대출희망금액

Cluster 0
: 연소득이 평균적으로 낮은 군집이기 때문에 대출 희망금액도 군집 중 평균적으로 낮음

Cluster 1
: 평균 연소득이 가장 높은 집단으로 대출 희망금액도 높음

Cluster 2
: 가장 평균적인 대출희망금액을 띔

Result

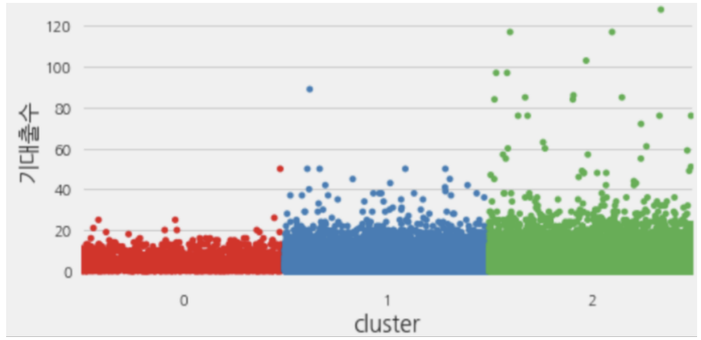
일반적으로 **연소득이 높은** 사람들이
대출 희망금액이 높음을 알 수 있다.

06.

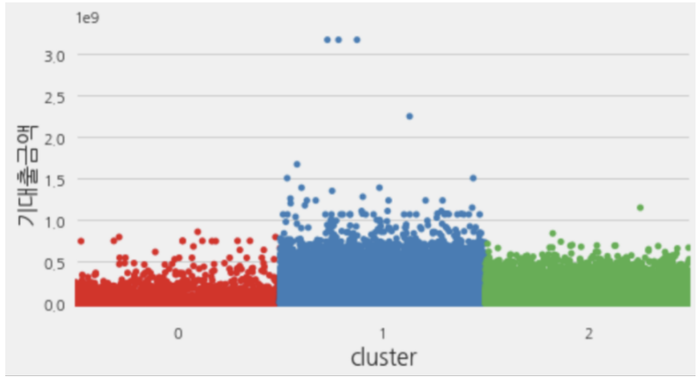
Clustering | Clustering 진행 후 각 Cluster별 EDA를 통한 결과 해석

기대출수 - 기대출금액

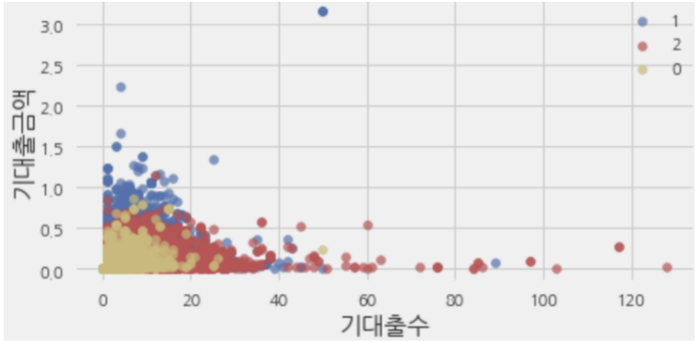
기대출수



기대출금액



기대출수 - 기대출금액



기대출수별 기대출금액

Cluster 0

: 연소득이 낮아 상환할 수 있는 대출액이 적기 때문에 기대출금액이 많이 낮음

Cluster 1

: 연소득이 가장 높은 군집으로 기대출액이 많음

Cluster 2

: 평균적인 군집으로 지표들이 다 평균값을 가짐을 알 수 있음

Result

연소득이 높은 Cluster 1은 상환가능 금액이 상대적으로 높기 때문에 타 군집보다 기대출금액이 높음

› 대출액이 크기 때문에 VIP 고객으로 전환하여 적절한 서비스 제공 가능

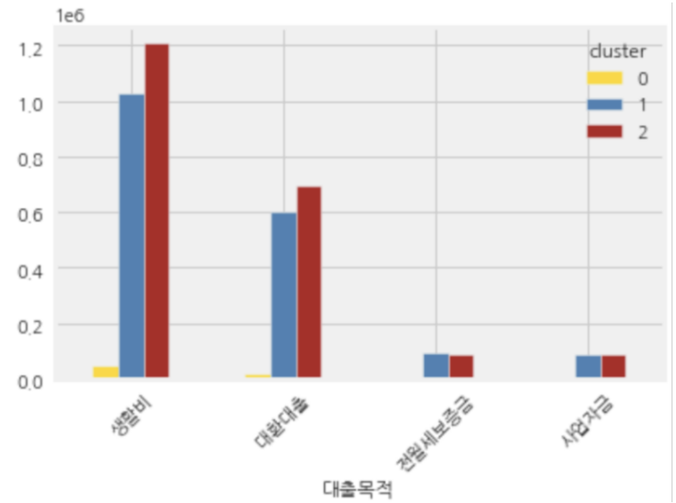
평균 연소득을 가진 Cluster 2는 타 군집보다 기대출수가 많은 것으로 보아 대출신청비율을 높음을 짐작할 수 있음

대출금액은 작지만 앱을 자주 이용하는 충신고객으로 분류 가능

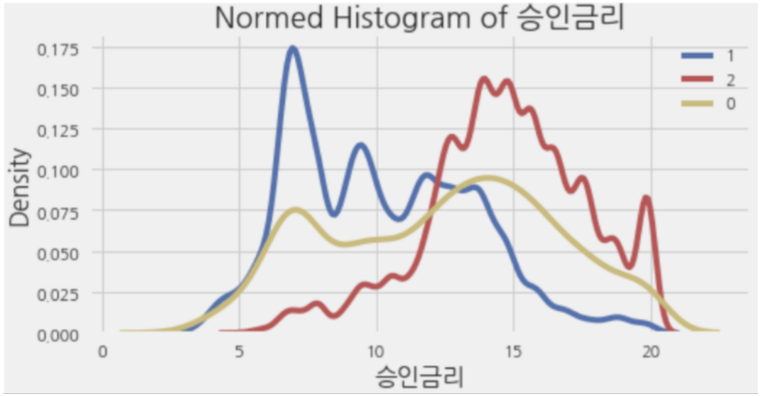
06.

Clustering | Clustering 진행 후 각 Cluster별 EDA를 통한 결과 해석

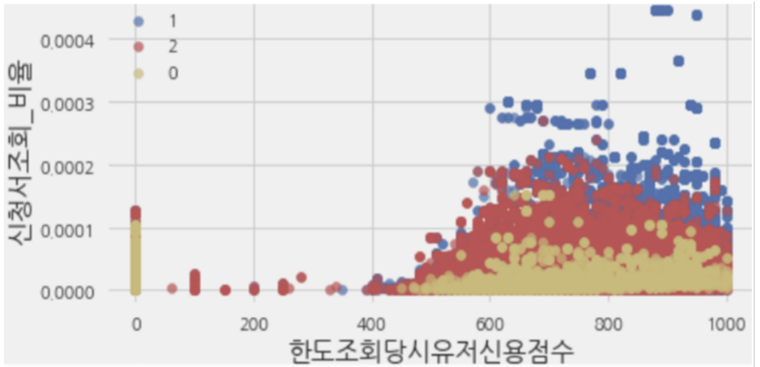
대출목적



승인금리



신청서조회비율



Result

일반적으로 생활비, 대환대출이 높은 비율을 차지하고 있음.
평균 연소득을 가지는 Cluster 2이 두 대출에 대해 높은 비율을 차지 함.
전월세보증금, 사업자금 같은 경우 연소득이 높은 Cluster 1이 앞 선 2개의 항목보다 더 높은 비율을 차지함.

Result

Cluster 2의 경우 생활비대출, 대환대출 비율이 높기 때문에 상대적으로 대출 상품 금리가 높다
› 보편적으로 생활비대출, 대환대출 금리가 상대적으로 높음
› 생활비 대출의 경우 크지 않은 금액을 대출할 경우 금리를 크게 생각하지 않고 대출을 신청하는 경우도 존재하기 때문이라고 판단
Cluster 1의 경우 연소득이 높은 집단이지만 금리가 낮은 상품을 신청
› 대출금액이 큰 경우 상대적으로 금리가 낮은 경향을 가지기 때문이라고 판단

Result

군집을 불문하고 신용점수가 높은 사람이 신청서 조회를 많이 하는 경향
›신용점수가 높은 사람들에게 해당 군집의 평균 대출금리보다 낮은 대출 상품을 제공한다면 높은 신청률 기대
›채권자 입장에서도 신용점수가 높은 사람에게 대출 승인 하는것을 선호

07.

서비스 제안

| 분석결과를 바탕으로 효과적인 서비스 방안 제시

01. CLUSTER 0의 경우 평균이하의 연소득을 가지며 이에 따라 기대출금액도 평균이하

대출금리에 민감하지 않고 생활비 대출에 비중이 큰 군집이기 때문에 금리가 낮고 이상적인 대출 상품보단 고객이 수용 가능한 금리로 꾸준히 대출 가능할 수 있는 상품을 제공한다면 중저액 대출 장기고객으로 유치할 수 있음

02. CLUSTER 1의 경우 평균 이상의 연소득을 가지며 기대출금액과 기대출횟수도 평균이상

이들은 본인들의 특정 상품에 대한 대출 가능 여부보단, 대출 상품의 금액이 얼마인지, 승인금리가 어떻게되는지에 예민하다. 특히 타 군집에 비해 사업자금, 전월세보증금 대출 비율이 더 큼. 따라서 해당 고객의 연소득, 대출목적을 파악해서 꾸준한 장기고객 보단, 한번 한번에 큰 금액의 대출을 할 수 있는 상품을 제공하는 것이 유리

03. CLUSTER 2의 경우 평균 정도의 연소득을 가지고 있음

대출목적 또한 생활비대출, 대환대출의 비중이 크고, 승인금리에 예민한 군집이기 때문에 이탈 고객이 많아질 수 있다. 따라서 분기 및 반기간 마케팅으로 고객의 지속적인 관심을 이끔

THANK YOU!