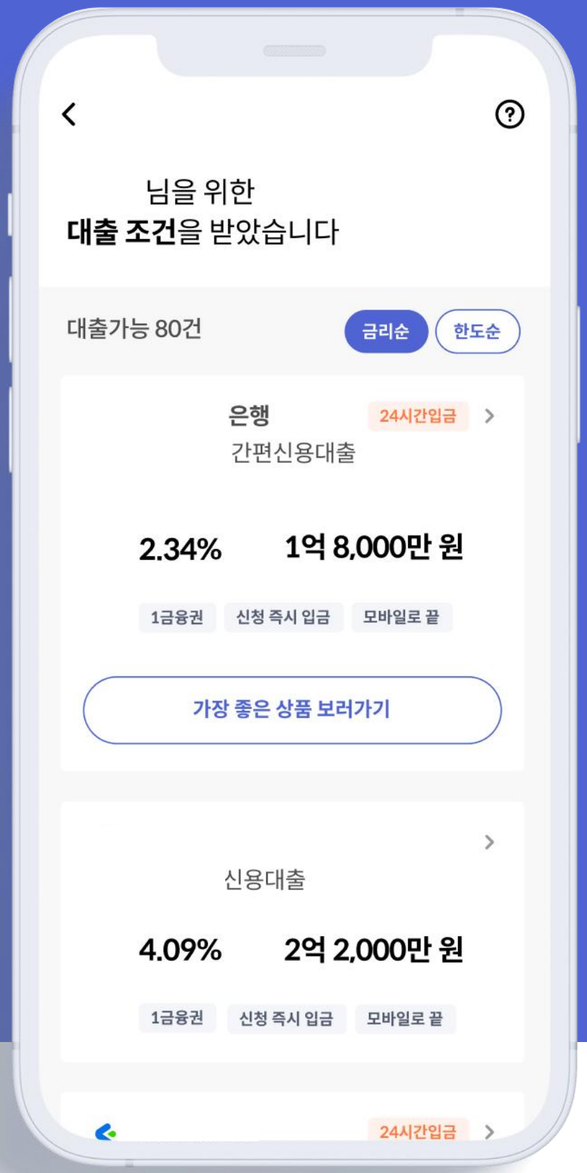


2022 빅콘테스트 퓨처스리그

앱 사용성 데이터를 통한 대출신청 예측분석

팀명 : 개짖는소리

이승준 : lsj7087@naver.com (팀장)
배정민 : vz0502@naver.com
이한재 : 09eoen@gmail.com
조용민 : evbf@naver.com

개요

분석 배경

핀다는 개인을 위한 금융상품 추천 프로덕트 및 서비스를 만들어나가는 국내 대표 핀테크 스타트업입니다.

대출, 예적금과 같은 은행 상품에서부터 신용카드, 보험 및 P2P 투자상품까지 개인 금융생활에 필수적인 다양한 금융상품의 정보를 객관적으로 제공하고, 인공지능 챗봇을 통한 데이터 기반 맞춤 추천 등을 개발하여, 개인 금융시장에서 고도화된 빅데이터 기술과 최적화된 사용자 경험을 제공하고 있습니다.

개요

분석 배경

누적 다운로드 수가 증가하면서 핀다를 이용한 대출 신청이 많아졌다.

이 대용량 데이터를 활용하기 위하여,
고객의 대출신청에 영향을 끼치는 요인을 알아보고자 한다.

추가적으로 고객의 군집을 세분화하고, 적절한 메시지를 추천함으로써

최적화된 사용자 경험을 제공하려한다.

2022년 6월 기준
핀다 누적 다운로드 200만 돌파

개요

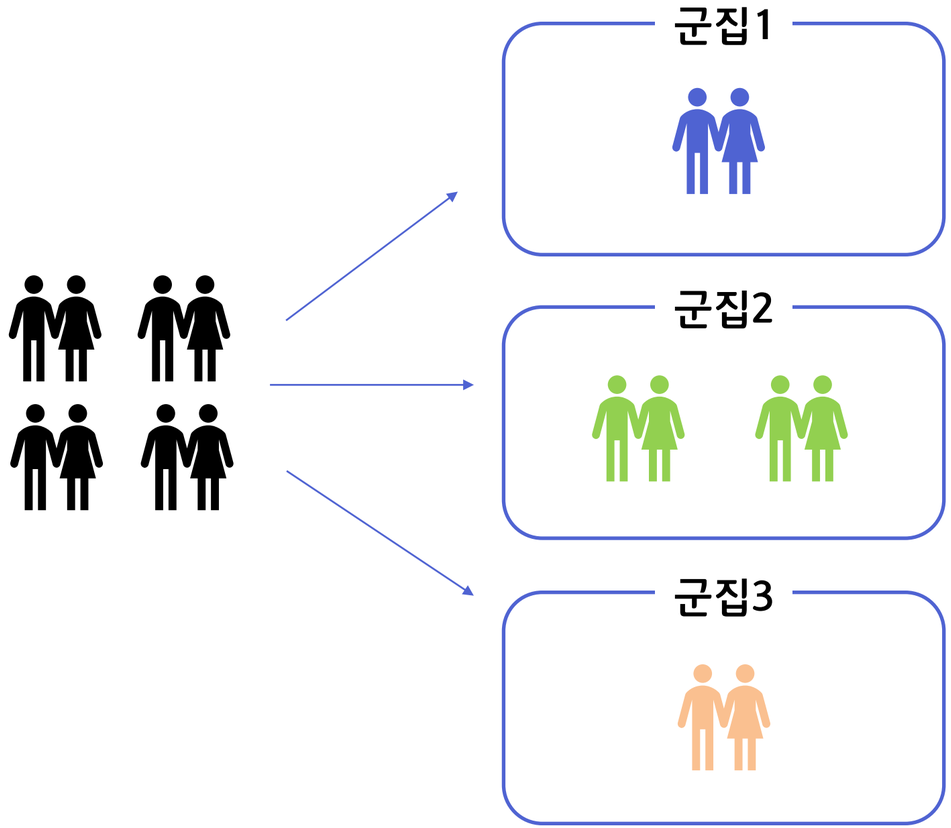
분석 과제

1. 핀다 홈화면 진입 고객 중 특정기간 안에 대출신청 고객 예측
(2022년 3~5월 데이터제공 / 2022년 6월 예측)

	application_id	loanapply_insert_time	bank_id	product_id	loan_limit	loan_rate	is_applied
13284	2157865	2022-05-09 08:44:59	54	235	20000000.0	16.5	1.0
13285	576643	2022-05-09 10:54:53	54	235	11000000.0	16.5	0.0
13286	576643	2022-05-09 10:54:53	11	118	3000000.0	20.0	0.0
13287	2136706	2022-05-09 10:41:06	42	216	10000000.0	13.5	0.0
13288	2136706	2022-05-09 10:41:07	25	169	22000000.0	15.9	0.0

	application_id	loanapply_insert_time	bank_id	product_id	loan_limit	loan_rate	is_applied
0	1748340	2022-06-07 13:05:41	7	191	42000000.0	13.6	NaN
1	1748340	2022-06-07 13:05:41	25	169	24000000.0	17.9	NaN

2. 핀다 홈화면 진입 고객의 모델 기반 고객 군집 분석



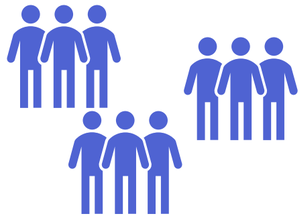
목차

데이터 시각화 및 전처리

- 데이터 설명 및 시각화
- 결측치 처리

모델링

- 모델링 구조 설명
- 1차 모델링 및 성능 평가
- 2차 모델링 및 성능 평가



군집화

- 군집 개요
- 군집 별 특성 파악
- 추천 메시지 제안

결론

- 예측 모델 해석 및 활용 방안
- 군집 결과 해석 및 활용 방안

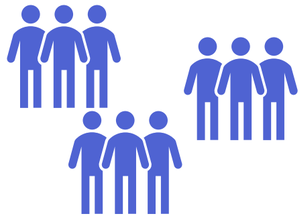
데이터 시각화 및 전처리

데이터 시각화 및 전처리

- 데이터 설명 및 시각화
- 결측치 처리

모델링

- 모델링 구조 설명
- 1차 모델링 및 성능 평가
- 2차 모델링 및 성능 평가



군집화

- 군집 개요
- 군집 별 특성 파악
- 추천 메시지 제안

결론

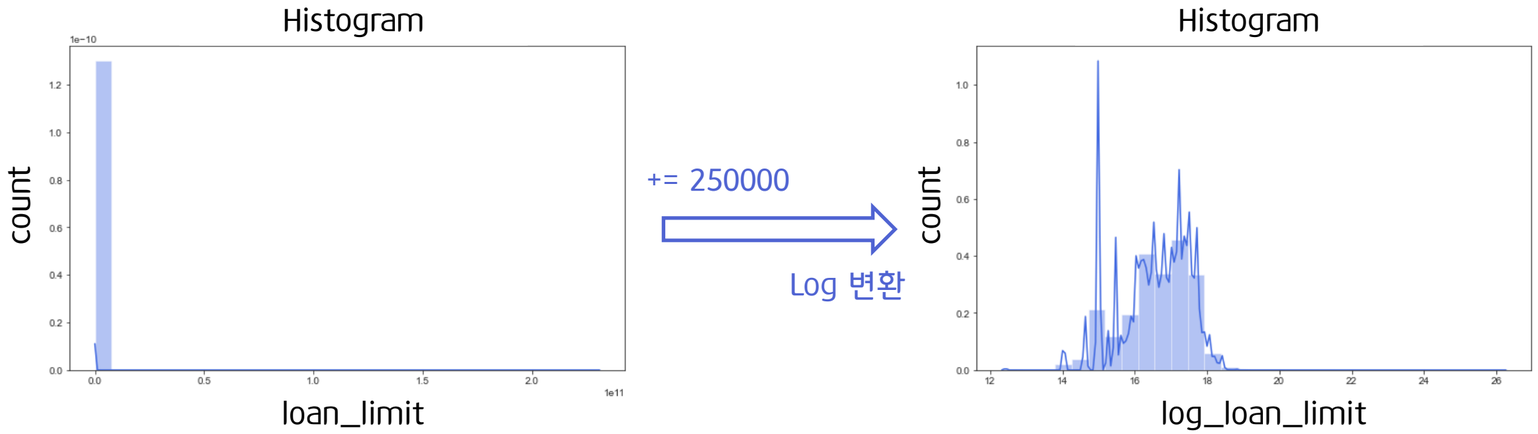
- 예측 모델 해석 및 활용 방안
- 군집 결과 해석 및 활용 방안

데이터 시각화 및 전처리

데이터 설명 및 시각화

loan_result 시각화 user_spec 시각화 log_data 시각화

No.	컬럼ID	컬럼명
1	application_id	신청서 번호
2	loanapply_insert_time	한도조회 일시
3	bank_id	금융사 번호
4	product_id	상품 번호
5	loan_limit	승인한도
6	loan_rate	승인금리
7	is_applied	신청 여부(Target)



loan_limit 의 값이 0인 행들이 존재함 -> 반올림 과정에서 0이상 500,000 미만의 값들이 0으로 변환됨

따라서, 전체 행에 대하여 250,000 을 더하여 0을 없애줌.

loan_limit 변수가 좌측으로 편향되어 있기 때문에 Log 변환

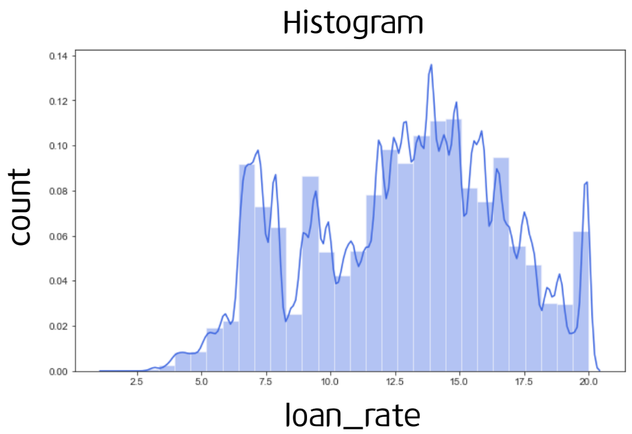
사용자가 신청한 대출별 금융사별 승인결과 <loan_result>

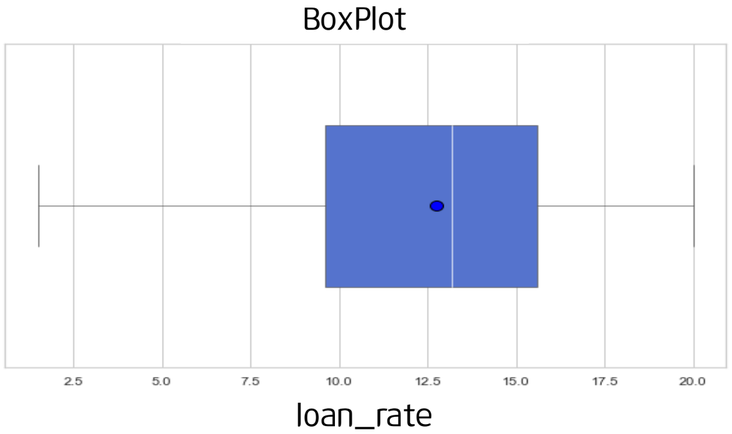
데이터 시각화 및 전처리

데이터 설명 및 시각화

loan_result 시각화 user_spec 시각화 log_data 시각화

No.	컬럼ID	컬럼명
1	application_id	신청서 번호
2	loanapply_insert_time	한도조회 일시
3	bank_id	금융사 번호
4	product_id	상품 번호
5	loan_limit	승인한도
6	loan_rate	승인금리
7	is_applied	신청 여부(Target)





최저 금리는 1.5, 최대 금리는 20, 평균 금리는 12.75

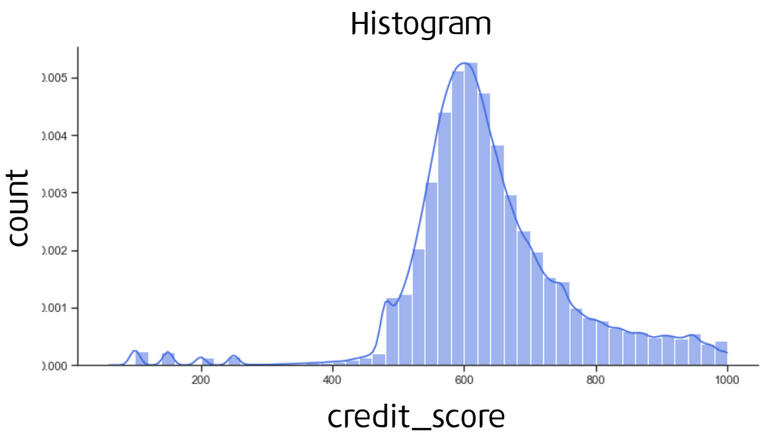
사용자가 신청한 대출별 금융사별 승인결과 <loan_result>

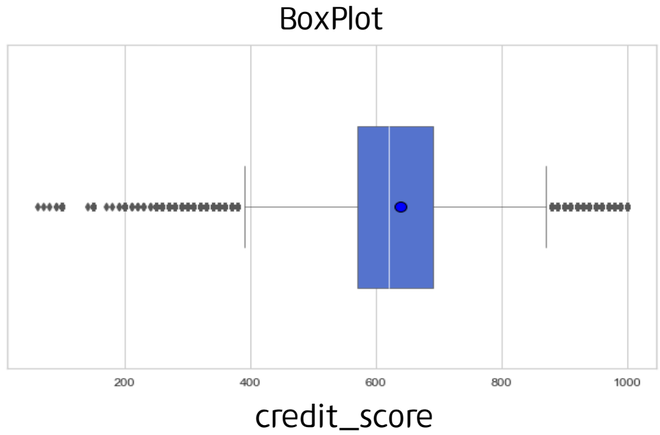
데이터 시각화 및 전처리

데이터 설명 및 시각화

loan_result 시각화 user_spec 시각화 log_data 시각화

No.	컬럼ID	컬럼명
1	application_id	신청서 번호
2	user_id	유저 번호
3	birth_year	유저 출생년도
4	gender	유저 성별
5	insert_time	생성일시
6	credit_score	한도조회 당시 유저 신용점수
7	yearly_income	연소득
8	income_type	근로형태
9	company_enter_month	입사연월
10	employment_type	고용형태
11	houseown_type	주거소유형태
12	desired_amount	대출희망금액
13	purpose	대출 목적
14	personal_rehabilitation_yn	개인회생자 여부
15	personal_rehabilitation_complete_yn	개인회생자 납입 완료 여부
16	existing_loan_cnt	기대출수
17	existing_loan_amt	기대출금액





credit_score 결측인 행 : 105115

Histogram이 종모양을 따름.

(Q3-Q1)*1.5 를 벗어나는 값들이 많음.

user 신용정보 <user_spec>

데이터 시각화 및 전처리

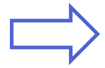
데이터 설명 및 시각화

No.	컬럼ID	컬럼명
1	application_id	신청서 번호
2	user_id	유저 번호
3	birth_year	유저 출생년도
4	gender	유저 성별
5	insert_time	생성일시
6	credit_score	한도조회 당시 유저 신용점수
7	yearly_income	연소득
8	income_type	근로형태
9	company_enter_month	입사연월
10	employment_type	고용형태
11	houseown_type	주거소유형태
12	desired_amount	대출희망금액
13	purpose	대출 목적
14	personal_rehabilitation_yn	개인회생자 여부
15	personal_rehabilitation_complete_yn	개인회생자 납입 완료 여부
16	existing_loan_cnt	기대출수
17	existing_loan_amt	기대출금액

"연월" 형식도 있고 "연월일" 형식도 존재

모두 "연월" 형식으로 바꿔줌.

	company_enter_month
0	20151101.0
1	20070201.0
2	20210901.0
3	20170101.0
4	20210901.0
...	
1394211	202106.0
1394212	NaN
1394213	200908.0
1394214	201705.0
1394215	201103.0



	company_enter_month
0	2015-11-01
1	2021-09-01
2	2021-09-01
3	2007-02-01
4	2021-09-01
...	
1394211	2021-06-01
1394212	NaT
1394213	2009-08-01
1394214	2017-05-01
1394215	2011-03-01

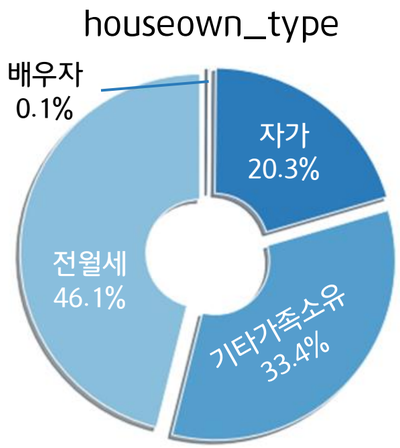
user 신용정보 <user_spec>

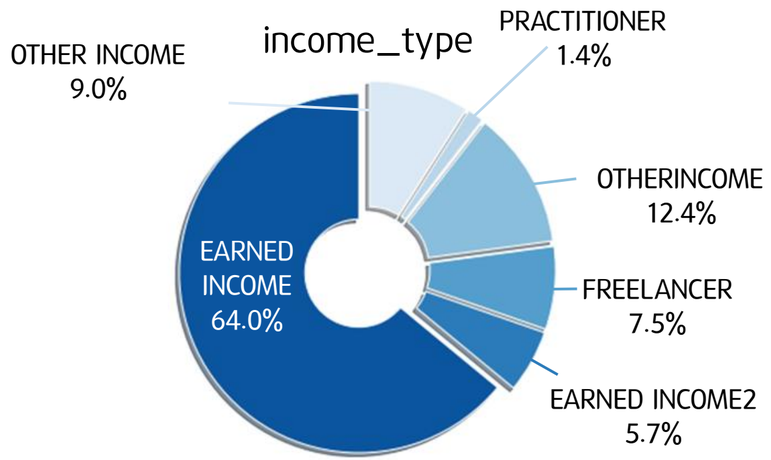
데이터 시각화 및 전처리

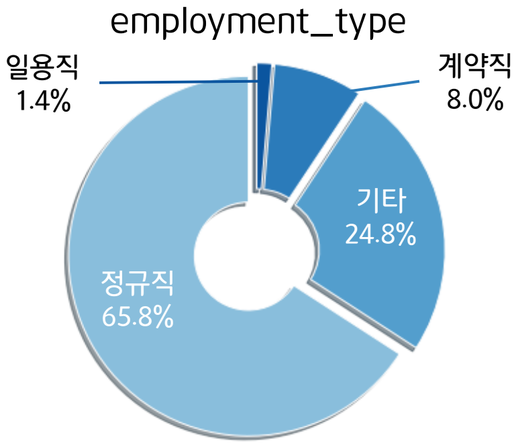
데이터 설명 및 시각화

loan_result 시각화 user_spec 시각화 log_data 시각화

No.	컬럼ID	컬럼명
1	application_id	신청서 번호
2	user_id	유저 번호
3	birth_year	유저 출생년도
4	gender	유저 성별
5	insert_time	생성일시
6	credit_score	한도조회 당시 유저 신용점수
7	yearly_income	연소득
8	income_type	근로형태
9	company_enter_month	입사연월
10	employment_type	고용형태
11	houseown_type	주거소유형태
12	desired_amount	대출희망금액
13	purpose	대출 목적
14	personal_rehabilitation_yn	개인회생자 여부
15	personal_rehabilitation_complete_yn	개인회생자 납입 완료 여부
16	existing_loan_cnt	기대출수
17	existing_loan_amt	기대출금액







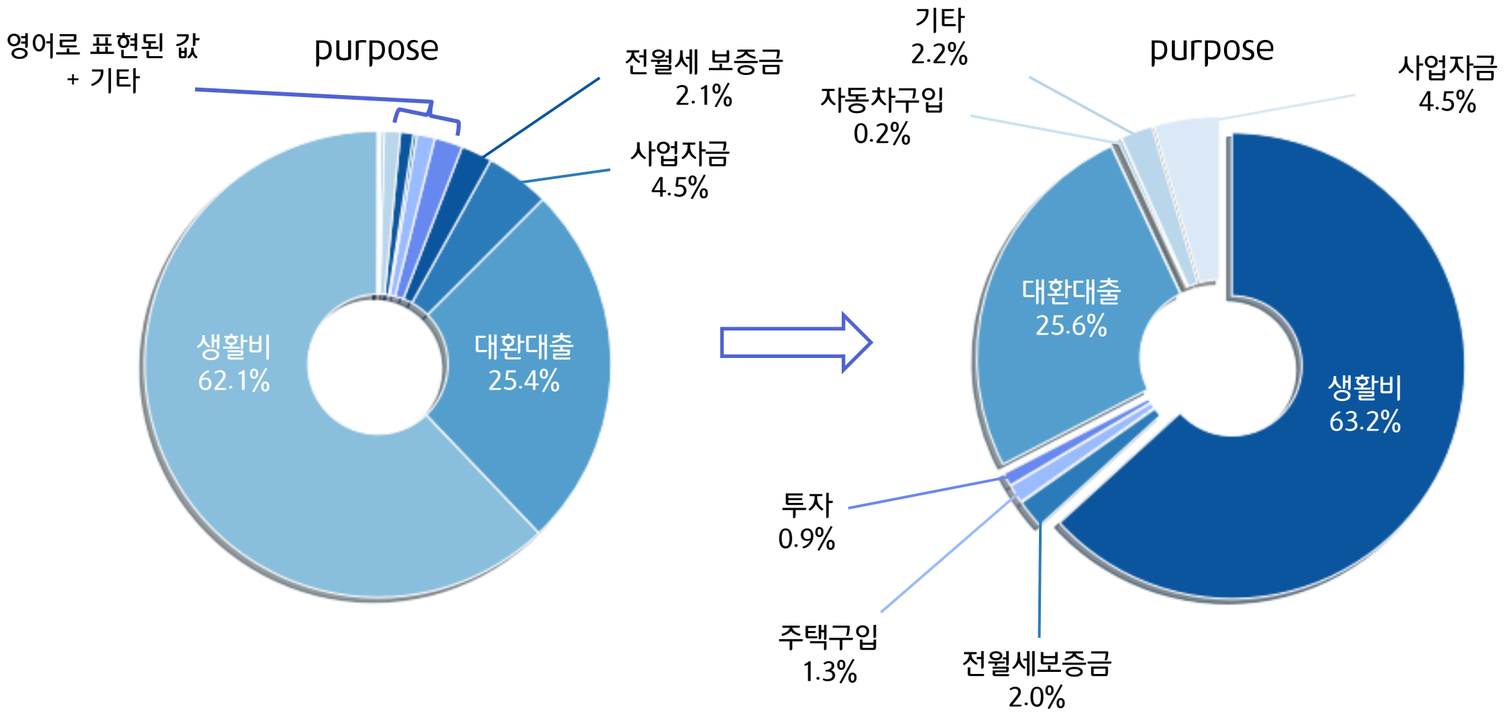
user 신용정보 <user_spec>

데이터 시각화 및 전처리

데이터 설명 및 시각화

loan_result 시각화 user_spec 시각화 log_data 시각화

No.	컬럼ID	컬럼명
1	application_id	신청서 번호
2	user_id	유저 번호
3	birth_year	유저 출생년도
4	gender	유저 성별
5	insert_time	생성일시
6	credit_score	한도조회 당시 유저 신용점수
7	yearly_income	연소득
8	income_type	근로형태
9	company_enter_month	입사연월
10	employment_type	고용형태
11	houseown_type	주거소유형태
12	desired_amount	대출희망금액
13	purpose	대출 목적
14	personal_rehabilitation_yn	개인회생자 여부
15	personal_rehabilitation_complete_yn	개인회생자 납입 완료 여부
16	existing_loan_cnt	기대출수
17	existing_loan_amt	기대출금액



대출 목적 변수에 영어와 한글이 혼용되어서 한글로 통일

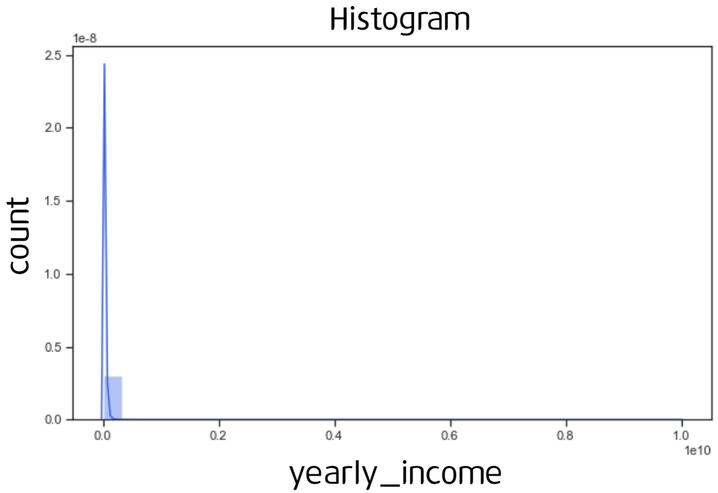
user 신용정보 <user_spec>

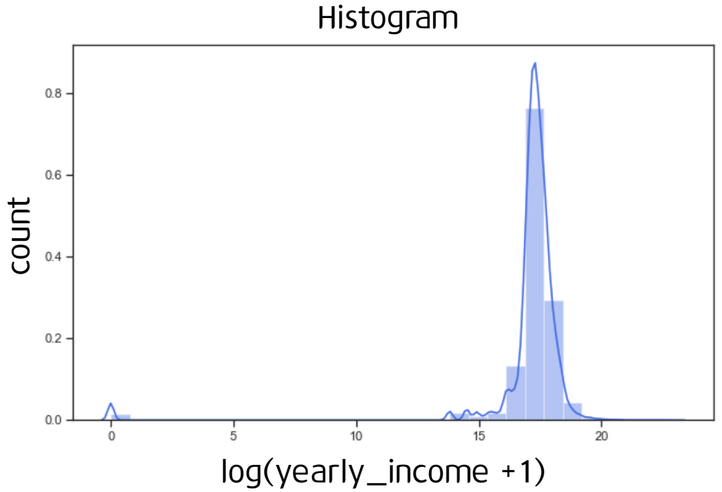
데이터 시각화 및 전처리

데이터 설명 및 시각화

loan_result 시각화 user_spec 시각화 log_data 시각화

No.	컬럼ID	컬럼명
1	application_id	신청서 번호
2	user_id	유저 번호
3	birth_year	유저 출생년도
4	gender	유저 성별
5	insert_time	생성일시
6	credit_score	한도조회 당시 유저 신용점수
7	yearly_income	연소득
8	income_type	근로형태
9	company_enter_month	입사연월
10	employment_type	고용형태
11	houseown_type	주거소유형태
12	desired_amount	대출희망금액
13	purpose	대출 목적
14	personal_rehabilitation_yn	개인회생자 여부
15	personal_rehabilitation_complete_yn	개인회생자 납입 완료 여부
16	existing_loan_cnt	기대출수
17	existing_loan_amt	기대출금액





year_income 도 좌측에 편향되어 있기 때문에, 로그 변환

0인 경우도 존재하기에 1을 더해줌

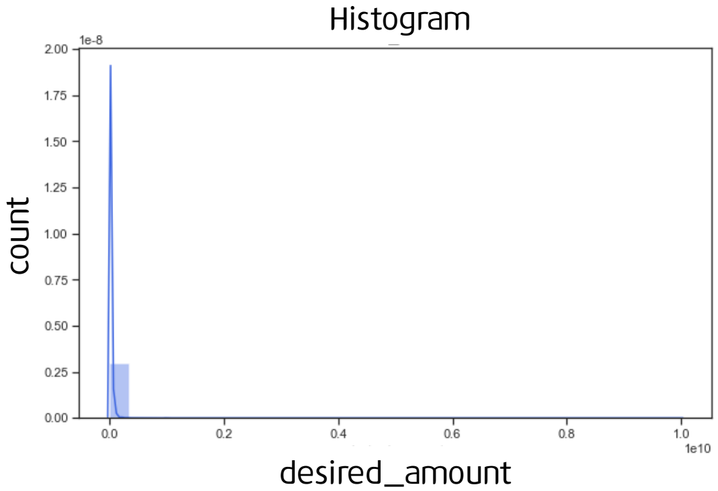
user 신용정보 <user_spec>

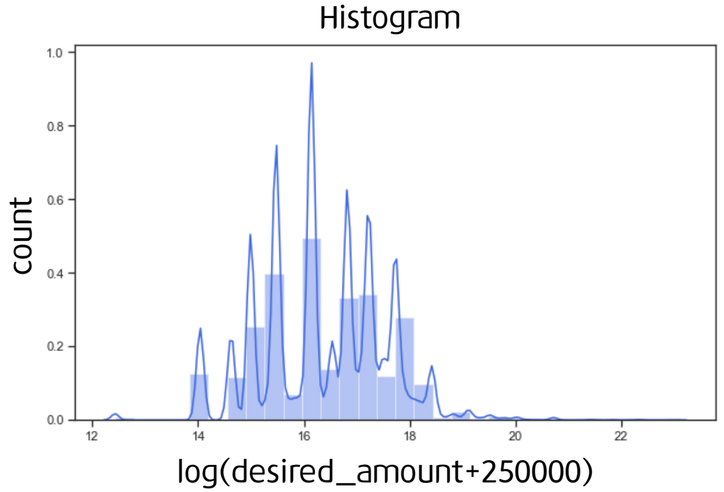
데이터 시각화 및 전처리

데이터 설명 및 시각화

loan_result 시각화 user_spec 시각화 log_data 시각화

No.	컬럼ID	컬럼명
1	application_id	신청서 번호
2	user_id	유저 번호
3	birth_year	유저 출생년도
4	gender	유저 성별
5	insert_time	생성일시
6	credit_score	한도조회 당시 유저 신용점수
7	yearly_income	연소득
8	income_type	근로형태
9	company_enter_month	입사연월
10	employment_type	고용형태
11	houseown_type	주거소유형태
12	desired_amount	대출희망금액
13	purpose	대출 목적
14	personal_rehabilitation_yn	개인회생자 여부
15	personal_rehabilitation_complete_yn	개인회생자 납입 완료 여부
16	existing_loan_cnt	기대출수
17	existing_loan_amt	기대출금액





desired_amount 도 좌측에 편향되어 있기 때문에, 로그 변환

0인 경우도 존재, 100만 단위로 반올림하는 과정에서 사라졌다고 판단.

전체 행에 대하여 250000 더해줌

user 신용정보 <user_spec>

데이터 시각화 및 전처리

데이터 설명 및 시각화

loan_result 시각화 user_spec 시각화 log_data 시각화

No.	컬럼ID	컬럼명
1	user_id	유저 번호
2	event	행동명
3	timestamp	행동일시
4	date_cd	일 코드

	user_id	event	timestamp	mp_os	mp_app_version	date_cd
0	576409	StartLoanApply	2022-03-25 11:12:09	Android	3.8.2	2022-03-25
1	576409	ViewLoanApplyIntro	2022-03-25 11:12:09	Android	3.8.2	2022-03-25
2	72878	EndLoanApply	2022-03-25 11:14:44	Android	3.8.4	2022-03-25
3	645317	OpenApp	2022-03-25 11:15:09	iOS	3.6.1	2022-03-25
4	645317	UseLoanManage	2022-03-25 11:15:11	iOS	3.6.1	2022-03-25

finda App 로그 정보 <log_data>

데이터 시각화 및 전처리

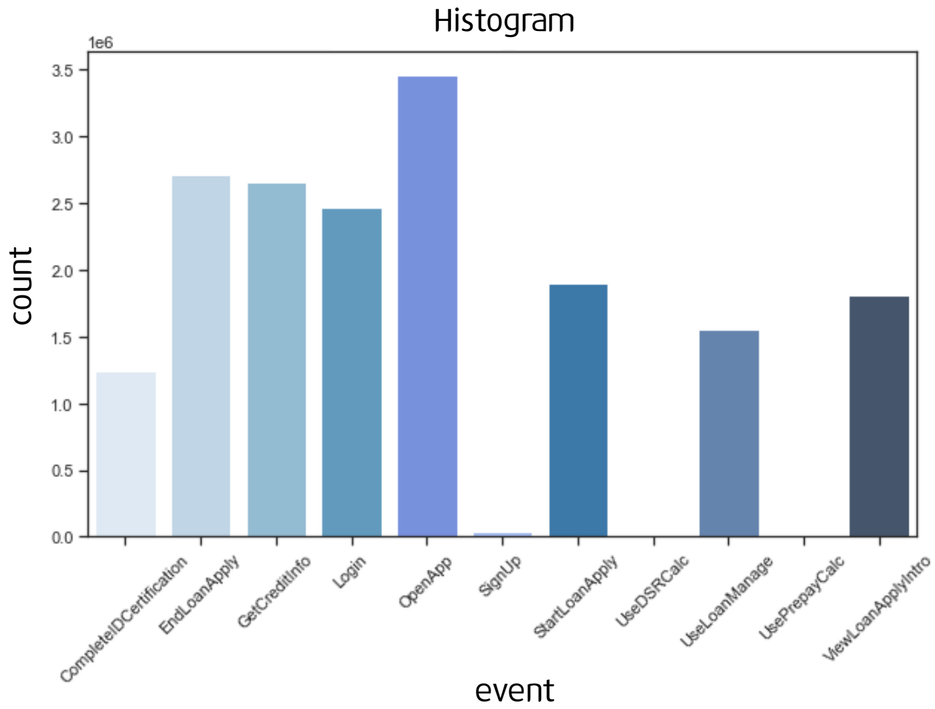
데이터 설명 및 시각화

loan_result 시각화 user_spec 시각화 log_data 시각화

No.	컬럼ID	컬럼명
1	user_id	유저 번호
2	event	행동명
3	timestamp	행동일시
4	date_cd	일 코드

대출 조회 관련 log들의 빈도가 많고

상대적으로 DSR 계산기, 여윳돈 계산기, 회원가입의 빈도가 낮음



finda App 로그 정보 <log_data>

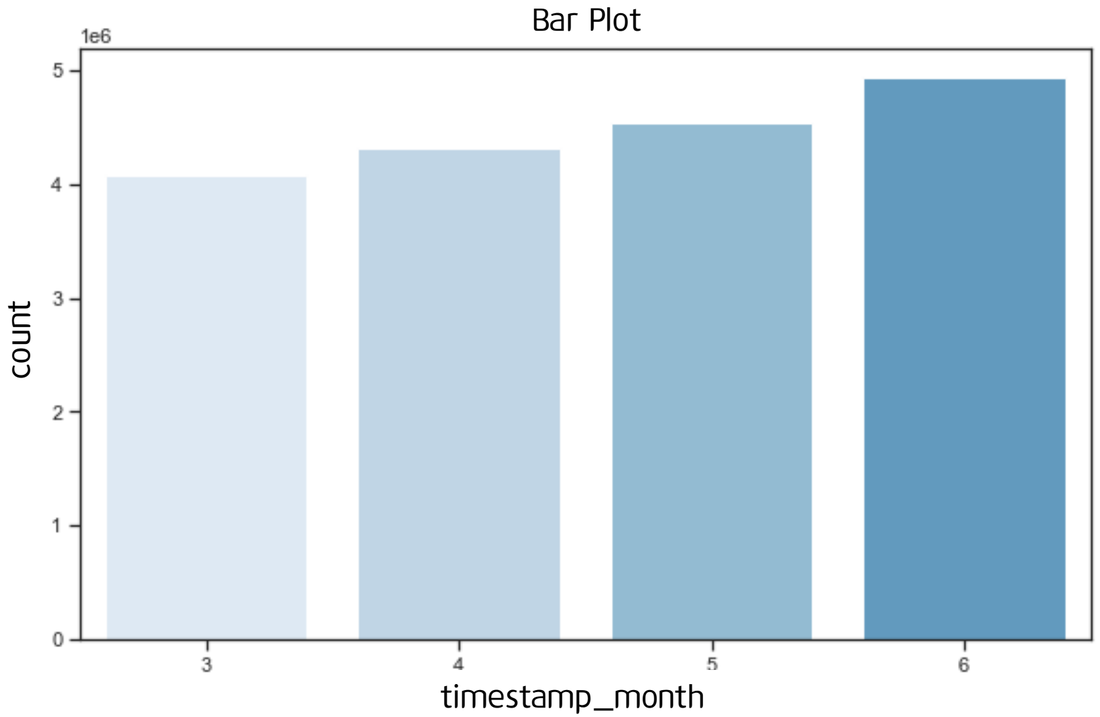
데이터 시각화 및 전처리

데이터 설명 및 시각화

No.	컬럼ID	컬럼명
1	user_id	유저 번호
2	event	행동명
3	timestamp	행동일시
4	date_cd	일 코드

3월의 log가 가장 적음.

시간에 따라 점점 늘어나며, 6월에 가장 많음.



finda App 로그 정보 <log_data>

데이터 시각화 및 전처리

결측치 처리

승인 한도, 금리 | 출생년도 및 성별 | 개인 입력 정보 | 입사연월 | 개인회생 정보 | 기대출 정보 | 신용점수 | 정리

No.	컬럼ID	컬럼명
1	application_id	신청서 번호
2	loanapply_insert_time	한도조회 일시
3	bank_id	금융사 번호
4	product_id	상품 번호
5	loan_limit	승인한도
6	loan_rate	승인금리
7	is_applied	신청 여부(Target)

7. loan_result.csv에서 loan_rate, loan_limit 컬럼에 결측치(nan)가 있는 경우는 무엇인가요?
- 금융사에서 값을 보내주지 않은 경우로, 해당 경우는 채점에서도 제외할 예정으니 무시하고 진행하시면 됩니다.

결측인 행은 전부 삭제 / 데이터분석리그 퓨처스부문 문제 및 데이터 자주 묻는 질문 (ver. 9/20) 7번

데이터 시각화 및 전처리

결측치 처리

No.	컬럼ID	컬럼명
1	application_id	신청서 번호
2	user_id	유저 번호
3	birth_year	유저 출생년도
4	gender	유저 성별
5	insert_time	생성일시
6	credit_score	한도조회 당시 유저 신용점수
7	yearly_income	연소득
8	income_type	근로형태
9	company_enter_month	입사연월
10	employment_type	고용형태
11	houseown_type	주거소유형태
12	desired_amount	대출희망금액
13	purpose	대출 목적
14	personal_rehabilitation_yn	개인회생자 여부
15	personal_rehabilitation_complete_yn	개인회생자 납입 완료 여부
16	existing_loan_cnt	기대출수
17	existing_loan_amt	기대출금액

1. 같은 user_id 인 다른 application_id를 이용하여 대체

	application_id	user_id	birth_year	gender
253	132643	49072	1985.0 NaN	0.0 NaN
266825	484786	49072	1985.0	0.0
266825	484786	49072	1985.0	0.0

결측인 행 수 : 12961개 -> 9724개

데이터 시각화 및 전처리

No.	컬럼ID	컬럼명
1	application_id	신청서 번호
2	user_id	유저 번호
3	birth_year	유저 출생년도
4	gender	유저 성별
5	insert_time	생성일시
6	credit_score	한도조회 당시 유저 신용점수
7	yearly_income	연소득
8	income_type	근로형태
9	company_enter_month	입사연월
10	employment_type	고용형태
11	houseown_type	주거소유형태
12	desired_amount	대출희망금액
13	purpose	대출 목적
14	personal_rehabilitation_yn	개인회생자 여부
15	personal_rehabilitation_complete_yn	개인회생자 납입 완료 여부
16	existing_loan_cnt	기대출수
17	existing_loan_amt	기대출금액

2. 대체할 수 없는 출생년도와 성별은 대체보단 범주화

만나이 = 2022 - 출생년도

cat_age : “청소년” “청년” “중년” “장년” “노년” “Missing_age”

삭제
법적 대출을 못 받는 청소년이며, loan_result에 존재하지 않음

cat_gender : “남성” “여성” “Missing_gender”

데이터 시각화 및 전처리

결측치 처리

No.	컬럼ID	컬럼명
1	application_id	신청서 번호
2	user_id	유저 번호
3	birth_year	유저 출생년도
4	gender	유저 성별
5	insert_time	생성일시
6	credit_score	한도조회 당시 유저 신용점수
7	yearly_income	연소득
8	income_type	근로형태
9	company_enter_month	입사연월
10	employment_type	고용형태
11	houseown_type	주거소유형태
12	desired_amount	대출희망금액
13	purpose	대출 목적
14	personal_rehabilitation_yn	개인회생자 여부
15	personal_rehabilitation_complete_yn	개인회생자 납입 완료 여부
16	existing_loan_cnt	기대출수
17	existing_loan_amt	기대출금액

6개 변수 모두 결측인 행 85개, loan_result에 존재하지 않는 application_id

-> 따라서 삭제하기로 결정

※ year_income만 결측인 행 5개 존재

이 중 1개의 행이 loan_result에 존재함

566158	1746224	670502	1981.0	1.0	···	930.0	0.0	OTHERINCOME	NaT	기타	기타가족소유	6000000.0	
597613	341149	670502	1981.0	1.0	···	930.0	NaN	OTHERINCOME	NaT	기타	기타가족소유	6000000.0	

사업자금	0.0	0.0	1.0	3000000.0	41.0	중년	남성
사업자금	0.0	0.0	1.0	3000000.0	41.0	중년	남성

year_income 외의 모든 정보가 일치 -> **따라서 같은 값으로 대체**

데이터 시각화 및 전처리

결측치 처리

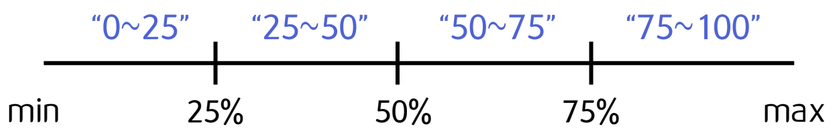
No.	컬럼ID	컬럼명
1	application_id	신청서 번호
2	user_id	유저 번호
3	birth_year	유저 출생년도
4	gender	유저 성별
5	insert_time	생성일시
6	credit_score	한도조회 당시 유저 신용점수
7	yearly_income	연소득
8	income_type	근로형태
9	company_enter_month	입사연월
10	employment_type	고용형태
11	houseown_type	주거소유형태
12	desired_amount	대출희망금액
13	purpose	대출 목적
14	personal_rehabilitation_yn	개인회생자 여부
15	personal_rehabilitation_complete_yn	개인회생자 납입 완료 여부
16	existing_loan_cnt	기대출수
17	existing_loan_amt	기대출금액

입사 나이(bef_enter_time) = 입사년도 - 출생년도

현재 경력(aft_enter_time) = 2022년(현재) - 입사년도

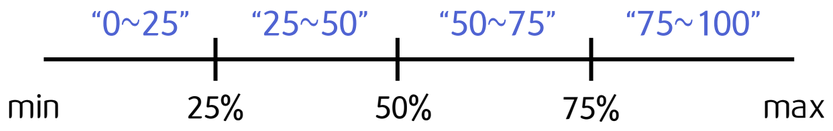
※ 결측인 행을 대체하기 보다는 percentile 기준으로 범주화

cat_bef_enter_time :



"cat_bef_enter_NA"

cat_aft_enter_time :



"cat_aft_enter_NA"

데이터 시각화 및 전처리

결측치 처리

No.	컬럼ID	컬럼명
1	application_id	신청서 번호
2	user_id	유저 번호
3	birth_year	유저 출생년도
4	gender	유저 성별
5	insert_time	생성일시
6	credit_score	한도조회 당시 유저 신용점수
7	yearly_income	연소득
8	income_type	근로형태
9	company_enter_month	입사연월
10	employment_type	고용형태
11	houseown_type	주거소유형태
12	desired_amount	대출희망금액
13	purpose	대출 목적
14	personal_rehabilitation_yn	개인회생자 여부
15	personal_rehabilitation_complete_yn	개인회생자 납입 완료 여부
16	existing_loan_cnt	기대출수
17	existing_loan_amt	기대출금액

개인회생자 여부	개인회생자 납입 완료 여부	의미 해석
0	-	개인회생자 아님
1	0	개인회생자 and 납부중
1	1	개인회생자 and 납부완료

개인회생자 여부	개인회생자 납입 완료 여부	Sample 수	
NA	NA	587360	결측치
1	0 or 1	12705	정상
0	NA	794005	정상
0	0 or 1	178145	비정상

⇨ NA의 원인을 살펴봄

⇩

개인회생자가 아닌 user에 대해서는
납입 완료 여부는 무시 / 메일 문의 답변 기준
-> **전부 0,NA로 간주**

데이터 시각화 및 전처리

결측치 처리

No.	컬럼ID	컬럼명
1	application_id	신청서 번호
2	user_id	유저 번호
3	birth_year	유저 출생년도
4	gender	유저 성별
5	insert_time	생성일시
6	credit_score	한도조회 당시 유저 신용점수
7	yearly_income	연소득
8	income_type	근로형태
9	company_enter_month	입사연월
10	employment_type	고용형태
11	houseown_type	주거소유형태
12	desired_amount	대출희망금액
13	purpose	대출 목적
14	personal_rehabilitation_yn	개인회생자 여부
15	personal_rehabilitation_complete_yn	개인회생자 납입 완료 여부
16	existing_loan_cnt	기대출수
17	existing_loan_amt	기대출금액

개인회생자 여부, 개인회생자 납입 완료 여부가 결측이 아닌 행이

처음 나타난 시점 : 4월 18일

그전까지는 전부 NA,NA

3.8.0 2022년 4월 19일

- 개인회생자 전용 대출이 출시되었어요.
- 몇몇 오류와 기능을 개선해서 더 좋은 고객 경험을 제공할 수 있게 되었어요. 불편한 점이 있다면 핀다 앱 > 1:1채팅하기로 편하게 말을 걸어주세요.

<핀다, AppStore, 버전 업데이트 기록>

"(NA,NA) 는 모르는 경우입니다."
<빅콘 퓨처스 질문 메일 답변>

따라서 결측인 이유는 개인회생자 여부 선택지가 없었기 때문으로 생각

-> 대체보다는 **결측을 포함한 범주화 (rehabilitation)**

데이터 시각화 및 전처리

No.	컬럼ID	컬럼명
1	application_id	신청서 번호
2	user_id	유저 번호
3	birth_year	유저 출생년도
4	gender	유저 성별
5	insert_time	생성일시
6	credit_score	한도조회 당시 유저 신용점수
7	yearly_income	연소득
8	income_type	근로형태
9	company_enter_month	입사연월
10	employment_type	고용형태
11	houseown_type	주거소유형태
12	desired_amount	대출희망금액
13	purpose	대출 목적
14	personal_rehabilitation_yn	개인회생자 여부
15	personal_rehabilitation_complete_yn	개인회생자 납입 완료 여부
16	existing_loan_cnt	기대출수
17	existing_loan_amt	기대출금액

집단 별 신용 점수의 차이가 있을까?

신용정보업 감독규정 제19조 ②항에 의하여 [채무자 회생 및 파산에 관한 법률]에 따라 회생절차가 진행 중인 자는 등록사유발생일로부터 최장 5년 이내에 활용 가능합니다.

위 감독규정에 의하여, 고객님의 장기연체정보는 해제일로부터 5년 이내, 개인회생정보는 발생일로부터 5년 이내에 평가에 활용됩니다.

따라서, 개인회생 공공정보(1301) 등록 또는 삭제시 신용도 산정에 영향이 있을수 있습니다.

<개인회생 여부에 따른 신용점수 차이에 대한 문의 답변 / KCB>

신용점수 산정 기준인 KCB 문의 결과,

개인회생자 여부에 따라서 신용점수에 영향을 미칠 수 있음.

데이터 시각화 및 전처리

결측치 처리

No.	컬럼ID	컬럼명
1	application_id	신청서 번호
2	user_id	유저 번호
3	birth_year	유저 출생년도
4	gender	유저 성별
5	insert_time	생성일시
6	credit_score	한도조회 당시 유저 신용점수
7	yearly_income	연소득
8	income_type	근로형태
9	company_enter_month	입사연월
10	employment_type	고용형태
11	houseown_type	주거소유형태
12	desired_amount	대출희망금액
13	purpose	대출 목적
14	personal_rehabilitation_yn	개인회생자 여부
15	personal_rehabilitation_complete_yn	개인회생자 납입 완료 여부
16	existing_loan_cnt	기대출수
17	existing_loan_amt	기대출금액

집단 별 신용 점수의 차이가 있을까? (검정)

※ 집단 별 신용 점수 정규성 검정 (귀무가설 : 정규성을 만족한다.)

P-value	Shapiro test	D'Agostino's K^2 Test	Kolmogorove-Smirnov test
rehabilitation_not	0	0	0
rehabilitation	0	0	0
rehabilitation_completed	1.04798401e-31	3.154716136e-47	0

결론 : 귀무가설 전부 기각 => 정규성을 만족하지 않음.

따라서, 모수적인 방법인 T-검정을 할 수 없기 때문에,

비모수적 방법인 Mann-Whitney U 검정을 사용

데이터 시각화 및 전처리

No.	컬럼ID	컬럼명
1	application_id	신청서 번호
2	user_id	유저 번호
3	birth_year	유저 출생년도
4	gender	유저 성별
5	insert_time	생성일시
6	credit_score	한도조회 당시 유저 신용점수
7	yearly_income	연소득
8	income_type	근로형태
9	company_enter_month	입사연월
10	employment_type	고용형태
11	houseown_type	주거소유형태
12	desired_amount	대출희망금액
13	purpose	대출 목적
14	personal_rehabilitation_yn	개인회생자 여부
15	personal_rehabilitation_complete_yn	개인회생자 납입 완료 여부
16	existing_loan_cnt	기대출수
17	existing_loan_amt	기대출금액

집단 별 신용 점수의 차이가 있을까? (검정)

※ Mann-Whitney U 검정

귀무가설 & 대립가설	P-value
H0 : rehabilitation_not <= rehabilitation H1 : rehabilitation_not > rehabilitation	0.0
H0 : rehabilitation_not <= rehabilitation_completed H1 : rehabilitation_not > rehabilitation_completed	8.944e-65
H0 : rehabilitation <= rehabilitation_completed H1 : rehabilitation > rehabilitation_completed	1.0

결론 : rehabilitation < rehabilitation_completed < rehabilitation_not

개인회생 납부 중, 개인회생 납부완료, 개인회생 아님 순서로 신용 점수가 높다고 나옴.

범주화 분류가 적절하다고 판단.

데이터 시각화 및 전처리

결측치 처리

No.	컬럼ID	컬럼명
1	application_id	신청서 번호
2	user_id	유저 번호
3	birth_year	유저 출생년도
4	gender	유저 성별
5	insert_time	생성일시
6	credit_score	한도조회 당시 유저 신용점수
7	yearly_income	연소득
8	income_type	근로형태
9	company_enter_month	입사연월
10	employment_type	고용형태
11	houseown_type	주거소유형태
12	desired_amount	대출희망금액
13	purpose	대출 목적
14	personal_rehabilitation_yn	개인회생자 여부
15	personal_rehabilitation_complete_yn	개인회생자 납입 완료 여부
16	existing_loan_cnt	기대출수
17	existing_loan_amt	기대출금액

기대출 수, 기대출 금액은 같은 user_id 내에서는 변하지 않음.

만약 대출 신청을 한 이력이 있더라도 기대출 수 and 기대출 금액은 동일함.

기대출수	기대출금액	Sample 수	
NA	NA	198507	기대출 없음
not_NA	not_NA	1080375	기대출 있음
not_NA	NA	115187	이상 값
not_NA	0	5130	반올림 과정 중 0

⇨ 0으로 대체

⇨ 기대출수가 전부 1임

⇩

실제 0값(기대출 없음)과 구분하기 위하여, 0~50만 중앙값인 25만을 결측이 아닌 행에 대하여 더해줌

데이터 시각화 및 전처리

결측치 처리

No.	컬럼ID	컬럼명
1	application_id	신청서 번호
2	user_id	유저 번호
3	birth_year	유저 출생년도
4	gender	유저 성별
5	insert_time	생성일시
6	credit_score	한도조회 당시 유저 신용점수
7	yearly_income	연소득
8	income_type	근로형태
9	company_enter_month	입사연월
10	employment_type	고용형태
11	houseown_type	주거소유형태
12	desired_amount	대출희망금액
13	purpose	대출 목적
14	personal_rehabilitation_yn	개인회생자 여부
15	personal_rehabilitation_complete_yn	개인회생자 납입 완료 여부
16	existing_loan_cnt	기대출수
17	existing_loan_amt	기대출금액

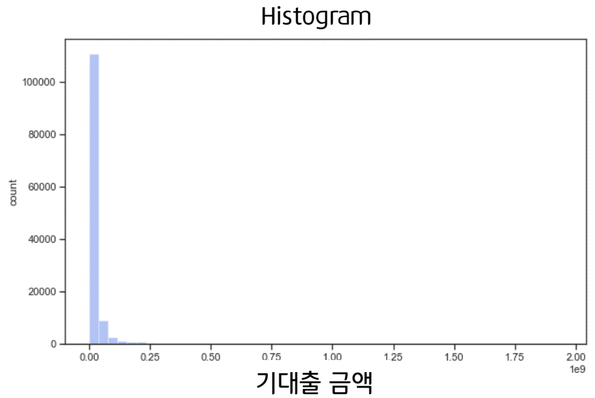
기대출 수	기대출 금액	Sample 수	
not_NA	NA	115187	이상 값

기대출 금액 정보를 받아오지 못하는 경우가 존재. / 메일 문의 답변 기준

(기대출 수 = 1) 정보는 살리며, 기대출 금액을 대체

기대출 수가 1인 데이터들의 기대출 금액을 이용하여 대체

좌측으로 편향된 분포이기 때문에 Median 선택!



기대출 수	기대출 금액
1	NA
1	NA
1	25000000.0
1	…

Median
10000000.0

기대출 수	기대출 금액
1	10000000.0
1	10000000.0
1	25000000.0
1	…

데이터 시각화 및 전처리

결측치 처리

No.	컬럼ID	컬럼명
1	application_id	신청서 번호
2	user_id	유저 번호
3	birth_year	유저 출생년도
4	gender	유저 성별
5	insert_time	생성일시
6	credit_score	한도조회 당시 유저 신용점수
7	yearly_income	연소득
8	income_type	근로형태
9	company_enter_month	입사연월
10	employment_type	고용형태
11	houseown_type	주거소유형태
12	desired_amount	대출희망금액
13	purpose	대출 목적
14	personal_rehabilitation_yn	개인회생자 여부
15	personal_rehabilitation_complete_yn	개인회생자 납입 완료 여부
16	existing_loan_cnt	기대출수
17	existing_loan_amt	기대출금액

신용 점수는 insert_time 순서로

- 초반만 결측이거나,
- 전체가 결측,
- 결측이 없는 경우만 존재

<결측인 이유>

신용점수 정보는 어플 내의 "나의 신용점수 확인하기"를 통하여 조회한 신용점수를 통해 가져옴. 따라서 아직 "나의 신용점수 확인하기"를 사용하지 않은 경우임. / 메일 문의 답변 기준

	application_id	user_id	insert_time	credit_score
413222	636758	93773	2022-04-07 16:09:53	NaN
674346	409035	93773	2022-04-07 16:17:54	NaN
806794	2039145	93773	2022-05-13 14:53:10	NaN
167198	548660	93773	2022-06-01 12:59:05	730.0
402585	446680	93773	2022-06-02 16:01:35	730.0
685981	1252877	93773	2022-06-07 16:53:58	730.0

	application_id	user_id	insert_time	credit_score
737345	984561	135941	2022-06-17 08:54:29	NaN
660062	257551	135941	2022-06-22 21:19:57	NaN

	application_id	user_id	insert_time	credit_score
449836	1026010	13157	2022-05-16 16:39:31	650.0
554468	2076598	13157	2022-05-17 16:01:16	650.0
95869	2090605	13157	2022-05-18 10:34:10	650.0
636658	880665	13157	2022-05-19 11:51:22	650.0
560461	462309	13157	2022-05-23 13:48:17	650.0

데이터 시각화 및 전처리

No.	컬럼ID	컬럼명
1	application_id	신청서 번호
2	user_id	유저 번호
3	birth_year	유저 출생년도
4	gender	유저 성별
5	insert_time	생성일시
6	credit_score	한도조회 당시 유저 신용점수
7	yearly_income	연소득
8	income_type	근로형태
9	company_enter_month	입사연월
10	employment_type	고용형태
11	houseown_type	주거소유형태
12	desired_amount	대출희망금액
13	purpose	대출 목적
14	personal_rehabilitation_yn	개인회생자 여부
15	personal_rehabilitation_complete_yn	개인회생자 납입 완료 여부
16	existing_loan_cnt	기대출수
17	existing_loan_amt	기대출금액

초반만 결측인 경우

	application_id	user_id	insert_time	credit_score	
413222	636758	93773	2022-04-07 16:09:53	NaN	730.0
674346	409035	93773	2022-04-07 16:17:54	NaN	730.0
806794	2039145	93773	2022-05-13 14:53:10	NaN	730.0
167198	548660	93773	2022-06-01 12:59:05	730.0	
402585	446680	93773	2022-06-02 16:01:35	730.0	
685981	1252877	93773	2022-06-07 16:53:58	730.0	

bfiil : 뒤 값으로 대체

결측 행 : 105054개 -> 87464개

데이터 시각화 및 전처리

No.	컬럼ID	컬럼명
1	application_id	신청서 번호
2	user_id	유저 번호
3	birth_year	유저 출생년도
4	gender	유저 성별
5	insert_time	생성일시
6	credit_score	한도조회 당시 유저 신용점수
7	yearly_income	연소득
8	income_type	근로형태
9	company_enter_month	입사연월
10	employment_type	고용형태
11	houseown_type	주거소유형태
12	desired_amount	대출희망금액
13	purpose	대출 목적
14	personal_rehabilitation_yn	개인회생자 여부
15	personal_rehabilitation_complete_yn	개인회생자 납입 완료 여부
16	existing_loan_cnt	기대출수
17	existing_loan_amt	기대출금액

전체가 결측인 경우

	application_id	user_id	insert_time	credit_score
737345	984561	135941	2022-06-17 08:54:29	NaN
660062	257551	135941	2022-06-22 21:19:57	NaN

<추가 변수>

application_id 별 승인 상품 log한도, 금리 통계량

('max', 'min', 'mean', 'median', 'count')

MissForest 모델 사용하여 결측치 추정

Valid RMSE : 67.1

user_id 별 평균 예측 신뢰점수를 둘째자리까지 반올림 후 일괄 적용.

데이터 시각화 및 전처리

결측치 처리

No.	컬럼ID	컬럼명
1	application_id	신청서 번호
2	user_id	유저 번호
3	birth_year	유저 출생년도
4	gender	유저 성별
5	insert_time	생성일시
6	credit_score	한도조회 당시 유저 신용점수
7	yearly_income	연소득
8	income_type	근로형태
9	company_enter_month	입사연월
10	employment_type	고용형태
11	houseown_type	주거소유형태
12	desired_amount	대출희망금액
13	purpose	대출 목적
14	personal_rehabilitation_yn	개인회생자 여부
15	personal_rehabilitation_complete_yn	개인회생자 납입 완료 여부
16	existing_loan_cnt	기대출수
17	existing_loan_amt	기대출금액

3. 출생년도, 4. 성별 : 같은 user_id 이용하여 대체 후 범주화

7, 8, 10, 11, 12, 13 : 결측일 수 없는 변수들이라 행 제거,
지울 수 없는 행은 같은 user_id 를 기준으로 대체

9. 입사연월 : 형식 맞춘 후, 파생 변수를 생성하여 범주화

14, 15. 개인회생자 변수 : 결측치 의미 정의하여, 범주화

16, 17. 기대출 변수 : 결측치 의미 파악 후, 대체

6. 신용점수 : NA행 다음으로 따라오는 행의 값으로 대체 후,
남은 행은 모델을 이용하여 대체

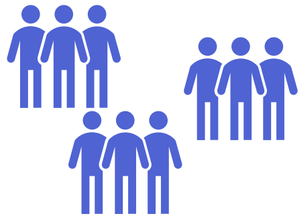
모델링

데이터 시각화 및 전처리

- 데이터 설명 및 시각화
- 결측치 처리

모델링

- 모델링 구조 설명
- 1차 모델링 및 성능 평가
- 2차 모델링 및 성능 평가



군집화

- 군집 개요
- 군집 별 특성 파악
- 추천 메시지 제안

결론

- 예측 모델 해석 및 활용 방안
- 군집 결과 해석 및 활용 방안

모델링

모델링 구조 설명

1차 모델링

신청할 application_id

application_id	is_applied_max
1864566	1
278753	1
1508331	1
1953536	1

신청하지 않을 application_id

application_id	is_applied_max
585773	0
59283	0

승인 상품이 없는 application_id -> 학습에 사용하지 않음

application_id	is_applied_max
5823	NA
53022	NA

신청할 application_id를 분류한 후,
해당 데이터에 대해서만 상품 신청 여부 예측

<2차 모델링의 기대효과>

1. 데이터 불균형 해소
2. 데이터 해석 및 예측 성능 향상

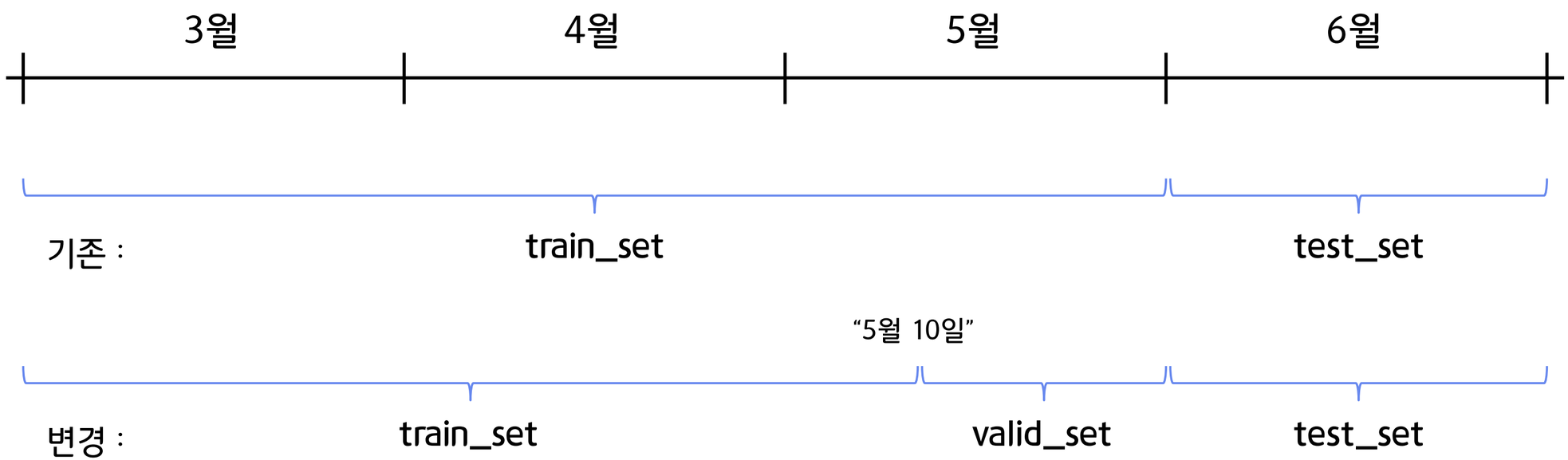
2차 모델링

신청할 product_id

application_id	product_id	is_applied
1864566	1	0
1864566	6	1
1864566	23	0
278753	52	1
278753	5	0
278753	43	0
278753	23	0
1508331	4	1
1508331	28	0
1508331	30	1
1953536	30	0
1953536	8	0
1953536	28	0
1953536	7	1

모델링

모델링 구조 설명



모델 평가 및 하이퍼파라미터 조정을 위하여 5월 10일 ~ 5월 31일까지의 데이터를 validation_set 으로 지정

모델링

1차 모델링 및 성능 평가

모델 구조 및 학습 모델 평가

CatboostClassifier

user_spec 내에 범주형 변수가 많기 때문에, one-hot encoding과 같은 방법으로 범주형 변수를 처리하는 것이 아닌 범주형 변수를 사용할 수 있는 Catboost 모델을 사용하기로 함

범주형 변수 : (cat_age, cat_gender, income_type, employment_type, houseown_type, purpose, cat_aft_enter_time, cat_bef_enter_time, rehabilitation, 9개)

Hyperparameter

- Iterations = 600
- Depth = 10
- Learning_rate = 0.09

모델링

1차 모델링 및 성능 평가

모델 구조 및 학습 모델 평가

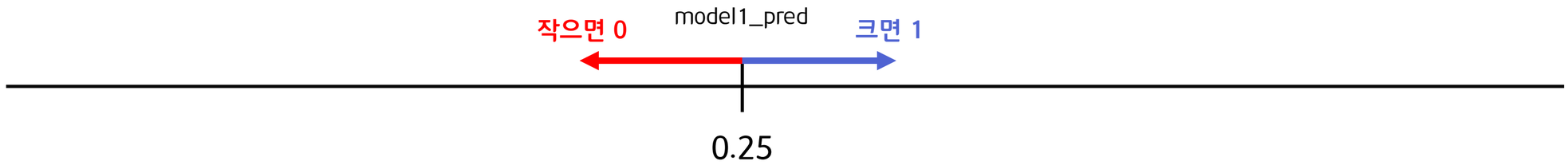
분류 결과를 0~1 사이의 확률 값으로 반환하여, 기준 값(default=0.5)을 조정하며 f1.5-score를 비교한다.

※ f1.5-score를 보는 이유 : 1차 모델에서는 실제 True인 값의 손실을 줄이기 위해 recall의 중요도가 높다고 판단함.

$$F_\beta = (1+\beta^2) \cdot \frac{\text{precision} \cdot \text{recall}}{(\beta^2 \cdot \text{precision}) + \text{recall}}$$

f1.5-score 최대!

기준 값	0.1	0.15	0.2	0.25	0.3	0.35	0.4	0.45	0.5
f1.5-score	…	…	…	…	…	…	…	…	…
recall	…	…	…	…	…	…	…	…	…



모델링

2차 모델링 및 성능 평가

모델 구조 및 학습

모델 평가

application_id	model1_pred
1864566	1
278753	1
1508331	1
1953536	1
585773	0
59283	0

0인 application_id의 상품들은
is_applied를 0으로 예측함

application_id	product_id	is_applied
1864566	1	0
1864566	6	1
1864566	23	0
278753	52	1
278753	5	0
278753	43	0
278753	23	0
1508331	4	1
1508331	28	0
1508331	30	1
1953536	30	0
1953536	8	0
1953536	28	0
1953536	7	1

Model 1. Xgboost Classifier

Model 2. LGBM Classifier

Model 3. Catboost Classifier

Ensemble!

모델링

2차 모델링 및 성능 평가

모델 구조 및 학습 모델 평가

Model 1. Xgboost Classifier

Tuned Hyper Parameter : n_estimator = 500, max_depth = 14, learning_rate = 0.03

Train f1-score : 0.609

Test f1-score : 0.376

Model 2. LGBM Classifier

Tuned Hyper Parameter : n_estimator = 500,
max_depth = 8, learning_rate = 0.1,
num_leaves = 200

Train f1-score : 0.567

Test f1-score : 0.398

Model 3. Catboost Classifier

Tuned Hyper Parameter : n_estimator = 400,
max_depth = 8, learning_rate = 0.02

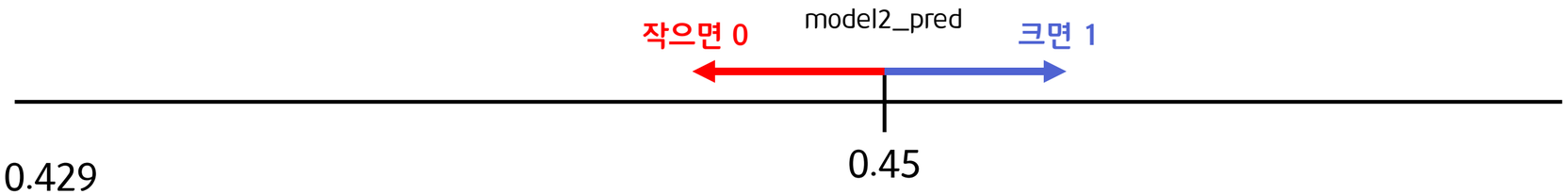
Train f1-score : 0.542

Test f1-score : 0.391

Ensemble 1 : 모델 별 예측 확률 평균을 바탕으로 최적의 기준 값 찾기

proba_xgb_02_BO_valid : Xgboost 예측 확률	proba_lgb_02_BO_valid : LGBM 예측 확률	proba_cat_02_BO_valid : Catboost 예측 확률	model2_prob_valid : 평균
0.7	0.8	0.9	0.8
0.2	0.02	0.02	0.08
0.8	0.6	0.4	0.6
0.1	0.2	0.15	0.15
…	…	…	…

기준 값	0.1	…	…	0.45	…	…	0.9
f1-score	0.79793	…	…	0.80249	…	…	0.70796

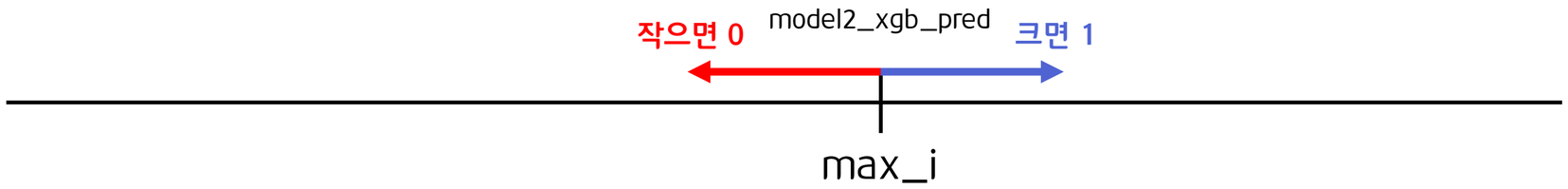


Valid_f1-score : 0.429

Ensemble 2 : 모델 별 최적의 기준 값으로 분류 후, 이 중 예측 값이 하나라도 1이면 최종 예측 값은 1

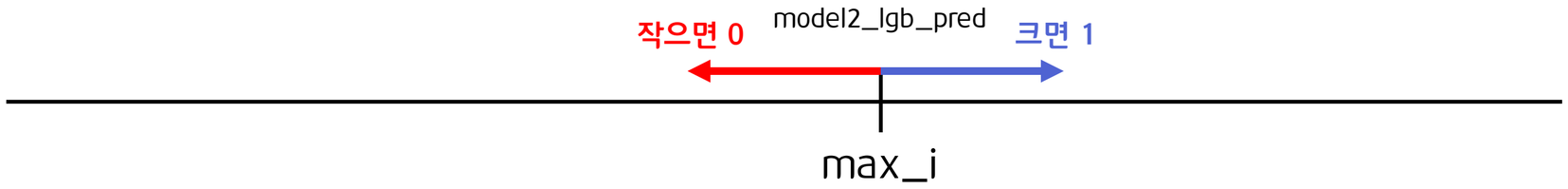
Xgboost

기준 값	0.1	…	…	…	max_i	…	…	0.9
f1-score	…	…	…	…	…	…	…	…



LGBM

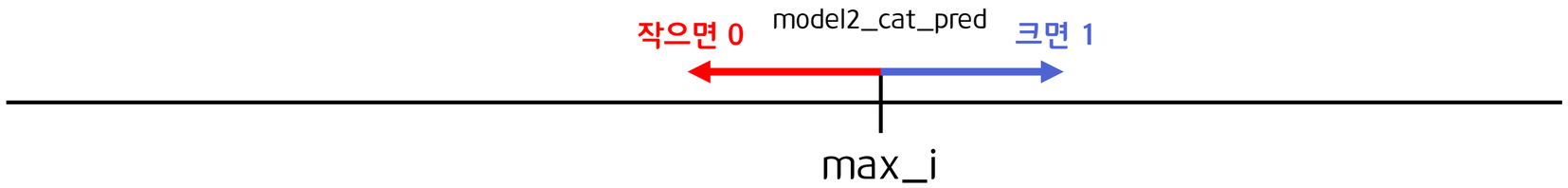
기준 값	0.1	…	…	…	max_i	…	…	0.9
f1-score	…	…	…	…	…	…	…	…



Ensemble 2 : 모델 별 최적의 기준 값으로 분류 후, 이 중 예측 값이 하나라도 1이면 최종 예측 값은 1

Catboost

기준 값	0.1	…	…	max_i	…	…	0.9
f1-score	…	…	…	…	…	…	…



Valid_f1-score : 0.417

모델링

2차 모델링 및 성능 평가

모델 구조 및 학습 모델 평가

Ensemble 3 : 모델 별 0.5 기준 값으로 분류 후, 이 중 예측 값이 하나라도 1이면 최종 예측 값은 1

pred_xgb_02_BO_valid : Xgboost 예측	pred_lgb_02_BO_valid : LGBM 예측	ppred_cat_02_BO_valid : Catboost 예측	model2_pred_valid : 최댓값
1	1	1	1
1	0	1	1
0	0	0	0
0	1	1	1
…	…	…	…

Valid_f1-score : 0.328

최종적으로 Ensemble 1 방법을 적용한다

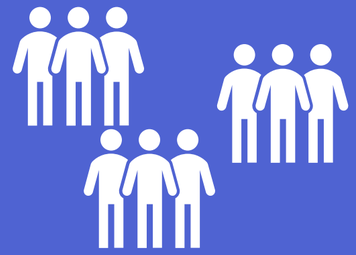
군집화

데이터 시각화 및 전처리

- 데이터 설명 및 시각화
- 결측치 처리

모델링

- 모델링 구조 설명
- 1차 모델링 및 성능 평가
- 2차 모델링 및 성능 평가



군집화

- 군집 구조 설명
- 군집 별 특성 파악
- 추천 메시지 제안

결론

- 예측 모델 해석 및 활용 방안
- 군집 결과 해석 및 활용 방안

군집화

군집 구조 설명

1차 군집화

정보 입력의 일관성이 떨어지는
사용자를 묶기 위한 군집화

신뢰성이 낮은 군집 외의 나머지 군집은 재차
군집화 진행

1차 군집화를 위해 필요한 변수

- n_company_enter_month
  : user_id 당 company_enter_month 의 갯수
- n_income_type
  : user_id 당 income_type 의 갯수
- n_imployment_type
  : user_id 당 imployment_type 의 갯수
- n_houseown_type
  : user_id 당 houseown_type 의 갯수
- var_work_experience_for_day
  : user_id 당 (2022/ 06/ 01 – company_enter_day) 의 분산

2차 군집화

사용자의 특성을 파악하는 군집화

2차 군집화를 위해 필요한 변수

- count
  : user_id 당 대출조회횟수
- Sign
  : 주어진 기간 내 첫 대출조회 시간부터 6월 1일까지의 기간
- Is_applied
  : user_id 당 평균 대출신청 횟수

군집화

군집별 특성 파악

1차 군집화

	n_company_enter_month	n_income_type	n_employment_type	n_houseown_type	var_work_experience_for_days
0	0.002454	0.005544	0.0000…	0.0000…	0.000068
1	0.033357	0.199219	3.517455e-01	3.246059e-02	0.000753
2	0.013257	0.019981	2.151369e-02	3.455655e-01	0.000401

변수 선택 이유

군집화 전, 신뢰할 만한 데이터로 정제하여 군집의 특성을 분석해야 한다고 생각했다.

신뢰성을 판단하는 기준으로 정보입력의 일관성을 선택했다.

이 때, 데이터에 존재하는 User_id 별 범주형 변수 마다 고유한 값의 빈도를 데이터로 활용하여 판단하고자 한다.

추가로 경력에 대한 변수를 추가한 이유는 경력의 변동성이 심한 사용자들을 분류하기 위함이다.

군집화

군집별 특성 파악

1차 군집화

	n_company_enter_month	n_income_type	n_employment_type	n_houseown_type	var_work_experience_for_days
0	0.002454	0.005544	0.0000…	0.0000…	0.000068
1	0.033357	0.199219	3.517455e-01	3.246059e-02	0.000753
2	0.013257	0.019981	2.151369e-02	3.455655e-01	0.000401

1차 군집설명

군집 0

2차 군집화 대상

정보입력이 매우 일관적인 집단

군집 2

2차 군집화 대상

정보입력이 어느 정도
일관적이지만 추가적인 확인은
필요한 집단

군집 1

정보 입력의 일관성이 떨어지는
고객 집단

Less_Consistency_Group
: 최종 군집 1

군집화

군집별 특성 파악

2차 군집화

cluster	count_std	sign_std	is_applied_std
0	-0.288913	-0.990750	-0.139219
1	0.025491	0.749961	-0.119884
2	2.854657	0.780155	3.172757

변수 선택 이유

User_id 별 대출조회 횟수와 평균 대출신청 횟수 변수가 대출관심고객층을 구분하는데 영향을 줄 수 있다고 판단했다.

이 때, 주어진 기간 내 첫 대출조회 시간부터 6월 1일까지의 시간을 대출조회 횟수와 같이 활용함으로써 핀다에 대한 관심도를 판단할 수 있다고 생각했다.

위의 파생 변수를 생성하는 과정에서 6월 1일을 기준으로 한 이유는 모델예측을 통해 생성된 is_applied 변수를 군집화에 사용하지 않기 위함이다.

군집화

군집별 특성 파악

2차 군집화

cluster	count_std	sign_std	is_applied_std
0	-0.288913	-0.990750	-0.139219
1	0.025491	0.749961	-0.119884
2	2.854657	0.780155	3.172757

Center 계산 : mean

cluster	count_std	sign_std	is_applied_std
0	-0.443234	-0.962350	-0.306615
1	0.236746	0.807416	-0.306615
2	1.828133	1.065507	3.038794

Center 계산 : median

2차 군집설명

군집 0

신규 사용자 군집

: 최종 군집 2

군집 1

잠재적 휴면 사용자 군집

: 최종 군집 3

군집 2

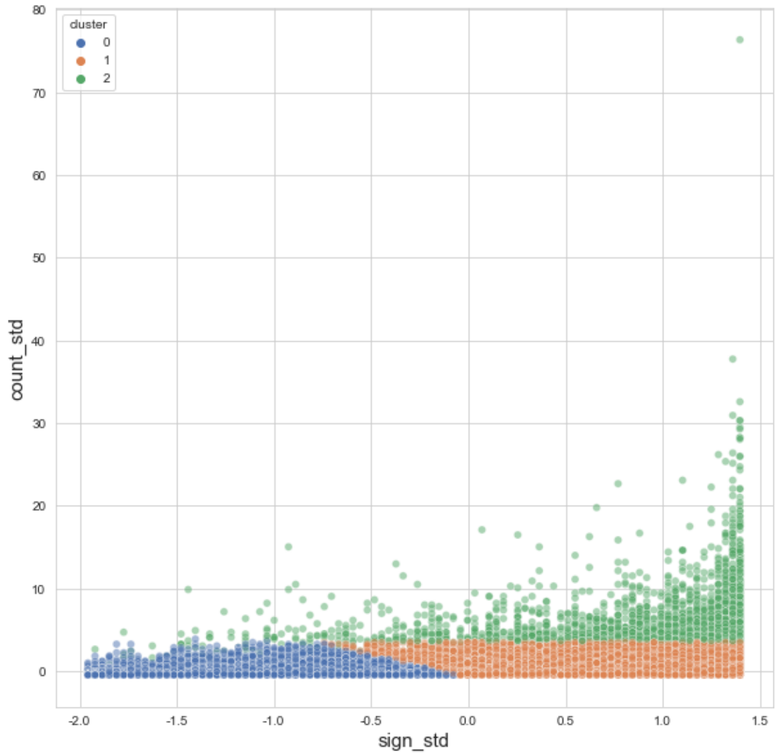
주 사용자 군집

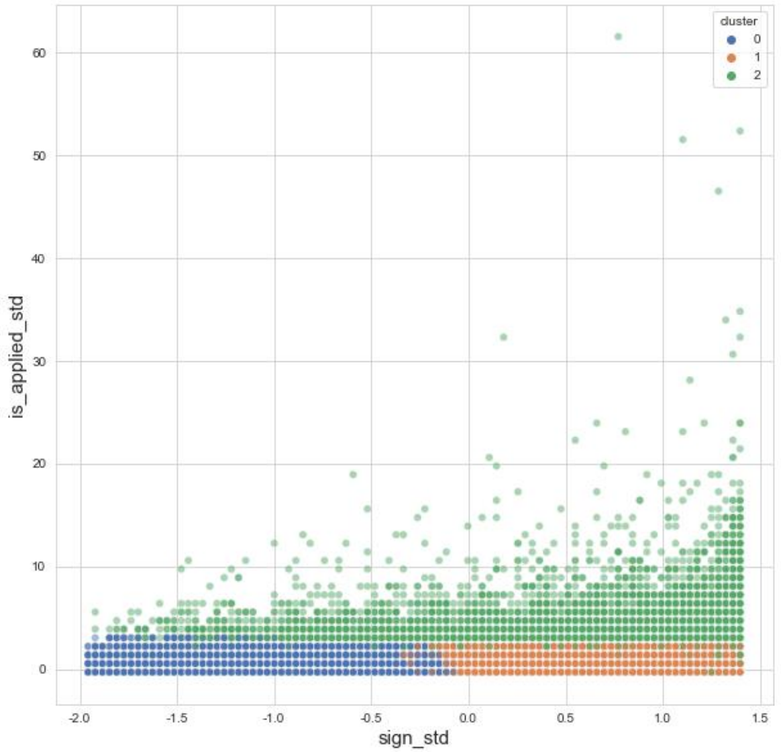
: 최종 군집 4

군집화

군집별 특성 파악

2차 군집 시각화





3차원의 Feature를 통해 3개의 군집을 형성했고, 2개씩의 Feature를 2차원 시각화로 각각 표현해보았다.

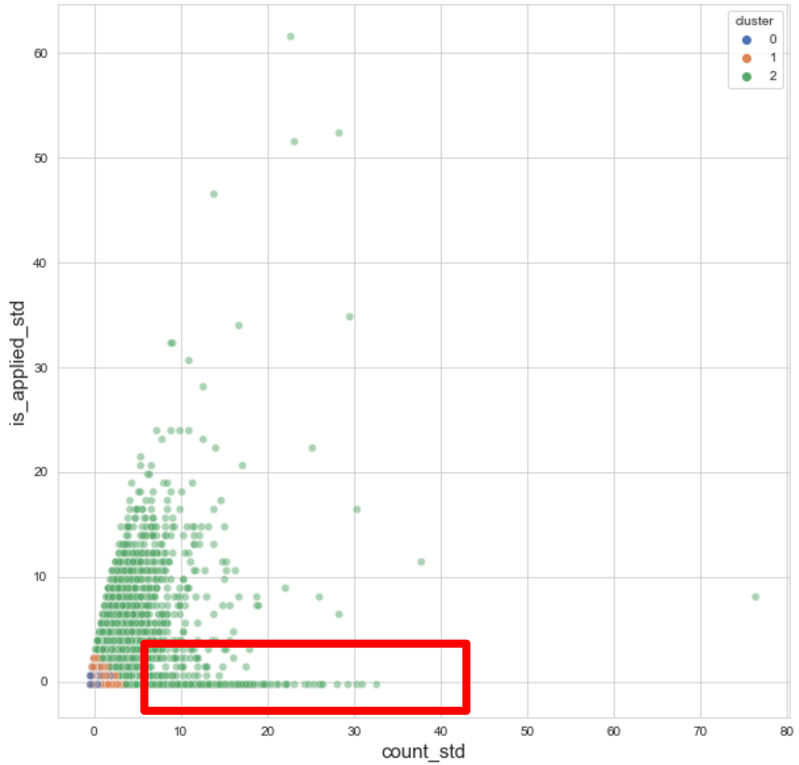
주어진 기간 내 첫 대출조회 시간부터 6월 1일까지의 시간이 군집 0과 1의 분류에 절대적인 영향을 준다.

대출조회 횟수와 평균 대출신청 횟수 각각의 특정 변수는 군집2와 군집 0,1 사이의 구분점에 영향을 준다.

군집화

군집별 특성 파악

2차 군집 시각화



군집0과 1에서는 대출조회 수 자체가 많지 않아 대출신청 수도 적게 나타난다.

군집2는 대출조회 수가 늘어남에 따라 자연스럽게 대출신청 수도 늘어나는 형태를 보인다.

그러나 군집2에서 대출조회 수가 많음에도, 대출신청 수는 0에 가까운 집단이 포함되어 있음을 볼 수 있다.

군집화

군집 정의 및 메시지 제안

최종 군집 1번 | 최종 군집 2번 | 최종 군집 3번 | 최종 군집 4번

최종 군집 1번

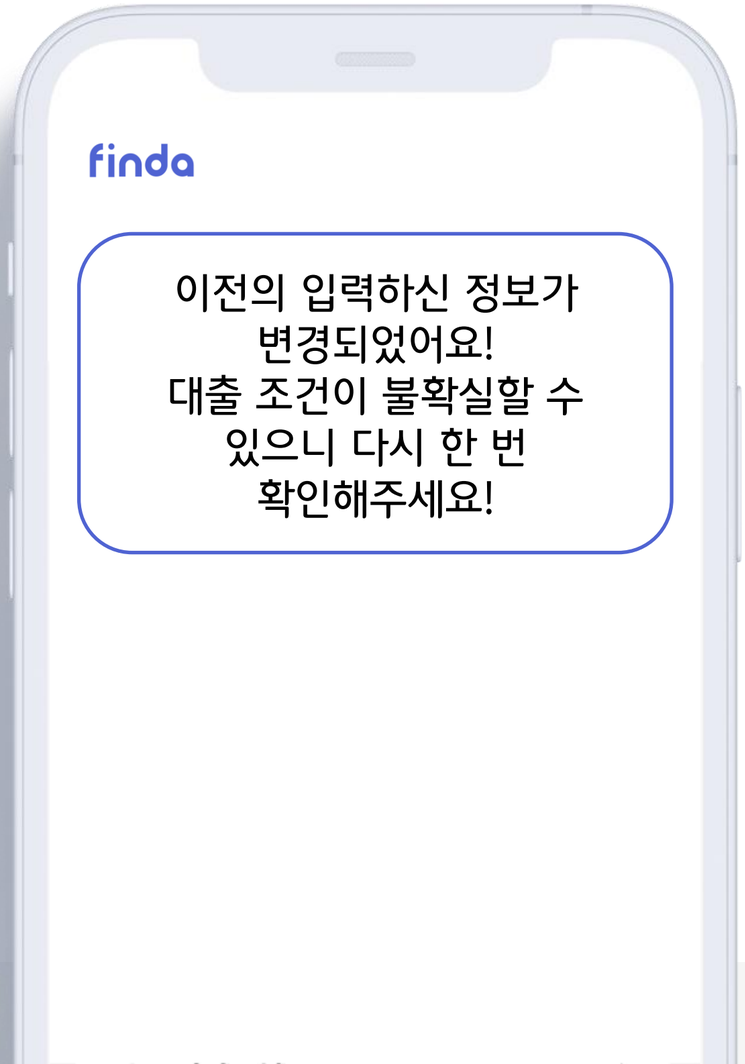
1차 군집화 과정에서 정보입력의 일관성이 떨어졌던 집단이 최종 군집 1이다.

이 군집은 정보를 있는 그대로 신뢰하기 어려운 고객층으로 구성되어 있다.

따라서 정보입력을 다시 한 번 확인할 수 있도록 기회를 제공하는 메시지를 제안한다.

이를 통해, 기업은 보다 양질의 데이터를 수집하여 질 높은 데이터베이스 구축이 가능하며,

사용자는 본인의 조건에 최적화된 대출 상품을 추천 받을 수 있다.

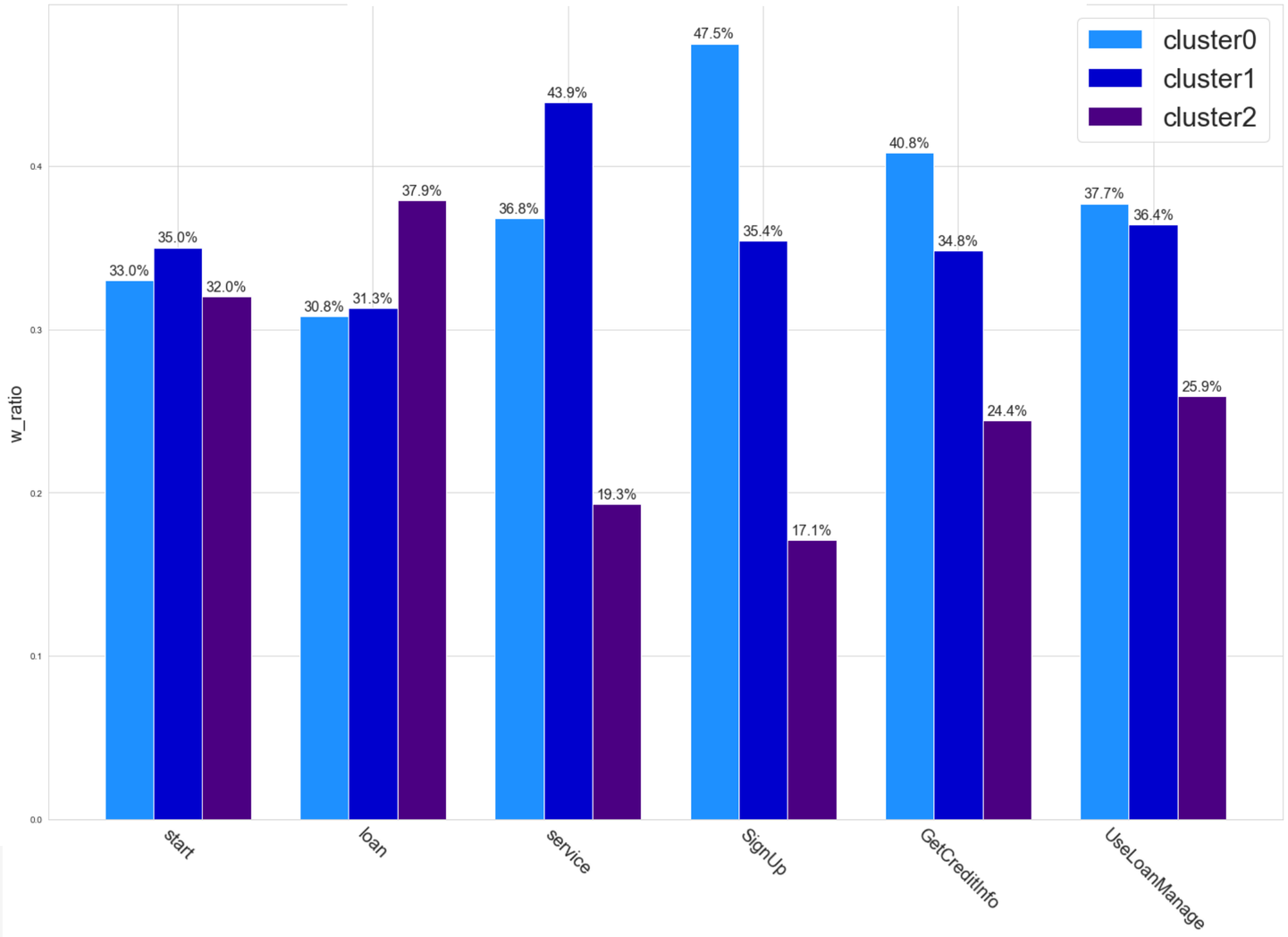


군집화

군집 정의 및 메시지 제안

최종 군집 1번 | 최종 군집 2번 | 최종 군집 3번 | 최종 군집 4번

2차 군집 별 로그 event 비율 비교 그래프



군집화

군집 정의 및 메시지 제안

최종 군집 1번 | 최종 군집 2번 | 최종 군집 3번 | 최종 군집 4번

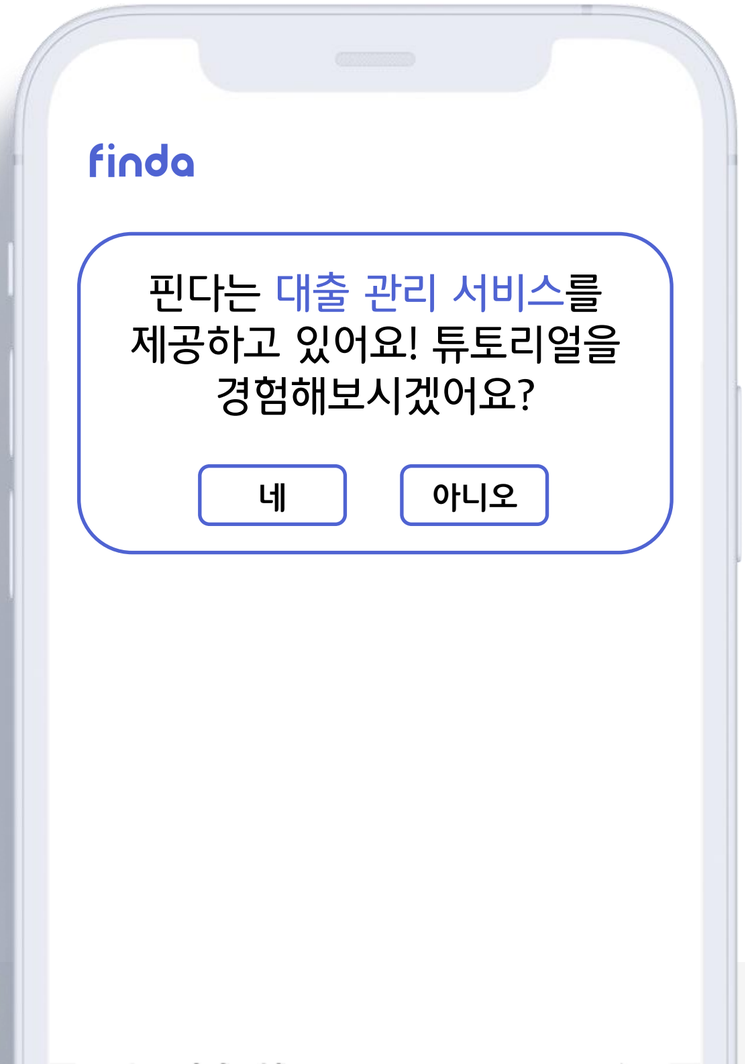
최종 군집 2번

1차 군집화 과정에서 일관성이 있다고 판단된 두 군집을 다시 2차 군집화 했다.

2차 군집화 과정에서 cluster0은 주어진 기간에서 첫 조회가 비교적 최근이며, 회원가입을 하는 로그 event 비율이 2차 군집화 과정 속 타 군집에 비해 가장 높았던 것으로 보아 유입된 지 얼마 되지 않은 신규 유입층으로 정의할 수 있다.

그렇기 때문에 여러가지 서비스를 전반적으로 경험해볼 수 있도록 각 서비스별 튜토리얼을 제공한다.

이를 통해, 신규 유입된 고객층이 전반적인 서비스를 경험함으로써 향후 핀다를 적극적으로 활용하는 고객 군집이 되도록 안내한다.



군집화

군집 정의 및 메시지 제안

최종 군집 3번

2차 군집화 과정에서 도출된 cluster 1은 주어진 기간에서 첫 조회가 비교적 오래되었으며,

그럼에도 대출조회 횟수나 대출 신청 횟수가 매우 적은 집단이다.

활동한 기간에 비해 로그가 상대적으로 적다는 것은 핀다 어플을 소극적으로 활용하는

집단이자 “잠재적인 휴면 고객층”을 의미한다.

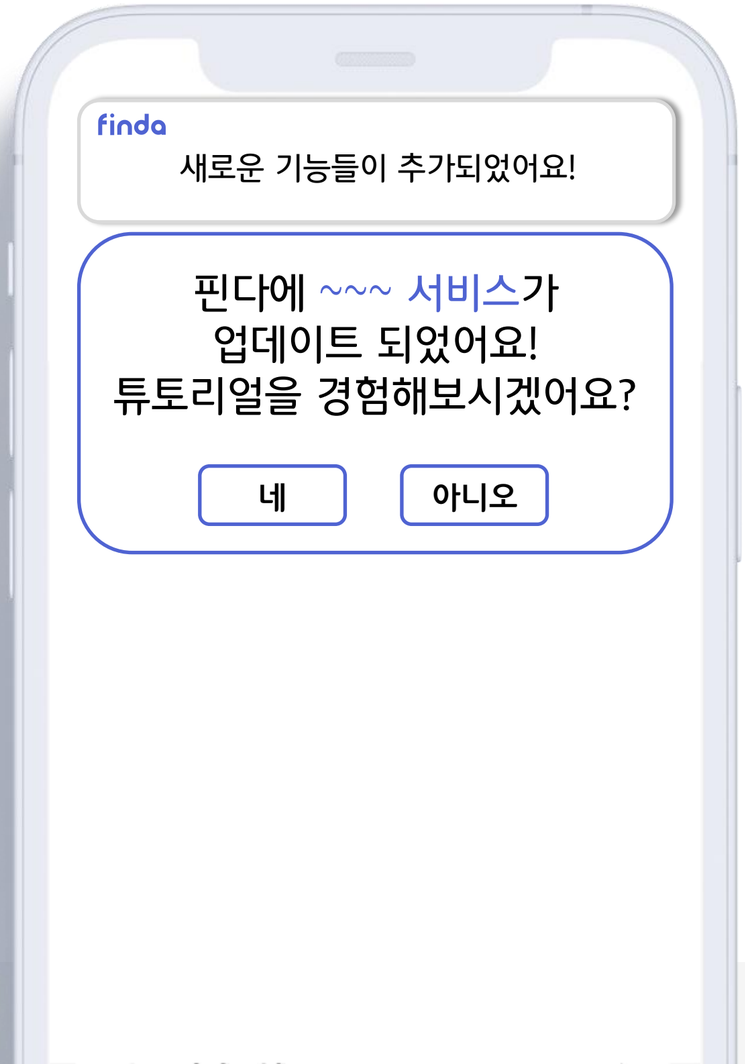
따라서 어플에 접속하여 홈화면에 메시지를 띄우는 것 보다는 팝업 알림을 활용하여 핀다

어플을 실행할 수 있도록 유도하는 것이 실용적이다.

어플에 접속한 사용자에 대해서는 그동안 핀다에 새로 업데이트된 기능들에 대한 튜토리얼을

제공하는 메시지를 제안한다.

이를 통해, 잠재적 휴면 고객들이 다시 핀다에 복귀하여 원활하게 활동하도록 유도할 수 있다.



군집화

군집 정의 및 메시지 제안

최종 군집 4번

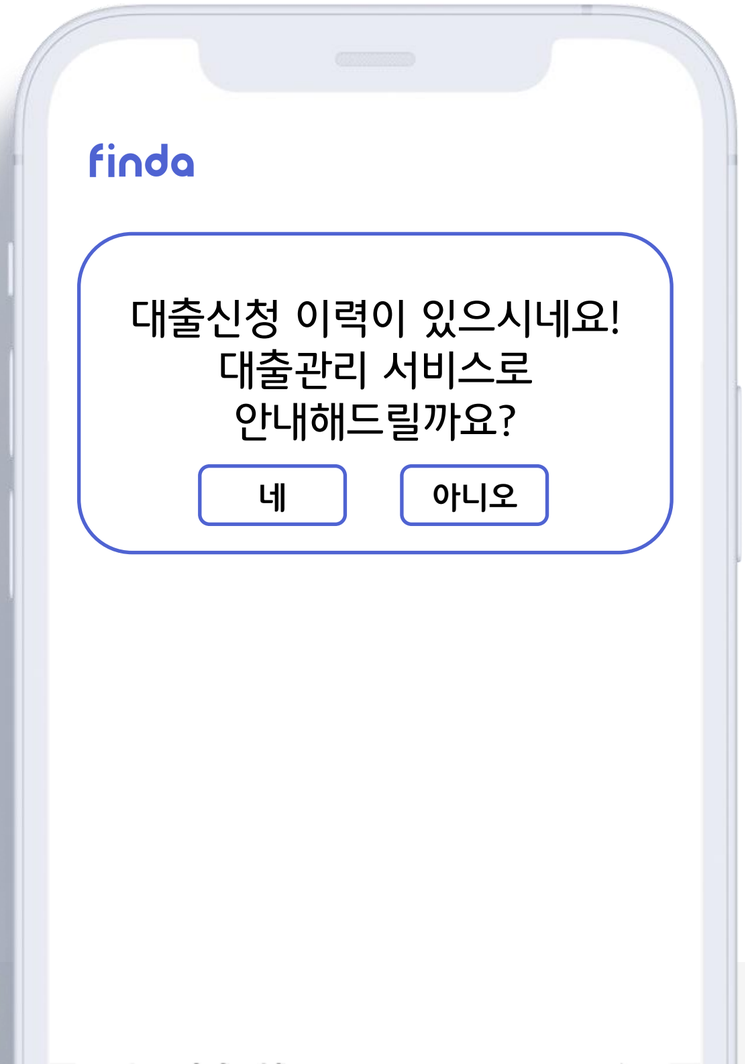
2차 군집화 과정에서 도출된 cluster2는 user_id 대비 가장 많은 로그 데이터를 갖고 있다.

해당 군집의 center 정보를 보면, 타 군집에 비해 대출신청 횟수와 대출관련 로그 event 비율이 월등하게 높다.

최종 군집 4번은 특히 대출관련 특징이 명확하여, “대출서비스 경험군“ 혹은 “대출 관심 고객층”이라고 정의할 수 있다.

따라서, 이 집단은 대부분 대출을 신청한 경험이 있기 때문에 사후관리 차원에서 대출관리 서비스를 이용하도록 제안하는 메시지를 띄운다.

이를 통해, 주 사용 군집인 해당 집단이 핀다에 대하여 지속적으로 관심을 유지할 수 있다.



군집화

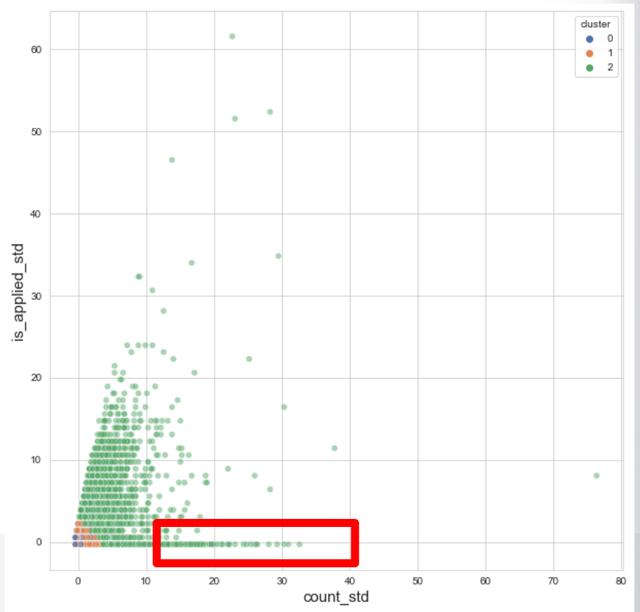
군집 정의 및 메시지 제안

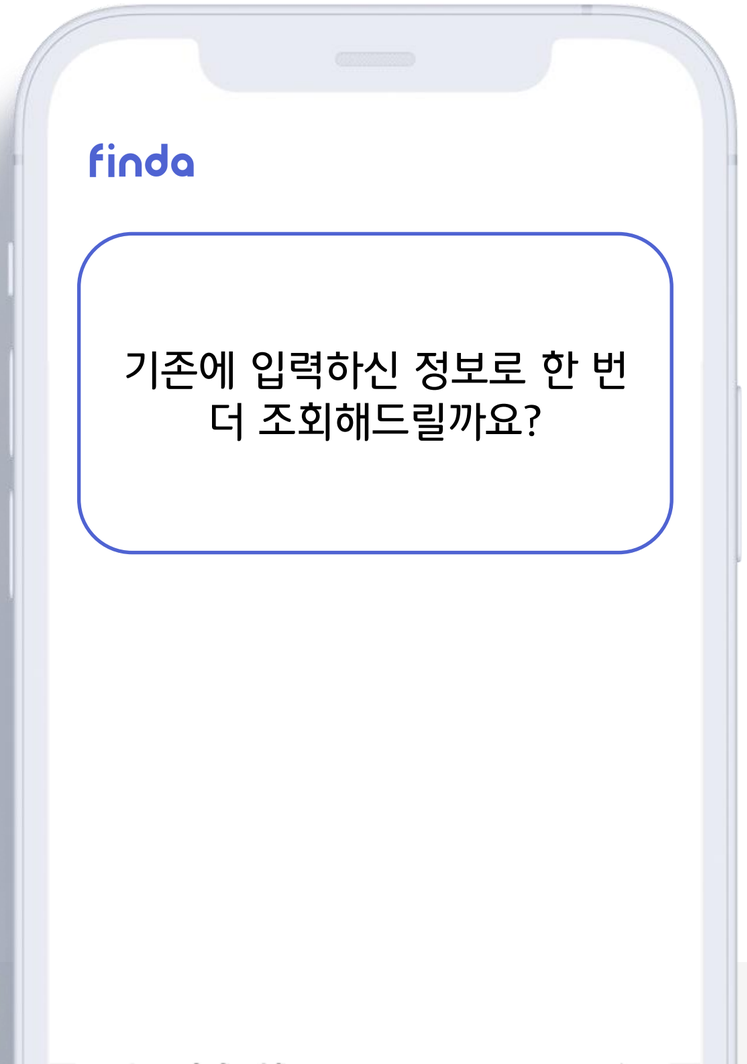
최종 군집 1번 최종 군집 2번 최종 군집 3번 최종 군집 4번

최종 군집 4번 (특히, 대출신청 횟수가 적은 집단)

하지만, 이 중에서 대출조회 횟수에 비해 대출신청 횟수가 적은 사용자들이 해당 군집에 속해 있다. 이들은 추천 받은 상품이 만족스럽지 않거나, 단순히 대출조회에만 관심이 있는 고객층이다. 따라서, 이러한 군집 내 집단에게는 새로운 대출조건을 확인할 수 있도록 대출조회를 유도하는 메시지를 제안한다.

이를 통해, 추천 받은 상품이 만족스럽지 않았던 고객들은 새로운 대출 조건을 통해 대출신청을 한 번 더 고민하는 기회가 생긴다.





결론

데이터 시각화 및 전처리

- 데이터 설명 및 시각화
- 결측치 처리

모델링

- 모델링 구조 설명
- 1차 모델링 및 성능 평가
- 2차 모델링 및 성능 평가

군집화

- 군집 개요
- 군집 별 특성 파악
- 추천 메시지 제안

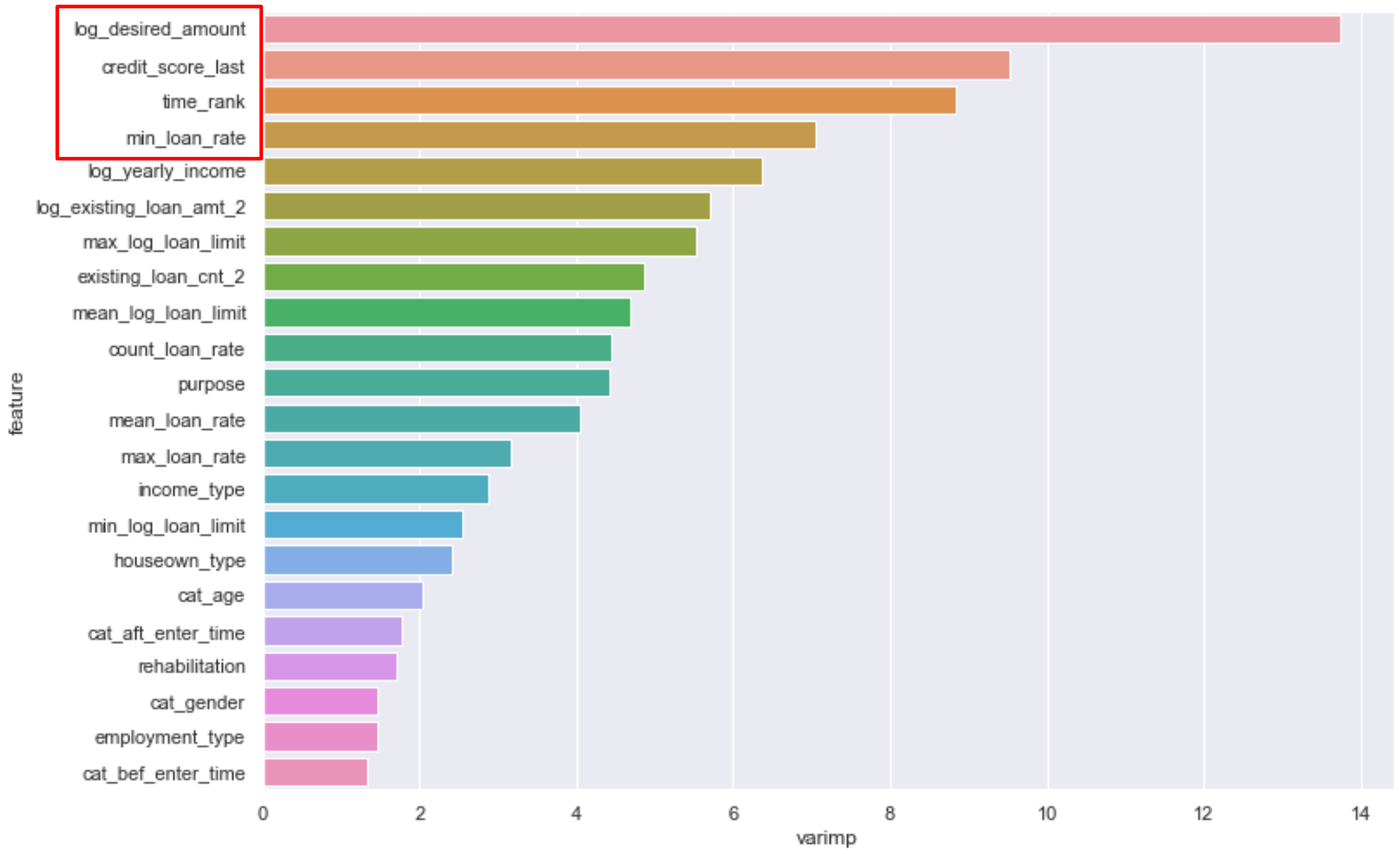
결론

- 예측 모델 해석 및 활용 방안
- 군집 결과 해석 및 활용 방안

결론

예측 모델 해석 및 활용 방안

1차 모델링 변수 중요도



1차 모델링은 user_spec 데이터를 기반으로 신청 여부를 예측하는 구조이다.

credit_score_last : 신용 점수 변수의 중요도와 min_loan_rate : 최소 금리 변수의 중요도가 높은 것을 보아, 신용 점수가 높은 사용자는 보다 좋은 조건의 금리를 받게 되며, 이를 통해 신청에 영향을 준다고 해석할 수 있다.

또한, log_desired_amount : 목적 금액 log 가 높은 것은 필요한 대출 금액이 큰 사람 일수록 대출 신청을 많이한다는 것을 의미한다.

결론

예측 모델 해석 및 활용 방안

1차 모델링 변수 중요도

time_rank : 이용자 당 대출 조회 순서

time_rank 당 신청 평균

1 : 0.5109825488655573	11 : 0.4100731962356221
2 : 0.5671068522441545	12 : 0.39449348465741907
3 : 0.5579266767809806	13 : 0.39056984367120523
4 : 0.5401594114040466	14 : 0.3765376537653765
5 : 0.5108175307084276	15 : 0.37224746592100666
6 : 0.4889536667689475	16 : 0.37124289195775795
7 : 0.4716916780354707	17 : 0.3682225365393682
8 : 0.4575984322109389	18 : 0.3533916849015317
9 : 0.44118653736451796	19 : 0.34308841843088417
10 : 0.4200939234381671	20 : 0.35724533715925394

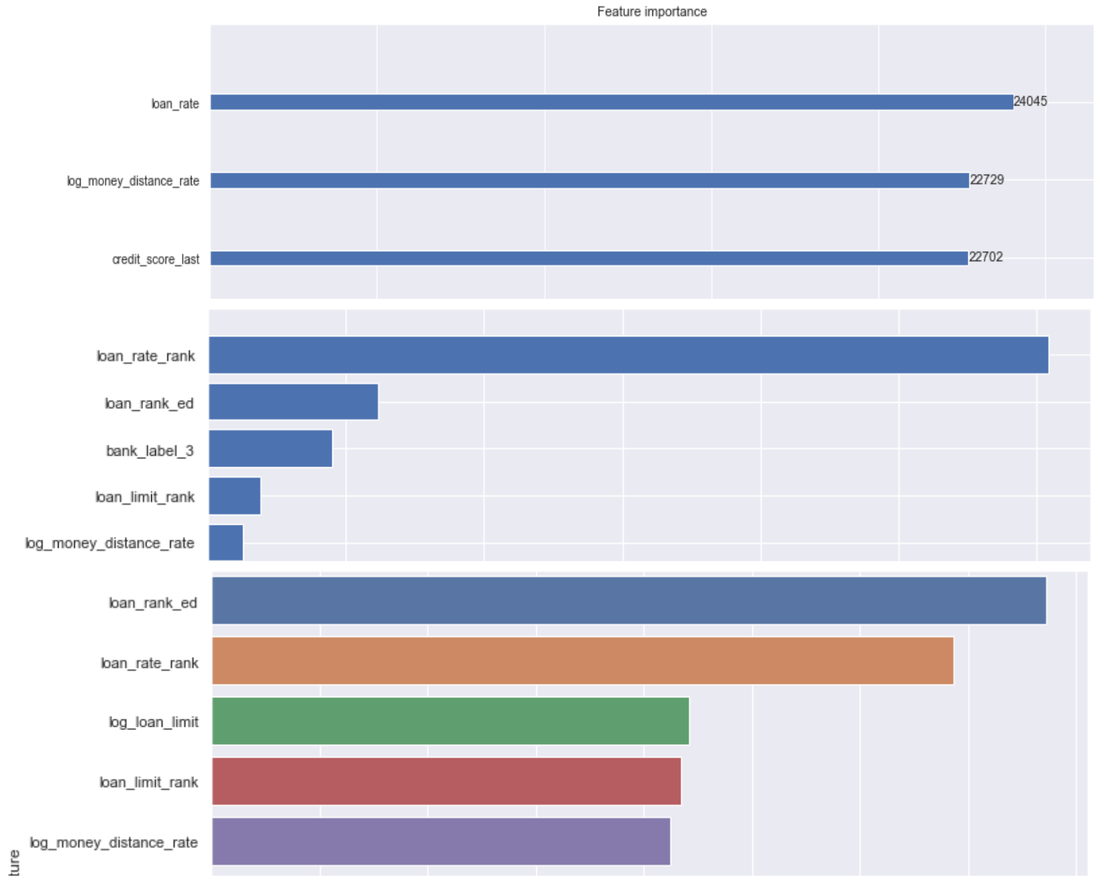
Time_rank가 작을 수록, 즉 첫 조회일수록 대출 신청하는 경향이 있다.

이를 통해, 첫 조회인 이용자들에 주목하여, 대출 신청을 경험하도록 유도하는 것이 중요함을 알 수 있다.

결론

예측 모델 해석 및 활용 방안

2차 모델링 변수 중요도



3가지 모델의 변수 중요도를 요약했을 때, loan_rate의 비율이 높음을 알 수 있다. 이는 이용자들이 대출 상품의 금리를 우선시한다는 것을 나타낸다.

또한 log_money_distance_rate : 대출 한도와 필요 금액의 차이 절댓값 log 변수의 중요도가 높음을 통해, 대출 상품의 한도는 필요로 하는 금액과 얼마나 차이가 나는지에 따라 대출 신청에 영향을 준다고 해석할 수 있다.

결론적으로 여러가지 대출 상품을 추천할 때, 이용자가 신청할 확률이 높은, 1. 최저 금리 상품, 2. 한도 목표 금액 근접 상품을 상단에 추천해줌으로써, 이용자들에게 보다 만족스러운 서비스를 제공할 수 있을 것이다.

결론

군집 결과 해석 및 활용 방안

1. 특정 군집 대상으로 맞춤형 깜짝 이벤트 준비

특정 군집을 대상으로는 맞춤형 이벤트를 제공하여 사용자들로 하여금 관심과 어플 서비스 참여를 독려한다.

예를 들어, 핀다에 대한 관심이 적은 최종 군집 3번에 대해서는 “출석 이벤트”를 진행하여 핀다에 참여를 독려한다. 이를 통해 해당 군집의 핀다 이탈율이 증가하는 속도를 늦추거나 막을 수 있다. 이 밖에도, 신규 유입층으로 분류된 2번 군집에 대해서는 “선착순 이벤트”를 활용해 다양한 기능에 대한 접근성을 자연스럽게 늘려준다면 충성 고객으로 발전할 가능성이 높다.

결론

군집 결과 해석 및 활용 방안

2. 충성 고객 확보

최종 군집 2번의 경우, 사용자들은 여러가지 서비스를 고르게 활용하고 신규 유입층 이기에 핀다에 관심을 가지는 충성 고객으로 발전할 가능성이 높다.

추가로 로그 데이터 분석에서 대출 조회변수와 대출 신청변수가 높은 군집이 어플을 사용하는 빈도가 높음을 확인하였다.

따라서, 이 군집의 사용자들에게 구체화된 메시지로 핀다의 강점인 대출조회로 유도한다면 충성 고객이 발전할 가능성이 높을 것이다.

또한, 최종 군집 4번의 경우, 현재 핀다 어플을 적극적으로 사용하는 충성 고객층이기 때문에 충성 고객층의 사용자들을 놓치지 않도록

사후 대출 관리서비스를 더 개선할 필요가 있다.

결론

군집 결과 해석 및 활용 방안

3. 신뢰할 수 있는 데이터 베이스 확보 기준 제안

신뢰성은 데이터분석과 데이터베이스 관리에 영향을 미치는 중요한 척도이다.

왜냐하면, 신뢰성이 만족되지 않는다면 이후 분석 결과는 모두 의미가 없기 때문이다.

1차 군집화에서 고객의 정보입력이 일관적인지 판단하고자 범주형 변수의 고유값의 개수를 분석에 활용했으며, 신뢰성을 평가하는 새로운 기준으로 채택했다.

위와 같은 개념으로 정보입력의 일관성을 체계화하여 핀다와 같은 핀테크 어플에 적용한다면 좋은 품질의 데이터베이스 관리에 도움이 될 것이다.

핀다 팀의 무궁한 발전을 기원합니다.

감사합니다.