사용자 특성과 대출 가심사 결과를 활용한 대출신청 여부 예측과 사용자 로그데이터를 활용한 고객 군집화

팀명

와빅병아리

팀장

이다영(leedy40259@naver.com)

팀원

강세정(sjkang6870@yonsei.ac.kr) | 박선종(ryan0507@yonsei.ac.kr)

박유찬(ucp0650@naver.com) | 이예림(dpfla_activity@daum.net)

1. Motivation

1 Motivation

1

[Prediction] 대출 신청 고객 예측

- 대출 신청 여부별 고객 특성 분석
- 실제 대출로 이어진 케이스 분석

2

[Clustering] 고객 군집 생성 & 군집별 서비스 메시지 제안

- 앱 기능 이용 활성화 및 고객 편의 증진 목적

2. Process

2 Process

Dataset

- 유저스펙테이블
- 대출상품결과테이블
- 유저로그데이터

- 외부 데이터

Experiments

- Machine Learning
  - CatBoostClassifier
  - LGBMClassifier
  - XGBClassifier

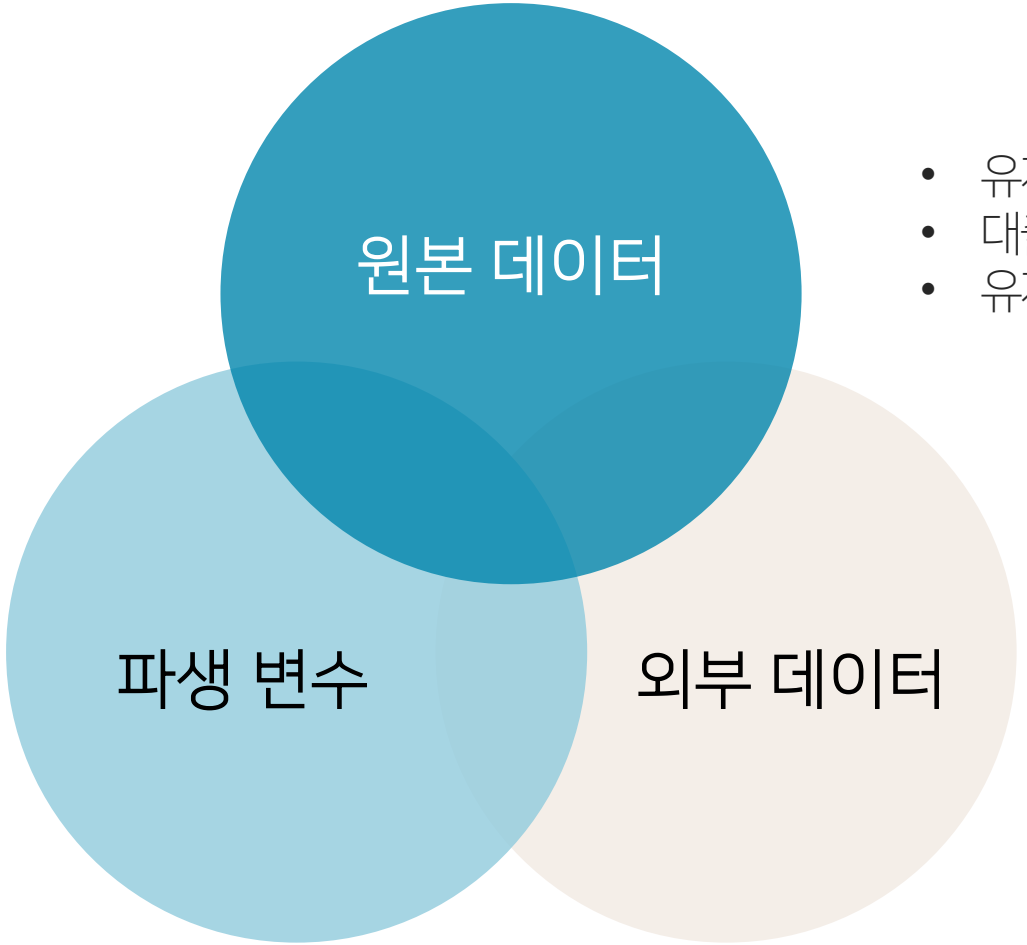
- Deep Learning
  - AutoEncoder

Model

- Machine Learning
  - LGBMClassifier

3. Data

3 Data



- 유저스펙테이블(user_spec)
- 대출상품결과테이블(loan_result)
- 유저로그데이터(log_data)

- [유저스펙테이블+대출상품결과테이블] application_id 별 loan_limit, loan_rate 의 rank
- [대출상품결과테이블] bank ratio
- [유저로그데이터] log당 count
- [유저로그데이터] user log count

- KOSPI

4. Preprocessing

4 Preprocessing

변수 선택 & 결측치 처리

- 'application_id', 'user_id' : 범주가 많은 변수는 overfitting의 위험이 있어 제외
- 'personal_rehabilitation_yn', 'personal_rehabilitation_complete_yn' : 결측치가 많았고, 여러 실험에서 해당 변수들을 제외했을 때 성능이 더 좋았음

파생변수 생성

- 'loan_limit_rank', 'loan_rate_rank' : 앱 화면상 대출 상품 정렬 기준을 반영하고자 생성
- 'bank ratio' : 은행 선택 기준
- 'log_length' : 앱 접속 빈도와 대출 신청 여부의 관련성을 가정하고 생성
- log별 column : log별 count와 대출 신청 여부의 상관관계를 가정하고 생성

외부 데이터 활용

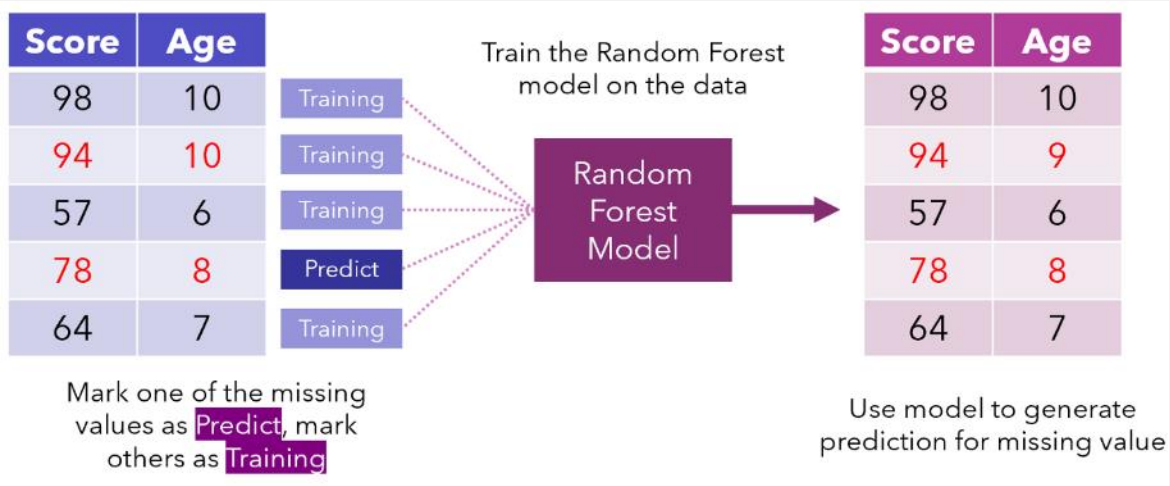
- KOSPI : 경기 변동으로 인한 대출 심리 변동을 고려하여, 경기 변동을 가장 잘 나타내는 지표인 KOSPI를 수집하여 예측에 활용

5. Experiments

5 Experiments

Preprocessing 기법

1. **결측치 처리**
   - 결측치 처리를 위해 **miss-forest** 기법 시도



2. **범주형 변수 OverSampling**
   - Unbalanced Target 문제를 해결하기 위해 Oversampling 을 시도했는데, 범주형 변수의 경우는 CT-GAN으로 수행

3. **유저별 대출 신청 여부 probability 및 bank 선호도 계산**

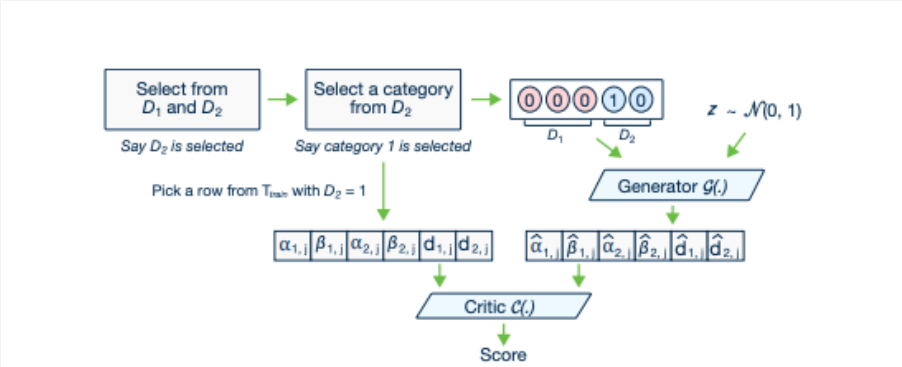
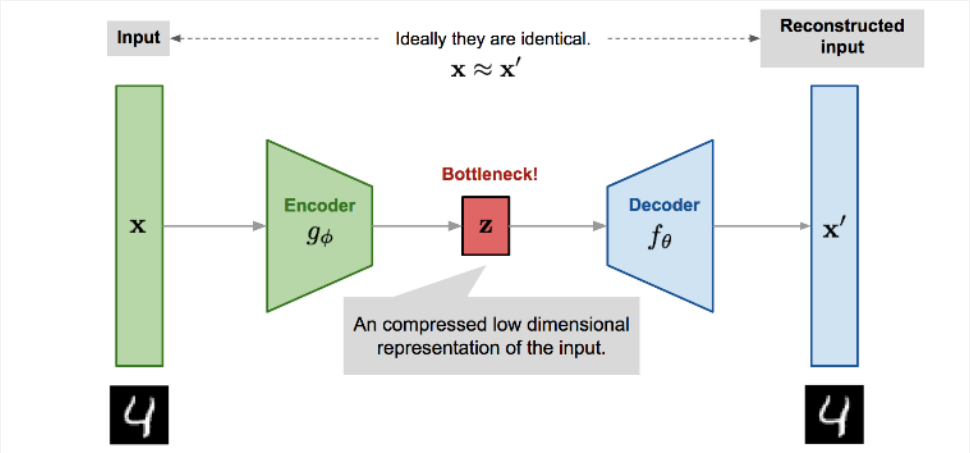


Figure 2: CTGAN model. The conditional generator can generate synthetic rows conditioned on one of the discrete columns. With training-by-sampling, the *cond* and training data are sampled according to the log-frequency of each category, thus CTGAN can evenly explore all possible discrete values.

5 Experiments

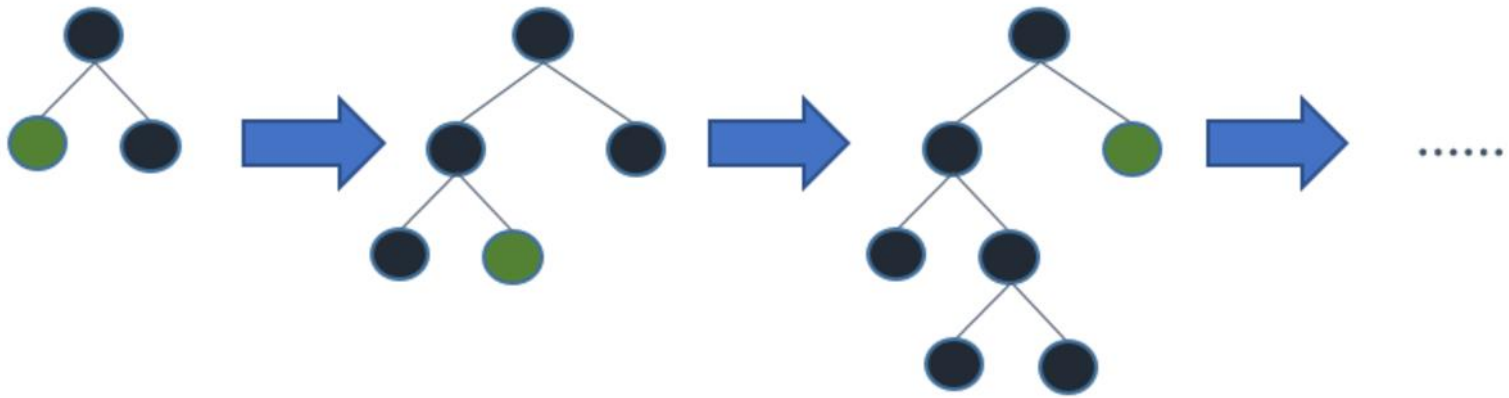
Models (ML / DL)

1. **다양한 ML 모델 적용**
   - 최종 모델은 LGBMClassifier
   - 이외에 RandomForest, CatBoost, XGBoost 등다양한 머신러닝 모델들을 적용하여 최적의 모델을 탐색함
2. **AutoEncoder**
   - 대출 상품 신청을 이상 행동이라고 간주하여 이를 탐지하는 AutoEncoder 모델을 구축
   - 비교적 좋은 성능을 보였으나, 최종 모델 성능에는 미치지 못했음



6. Model

6 Model



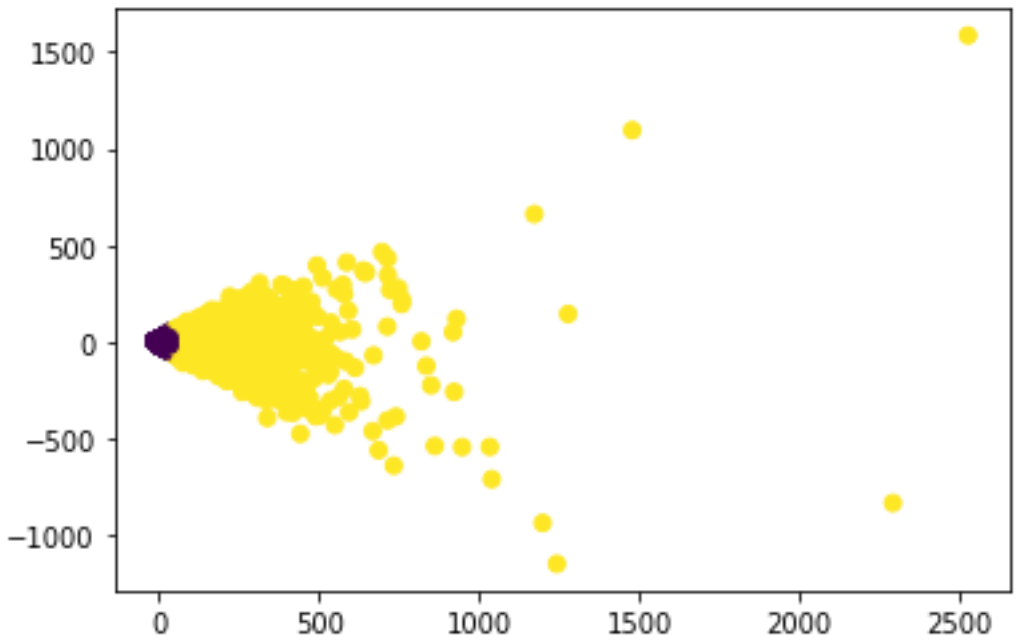
Leaf-wise tree growth

- Gradient Boosting Model은 여러개의 Tree를 만들고, Loss Function을 작아지는 방향으로 가공하기 때문에 높은 성능의 결과를 얻을 수 있다.
- GBM 방식의 모델에는 LightGBM과 XGBoost가 존재하며, leaf-wise 방향으로 확장하는 LightGBM의 경우 Prediction Error 값을 더욱더 최소값을 만들 수 있다.
- LightGBM은 Training에 소요되는 시간에 비해 훨씬 좋은 성능을 보이기 때문에, XGBoost 대신 LightGBM을 머신러닝 기법의 최종 모델로 선정하였다.

7. Clustering

7 Clustering

베이스라인 모델

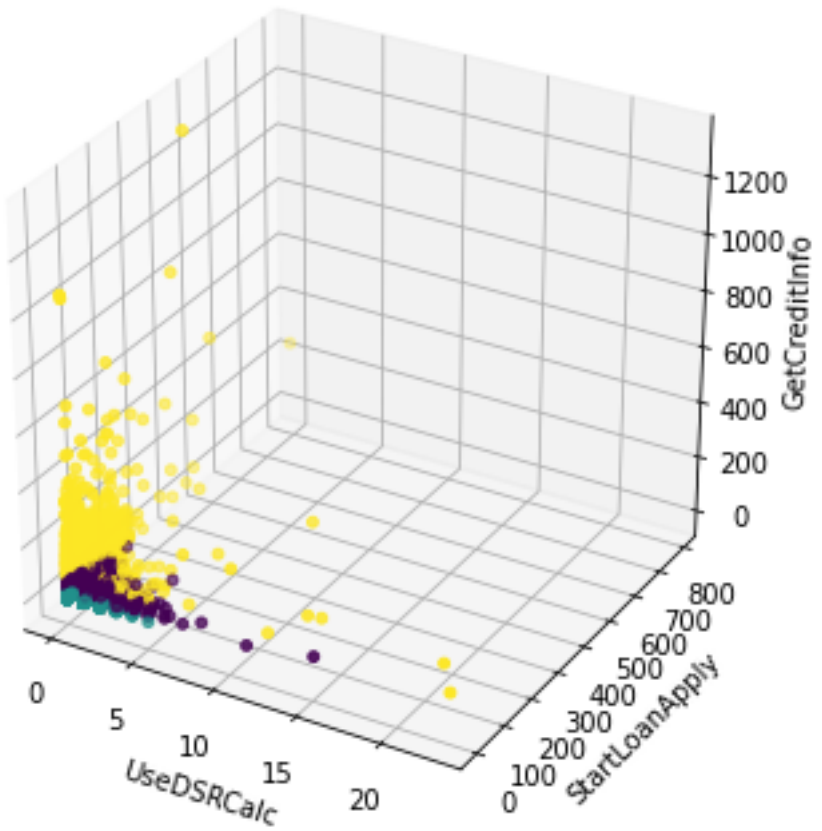


PCA + K-means Clustering

1. 대표적인 고차원 데이터의 군집화 방법인 PCA 차원 축소와 K-means clustering을 이용해 군집화 시도
2. 앱 사용 목적별로 클러스터링되지 않고, log가 적은 사람과 log가 많은 사람들(앱을 자주 이용하는 사람들)로 군집 생성
3. 앱 사용자들이 앱에서 제공하는 기능들을 골고루 이용하는 것이 아니라 한도 조회를 위주로 이용하기 때문 → Feature Extraction이 아닌 Feature Selection Approach를 채택

7 Clustering

최종 모델



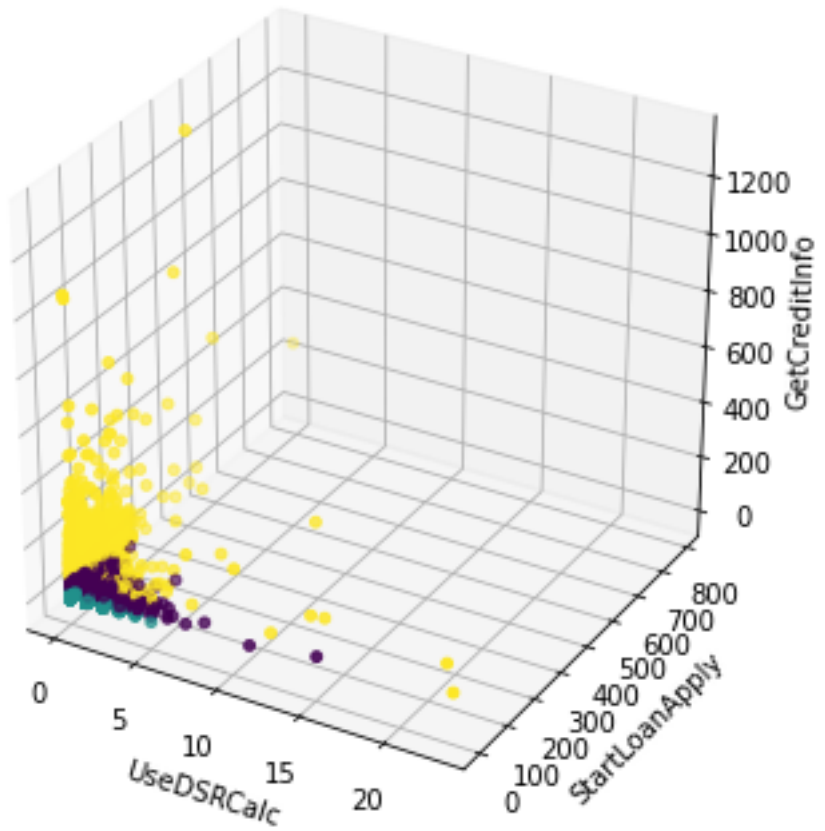
Feature Selection + K-means Clustering

1. Feature Selection
- 핀다가 제공하는 기능을 3가지 —마이데이터, 신용조회, 대출신청— 로 분류
- 각 기능을 대표할 수 있는 Feature 추출 — UserDSRCalc, GetCreditScore, StartLoanApply

2. K-Means Clustering 실시

7 Clustering

Clustering 결과에 따른 서비스 메시지 제안



Group 1. 신용정보조회

- 신용정보 조회를 비롯해 어플을 활발하게 이용
- "내 신용등급에 최상의 조건을 제공하는 상품 보러가기"

Group 2. 대출신청시작

- 신용등급 조회 이전 대출신청 시작 페이지에서 망설이는 고객군
- "핀다와 함께라면 두렵지 않죠. 대출 신청 마저 하러 가기"

Group 3. DSR 계산기

- 마이데이터 연결을 통해 DSR을 계산한 고객군
- "연동된 대출계좌로 나에게 꼭 맞는 금융혜택 보러가기"

8. Improvements

8 Improvements

Task 1

Limitation [Prediction]

- 대출을 신청할 probability에 대한 threshold를 일일이 지정
  - 현재로서는, 미래 대출 신청 비율 예측 과정이 필수
  - threshold 선정 근거 설득력 부족

- 직관적으로 중요한 변수를 활용하지 않은 점
  - 개인회생자 여부('personal_rehabilitation_yn') 및 개인회생자 납입 완료 여부('personal_rehabilitation_compete_yn')는 대출 신청 여부와 상관관계가 있을 것으로 기대
  - 본 프로젝트에서는 두 변수 모두 분석 과정에서 제외

8 Improvements

Task 1

Future Work [Prediction]

- 대출 신청 여부에 영향을 미치는 외부 데이터 추가 수집
- 대출 신청 비율 예측에 기반한 threshold 조정
- Unbalanced Data 분석을 위한 다양한 기법 시도

THANK YOU