

유전 알고리즘을 활용한 이탈유저 예측 및 결제고객 파악

Team BMS
강경수, 진교훈



NCSoft®



Contents

I. Approach Problem

분석 방향 설정

II. Analysis 1

전체 데이터 파악 및 전처리

III. Analysis 2

세부 데이터별 파악 및 전처리

IV. Modeling 1

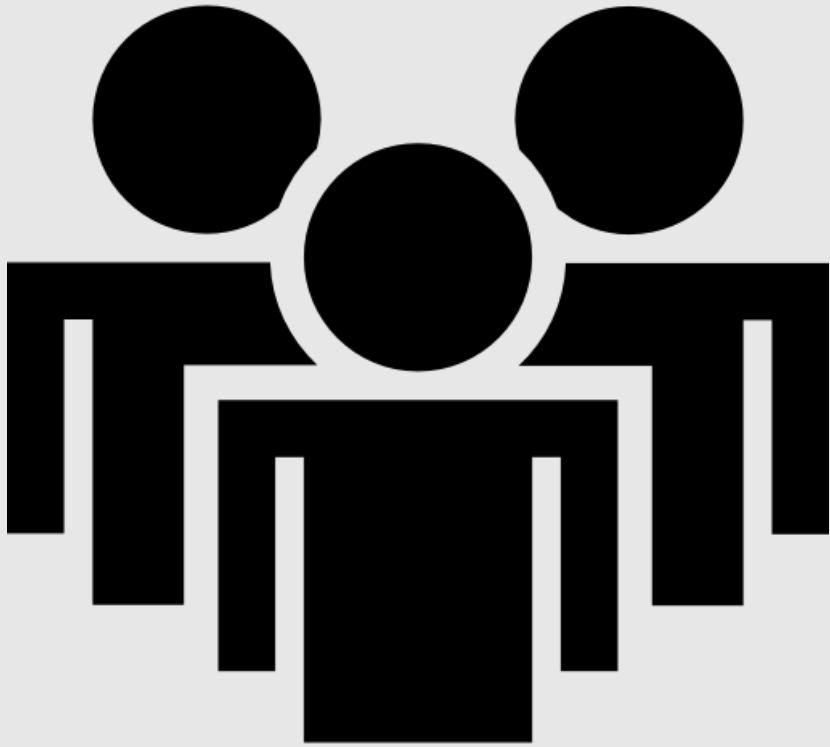
유전 알고리즘을 활용한 파라미터 튜닝 및 예측모형 구축

IV. Modeling 2

추가적인 분석 진행

VII. Total Conclusion

최종 결론 및 느낀점



Team Introduce

Team Introduce



강경수

(국립창원대학교 경영학과 박사수료)

- 관심분야: 일정계획, 조합 최적화, 메타 휴리스틱, 데이터 분석



진교훈

(가천대학교 응용통계학과 학사과정)

- 관심분야: 객체 탐지(R-CNN), 이미지 분류, 앙상블 기법, 데이터 분석



I. Approach Problem

Description



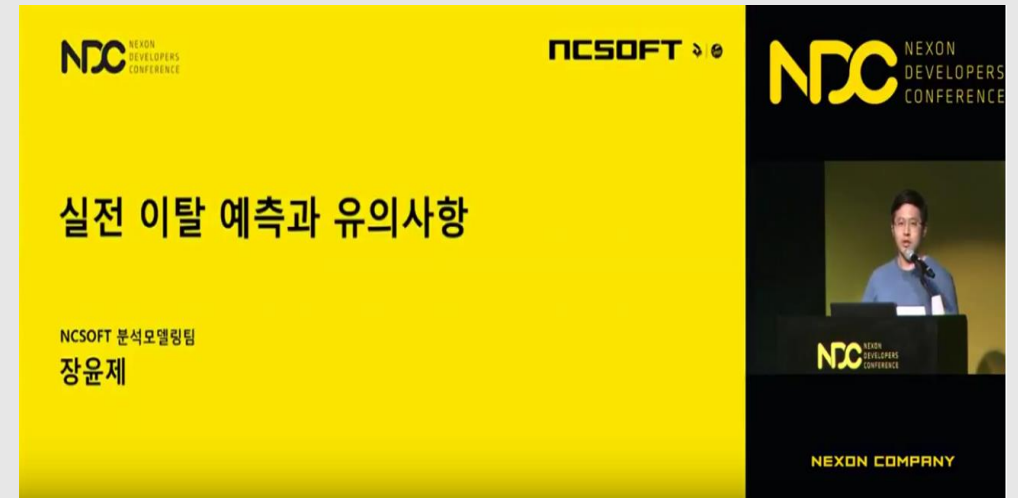
2018년도 빅콘테스트 분석분야 중 챔피언리그 주제는
'블레이드 앤 소울' 게임 유저 이탈 예측 모형 구축 및 이탈/비이탈 원인 분석

Analysis Planning

[NDC2018]

실전 이탈 예측과 유의사항 – 장윤제

- **이탈예측을 하는 이유는?**
 - 데이터를 기반으로 한 의사결정
- **이탈예측이 중요한 이유는?**
 - 신규 유저 유입 비용보다 기존 유저 유지의 비용이 적음



https://www.youtube.com/watch?v=kcE_1n41xdk

Analysis Planning

고객이탈이 기업에 미치는 영향?

- 기업 수익의 손실
- 기업 이미지 하락

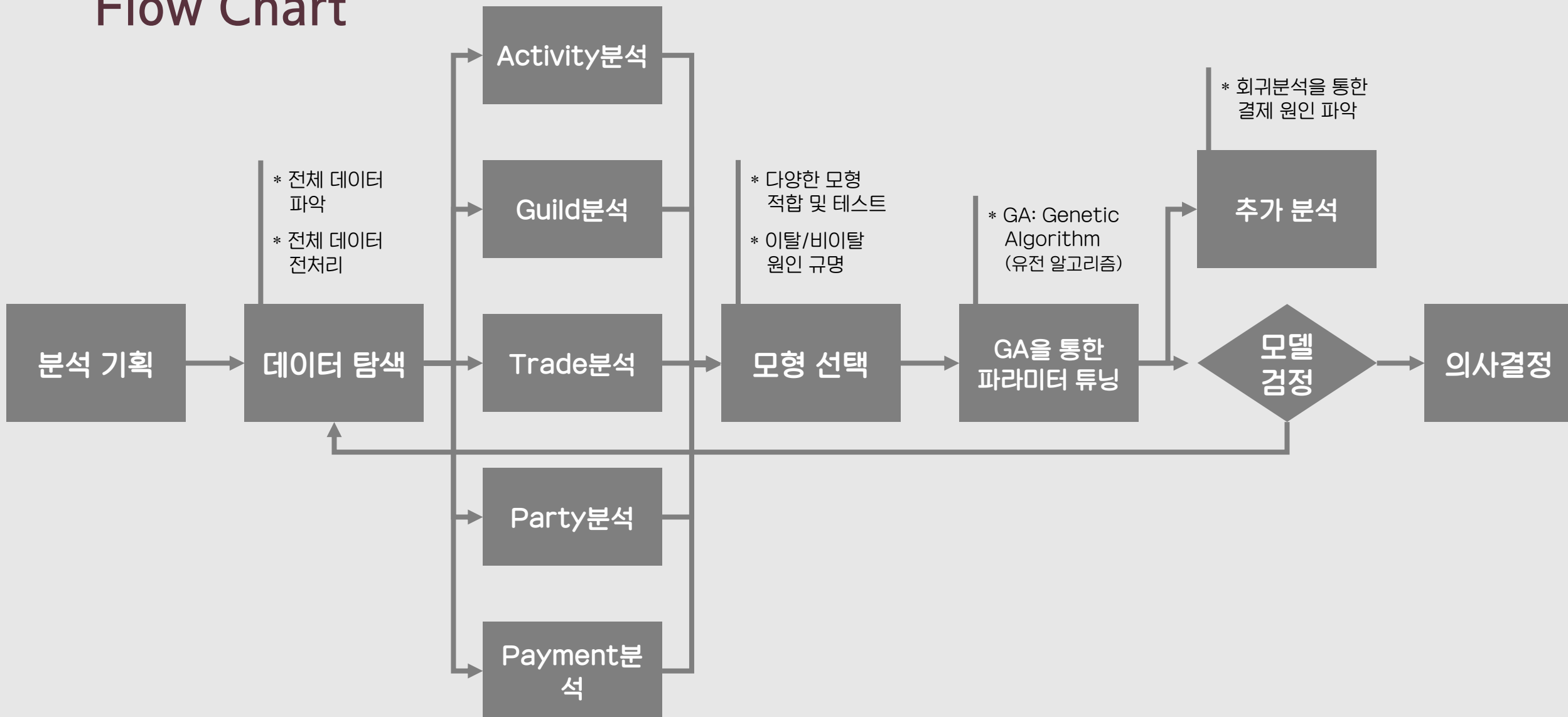
고객이탈을 방지하기 위해서는?

- 고객 이탈의 원인 파악
- 정확한 고객이탈 예측



➔ 기업 수익에 관심을 두고 분석을 진행

Flow Chart





II. Analysis 1

Analysis 1 Flow Chart



전체 데이터 파악

- 변수 확인
- 결측치 확인
- 데이터 개수 확인

목표변수 파악

- 목표변수의
형태 및 분포 확인
- 데이터불균형 확인

특징변수 파악

- 특징변수의
형태 및 분포 확인
- 표준화 여부 확인
- 시계열성 확인
- 이상치 확인 등-

Data Describe



Activity

- Train Data
Columns: 38
Rows: 440323
NA's: 0
- Test Data
Columns: 38
Rows: 175631
NA's: 0



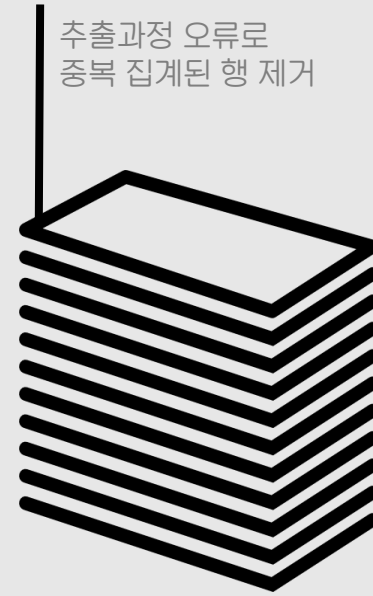
Guild

- Train Data
Columns: 2
Rows: 9963
NA's: 0
- Test Data
Columns: 2
Rows: 5906
NA's: 0



Party

- Train Data
Columns: 7
Rows: 6962341
NA's: 0
- Test Data
Columns: 7
Rows: 4121512
NA's: 0



Trade

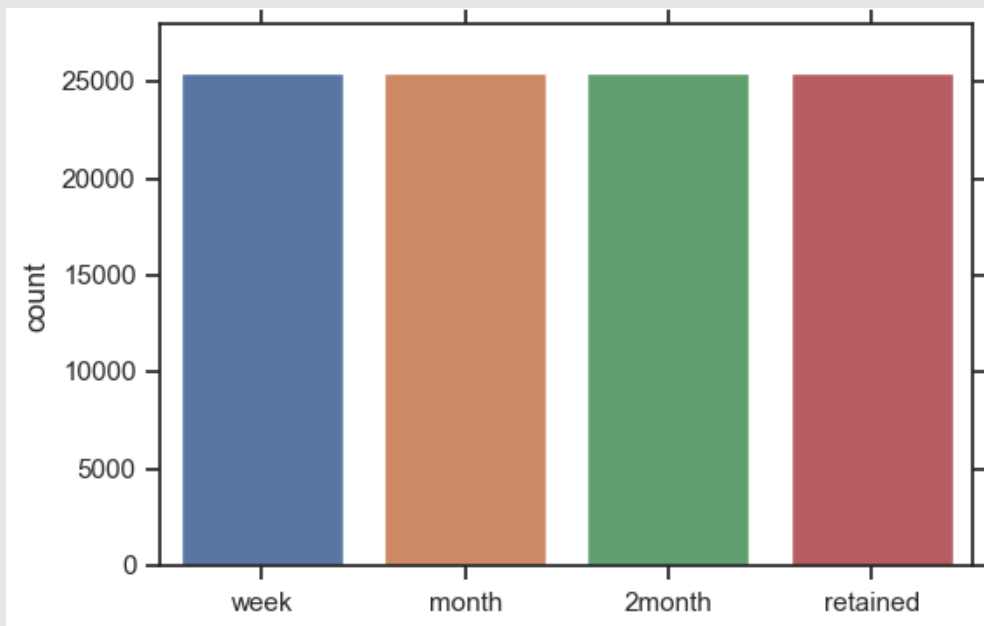
- Train Data
Columns: 7
Rows: 10414351
NA's: 0
- Test Data
Columns: 7
Rows: 3873536
NA's: 0



Payment

- Train Data
Columns: 3
Rows: 800000
NA's: 0
- Test Data
Columns: 3
Rows: 320000
NA's: 0

Target Variable

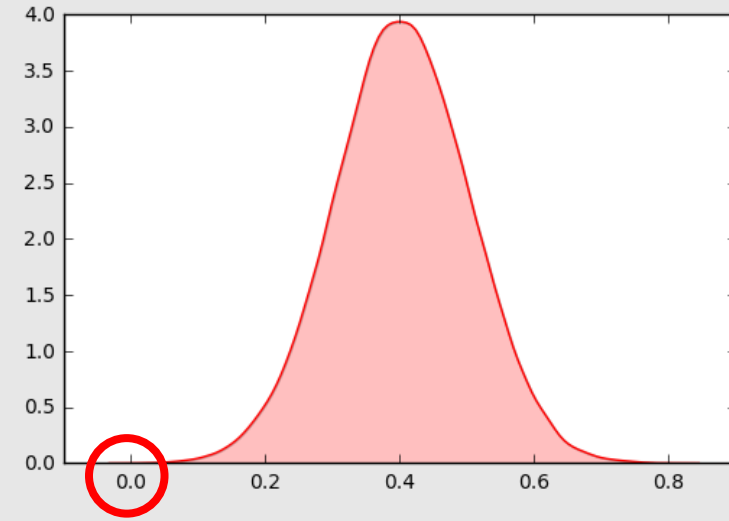
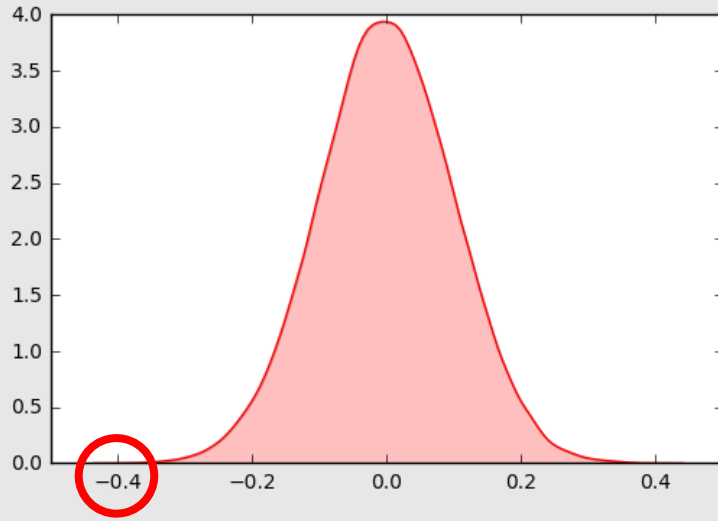


〈데이터 내 Label 분포〉

Train_Label

- Week, month, 2month, retained
- 각 25,000개씩 분포
- Sampling데이터기 때문에
실제데이터가 어떤 분포인지 알 수 없음

Explanatory Variable



- 데이터는 이미 표준화된 상태
 - 합계나 나누기 같은 연산에 영향을 미치지 때문에 전처리 수행
 - 축을 옮겨서 최소값이 0이 되도록 맞추어줌

Explanatory Variable

최소값 보정

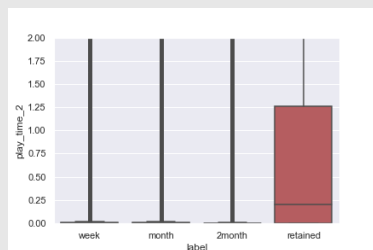
- 플레이 시간은 접속횟수가 아니라면
최소값이 0이 될 수 없음
- 때문에 최소값이 0.01이 되돌고 보정
- 전투참여시간도 마찬가지로 이유로
최소값을 보정



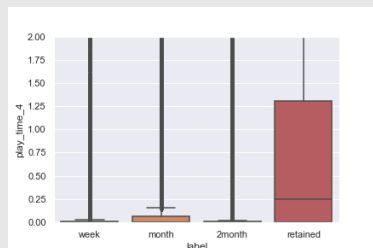
Explanatory Variable

주어진 데이터는 8주간 각 활동이 기록된 로그데이터

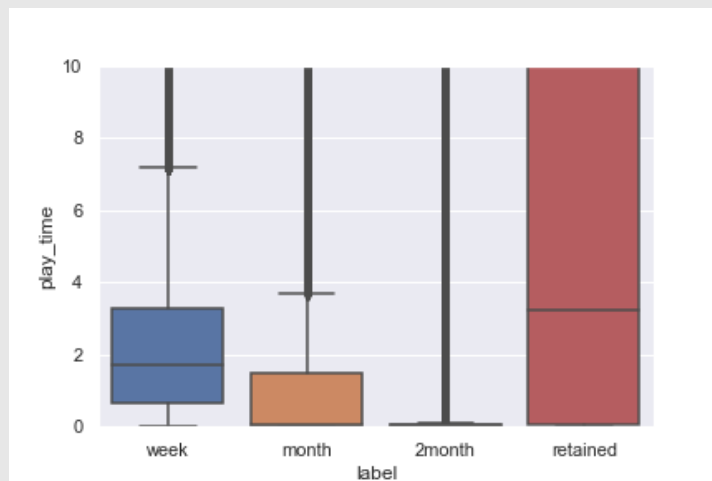
- acc_id기준으로 처리시 시계열 정보 확인이 어려움
- 변수들을 주차 별로 볼 필요가 있음



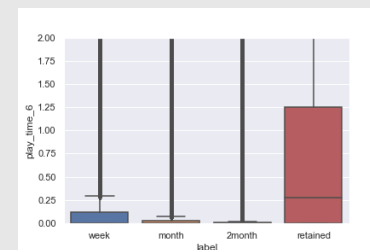
〈플레이 시간 2주차〉



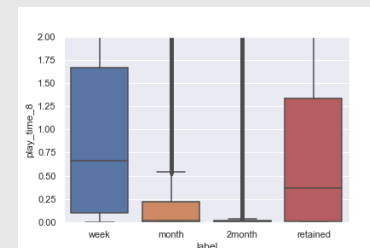
〈플레이 시간 4주차〉



〈플레이 시간 총합〉



〈플레이 시간 6주차〉



〈플레이 시간 8주차〉

Explanatory Variable

- 시간 간격이 일주일로 매우 넓음
- 8간격의 데이터로 시계열 분석도 용이하지 못함

→ 전체 데이터를 1열로 펼쳐서 분석 진행

아이디	주차
A	1주차
A	2주차
A	3주차
A	...



	1주차	2주차	...
A			
B			
C			
...			



III. Analysis 2

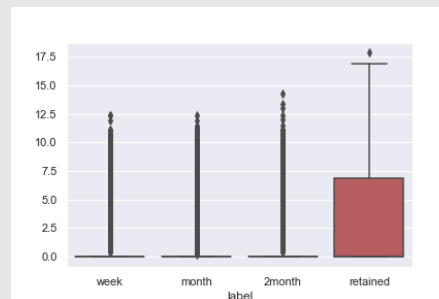
Activity Data



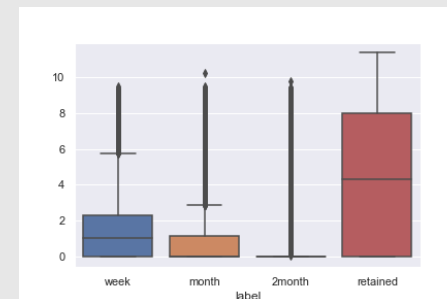
Activity Data

Activity Data

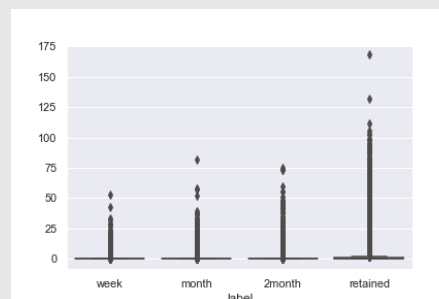
- **입장횟수와 승리횟수**
 - 승리횟수는 입장횟수에 비례해서 증가
 - 입장횟수와 승리횟수의 상관계수는 0.9
→ 변수를 줄이거나 파생변수 생성 필요
→ 다중공선성 문제
- **승률 데이터 생성**
 - **승리횟수/입장횟수**
→ 입장횟수에 따른 승리횟수 확인



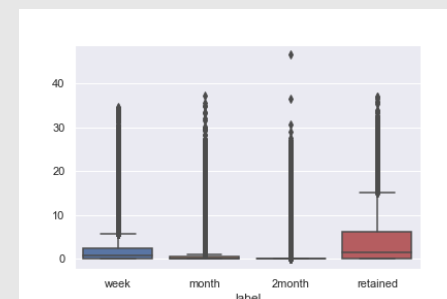
〈Solo Inzone Clear Rate〉



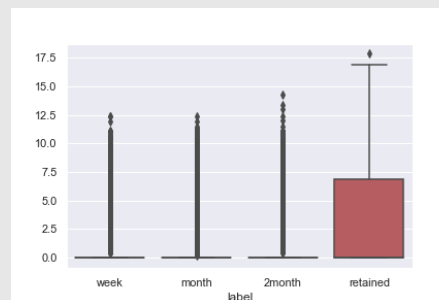
〈LightInzone Clear Rate〉



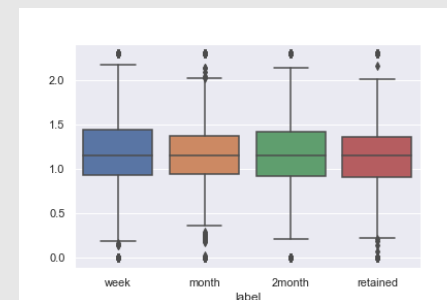
〈SkilledInzone Clear Rate〉



〈Normal Inzone Clear Rate〉



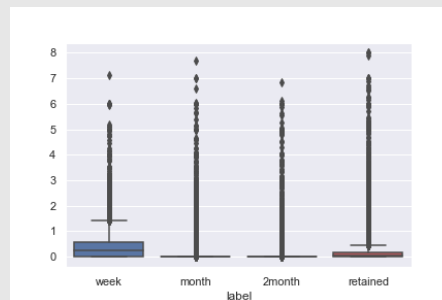
〈Raid Clear Rate〉



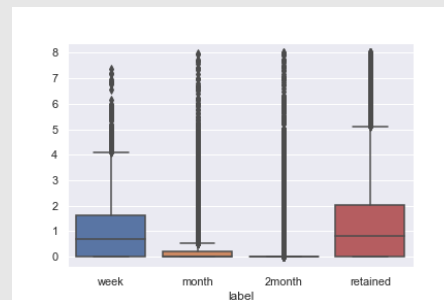
〈Duel Win Rate〉

Activity Data

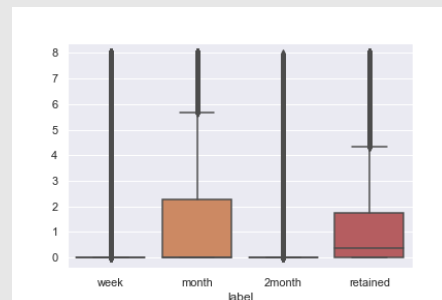
- **다양한 채팅변수**
 - 채팅데이터도 횟수 그 자체로도 의미가 있지만 이 또한 비율로 봐도 유의할 것이라고 판단
- **채팅 비율 데이터 생성**
 - **해당채팅/전체채팅**
→ 전체채팅 횟수에 따른 개인 채팅 확인



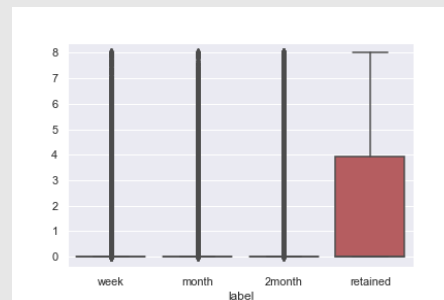
〈Normal Chat Ratio〉



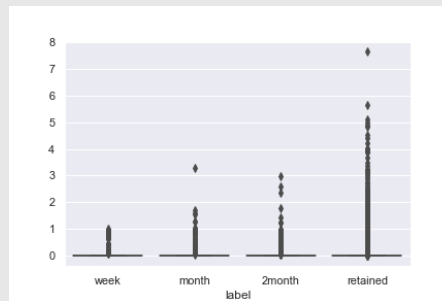
〈Party Chat Ratio〉



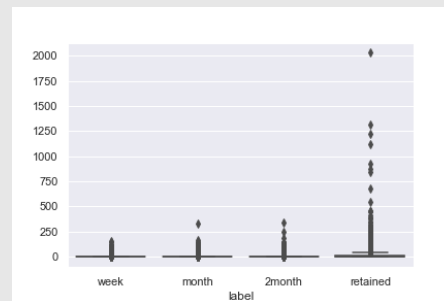
〈Whisper Chat Ratio〉



〈Guild Chat Ratio〉



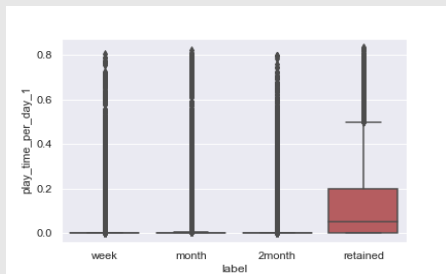
〈Faction Chat Ratio〉



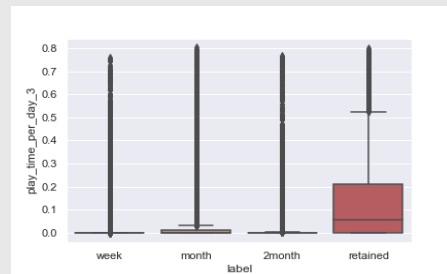
〈All Chat〉

Activity Data

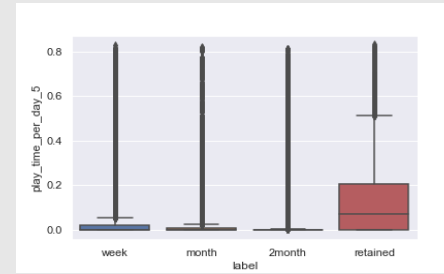
단순한 플레이시간이 아닌 일별 플레이 시간 데이터 생성



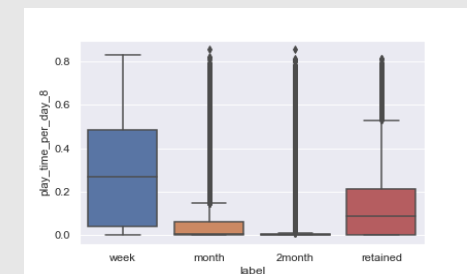
<1주차 일별 플레이시간>



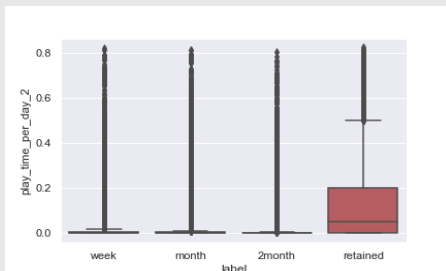
<3주차 일별 플레이시간>



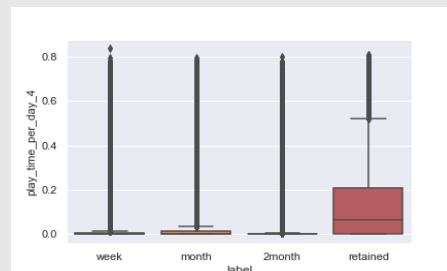
<5주차 일별 플레이시간>



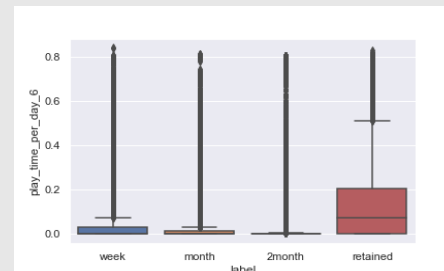
<7주차 일별 플레이시간>



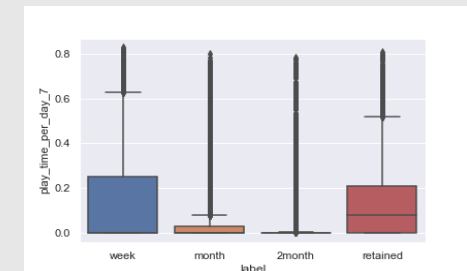
<2주차 일별 플레이시간>



<4주차 일별 플레이시간>



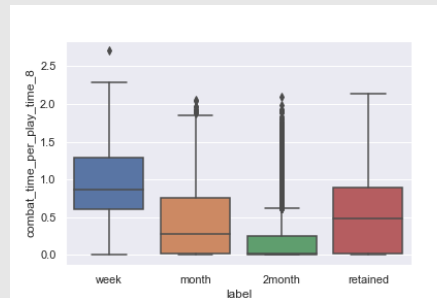
<6주차 일별 플레이시간>



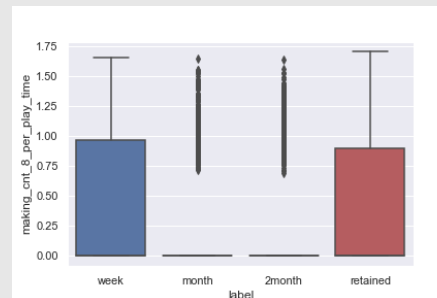
<8주차 일별 플레이시간>

Activity Data

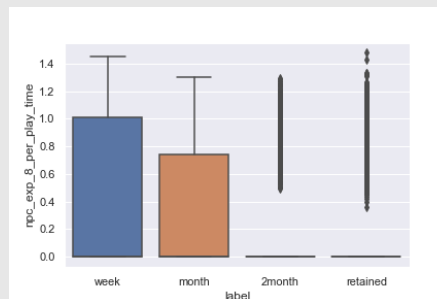
- **Activity데이터들의 특징**
 - Activity데이터들은 플레이시간이 많다면 필연 증가함
- **시간 대비 Activity데이터 생성**
 - **해당 Activity데이터/플레이 시간**
→ 플레이 시간에 따른 해당 Activity 확인



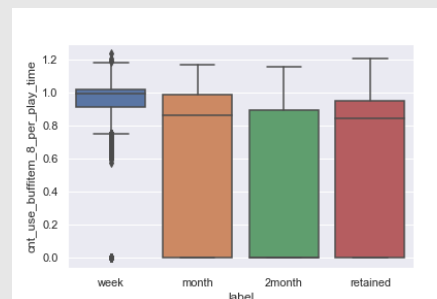
〈Combat Time per Play Time〉



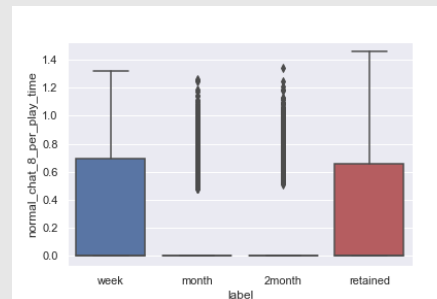
〈Making CNT per Play Time〉



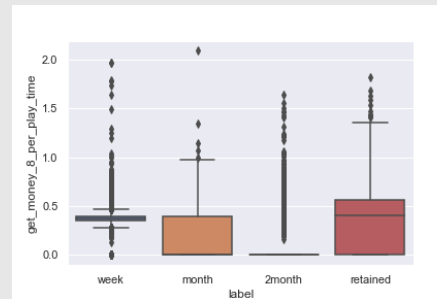
〈NPC_Exp per Play Time〉



〈Buffitem per Play Time〉



〈Normal Chat per Play Time〉



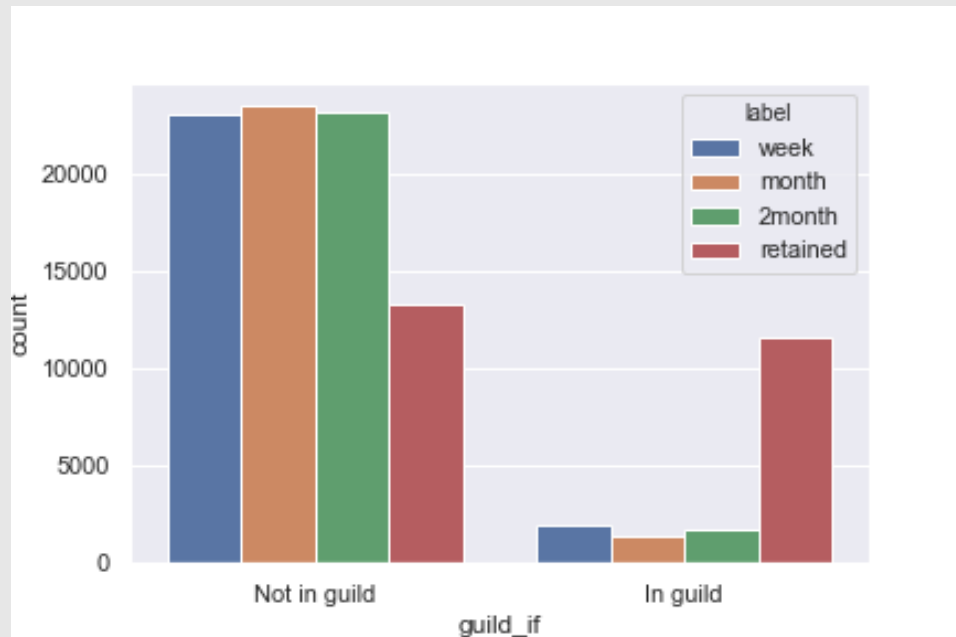
〈Get Money per Play Time〉

Guild Data



Guild Data

Guild Data

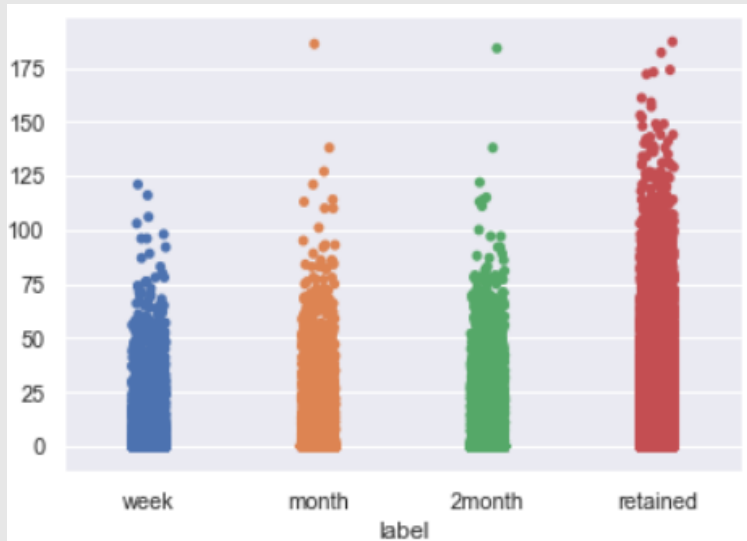


<Label별 길드가입여부>

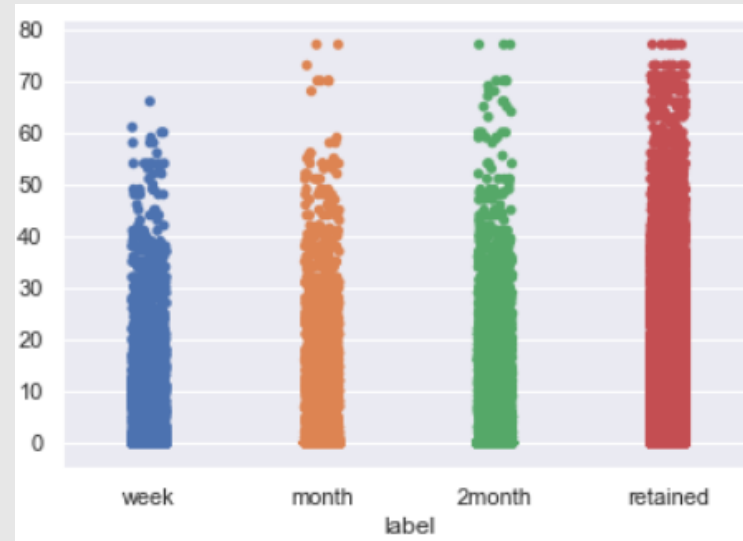
길드 가입 여부

- Week, month, 2month간에는 큰 차이 없음
- Retained와는 확연히 차이남
- 길드 가입 유저들의 대다수가 비이탈 유저

Guild Data



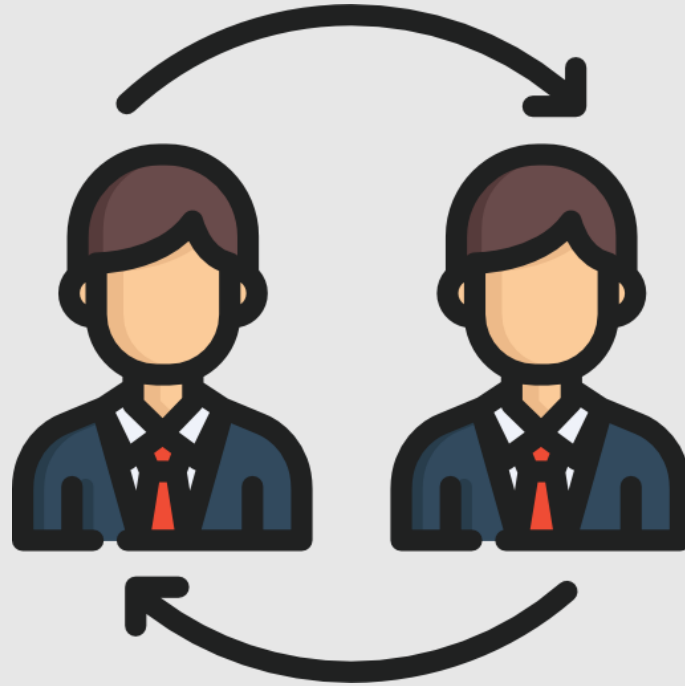
〈아이디 별 길드원 수 합계〉



〈아이디 별 길드원 수 평균〉

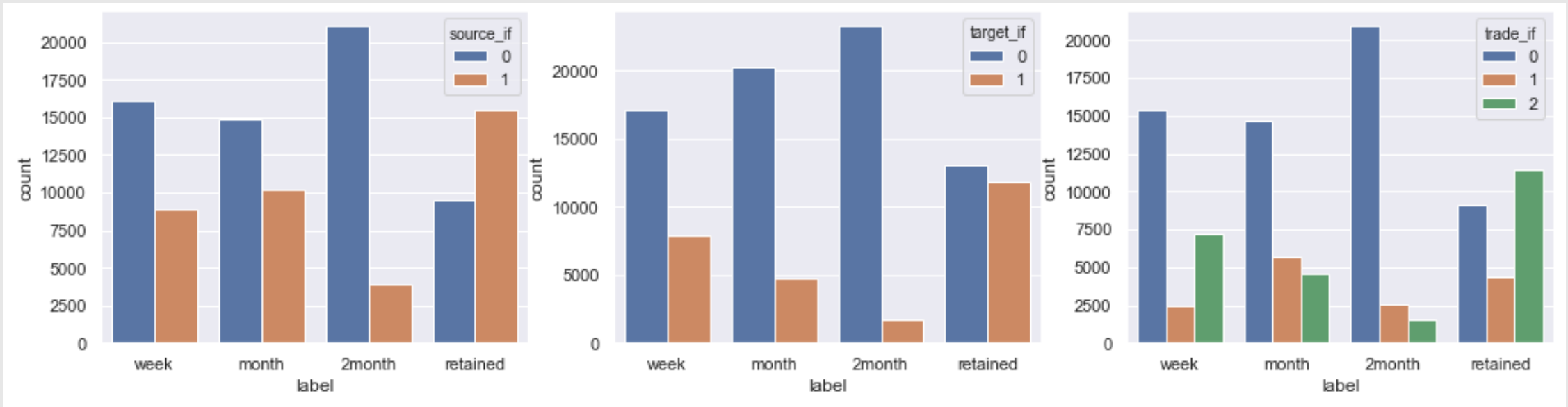
- 길드원 정보를 제외한 다른 길드 정보는 없음
 - 한 아이디는 여러 서버의 길드에 가입할 수 있음
 - 해당 아이디가 속해있는 길드의 규모를 보기 위해서
소속된 길드의 멤버수 합과 평균 데이터 생성

Trade Data



Trade Data

Trade Data



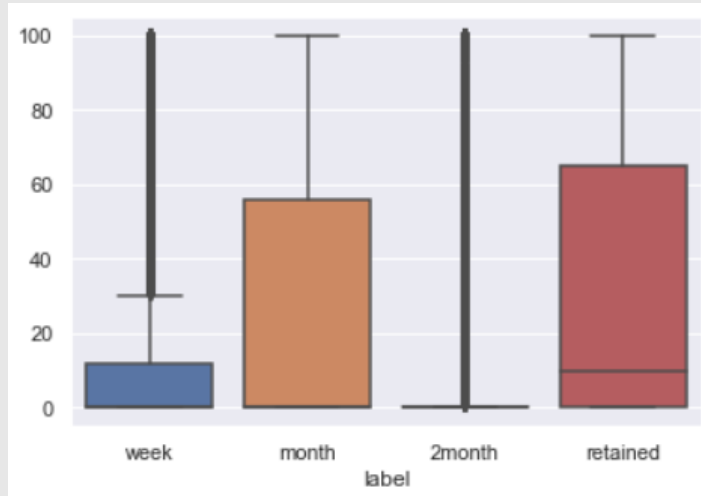
〈Label별 주는 거래 여부〉

〈Label별 받는 거래 여부〉

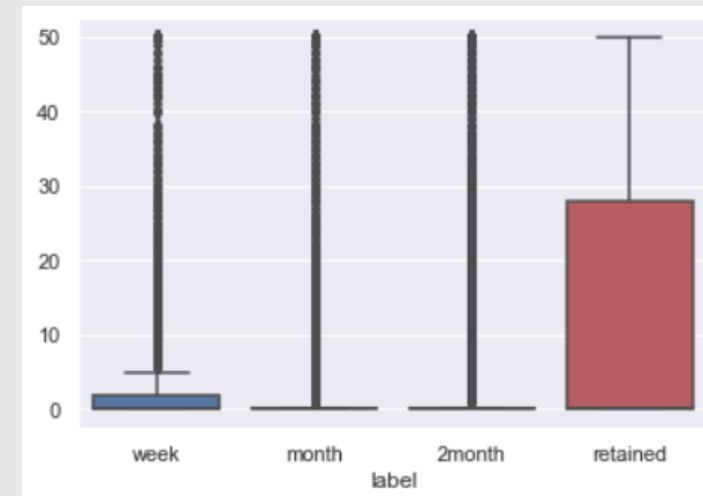
〈Label별 거래 여부〉

- Source 및 Target거래 여부 데이터 생성
 - 2Month가 대체적으로 거래 횟수가 적음
 - Retained가 대체적으로 거래 횟수가 많음

Trade Data



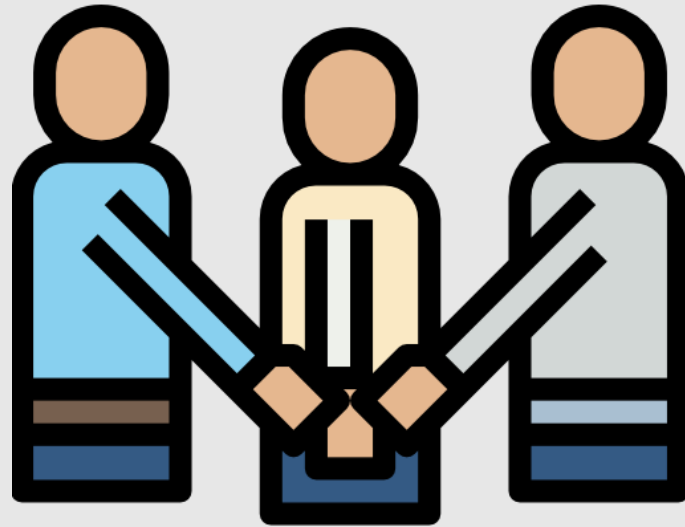
〈Label별 주는 거래 횟수〉



〈Label별 받는 거래 횟수〉

- Source 및 Target거래 횟수 데이터 생성
 - 거래 여부 데이터와 비슷하지만 Month의 거래량도 많음
 - Month데이터가 주는 거래는 많지만 받는 거래는 없음
 - 게임 이탈 전 개인 아이템 정리?

Party Data



Party Data

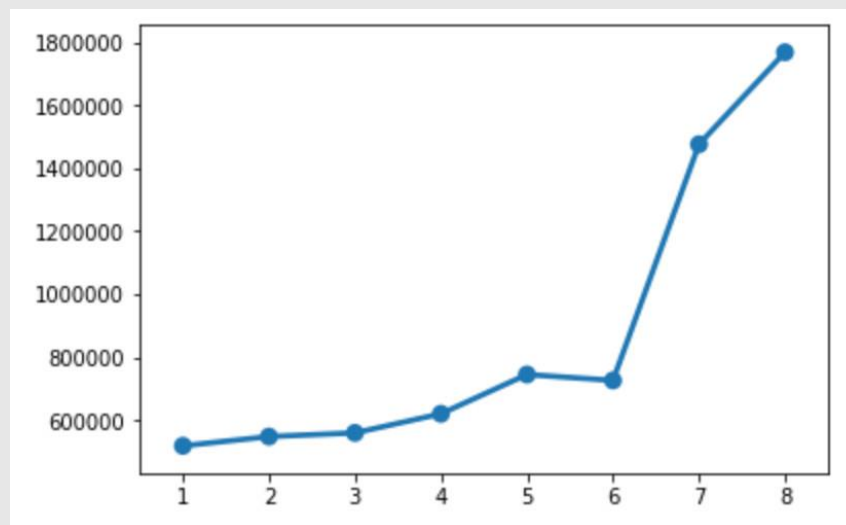
Party Data

- Party데이터를 통해 알고 싶었던 것
 - 얼마나 많은 파티 플레이를 하였는가
 - 얼마나 긴 시간 동안 파티 플레이를 하였는가
- Party데이터 내 특이값
 - 지속시간이 7일인 파티
 - 파티 멤버가 400명이 넘는 파티
 - 파티자체가 지속된 시간만큼의 데이터이기 때문에 생기는 문제점



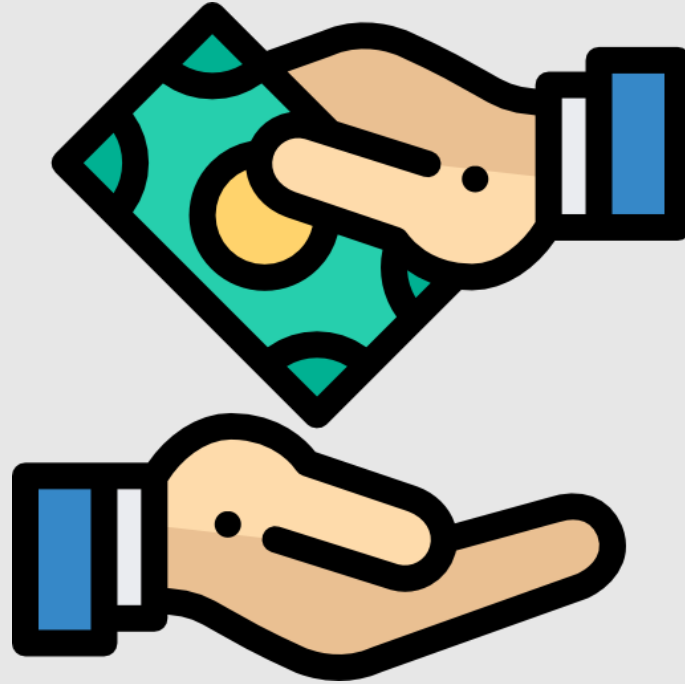
→ 알고 싶었던 내용에 대한 당위성 부족

Party Data



- 7주 이전 데이터와 이후 데이터의 차이가 극명함
 - 날짜 정보 부재로 이벤트 여부나 외부적 요인 파악 불가
 - 7-8주차에 지수적으로 증가한 원인 분석 불가

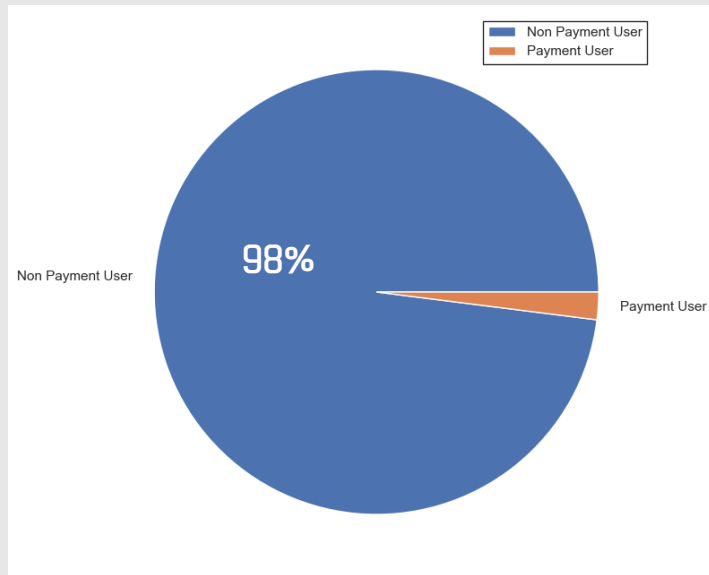
Payment Data



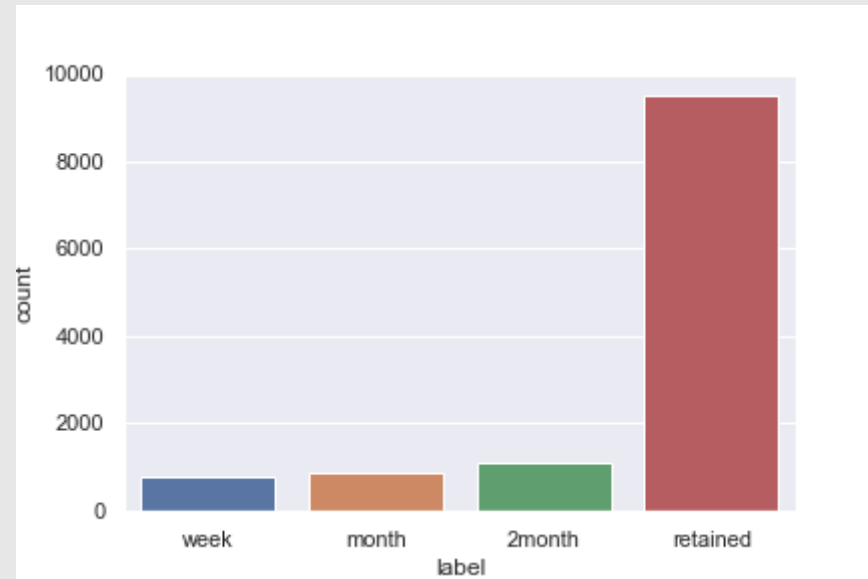
Payment Data

Payment Data

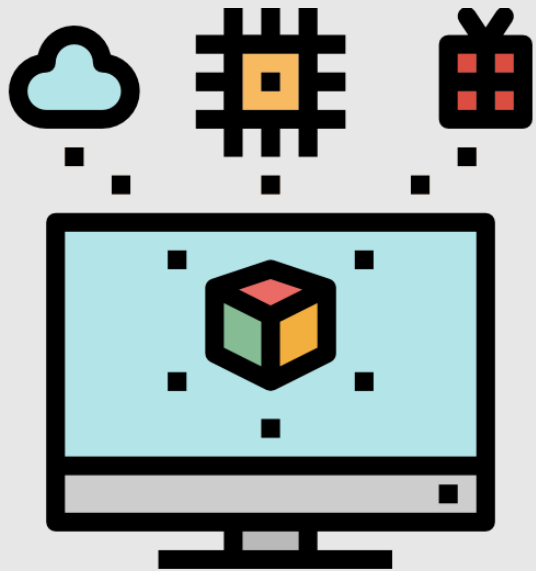
- 전체 데이터의 단 2%만이 결제이력이 있는 데이터
- 결제하는 유저들 대다수가 Retained
- 차후 분석을 통해서 상세히 살펴볼 예정



〈결제 이력 여부〉



〈Label별 결제 이력 여부〉

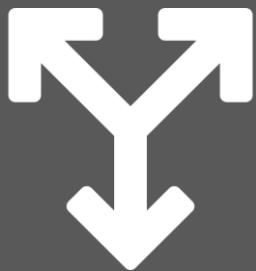


IV. Modeling 1

Modeling Flow Chart



LLR을 통한
이탈/비이탈
중요 변수 파악



Label별
모형 적합



전체
모형 적합



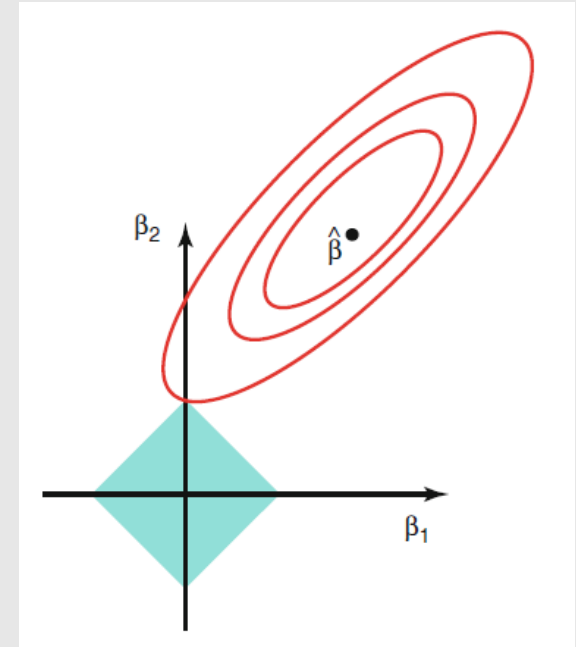
GA을 통한
파라미터 튜닝



변수 중요도를
통한
중요 변수 파악

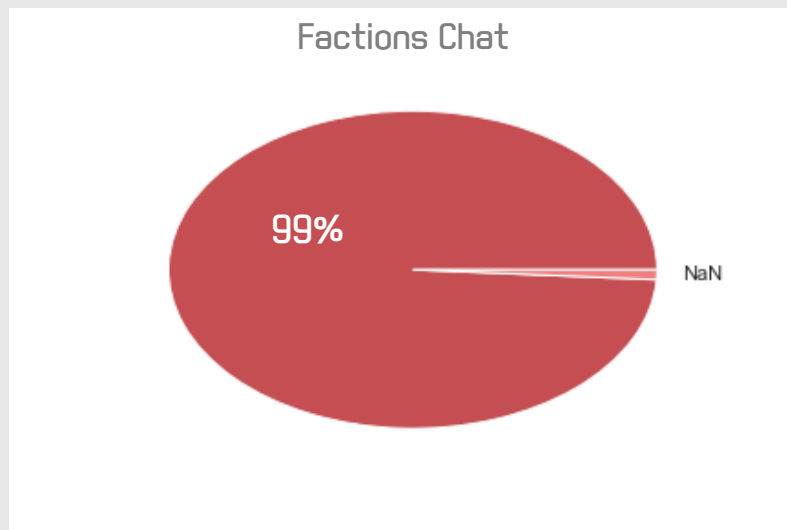
Lasso Logistic Regression

- 이탈/비이탈 유저간의 차이 파악을 위한 LLR실시 (LLR: Lasso Logistic Regression)
 - 어떠한 변수가 제일 중요한지 확인
 - 유의하지 않은 변수 제거
- 유의한 변수
 - 일별 플레이시간, 전장참여 시간, 플레이시간 대비 재화 획득량, 솔로 인던 클리어 순으로 유의함
- 제거된 변수
 - 지역 채팅, 세력 채팅, 파티플레이 시간, 레이드, 등-총 78개의 변수

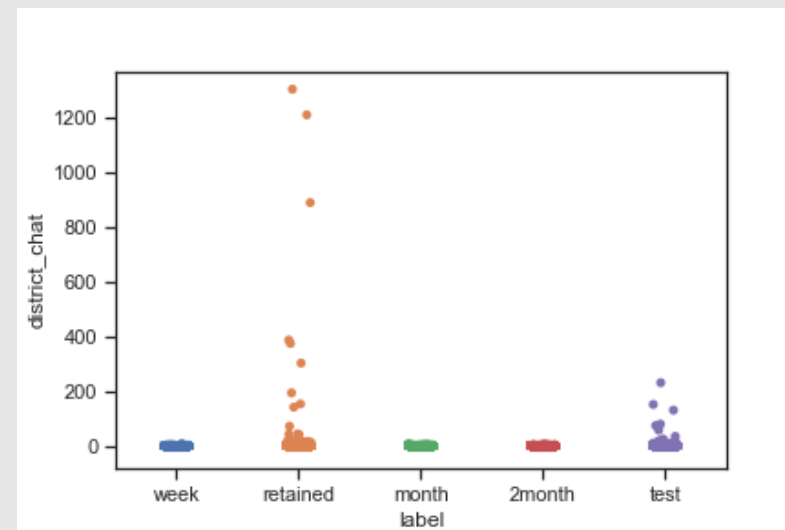


➔ 제거된 변수를 더 살펴보기로 함

Lasso Logistic Regression



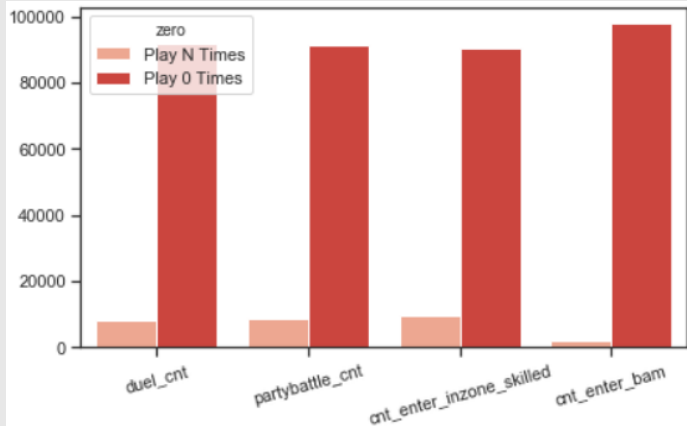
〈세력 채팅 NaN비율〉



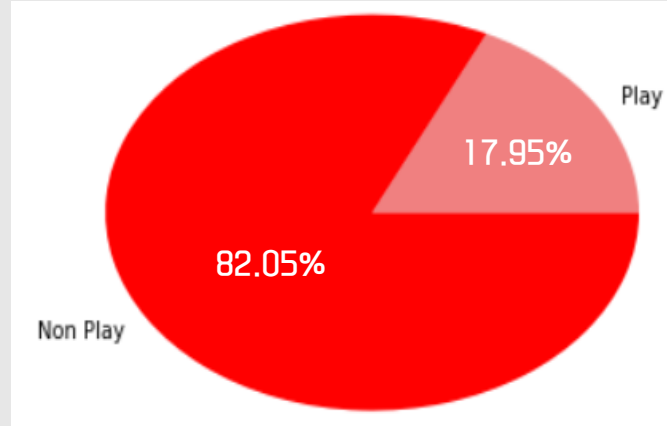
〈지역 채팅 Label별 분포〉

- 세력 채팅이 제거된 이유
 - 세력 채팅을 한 이력이 있는 데이터가 전체에서 단 1%에 불과
- 지역 채팅이 제거된 이유
 - 극단값 제거 시 Label별 분포의 차이가 없음

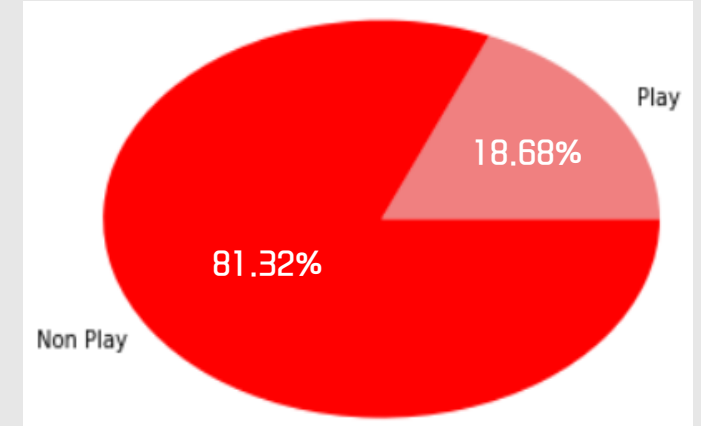
Lasso Logistic Regression



<던전별 플레이 여부>



<1주차 플레이 여부>

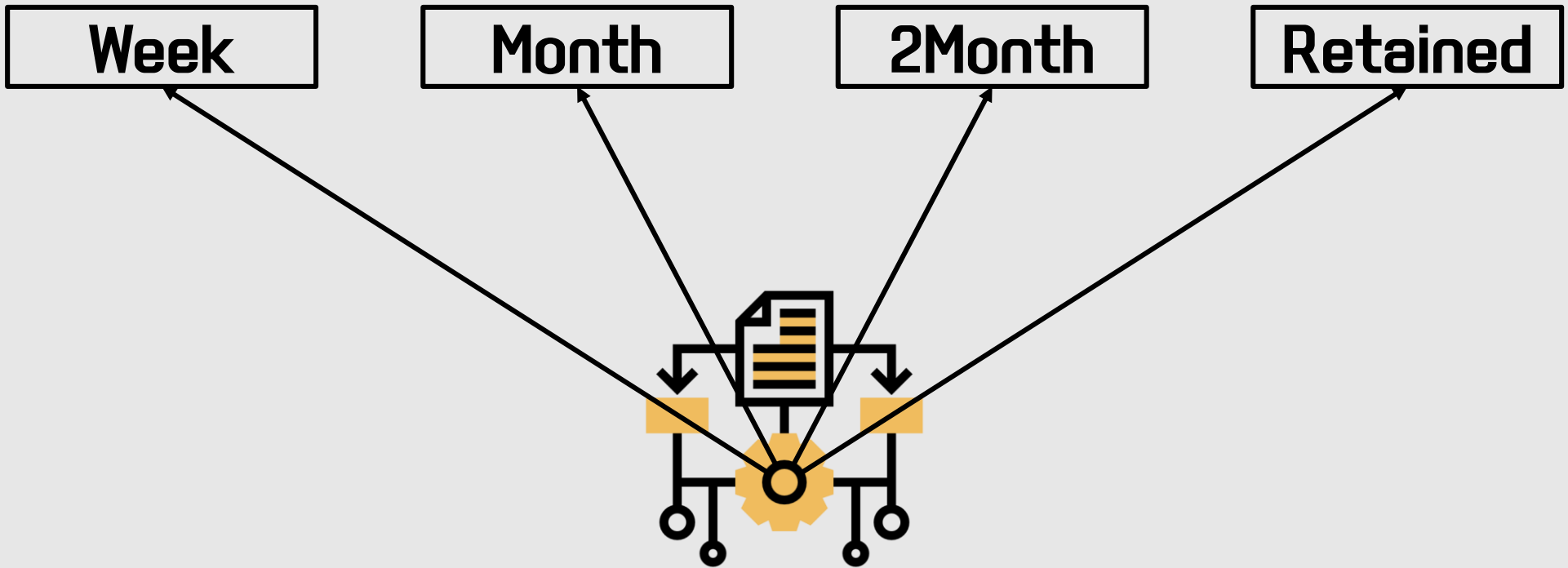


<2주차 플레이 여부>

- 특정 던전관련 변수가 제거된 이유
 - 특정 던전을 클리어한 경험이 있는 데이터가 많지 않음
- 1,2주차의 많은 활동 데이터가 삭제된 이유
 - 1주차, 2주차때 플레이한 이력이 있는 데이터가 전체의 20%이내이다.

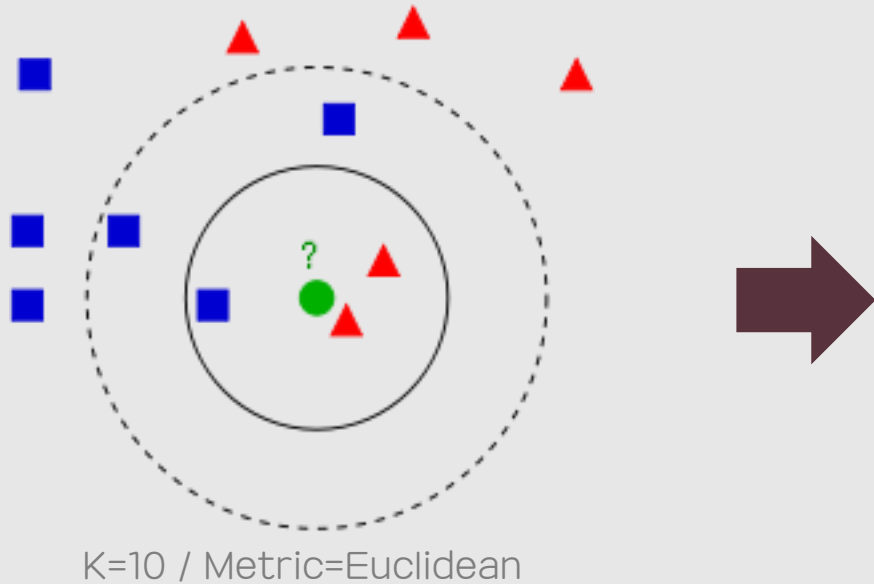
➔ 다양한 콘텐츠를 꾸준히 즐기는 유저가 부족

Label Separated Modeling



큰 차이가 없는 특징변수들을 선별 후
각 Label별로 이진화하여 분류모형 구축

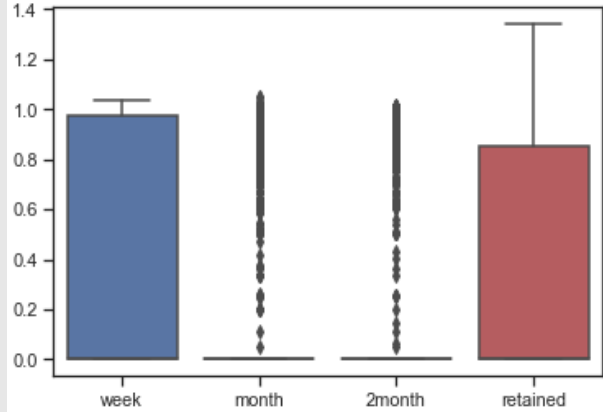
Label Separated Modeling



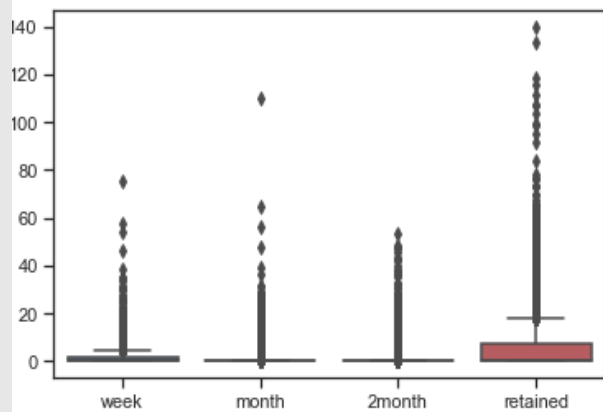
변수	Accuracy
Week	0.8634
Month	0.7637
2Month	0.8012
Retained	0.8429

- 머신러닝에서 가장 기본적인 모델인 kNN사용
 - Week과 Retained의 분류는 준수한 편
 - Month와 2Month의 분류 정확도가 좋지 못함

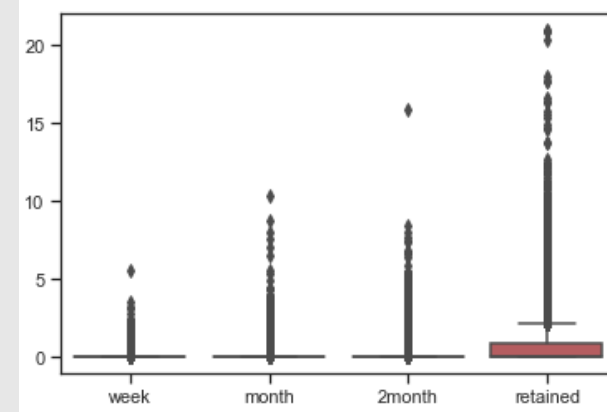
Label Separated Modeling



〈8주차 솔로 인던 클리어율〉



〈아이템 제작횟수 합〉



〈길드 채팅 합〉

...

- 많은 변수들이 Month와 2Month간에 큰 차이를 보이지 않음
 - Week과 Retained의 분류는 준수한 편
 - Month와 2Month의 분류에 초점을 두고 변수를 선택

Total Modeling

변수	파라미터	Accuracy	Precision	Recall
k-NN	K=10	0.6226	0.6192	0.6231
Decision Tree	Feature 및 Node 수 조절	0.6729	0.6751	0.6735
Naïve Bayes	Gaussian Naïve Bayes	0.3619	0.5263	0.3616
Logistic Regression	Lasso (Penalty='L1')	0.6571	0.6653	0.6572

- 의사결정나무(Decision Tree)의 성능이 제일 준수함
- 더 높은 성능을 위해서 앙상블모형에 적합해 보기로 함

Total Modeling

- 주어진 데이터는 여러 사람의 로그데이터
 - 사냥만 하는 유저
 - 길드를 통한 사회활동에 주력하는 유저
 - 채팅으로 많은 시간을 쓰는 유저 등
 - 유저 별로 플레이 스타일이 다를 것으로 판단
- 선행연구나 테스트 결과에도 트리구조가 성능이 우수하기 때문에 트리 모형 중 가장 최신 기법인 **XGBoost** 사용



Total Modeling

• XGBoost: A Scalable Tree Boosting System

1. CART 의사결정나무를 앙상블기법에 사용
2. Boosting기법을 통한
약한 분류기에 가중치를 주어 정확도를 향상
3. 병렬처리와 scikit-learn API로 쉽게 사용 가능
4. 유연성이 좋고 평가함수를 포함한
다양한 커스텀 최적화 옵션 제공

➔ 이번 분석은 유전 알고리즘을 통한 하이퍼 파라미터 튜닝 진행

XGBoost: A Scalable Tree Boosting System

Tianqi Chen
University of Washington
tqchen@cs.washington.edu

Carlos Guestrin
University of Washington
guestrin@cs.washington.edu

ABSTRACT

Tree boosting is a highly effective and widely used machine learning method. In this paper, we describe a scalable end-to-end tree boosting system called XGBoost, which is used widely by data scientists to achieve state-of-the-art results on many machine learning challenges. We propose a novel sparsity-aware algorithm for sparse data and weighted quantile sketch for approximate tree learning. More importantly, we provide insights on cache access patterns, data compression and sharding to build a scalable tree boosting system. By combining these insights, XGBoost scales beyond billions of examples using far fewer resources than existing systems.

Keywords

Large-scale Machine Learning

1. INTRODUCTION

Machine learning and data-driven approaches are becoming very important in many areas. Smart spam classifiers protect our email by learning from massive amounts of spam data and user feedback; advertising systems learn to match the right ads with the right context; fraud detection systems protect banks from malicious attackers; anomaly event detection systems help experimental physicists to find events that lead to new physics. There are two important factors that drive these successful applications: usage of effective (statistical) models that capture the complex data dependencies and scalable learning systems that learn the model of interest from large datasets.

Among the machine learning methods used in practice, gradient tree boosting [10]¹ is one technique that shines in many applications. Tree boosting has been shown to give state-of-the-art results on many standard classification benchmarks [16]. LambdaMART [5], a variant of tree boosting for ranking, achieves state-of-the-art result for ranking

¹Gradient tree boosting is also known as gradient boosting machine (GBM) or gradient boosted regression tree (GBRT)

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

ACM 978-1-4503-4232-3/16/08...\$15.00
DOI: <https://doi.org/10.1145/2839672.2839785>

problems. Besides being used as a stand-alone predictor, it is also incorporated into real-world production pipelines for ad click through rate prediction [15]. Finally, it is the de-facto choice of ensemble method and is used in challenges such as the Netflix prize [3].

In this paper, we describe XGBoost, a scalable machine learning system for tree boosting. The system is available as an open source package². The impact of the system has been widely recognized in a number of machine learning and data mining challenges. Take the challenges hosted by the machine learning competition site Kaggle for example. Among the 29 challenge winning solutions³ published at Kaggle's blog during 2015, 17 solutions used XGBoost. Among these solutions, eight solely used XGBoost to train the model, while most others combined XGBoost with neural networks in ensembles. For comparison, the second most popular method, deep neural nets, was used in 11 solutions. The success of the system was also witnessed in KDDCup 2015, where XGBoost was used by every winning team in the top-10. Moreover, the winning teams reported that ensemble methods outperform a well-configured XGBoost by only a small amount [1].

These results demonstrate that our system gives state-of-the-art results on a wide range of problems. Examples of the problems in these winning solutions include: store sales prediction; high energy physics event classification; web text classification; customer behavior prediction; motion detection; ad click through rate prediction; malware classification; product categorization; hazard risk prediction; massive on-line course dropout rate prediction. While domain dependent data analysis and feature engineering play an important role in these solutions, the fact that XGBoost is the consensus choice of learner shows the impact and importance of our system and tree boosting.

The most important factor behind the success of XGBoost is its scalability in all scenarios. The system runs more than ten times faster than existing popular solutions on a single machine and scales to billions of examples in distributed or memory-limited settings. The scalability of XGBoost is due to several important systems and algorithmic optimizations.

These innovations include: a novel tree learning algorithm for handling sparse data; a theoretically justified weighted quantile sketch procedure enables handling instance weights in approximate tree learning. Parallel and distributed computing makes learning faster which enables quicker model exploration. More importantly, XGBoost exploits out-of-core

²<https://github.com/dmlc/xgboost>

³Solutions come from top-3 teams of each competitions.

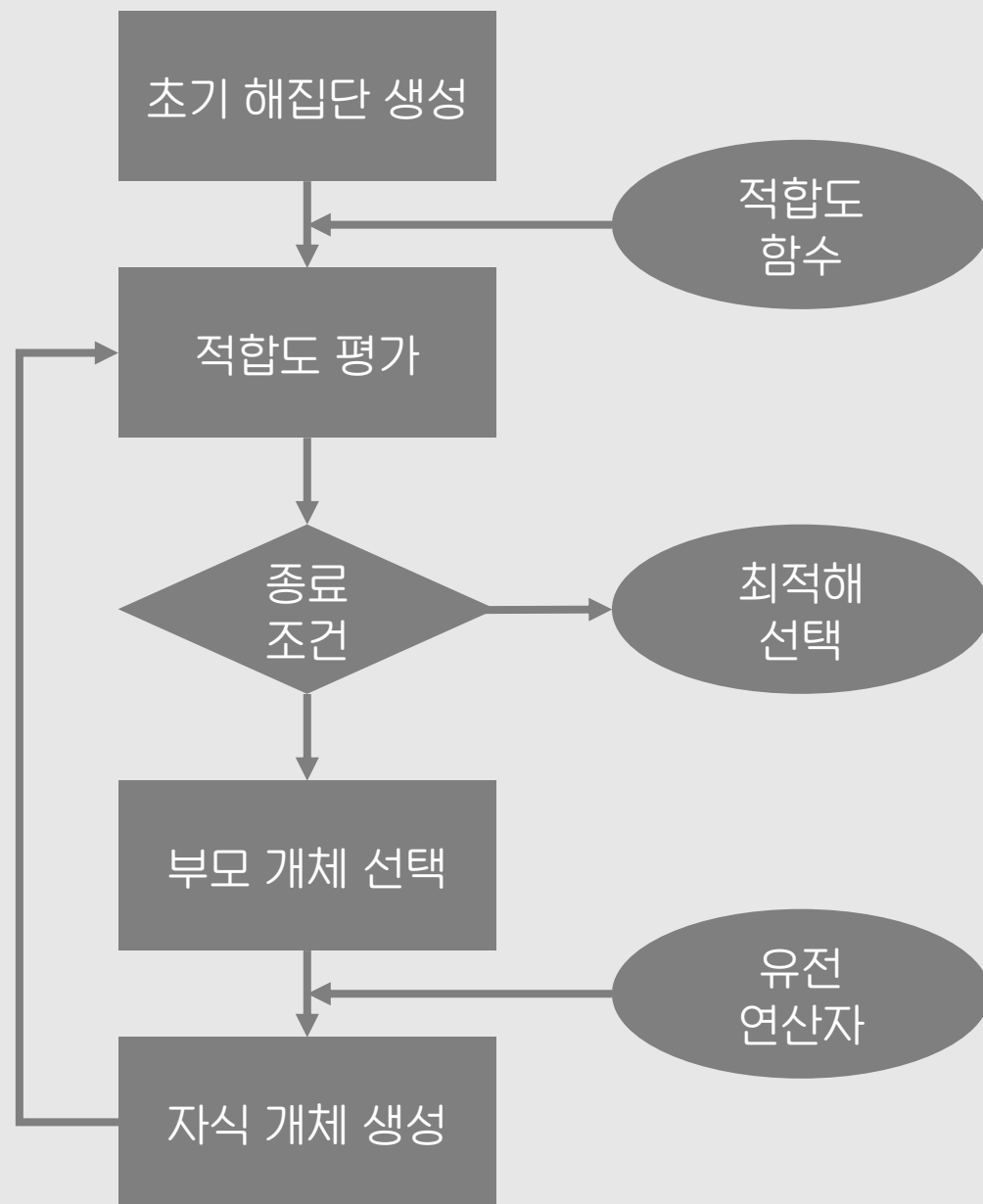
Tianqi Chen et al. 2016

Genetic Algorithm

- Holland(1975)가 개발한
생물의 진화를 모방한 집단 기반의 확률적 탐색 기법

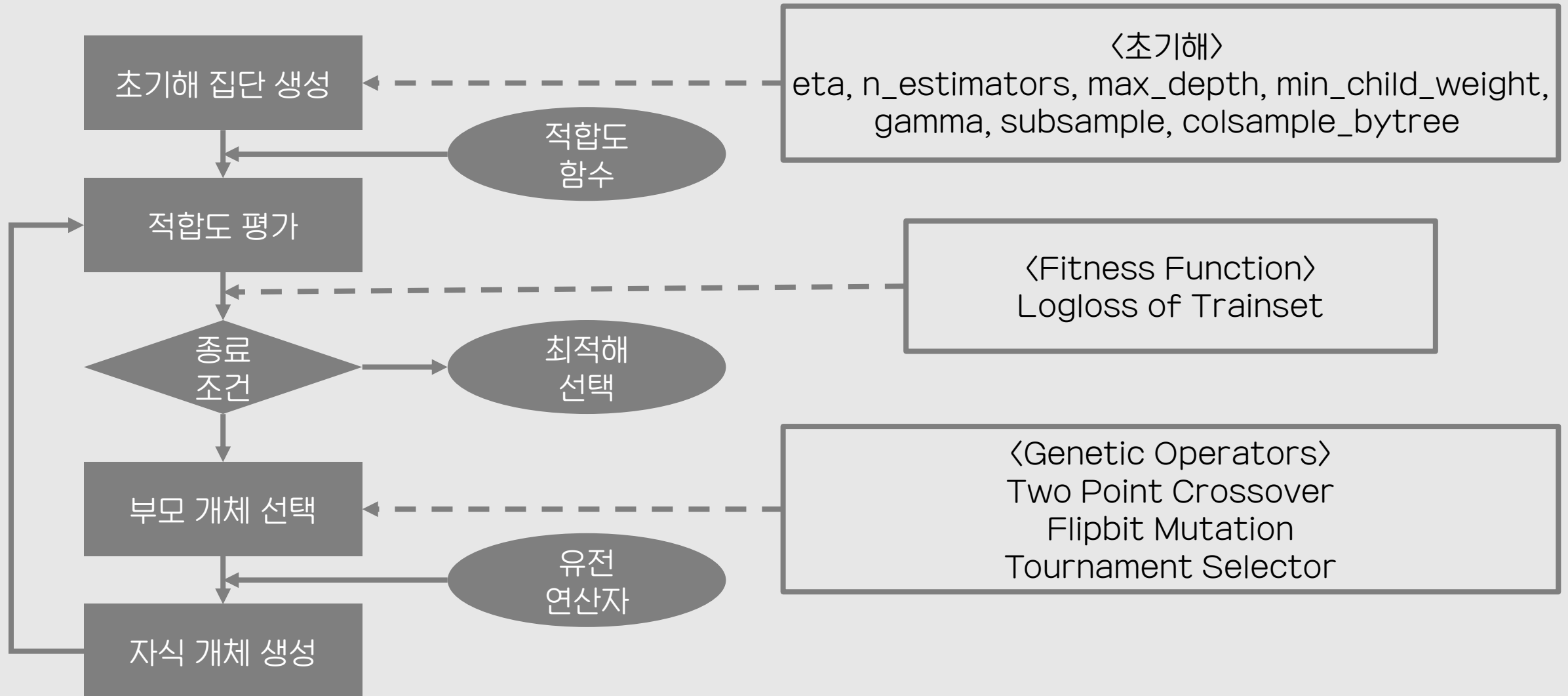
◆ 생물의 진화

- 염색체의 유전자가 개체 정보 코딩
- 적자생존과 자연선택
 - 환경에 적합도가 높은 개체의 높은 생존 및 후손 번식 가능성
 - 우수 개체들의 높은 자손 증식 기회
 - 열등 개체들도 작지만 증식 기회
- 해집단의 진화
 - 세대를 거듭할 수록 집단이 변화
- 형질 유전과 변이
 - 부모 유전자들의 교차상속
 - 돌연변이에 의한 변이



〈Genetic Algorithm Flow Chart〉

Genetic Algorithm



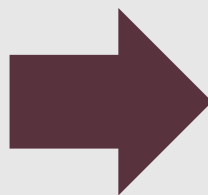
〈Genetic Algorithm Flow Chart〉

Genetic Algorithm

유전 알고리즘을 통한 하이퍼 파라미터 튜닝 후
자율평가 기준

0.6721 → 0.7319

성능 향상

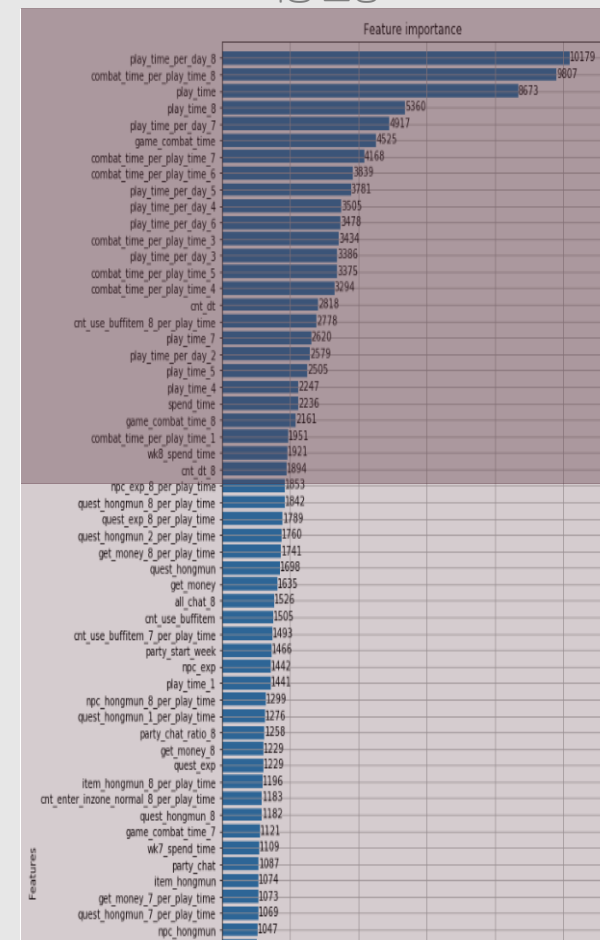


Genetic Algorithm

- 기존 유저
 - 유의한 변수들 대다수가 Play_time과 Combat_time
 - 그 중에서도 7,8주차의 정보가 상위권
- 신규 유저
 - 차순위로 유의한 변수들은 Exp, Quest와 연관이 있다.
 - 신규 유저들의 활동으로 보이나 이들 대다수는 이탈 유저이다.

➔ 전투와 경험치에 따른 기존/신규 유저 파악 가능

최종 모형



〈Feature Importance〉

Conclusion

- 플레이 시간과 전투참여시간이 유의
- 일부 콘텐츠는 사용 빈도가 매우 적음
- Month와 2Month의 분류에 어려움
- 유전 알고리즘을 활용한 파라미터튜닝을 통해 성능향상
- 자율평가기준 73%정도의 성능

이탈 고객	비이탈 고객
신규 유저	지속 플레이 유저
주는 거래 많음	받는 거래 많음
길드 미가입	길드 가입
퀘스트, 홍문 유의	퀘스트, 홍문 비유의
결제 적음	결제 많음
던전 클리어율 적음	던전 클리어율 높음

Conclusion

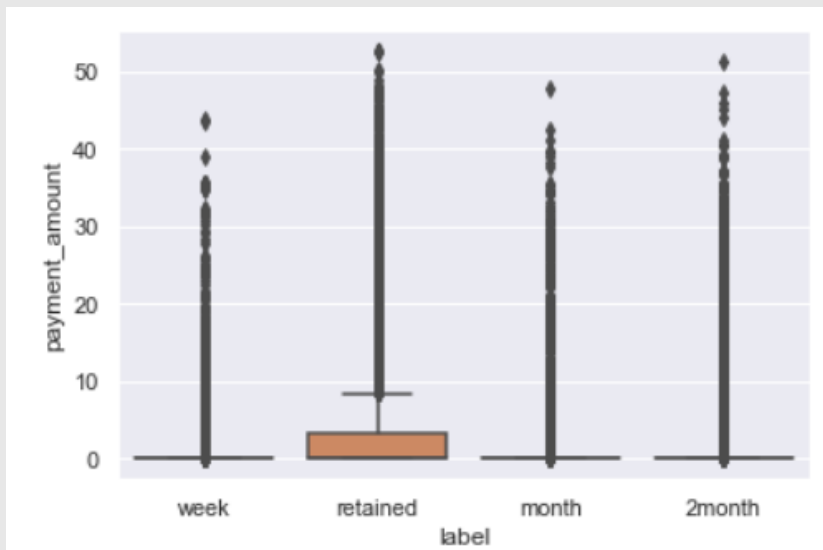
- 현재 블레이드앤소울 내 특정 콘텐츠(레이드, 등)는 진입장벽이 높음
 - 오래된 게임들의 속명
 - 신규유저가 캐시아이템 없이 해당 콘텐츠를 즐기기가 어려움
 - 결제를 유도할 만큼의 재미요소가 되지도 못함
- Play time과 Combat time변수가 이탈예측에 매우 유의함
 - 하지만 인과성이 아닌 상관성으로 생각됨
- Quest가 예상외로 유의함
 - 신규 유저 변수로 볼 수 있는데 신규유저가 게임에 적응하지 못하고 이탈

➔ 특정 콘텐츠의 개선 필요 및 신규 유저의 이탈에 초점을 두어야 함

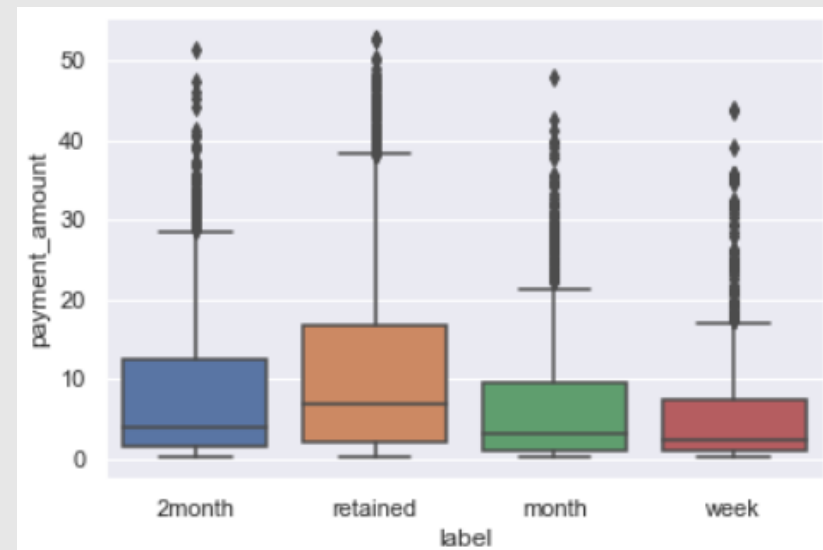


V. Modeling 2

Payment by Label



<Label별 Payment합 (0포함)>



<Label별 Payment합 (0제외)>

- Label별로 Payment의 합 시각화
 - Retained군집이 더 많은 결제를 하고 있음
 - 이탈유저들의 결제는 적음

Analysis Plan

- 이탈/비이탈 별 결제 금액의 차이 확인 결과
 - Retained군집이 더 많은 결제를 함
 - Retained군집의 결제에 영향을 주는 요인변수 파악 필요

➔ 인과관계 규명을 위한 회귀분석 실시



Analysis Plan

- 변수의 수가 너무 많음
 - 선형관계규명이 어려움
 - 모형에 대한 해석 불가
- 해결 방안은?
 - Principal Component Analysis
- **사회과학 측면**에서 보게 되면 특징변수의 배후에 숨겨진 공통인자 존재 확인
(예시)
 - 경험치_주성분1: NPC_Exp, Quest_Exp가 묶임 → **신규 유저 콘텐츠**
 - 경험치_주성분2: NPC_hongmun, Quest_hongmun, Item_hongmun → **기존 유저 콘텐츠**

Variable Select

- Activity데이터에서 변수를 총 5개로 분류
 - 각각의 분류에 대하여 PCA진행
 - 주성분 선택 기준은 설명률 80%



[EXP]
NPC_Exp
NPC_Hongmun
Quest_Exp
Quest_Hongmun
Item_Hongmun



[Inzone]
Inzone
Raid
Duel
Partybattle



[Chat]
Normal Chat
Whisper Chat
District Chat
Party Chat
Guild Chat
Faction Chat



[Play Time]
CNT_DT
Play_Time
WK




[ETC]
Get_Money
CNT_Use_Buff
Gathering_CNT
Making_CNT

Variable Select

- Guild데이터
 - Guild가입여부와 가입된 길드의 규모 파악 필요
 - 서버당 가입된 길드의 멤버 수 평균 확인
- Trade데이터
 - 아이디 별 거래횟수 확인



Linear Regression



Variable	Coef	P-Value
Exp_1	0.2499	0.000
Exp_2	-0.3249	0.000
Inzone_1	0.04818	0.000
Inzone_2	-0.0172	0.000
Inzone_3	0.00259	0.000
Chat_1	0.0578	0.000
Chat_2	-0.0723	0.000
Chat_3	-0.0031	0.367
Playtime_1	0.0171	0.000
ETC_1	-0.02955	0.000
ETC_2	0.05899	0.000
Guild	0.04178	0.000
Trade	-0.0002	0.846

모형 자체의 P-Value는 0.00으로
해당 모형은 유의함

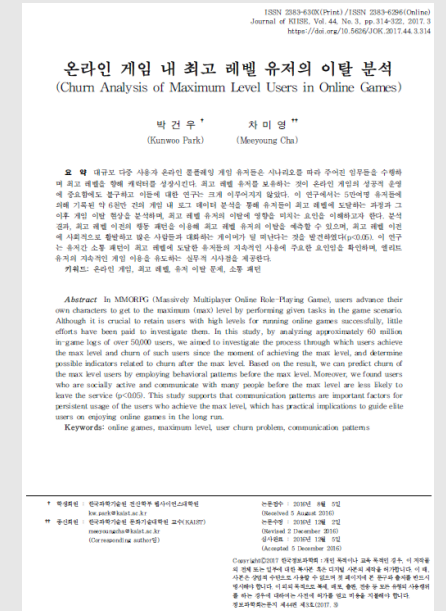
R-Squared값과
Adjusted R-Squared값 모두 0.466

회귀분석 결과 Chat3과 Trade를 제외한
모든 변수가 유의함

그 중에서도 Exp변수의 계수가 제일 크다.

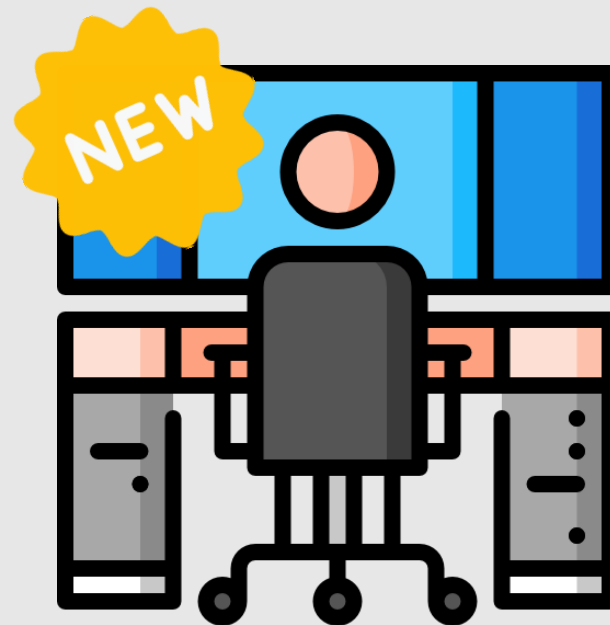
Conclusion

- Chat과 Guild변수가 유의함
 - 게임 내 사회활동을 많이 하는 유저의 결제량이 많다
 - 해당 유저들은 의상 아이템에 소비했을 것
 - 고 레벨 유저들의 사회활동은
관련논문을 통해서도 확인 가능



Conclusion

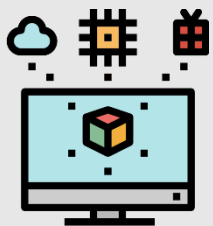
- Exp, Quest관련 변수의 유의함
 - 고 레벨 유저들은 Exp에 크게 영향받지 않음
 - 이는 곧 신규유저들의 결제라는 것
- 던전 관련 변수의 유의함
 - 현재 '블레이드앤 소울'은 신규 유저들의 진입장벽이 매우 높음
(캐시 아이템 없이 던전 클리어가 매우 힘들다.)
 - Exp, Quest와 마찬가지로 신규유저들의 결제





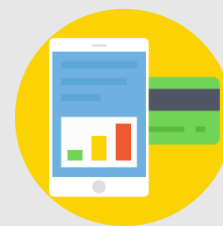
VI. Total Conclusion

Total Conclusion



Modeling 1
〈이탈고객 예측〉

- 신규 유저들에게 적합한 콘텐츠 부족
 - 특정 콘텐츠(숙련 던전 등)의 사용을 저조 (제한된 콘텐츠)
- ➔ 이탈의 주 원인은 신규 유저들이 게임에 쉽게 적응하지 못함



Modeling 2
〈결제원인 파악〉

- 결제의 원인은 사회활동, 플레이시간 등으로 다양함
 - 그 중에서도 경험치(Exp_1, Exp_2)가 제일 유의함
- ➔ 신규유저들이 콘텐츠를 즐기기 위해서 결제를 시도함

- 신규 유저들이 게임에 쉽게 적응하지 못하고 이탈하고 있음
 - 신규 유저들 중 이탈하지 않는 사람들이 주로 결제를 하고 있음
- ➔ 신규유저들이 게임에 적응할 수 있도록 하는 방안 필요

Total Conclusion

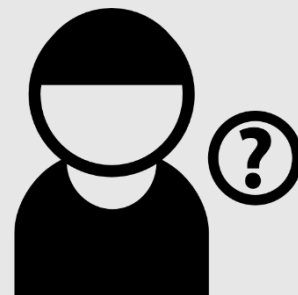
What-If

- 이미 표준화된 값으로 인한 정확한 수치 파악 불가
- 정확한 날짜 정보 부재로 인한 외부 데이터 활용 불가



- 주 단위로 주어진 데이터

- 길드 생성일, 문파 계급, 길드장 등 길드 정보 부족



- 개인 정보 보호로 인한 성별, 계정생성일 등 신상정보 부족

추가 데이터가 주어질 경우 더욱 정교한 분석 가능

감사합니다.