

PUBG Finish Placement Prediction

런닝 머신 조

조장 : 김준희

정일호, 문성만

정해준, 임도근

CONTENTS

I

서론

- PUBG란?
- 프로젝트 개요
- 프로젝트 시스템 구성도

2

본론

- 데이터 전처리(EDA)
- 개발 환경 (인프라)
- Final 1~5 개선과정

3

결론

- 결론 및 향후 계획

조원 소개

Kaggle competition 문제해결 기반 프로젝트



김준회

조장

데이터 전처리



임도균

조원

머신러닝 분석



정해준

조원

머신러닝 분석



문성만

조원

머신러닝 분석



정일호

조원

PPT

CONTENTS



서론

- PUBG란?
- 프로젝트 개요
- 프로젝트 시스템 구성도

PUBG

배틀 그라운드 란?



PUBG란?

배틀 그라운드 소개 영상

<https://www.youtube.com/watch?v=sv3jcGYPfYw>

100-1명까지 17초까지

<https://www.youtube.com/watch?v=scqEFKUY5ZQ>

게임 플레이 광고 영상

게임 한 번에 최대 100명의 플레이어와 전투를 벌이는 배틀 로얄 형식의 슈팅 비디오 게임

전투에 참가하게 된 플레이어들은 1인, 2인, 4인 타입에 따라 생존 전략과 기술을 활용해 전장의 최후 1인을 선정한다.

프로젝트 개요

Kaggle competition 문제해결 기반 프로젝트



각 플레이어 데이터 셋
(train_650Mb, test_250Mb)



MODEL



1 ~ 100

생성한 모델을 통해 실제 등수와 예측 등수의 오차를 최소화하는 것을 목적

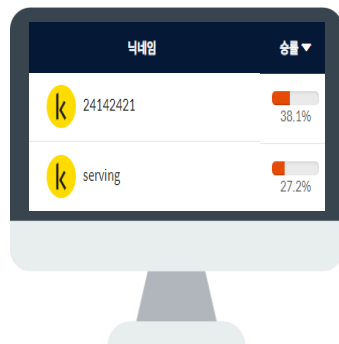
Competition 참여를 통해 생성한 등수 예측 모델을 기반으로 한 "배틀그라운드 플레이어 등수 예측" 서비스를 개발한다.

프로젝트 시스템 구성도

서비스 내용



client



플레이어 등수 예측 서비스

플레이어 게임 결과 데이터 시각화 서비스

실 유저 데이터 입력 시 결과 예측 서비스

프로젝트 시스템 구성도

서비스 도식화 및 소프트웨어 스택



소프트웨어 스택

분야	패키지	버전	용도
Library	Tensorflow		모델 생성 및 플레이어 등수 예측
	numpy, pandas, matplotlib, Seaborn		Python 데이터 분석
	D3.js	5.7.0	실유저 데이터를 활용한 시각화
Framework	Django	2.1.3	웹 서버, 서비스를 제공 풀 스택 프레임 워크
P/L	Python	3.6	데이터 분석, 서비스 구현
	R	3.5.1	데이터 분석
Network	JSON		Server-Client 간 송수신 데이터 타 입
DBMS	ElasticSearch	6.5.1	실 유저 데이터 수집 및 학습 데이 터
OS	Windoww10		

CONTENTS



본문

- 데이터 전처리(EDA)
- Final 1~5 개선과정

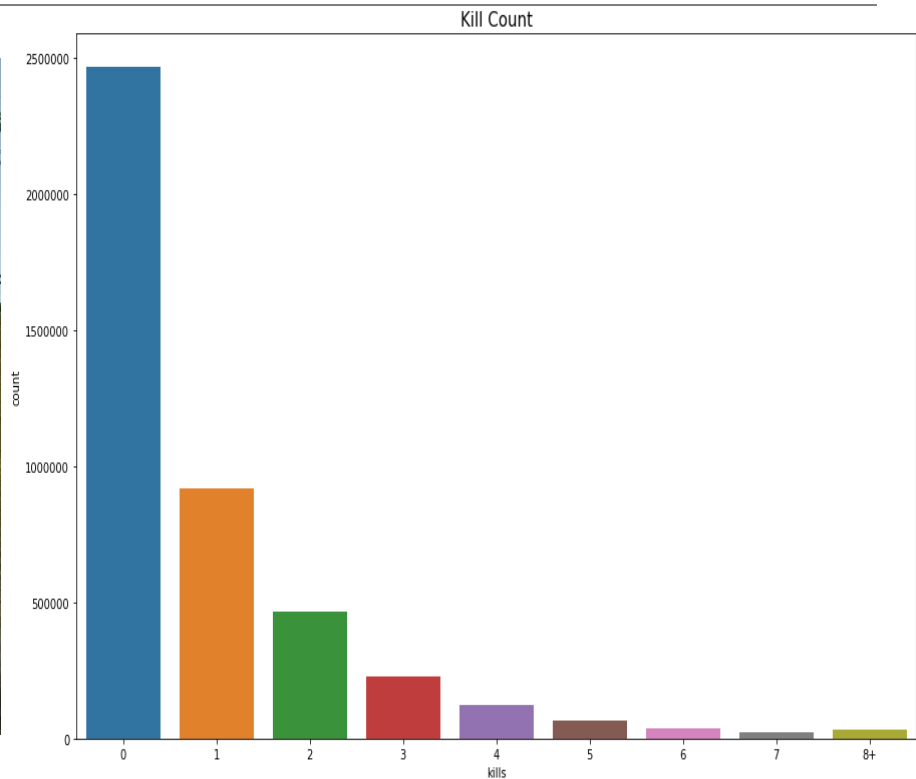
데이터 전처리(EDA)

데이터 변수 설명

변수	용도
boosts	에너지드링크, 진통제 등 부스팅 아이템 사용한 수
damageDealt	딜량
headshotKills	헤드샷으로 죽인 수
heals	구급상자, 붕대 등 힐링 아이템
killStreaks	짧은 시간 내에 가장 많이 낸 킬 수
kills	킬 수
longestKill	가장 먼 사격거리
matchId	매치 아이디

변수	용도
matchType	게임 타입(솔로, 듀오, 스쿼드, 기타)
revives	기절당한 아군을 부활시킨 수
swimDistance	탈 것 이용해서 이동한 총 거리
swimDistance	총 수영한 거리
walkDistance	걸어다닌 총 거리
weaponsAcquired	획득한 무기 개수
groupId	그룹 아이디
winPlacePerc	게임 등수(0~1 사이 값, 1등 : 1, 꼴등 : 0) - 예측하기 위한 변수

데이터 전처리(EDA) - 킬러

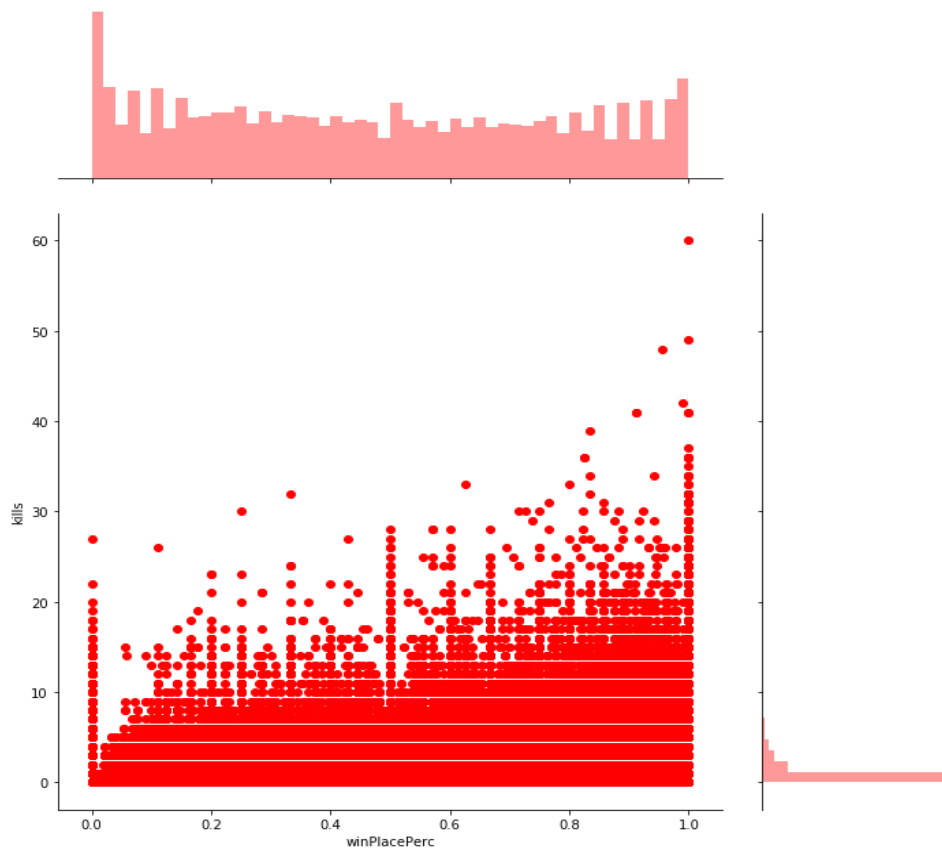


플레이어 1인당 평균 : 0.9345킬

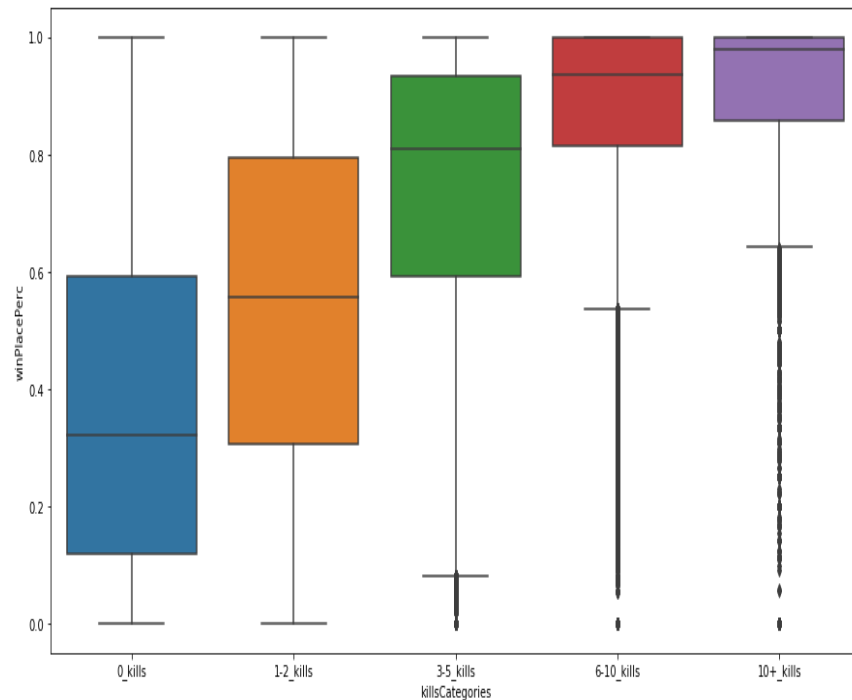
1% 사람만이 8킬 이상을 함 - 99%의 사람은 7킬 이하

최고 킬 수는 60 - 에임핵 가능성

데이터 전처리(EDA)

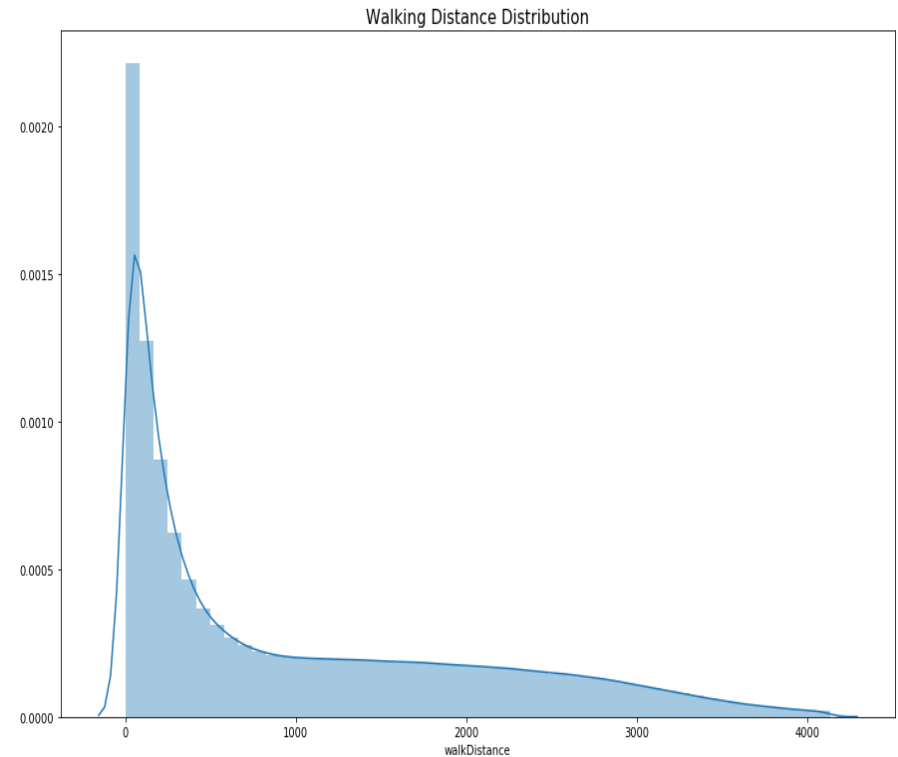


0.5%의 사람들이 1킬도 없이 우승



킬이 많아지면 우승할 확률이 높음

데이터 전처리(EDA) - 마라토너

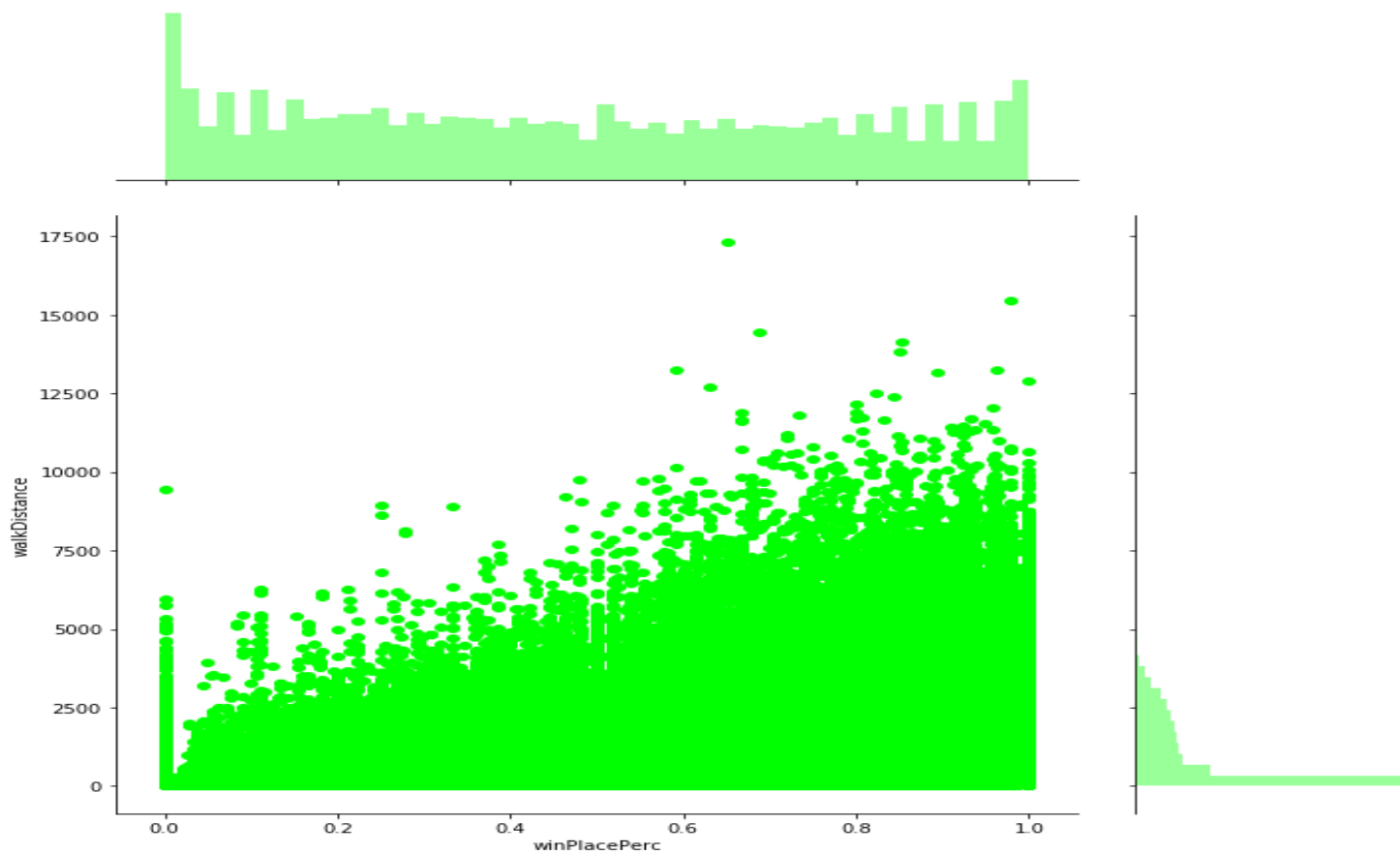


플레이어 1인당 평균 : 1055.1m

1% 사람만이 4200m 이상

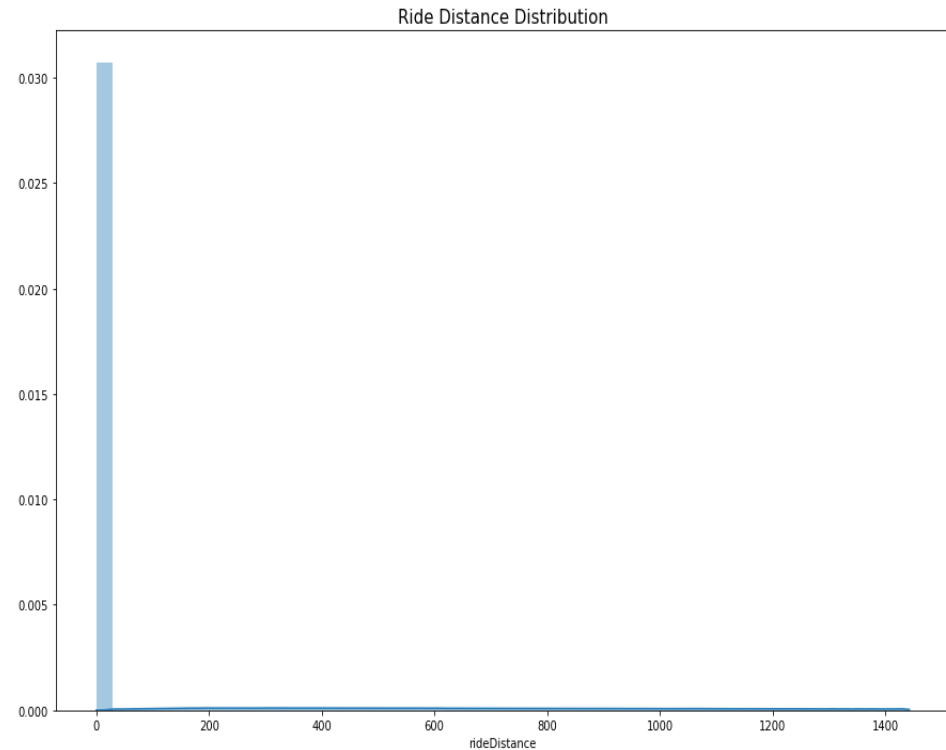
최고의 마라토너는 17300m - 스피드핵 가능성

데이터 전처리(EDA)



많이 걸었다는 것은 오래 살아남았다는 것을 의미

데이터 전처리(EDA) - 드라이버



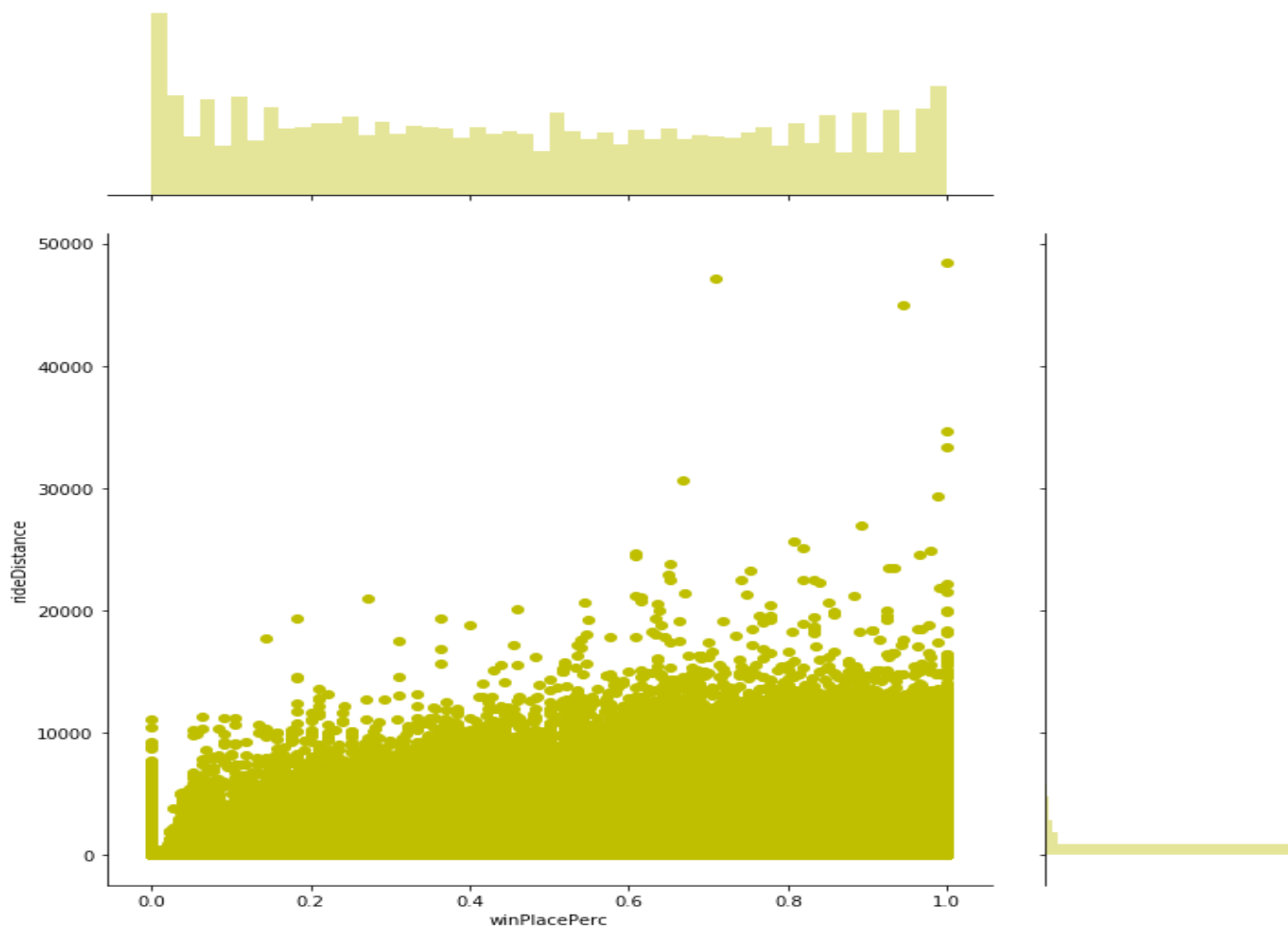
플레이어 1인당 평균 : 420m

1% 사람만이 6200m 이상

최고의 드라이버는 48390m

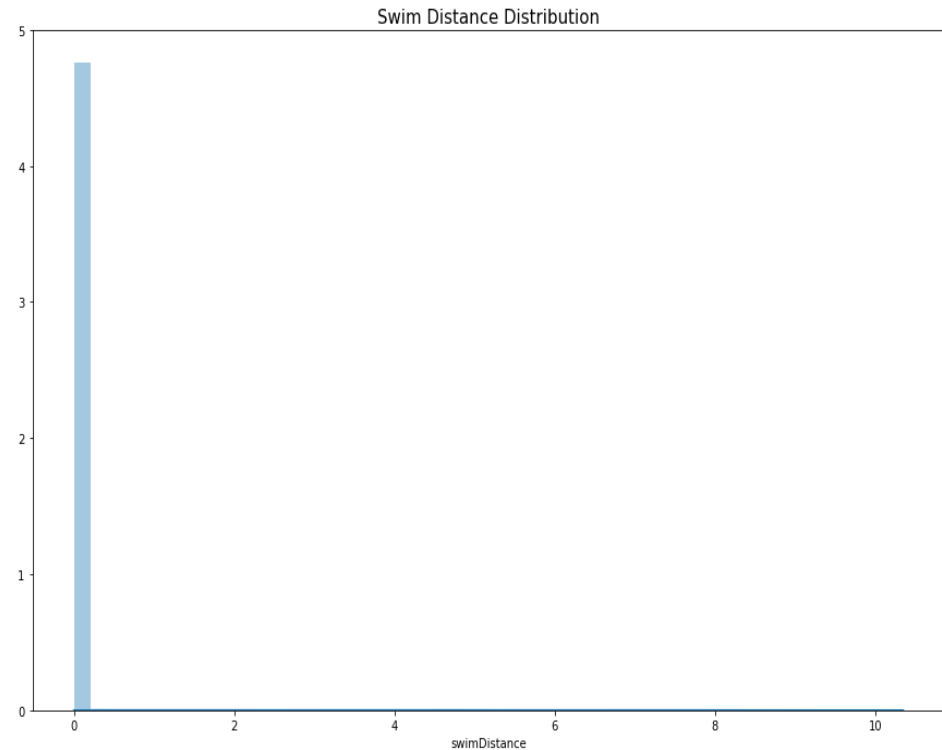
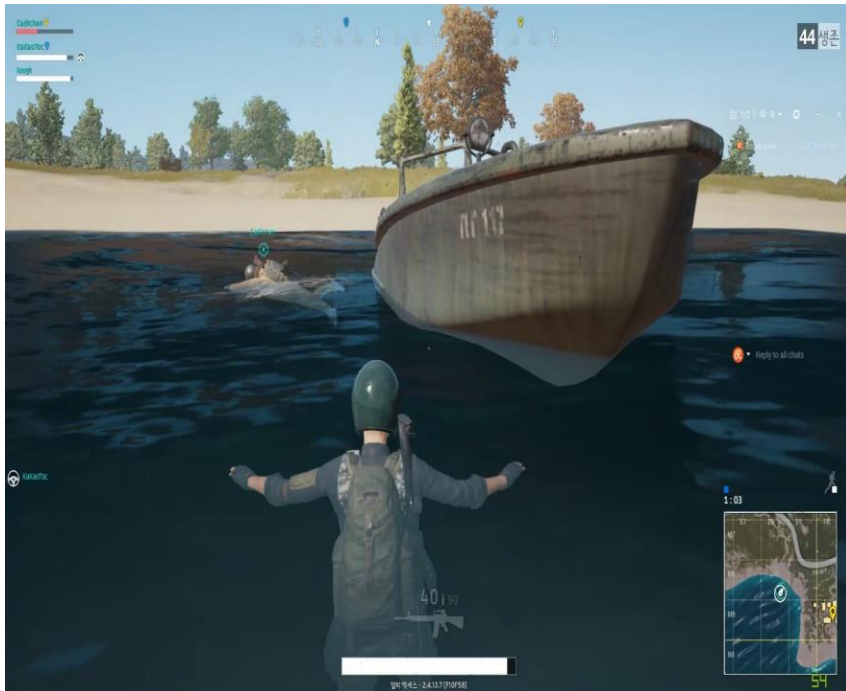
차량 개수는 정해져 있기 때문에, 대부분의 사람들이 차를 이용하지 못함

데이터 전처리(EDA)



약간의 상관관계는 있으나, 비교적 균등분포

데이터 전처리(EDA) — 바다의 왕자

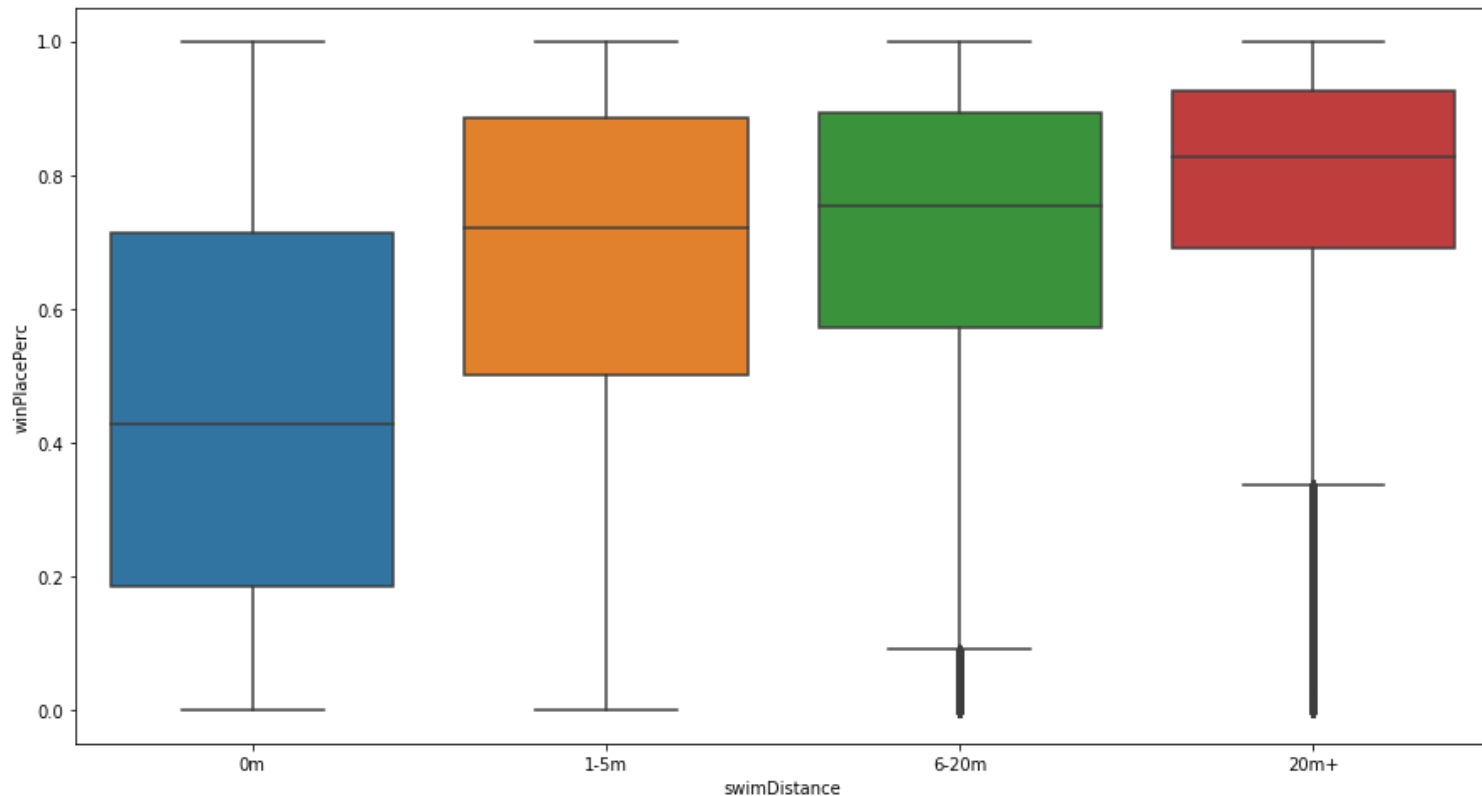


플레이어 1인당 평균 : 4m

1% 사람만이 115m 이상

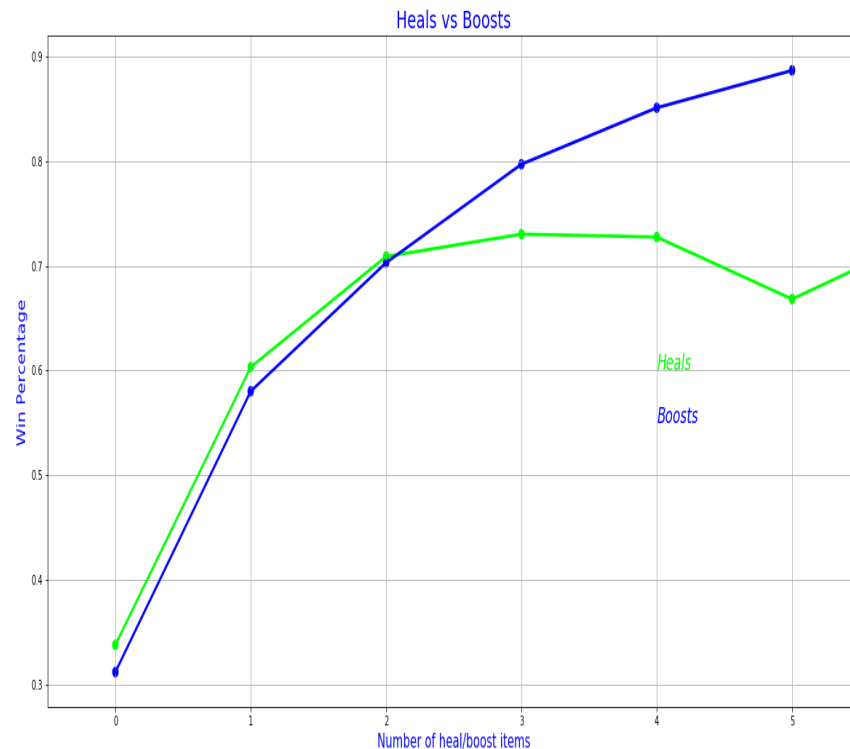
최고의 바다의 왕자는 5286m

데이터 전처리(EDA)



수영을 할 경우 등수가 올라감 – 오래 살아 남았기 때문

데이터 전처리(EDA) — 의사

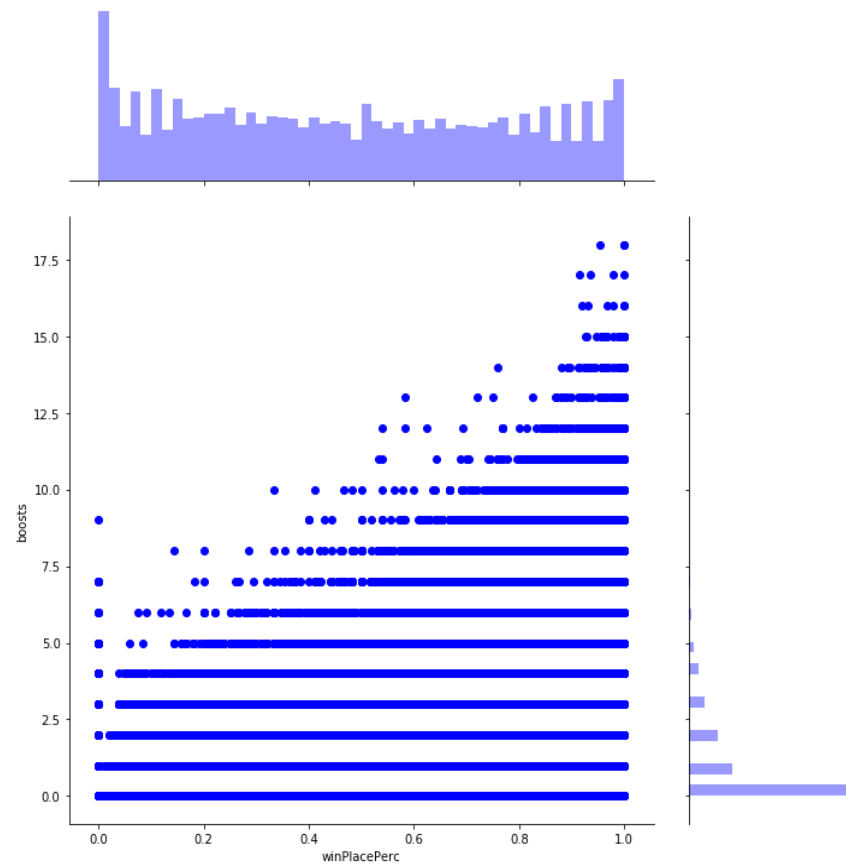
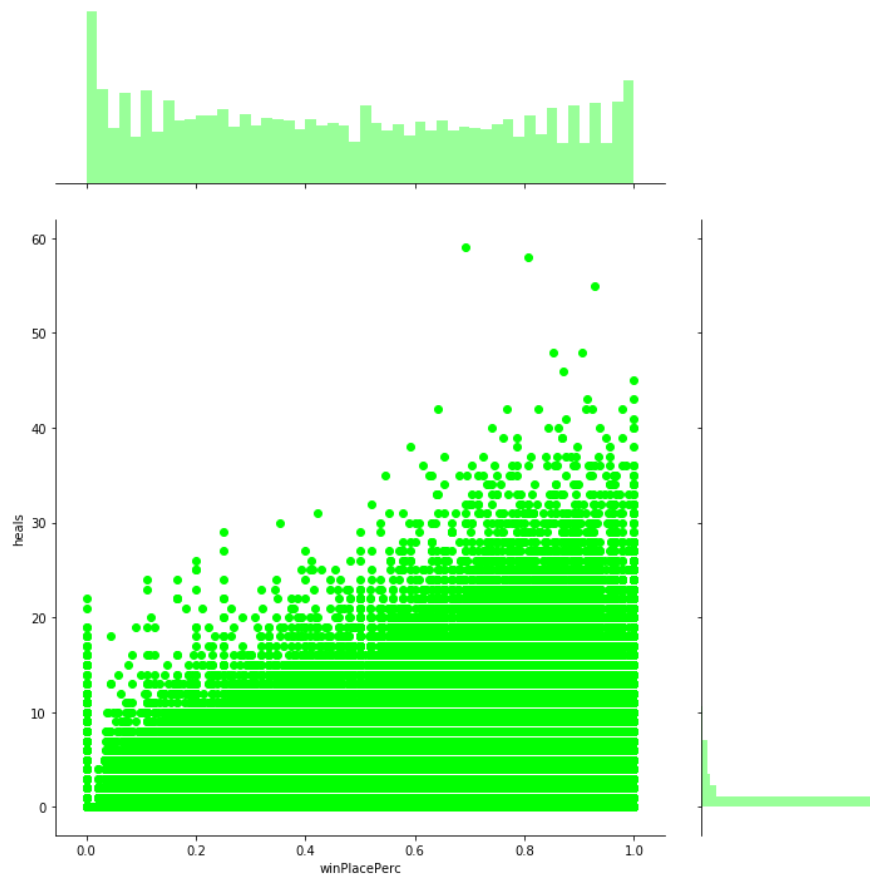


플레이어 1인당 평균 : 1.2개(힐) / 1개(부스트)

1% 사람만이 11개 이상(힐) / 7개(부스트)

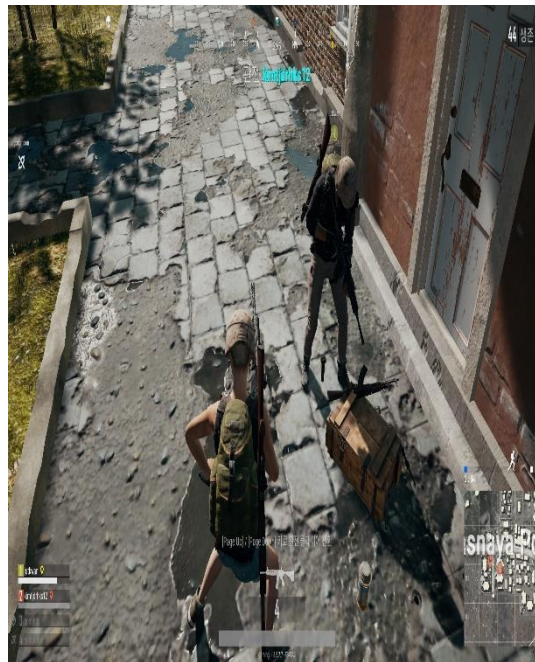
최고의 의사는 59개(힐) / 18개(부스트)

데이터 전처리(EDA)



힐과 부스팅은 생명력이 직결됨

데이터 전처리(EDA) – 게임 타임

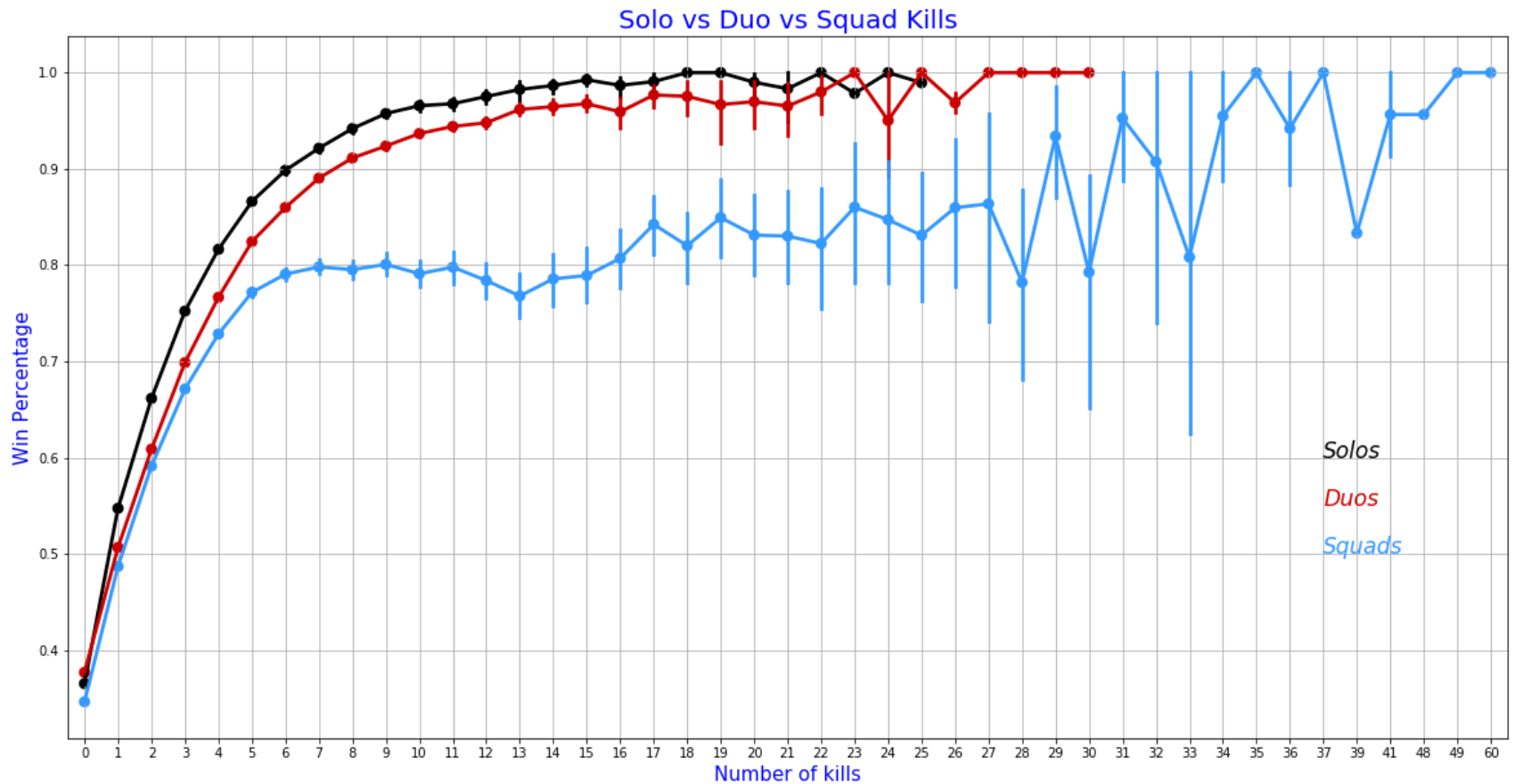


솔로 게임(1인) : 563279 (13%)

듀오 게임(2인) : 3070150(70%)

스쿼드 게임(4인) : 723907(16%)

데이터 전처리(EDA)



솔로, 듀오는 비슷한 양상. 스쿼드는 다름

CONTENTS



본문

- 데이터 전처리(EDA)
- **Final 1~5 개선과정**

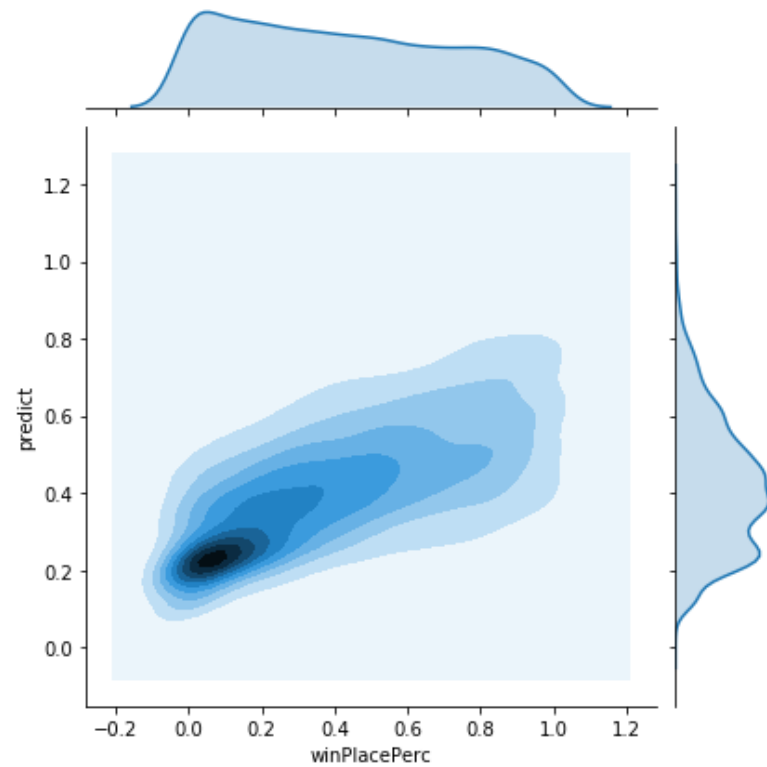
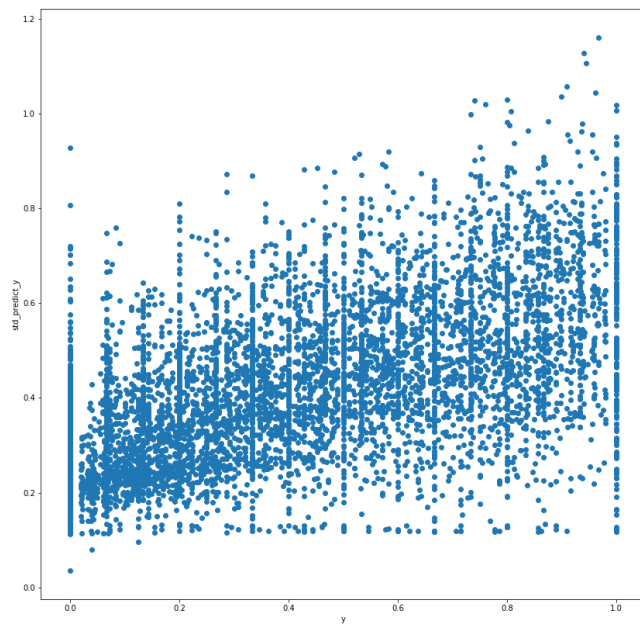
FINAL 1

FINAL 1 EDA 사항

- 데이터 분리
 - > 게임 타입별로 solo, duo, squad, other 분리해서 분석
- 변수제거
 - groupId, matchId, killplace, damagedealt, matchduration, numgroups, roadkills, teamkill, vehicle destroys, maxplace
 - killpoint, rankpoint, winpoint => 가공
- 이상치 제거
 - > Kill, assist, boost, ETC – 0값 제외하고 IQR
 - > Kill, assist, boost, ETC– Z-Score
 - => max(IQR, Z-score) 값으로 보수적으로 아웃라이어 제거
 - > SOLO 데이터의 경우 walkingdistance==0이고 winplaceperc >=0 은 제거
- 데이터 단위 변환
 - > Walkingdistance, swimdistance, ridingdistance, longestkill – log 변환
- 데이터 변환
 - > Headshotkills => headshotkillsrate = headshotkill/kil

FINAL 1

FINAL 1 분석 결과 – Multi Linear 사용



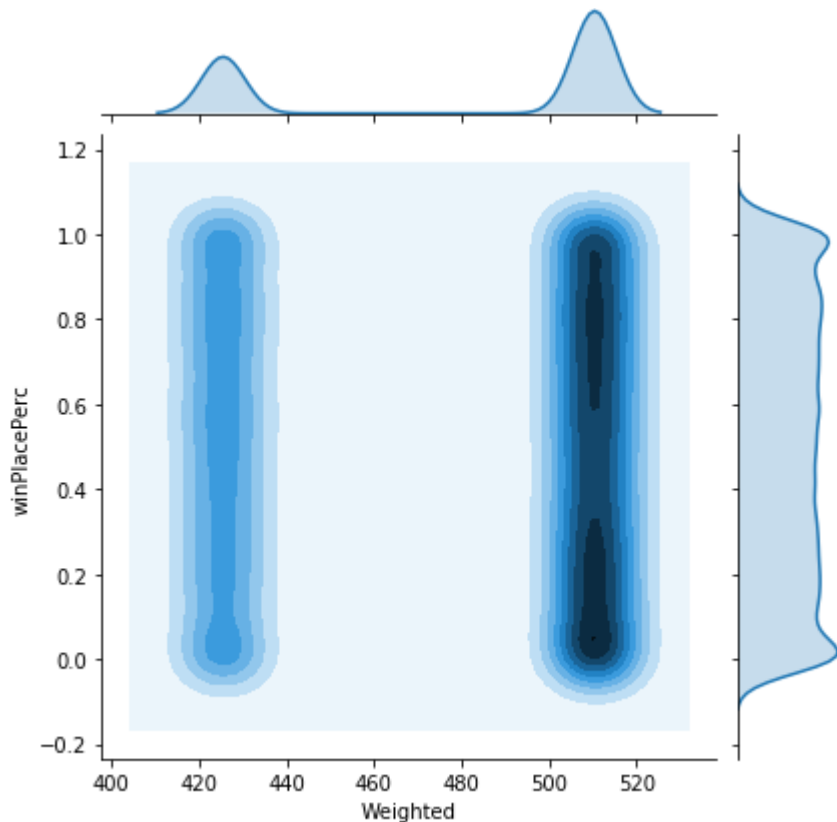
FINAL 1

FINAL 1 분석 결과 – **Multi Linear** 사용

	MAE(Mean Absolute Error)
Multi linear	0.247369
Kaggle 1등	0.0197

FINAL 1

FINAL 1 EDA의 고찰



- Weighted 데이터 무의미
- Kill의 outlier 수치가 7킬. 충분히 가능하다고 판단
- => 더 높게 잡을 필요성
- 솔로 게임에 그룹이 2~3명
- => 그룹아이디 별 평균값 사용
- 어떤 플레이어가 같이 게임하냐에 따라 달라짐
- => 매치 평균값 필요성
- 2명이 참가하면 0 아니면 1 등수
- => 몇 명이 참가했는지 필요

FINAL 2

FINAL 2 EDA 변경 사항

- 변수제거

- killpoint, rankpoint, winpoint 추가 제거

- 이상치 제거

- Kill, assist, boost, ETC – 상위 0.005%(ex. Kill - 13) 제거

- 데이터 단위 변환

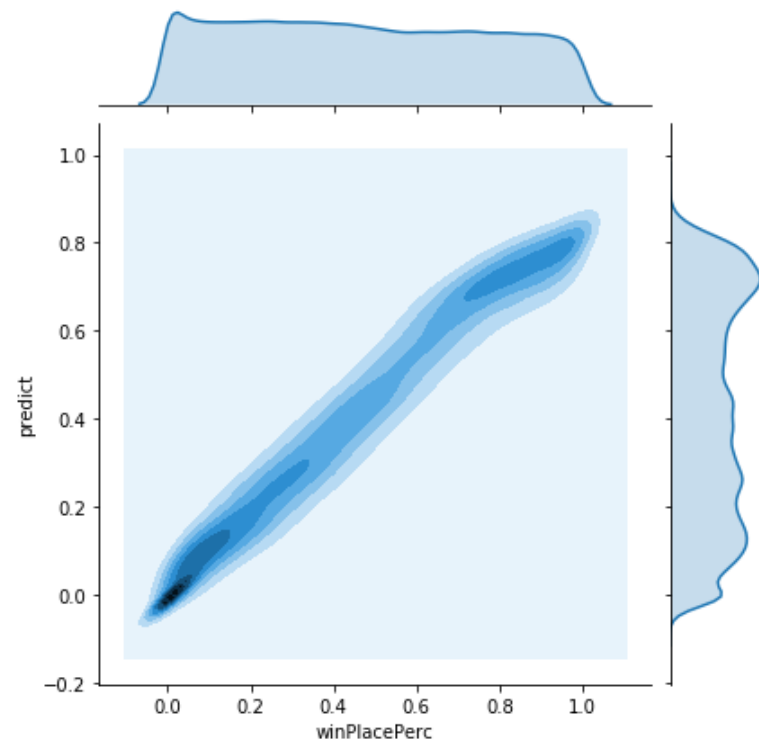
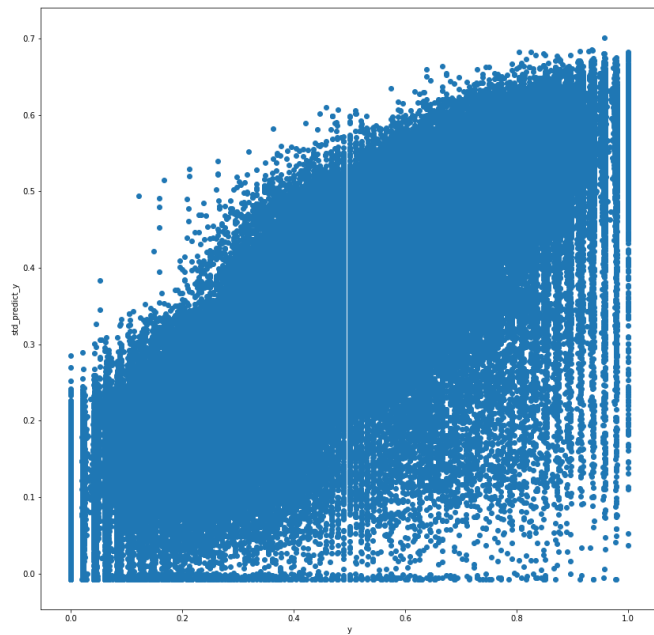
- 변경사항 없음

- 데이터 변환

- matchId, groupId 별로 평균값으로 컬럼 대체
- match 평균 변수 추가
- Matchsize – 몇 명 참가했는지

FINAL 2

FINAL 2 분석 결과 – NN 사용



FINAL 2

FINAL 2 분석 결과 – **NN**(activation – Leaky RELU) 사용

	MAE(Mean Absolute Error)
NN	0.09
Multi Linear	0.1913
Kaggle 1등	0.0197

FINAL 2

FINAL 2 EDA의 고찰

- 컬럼 개수가 많아짐 (Final 1 - 29개 = > Final 2 - 58개)
- 결과는 개선 됐으나 파라미터 변경 혹은 데이터 전처리 변경을 통한 개선 필요

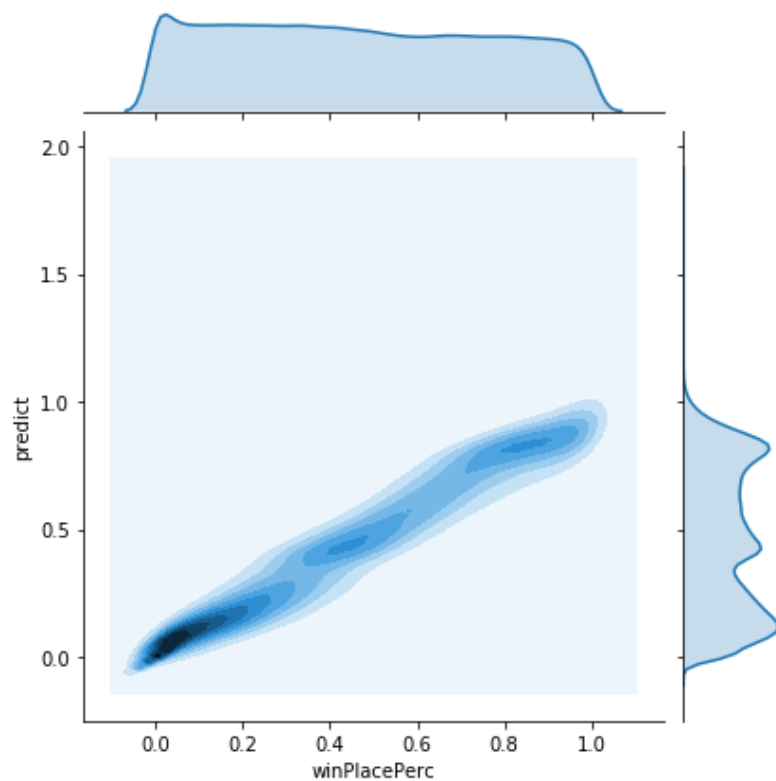
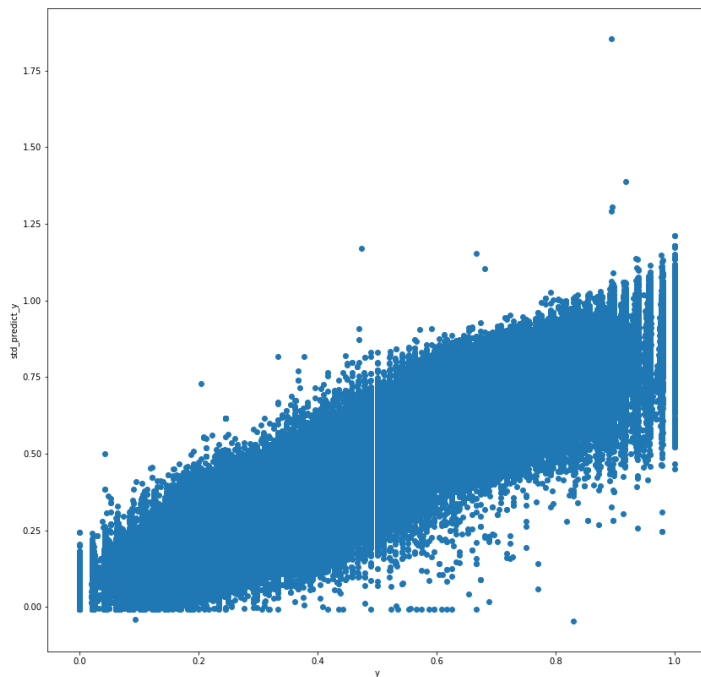
FINAL 3

FINAL 3 EDA 변경 사항

- 변수제거
 - 변경사항 없음
- 이상치 제거
 - 변경사항 없음
- 데이터 단위 변환
 - 변경사항 없음
- 데이터 변환
 - 각 변수 / 각 match_mean 변수 => 컬럼개수 줄이고, 해당 판의 상대적인 값

FINAL 3

FINAL 3 분석 결과 – ML, **NN**, LightGBM,



FINAL 3

FINAL 3 분석 결과 – ML, **NN**, LightGBM

	MAE(Mean Absolute Error)
Multilinear	0.200827
NN	0.149
LightGBM	0.2565
Kaggle 1등	0.0197

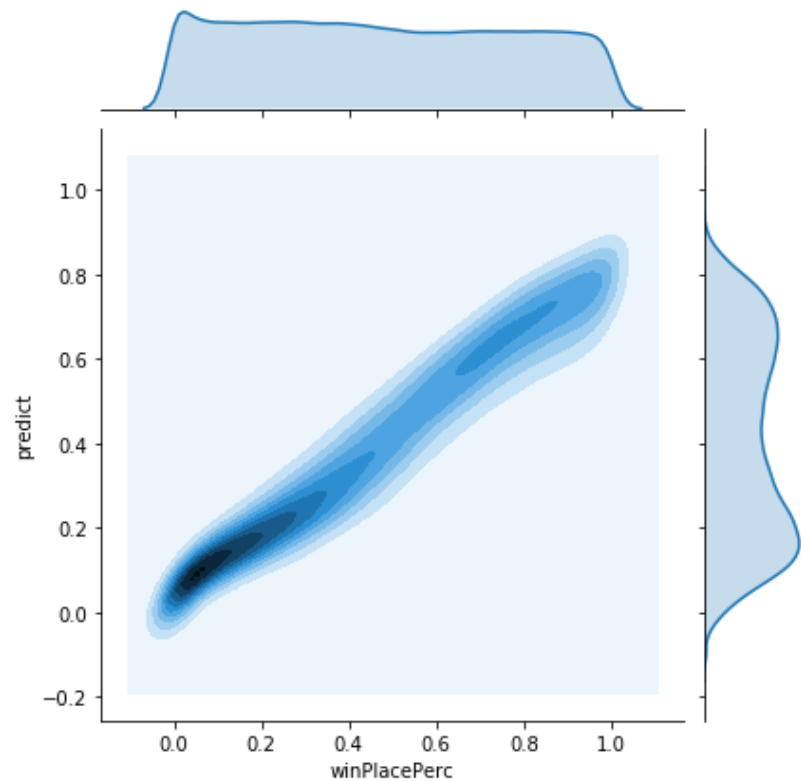
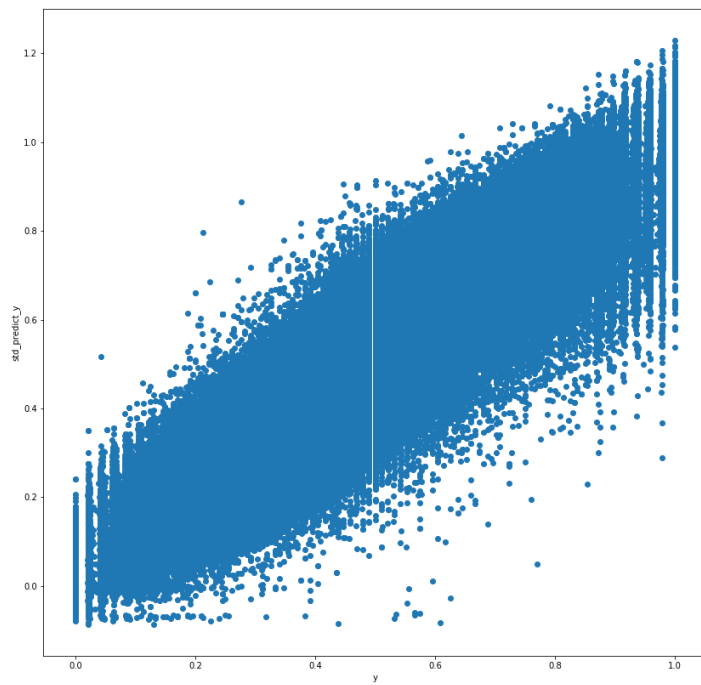
FINAL 4

FINAL 4 EDA 변경 사항

- 변수제거
 - 변경사항 없음
- 이상치 제거
 - 변경사항 없음
- 데이터 단위 변환
 - 변경사항 없음
- 데이터 변환
 - 비율과 함께 각_mean 변수 그대로 적용

FINAL 4

FINAL 4 분석 결과 – ML, NN, LightGBM, **RandomForest**



FINAL 4

FINAL 4 분석 결과 – ML, NN, LightGBM, **RandomForest**

	MAE(Mean Absolute Error)
RandomForest	0.0464
NN	0.09
LightGBM	0.25
Kaggle 1등	0.0197

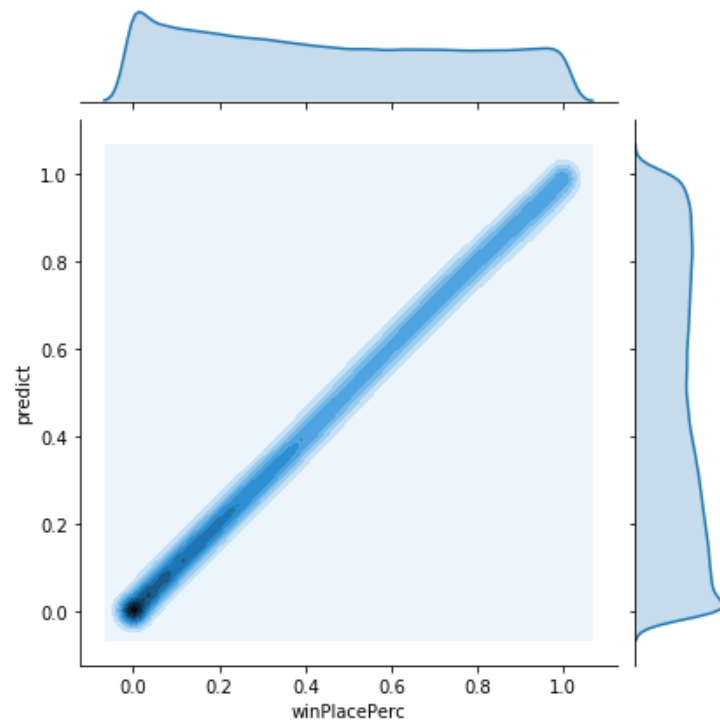
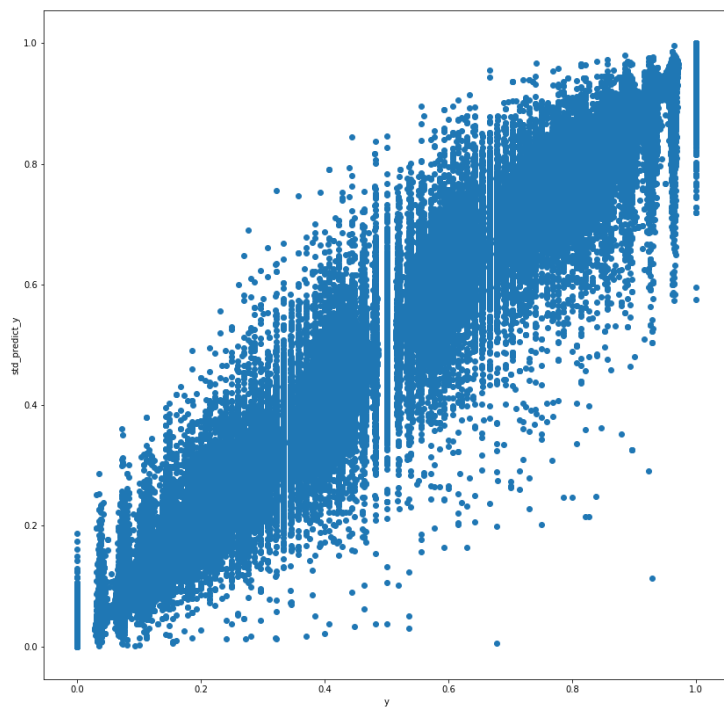
FINAL 5

FINAL 5 EDA 변경 사항

- 변수제거
 - 변경사항 없음
- 이상치 제거
 - 변경사항 없음
- 데이터 단위 변환
 - 변경사항 없음
- 데이터 변환
 - 해당 매치의 min, max값 추가 => 컬럼 개수 56개 증가

FINAL 5

FINAL 5 분석 결과 – ML, NN, LiatGBM, **RandomForest**



FINAL 5

FINAL 5 분석 결과 – ML, NN, LightGBM, **RandomForest**

	MAE(Mean Absolute Error)
RandomForest	0.041575
NN	0.155525
LightGBM	0.263425
Kaggle 1등	0.0197

CONTENTS

3






결론

- 결론 및 향후 계획

Kaggle 에서 제공해준 데이터를 바로 사용하기에는 무의미한 데이터가 많아, predictor 간의 관계를 찾아 새로히 유의미한 데이터로 가공하는것에 상당한 시간과 노력이 필요했고, 향후, 더 나은 관계를 찾을수 있다면 데이터 가공을 할 예정.

다양한 머신러닝 기법들을 지도학습을 기반으로(기본선형모델부터 Tree기반, Boosting기반, Deep Neural Network) 사용하여 RandomForest, Tree기반의 머신러닝 방법이 가장 좋은 결과를 예측.

Kaggle 최종 순위

473	▼ 66	Gion		0.0653	5	16d
474	new	Ansh Goyal		0.0654	5	4d
475	new	imdk		0.0657	6	39m
<div> <div>←</div> <div>Your Best Entry ↑</div> <div>→</div> </div> <div>Your submission scored 0.0733, which is not an improvement of your best score. Keep trying!</div>						
476	▼ 68	shigejun		0.0660	1	1mo
477	▼ 68	Alexandr M		0.0661	1	9d

RandomForest, Tree기반의 머신러닝 방법이 가장 좋은 결과를 예측.

향후, 아직 사용해 보지 못한 방법들과, 더 좋은 기법이 나온다면 써볼예정