

PUBG(BATTLEGROUNDS) EDA 및 예측모델 만들기

목표

- 2018년 8월경 배틀그라운드 게임 데이터를 이용해 EDA 및 최종 순위 예측모델 만들기

데이터

- train.csv – 순위 예측을 위한 여러가지 요소의 훈련 데이터
- test.csv – 순위 예측을 위한 여러가지 요소의 테스트 데이터
- sample_submission.csv – 샘플 제출 파일

데이터 필드

- DBNOs - Number of enemy players knocked.
- assists - Number of enemy players this player damaged that were killed by teammates.
- boosts - Number of boost items used.
- damageDealt - Total damage dealt. Note: Self inflicted damage is subtracted.
- headshotKills - Number of enemy players killed with headshots.
- heals - Number of healing items used.
- Id - Player's Id
- killPlace - Ranking in match of number of enemy players killed.
- killPoints - Kills-based external ranking of player. (Think of this as an Elo ranking where only kills matter.) If there is a value other than -1 in rankPoints, then any 0 in killPoints should be treated as a "None".
- killStreaks - Max number of enemy players killed in a short amount of time.
- kills - Number of enemy players killed.
- longestKill - Longest distance between player and player killed at time of death. This may be misleading, as downing a player and driving away may lead to a large longestKill stat.
- matchDuration - Duration of match in seconds.
- matchId - ID to identify match. There are no matches that are in both the training and testing set.

- matchType - String identifying the game mode that the data comes from. The standard modes are "solo", "duo", "squad", "solo-fpp", "duo-fpp", and "squad-fpp"; other modes are from events or custom matches.
- rankPoints - Elo-like ranking of player. This ranking is inconsistent and is being deprecated in the API's next version, so use with caution. Value of -1 takes place of "None".
- revives - Number of times this player revived teammates.
- rideDistance - Total distance traveled in vehicles measured in meters.
- roadKills - Number of kills while in a vehicle.
- swimDistance - Total distance traveled by swimming measured in meters.
- teamKills - Number of times this player killed a teammate.
- vehicleDestroys - Number of vehicles destroyed.
- walkDistance - Total distance traveled on foot measured in meters.
- weaponsAcquired - Number of weapons picked up.
- winPoints - Win-based external ranking of player. (Think of this as an Elo ranking where only winning matters.) If there is a value other than -1 in rankPoints, then any 0 in winPoints should be treated as a "None".
- groupId - ID to identify a group within a match. If the same group of players plays in different matches, they will have a different groupId each time.
- numGroups - Number of groups we have data for in the match.
- maxPlace - Worst placement we have data for in the match. This may not match with numGroups, as sometimes the data skips over placements.
- winPlacePerc - The target of prediction. This is a percentile winning placement, where 1 corresponds to 1st place, and 0 corresponds to last place in the match. It is calculated off of maxPlace, not numGroups, so it is possible to have missing chunks in a match.

일부 코드

```
1 train.head()
```

	id	groupId	matchId	assists	boosts	damageDealt	DBNOs	headshotKills	heals	killPlace	...	revives	rideDistance	road
0	7f96b2f878858a	4d4b580de459be	a10357fd1a4a91	0	0	0.00	0	0	0	60	...	0	0.0000	
1	ee90569b9d03c	684d5656442f9e	aeb375fc57110c	0	0	91.47	0	0	0	57	...	0	0.0045	
2	1ea90ac73de72	6a4a42c3245a74	110163d8bb94ae	1	0	68.00	0	0	0	47	...	0	0.0000	
3	4616d365dd2853	a930a9c79cd721	f1f1f4ef412d7e	0	0	32.90	0	0	0	75	...	0	0.0000	
4	315c96c26c9aac	de04010b3458dd	6dc8ff871e21e6	0	0	100.00	0	0	0	45	...	0	0.0000	

5 rows × 29 columns

```
1 train.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4446966 entries, 0 to 4446965
Data columns (total 29 columns):
#   Column              Dtype
---  -
0   Id                   object
1   groupId             object
2   matchId             object
3   assists             int64
4   boosts              int64
5   damageDealt         float64
6   DBNOs              int64
7   headshotKills       int64
8   heals              int64
9   killPlace          int64
10  killPoints          int64
11  kills               int64
12  killStreaks         int64
13  longestKill         float64
14  matchDuration       int64
15  matchType           object
16  maxPlace            int64
17  numGroups           int64
18  rankPoints          int64
19  revives             int64
20  rideDistance        float64
21  roadKills           int64
22  swimDistance        float64
23  teamKills           int64
24  vehicleDestroys     int64
25  walkDistance        float64
26  weaponsAcquired     int64
27  winPoints           int64
28  winPlacePerc        float64
dtypes: float64(6), int64(19), object(4)
memory usage: 983.9+ MB
```

Match Types

```
1 print('Match Counts :',train["matchId"].nunique())
```

Match Counts : 47965

```
1 mat_type = train["matchType"].value_counts().to_frame().reset_index()
2 mat_type.columns = ["Type","Count"]
3 mat_type
```

	Type	Count
0	squad-fpp	1756186
1	duo-fpp	996691
2	squad	626526
3	solo-fpp	536762
4	duo	313591
5	solo	181943
6	normal-squad-fpp	17174
7	crashfpp	6287
8	normal-duo-fpp	5489
9	flarefpp	2505
10	normal-solo-fpp	1682
11	flarefpp	718
12	normal-squad	516
13	crashtpp	371
14	normal-solo	326
15	normal-duo	199