Team 1

League of legends api 데이터 분석

팀 유저 이름

강민수, 정영화

발표일자

2023-12-04

시작

게임의 승패를 예측할 수 없을까?

시작

데이터 추출

데이터 전처리

데이터 분석
머신러닝

목차

목차를 입력하세요.

챌린저 유저

puuid 값

매치 정보

상세 매치

챌린저 300명의 유저정보

summonerId

챌린저 유저

puuid 값

매치 정보

상세 매치

매치 api 사용을 위한 puuid 값 호출

챌린저 유저 | puuid 값 | 매치 정보 | 상세 매치

매치 고유값 호출

챌린저 유저

puuid 값

매치 정보

매치 상세

매치 상세 데이터값 호출

```
function getUserApi(number) {
 var url =
'https://kr.api.riotgames.com/lol/league/v4/grandmasterleagues/by-queue/RANKED_SOLO_5x5';
 var queryParams = {
   'api_key': api_key
 };
 $.ajax({
   url: url,
   type: 'GET',
   data: queryParams,
   success: function (json) {
     var max_number = number+20;
     for (number; number < max_number; number++) {
       var summoner_id = json.entries[number].summonerId;
       getPuuidApi(summoner_id);
     }
   },
 });

 setTimeout(function (){
   if (call_count !=100){
     call_count+=20;
     getUserApi(call_count)
   }
   else
   {
     filter_match_ids = [...new Set(match_ids)];
     duplicationMatchIds();
   }
 },40000);
}
```

```
function getPuuidApi(summoner_id) {
 var url =
'https://kr.api.riotgames.com/lol/summoner/v4/summoners/' + summoner_id;
 var queryParams = {
   'api_key': api_key
 };
 $.ajax({
   url: url,
   type: 'GET',
   data: queryParams,
   success: function (json) {
     var puuid = json.puuid;
     getMatchCode(puuid);
   }
 });
}

function getMatchCode(puuid) {
 var url =
'https://asia.api.riotgames.com/lol/match/v5/matches/by-puuid/' + puuid + '/ids';
 var queryParams = {
   'count': '5',
   'api_key': api_key
 };
 $.ajax({
   url: url,
   type: 'GET',
   data: queryParams,
   success: function (json) {
     for (var i = 0; i < json.length; i++) {
       match_ids.push(json[i]);
     }
   }
 });
}
```

```
function duplicationMatchIds(){

 var max_match_count = match_count+20;
 var check = true;
 setTimeout(function (){
 for (match_count; match_count < max_match_count; match_count++) {
   if (match_count!=filter_match_ids.length){

getDetailMatch(filter_match_ids[match_count]);
   }
   else
   {

downloadCSV(return_data,'last_data.csv');
     check = false;
   }
 }
 if (check){
   duplicationMatchIds();
 }
 },40000)
}
```

```
function getDetailMatch(match_id) {
 //
https://asia.api.riotgames.com/lol/match/v5/matches/undefined?api key=RGAPI-e93d6d79-8b48-48fa-a2c7-0a84c063c97c
 var url =
'https://asia.api.riotgames.com/lol/match/v5/matches/' + match_id;
 var queryParams = {
   'api_key': api_key
 };
 $.ajax({
   url: url,
   type: 'GET',
   data: queryParams,
   success: function (json) {
     for (var i = 0; i < json.info.participants.length; i++) {
       if (i%5==0){
         var data = {};
         data.gametime = data.gametime =(parseInt(json.info.gameEndTimestamp)-parseInt(json.info.gameStartTimestamp))/1000/60;
         data.baron = 0;
//아군 바론 처치수 -> 팀 바론처치수
         data.damagePerMinute =0; // 분당딜량
         data.dealttakenPerMinute = 0;
// dealttaken 챔피언에게 받은 데미지 -> 팀원들 챔피언 받은 데미지 합
         data.killsPerMinute = 0;
// kills 팀원들 킬수 합
         data.deathsPerMinute = 0;
// deaths 데스 수 -> 팀원 데스 수
         data.dragon = 0;          //
dragon 아군 드래곤 처치 수 -> 팀 드래곤 처치 수
```

사용 데이터 테이블

게임 찾기

No	데이터명	데이터의미
1	baron	팀 바론 처치 수
2	herald	팀 전령 처치 수
3	dragon	팀 드래곤 처치 수
4	elderdragon	팀 장로드래군 처치 수
5	killsPerMinute	팀 분당 킬 수
6	deathsPerMinute	팀 분당 데스 수
7	goldPerMinute	팀 분당 골드 획득량
8	csPerMinute	팀 분당 cs수급량
9	visionwardsbuyPerMinute	팀 분당 제어와드 구매 수
10	visionScorePerMinute	팀 분당 시야점수
11	damagePerMinute	팀 분당 가한 피해량
12	dealttakenPerMinute	팀 분당 받은 피해량
13	ftkill	팀 퍼스트포탑 유무
14	fbkill	팀 퍼스트블러드 유무
15	turretkills	팀 포탑 부순 갯수

사용 데이터

게임 찾기

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	gametime	baron	damagePerMinute	dealttakenPerMinute	killsPerMinute	deathsPerMinute	dragon	ftkill	fbkill	goldPerMinute	herald	elderdragon	turretkills	visionwardsbuyPerMinute	visionScorePerMinute	csPerMinute	win
2	36.58578333	1	4440.037673	4638.195073	1.093320857	1.1479869	2	0	1	2038.606795	0	0	6	0.737991579	6.289873579	20.77309629	1
3	36.58578333	1	3867.484106	5718.12275	1.1479869	1.093320857	3	0	0	1978.878879	2	0	5	0.737991579	8.36461626	21.01909348	0
4	25.44975	1	3608.001771	4517.293883	1.375259089	1.178793505	1	0	1	2264.121406	1	0	7	0.471517402	5.467996163	23.53657698	1
5	25.44975	0	3614.617789	4500.515722	1.178793505	1.375259089	2	1	0	1927.678986	1	0	4	0.510810519	4.833687893	22.20061101	0
6	34.04175	1	4021.780635	4082.898206	0.851895099	0.793143713	3	1	1	2032.958218	1	0	7	0.734392327	7.256807286	26.320621	1
7	34.04175	0	3415.620284	4890.024749	0.793143713	0.851895099	2	0	0	1924.881959	1	0	5	1.14565203	7.664633052	25.67435575	0
8	15.28125	0	3525.299983	3451.092025	2.290388548	0.523517382	2	1	1	2441.659817	1	0	3	0.654396728	5.927262621	24.67075665	1
9	15.28125	0	2552.225804	4338.846626	0.523517382	2.290388548	0	0	0	1606.699087	0	0	0	0.654396728	4.668472574	18.38854806	0
10	19.96595	0	3502.15616	3528.557369	1.502558105	0.701193782	2	0	1	2307.164096	1	0	6	0.851449593	6.283163263	26.49510792	1
11	19.96595	0	2627.778169	4518.993587	0.701193782	1.502558105	0	1	0	1855.381053	1	0	2	0.701193782	5.918583303	22.78879793	0
12	37.66195	1	3858.922007	3977.781289	0.982423905	0.823111921	4	0	1	2062.16872	2	0	9	0.92931991	8.81109206	22.7816138	1
13	37.66195	2	3357.263517	4664.336286	0.823111921	0.982423905	1	0	0	1968.621856	0	0	6	0.955871908	6.922309726	25.17129357	0
14	23.74923333	0	6786.146997	7494.220866	2.694823833	2.147437742	0	0	1	3513.79548	0	0	2	0	0	13.26358605	0
15	23.74923333	0	6678.103148	7427.776616	2.063224497	2.694823833	0	1	0	3455.512857	0	0	4	0	0.217531655	12.50566685	1
16	15.69048333	0	2748.572385	3124.314207	1.338390893	0.701061896	1	0	1	2173.791967	1	0	2	0.828527696	4.899511047	26.76781786	1
17	15.69048333	0	2143.787123	3362.611519	0.701061896	1.338390893	1	0	0	1751.439621	0	0	0	0.637328997	5.259943223	24.40970057	0
18	31.5384	1	3130.464713	3965.578469	0.697562337	1.078050884	0	0	0	1883.923148	0	0	4	1.078050884	7.448035447	23.01955711	0
19	31.5384	1	3091.406363	4130.425133	1.078050884	0.697562337	5	1	1	2211.53629	2	1	10	0.760977095	6.712428137	24.57321868	1
20	16.20016667	0	3026.029915	3405.829158	1.35801072	0.55554984	2	0	0	2206.069982	2	0	3	0.9259164	5.090344122	28.95031944	1
21	16.20016667	0	2288.603089	3840.022222	0.49382208	1.35801072	0	0	1	1718.678807	0	0	0	0.55554984	5.647449777	25.06147056	0
22	30.1494	0	5574.609555	4922.751365	1.724744108	0.961876522	2	0	1	2276.914858	0	0	7	0.796035742	7.943101551	21.95731922	1
23	30.1494	1	3919.533504	6392.929876	0.961876522	1.724744108	2	1	0	2093.218666	2	0	3	1.160885457	7.080332422	24.0800812	0
24	27.78091667	0	7222.7372	9214.670742	2.375731542	2.555711205	0	1	0	3419.871822	0	0	2	0	0	13.03052755	0
25	27.78091667	0	8428.715961	7915.613536	2.555711205	2.375731542	0	0	1	3657.782754	0	0	5	0	0	14.75833231	1
26	14.67343333	0	5369.721763	8083.043505	2.657864667	3.680120308	0	0	1	3738.135727	0	0	0	0	0	11.04036092	0
27	14.67343333	0	7813.723489	6017.337456	3.680120308	2.657864667	0	0	0	3996.088805	0	0	4	0	0	12.67596995	1
28	29.33455	1	2992.147866	4105.602438	0.954505864	0.954505864	1	0	0	2112.648482	1	0	6	0.818147884	8.402588702	24.74897348	1
29	29.33455	0	2850.029486	3902.019973	0.954505864	0.954505864	3	1	1	1848.254902	1	0	2	1.022684855	6.541270627	24.20354156	0
30	18.68678333	0	6264.626125	7926.671881	2.408119107	2.889742929	0	1	0	3432.076459	0	0	1	0	0	12.14762305	0
31	18.68678333	0	7613.600968	6698.798705	2.889742929	2.408119107	0	0	1	3628.504224	0	0	3	0	0.84042432	12.09410929	1
32	18.4982	0	8943.322967	8483.31189	3.405736774	3.189499519	0	0	0	3852.380731	0	0	4	0	0	11.78493043	1
33	18.4982	0	7606.486305	9368.370977	3.189499519	3.405736774	0	1	1	3756.673759	0	0	2	0	0.108107458	12.10928631	0
34	32.0389	2	5029.48792	5877.043219	1.217270256	1.435754661	1	0	0	2320.69742	1	0	7	0.967573793	5.795241285	20.53753406	1
35	32.0389	0	4995.732782	6018.184145	1.435754661	1.217270256	3	0	1	2251.093255	0	0	6	0.905149677	5.595448681	21.4738958	0

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 1550 entries, 0 to 1559
Data columns (total 16 columns):
 #   Column                    Non-Null Count  Dtype
---  ------                    --------------  -----
 0   baron                     1550 non-null   int64
 1   damagePerMinute           1550 non-null   float64
 2   dealttakenPerMinute       1550 non-null   float64
 3   killsPerMinute            1550 non-null   float64
 4   deathsPerMinute           1550 non-null   float64
 5   dragon                    1550 non-null   int64
 6   ftkill                    1550 non-null   int64
 7   fbkill                    1550 non-null   int64
 8   goldPerMinute             1550 non-null   float64
 9   herald                    1550 non-null   int64
 10  elderdragon               1550 non-null   int64
 11  turretkills               1550 non-null   int64
 12  visionwardsbuyPerMinute   1550 non-null   float64
 13  visionScorePerMinute      1550 non-null   float64
 14  csPerMinute               1550 non-null   float64
 15  win                       1550 non-null   int64
dtypes: float64(8), int64(8)
memory usage: 238.1 KB
```

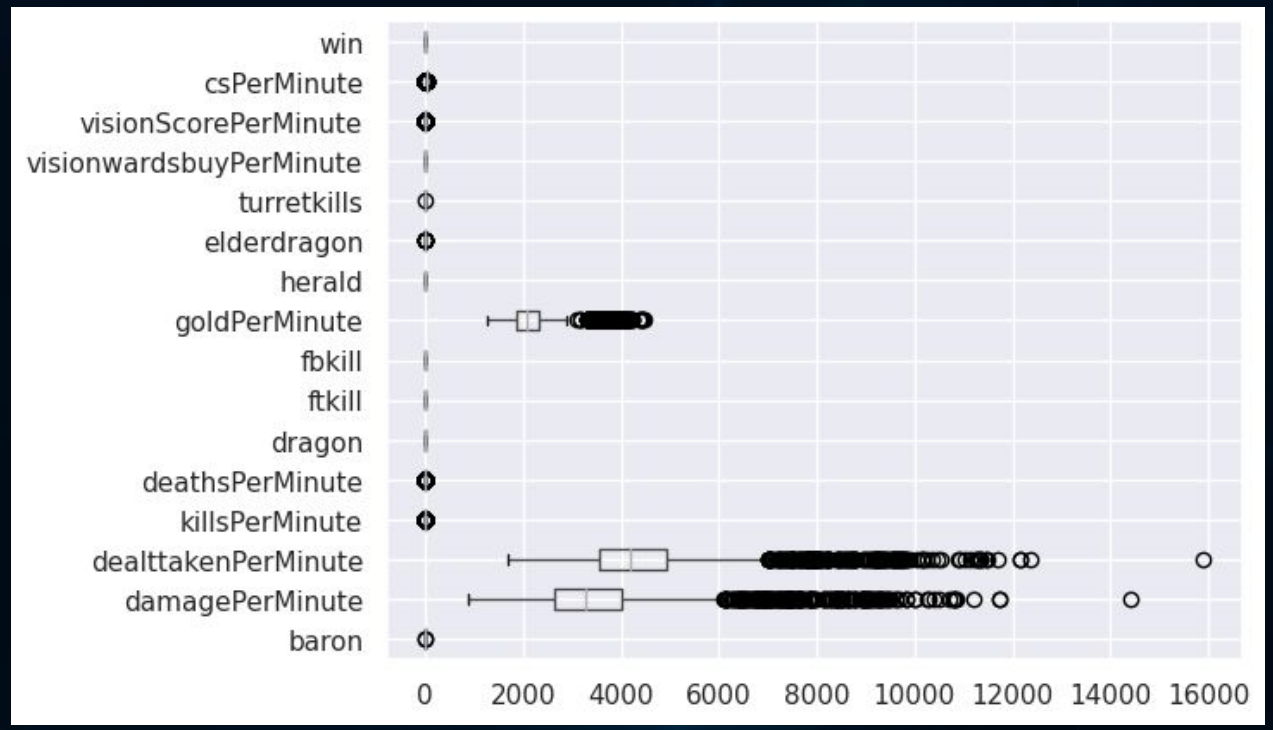
df.head()

	baron	damagePerMinute	dealttakenPerMinute	killsPerMinute	deathsPerMinute	dragon	ftkill
0	1	4440.037673	4638.195073	1.093321	1.147987	2	0
1	1	3867.484106	5718.122750	1.147987	1.093321	3	0
2	1	3608.001771	4517.293883	1.375259	1.178794	1	0
3	0	3614.617789	4500.515722	1.178794	1.375259	2	1
4	1	4021.780635	4082.898206	0.851895	0.793144	3	1

df.describe()

	baron	damagePerMinute	dealttakenPerMinute	killsPerMinute	deathsPerMinute
count	1550.000000	1550.000000	1550.000000	1550.000000	1550.000000
mean	0.316129	3781.182529	4648.441629	1.303331	1.307864
std	0.556142	1853.051020	1757.378037	0.865524	0.867726
min	0.000000	890.937000	1692.577669	0.000000	0.000000
25%	0.000000	2635.020911	3563.961183	0.744630	0.749311
50%	0.000000	3279.329956	4207.127224	1.050753	1.056637
75%	1.000000	4021.455661	4928.463128	1.470934	1.471557
max	3.000000	14406.518766	15895.732449	5.038774	5.038774

df.boxplot(vert = False)



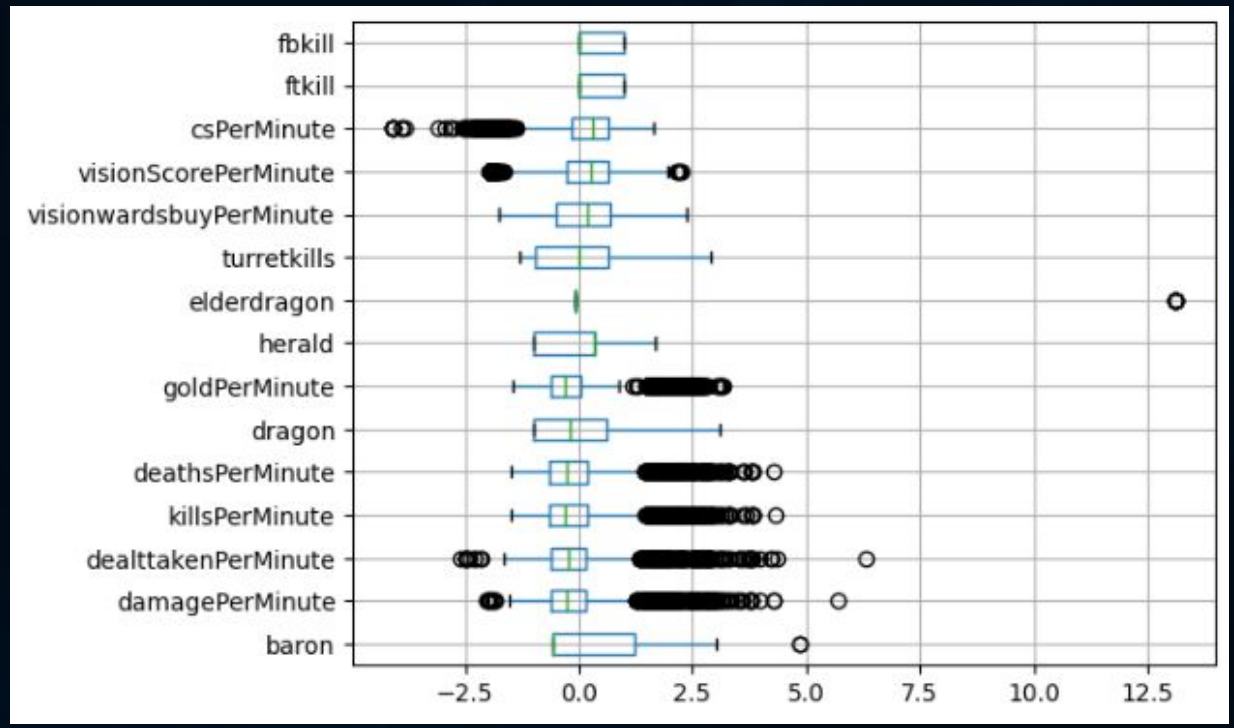
데이터 간 규모 차이가 너무 크다

검색 가나다순

재료

StandardScaler()

df.boxplot(vert = False)



정규화 후의 데이터

장로 드래곤의 이상치

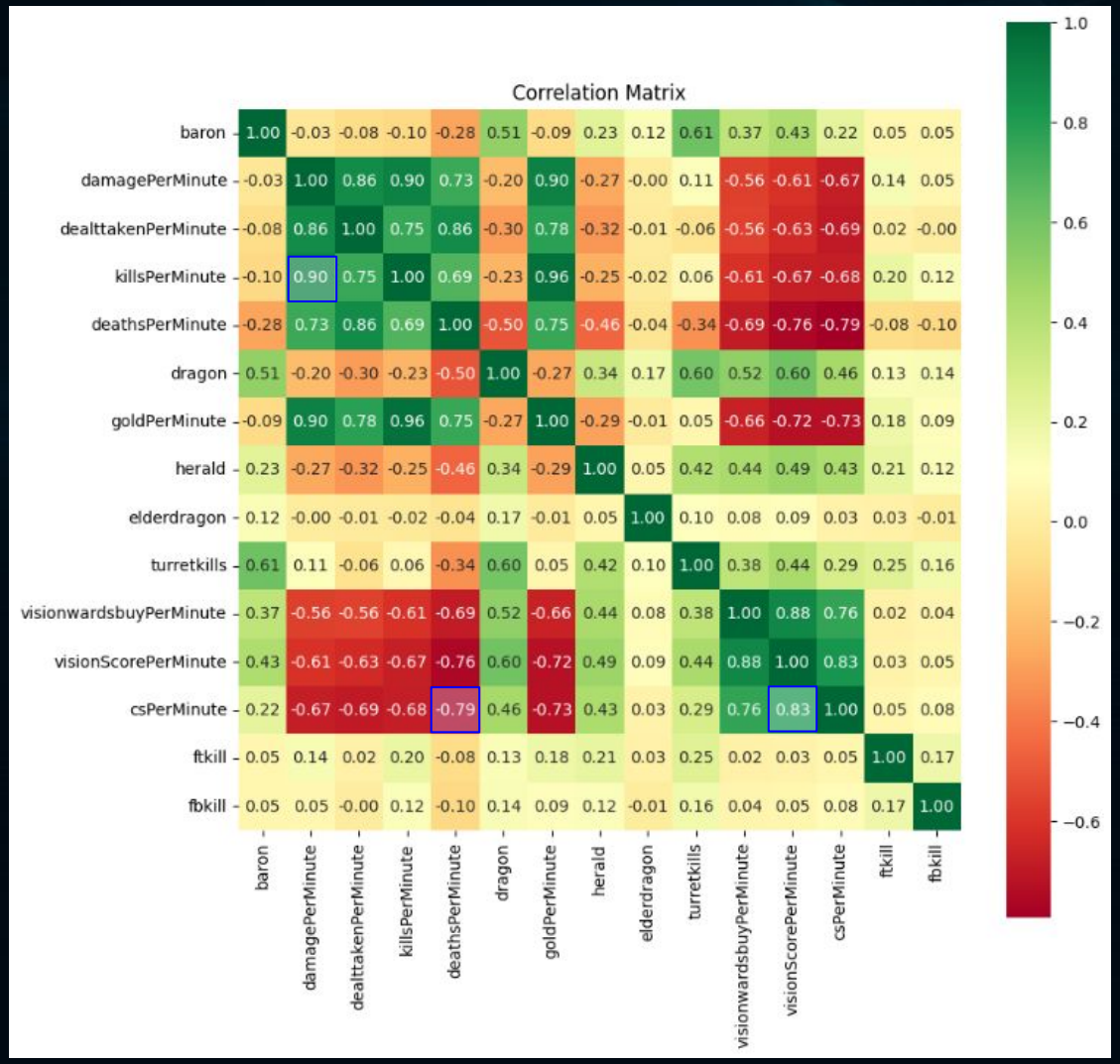
1273 65 0 63

승리

소환사의 협곡 · 개인/2인 랭크 게임 · 30:51 · 2023/12/01 · 게임 ID

점수표 개요 통계 그래프 룬

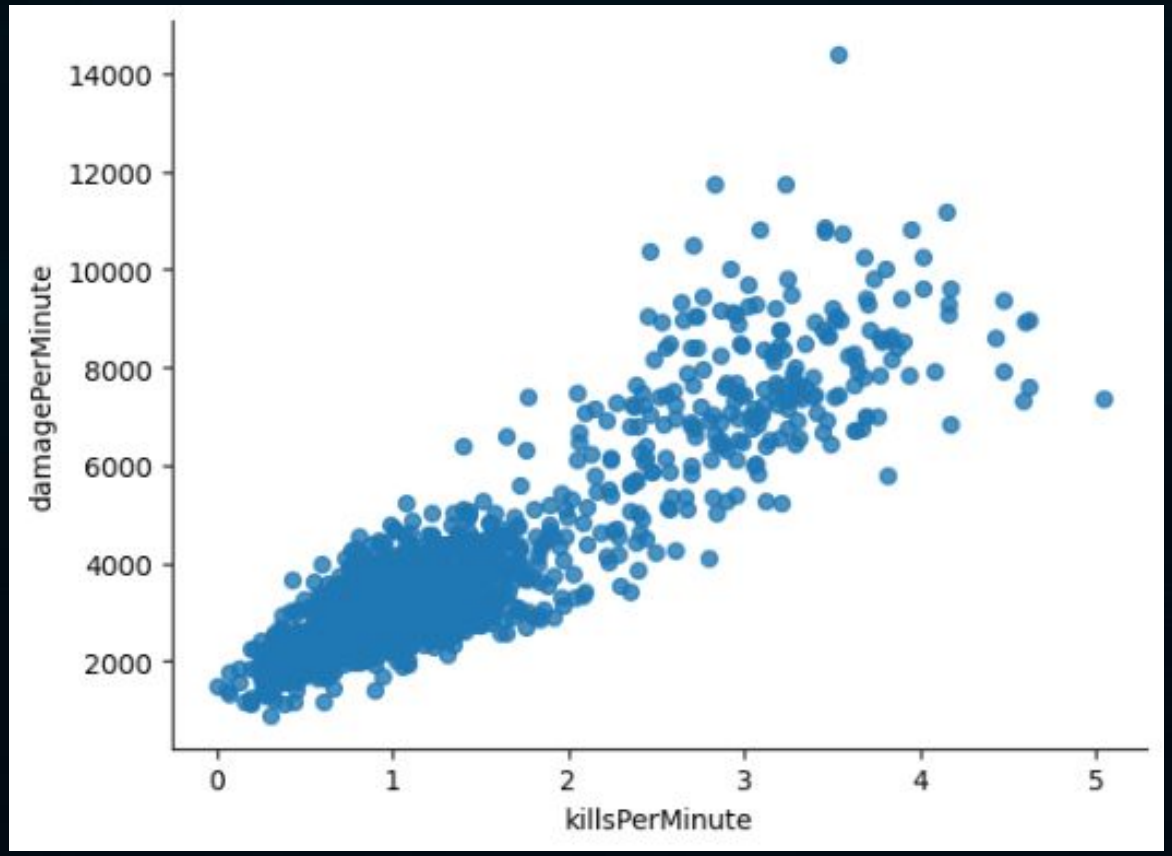
- [x] 상관계수
  - [x] sns.heatmap
  - [ ] scatter
    - [ ] baron
    - [ ] damagePerMinute
    - [ ] dealttakenPerMinute
    - [ ] killsPerMinute
    - [ ] deathsPerMinute
    - [ ] dragon
    - [ ] fbkill
    - [ ] goldPerMinute
    - [ ] herald
    - [ ] elderdragon
    - [ ] turretkills
    - [ ] visionwardsbupPerMinute
    - [ ] visionScorePerMinute
    - [ ] csPerMinute
    - [ ] tfkilil



승리
소환사의 협곡 · 개인/2인 랭크 게임 · 30:51 · 2023/12/01 · 게임 ID

점수표 개요 통계 그래프 룬

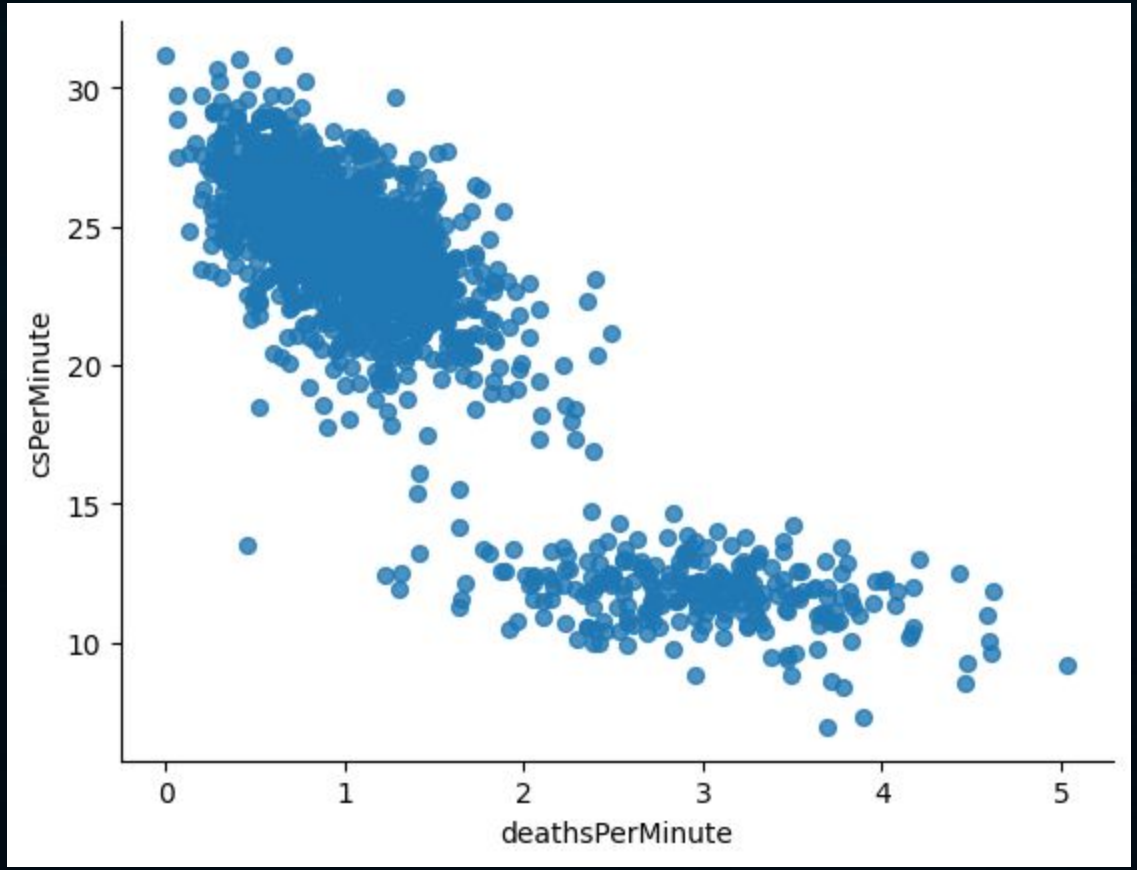
- [x] 상관계수
- [ ] sns.heatmap
- [x] scatter
  - [ ] baron
  - [x] damagePerMinute
  - [ ] dealttakenPerMinute
  - [x] killsPerMinute
  - [ ] deathsPerMinute
  - [ ] dragon
  - [ ] fbkill
  - [ ] goldPerMinute
  - [ ] herald
  - [ ] elderdragon
  - [ ] turretkills
  - [ ] visionwardsbupPerMinute
  - [ ] visionScorePerMinute
  - [ ] csPerMinute
  - [ ] tfkilil



승리
소환사의 협곡 · 개인/2인 랭크 게임 · 30:51 · 2023/12/01 · 게임 ID

점수표 개요 통계 그래프 룬

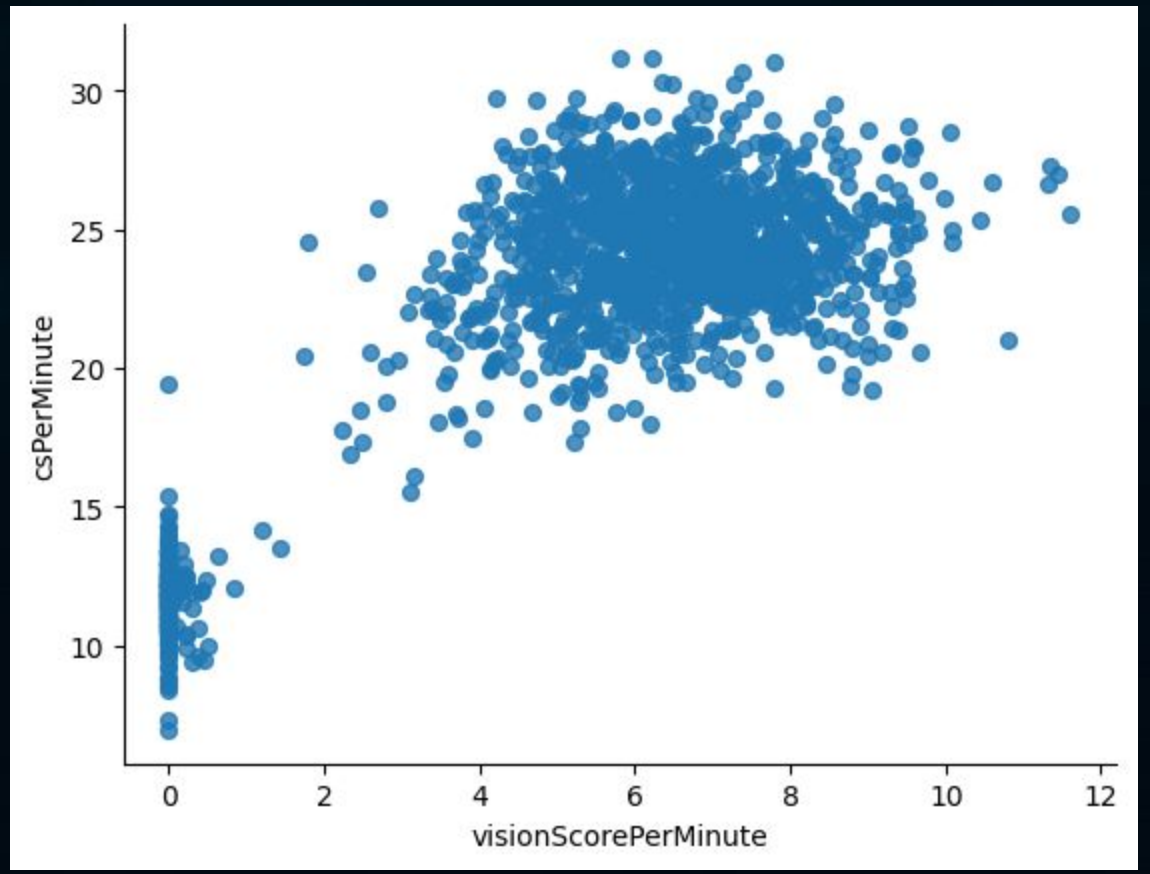
- [x] 상관계수
  - [ ] sns.heatmap
  - [x] scatter
    - [ ] baron
    - [ ] damagePerMinute
    - [ ] dealttakenPerMinute
    - [ ] killsPerMinute
    - [x] deathsPerMinute
    - [ ] dragon
    - [ ] fbkill
    - [ ] goldPerMinute
    - [ ] herald
    - [ ] elderdragon
    - [ ] turretkills
    - [ ] visionwardsbupPerMinute
    - [ ] visionScorePerMinute
    - [x] csPerMinute
    - [ ] tfkilil



승리
소환사의 협곡 · 개인/2인 랭크 게임 · 30:51 · 2023/12/01 · 게임 ID

점수표 개요 통계 그래프 룬

- [x] 상관계수
  - [ ] sns.heatmap
  - [x] scatter
    - [ ] baron
    - [ ] damagePerMinute
    - [ ] dealttakenPerMinute
    - [ ] killsPerMinute
    - [ ] deathsPerMinute
    - [ ] dragon
    - [ ] fbkill
    - [ ] goldPerMinute
    - [ ] herald
    - [ ] elderdragon
    - [ ] turretkills
    - [ ] visionwardsbupPerMinute
    - [x] visionScorePerMinute
    - [x] csPerMinute
    - [ ] tfkilil



RandomForest

여러 개의 의사결정트리를 만들어 그 결과를 평균내는 앙상블 모델

XGBoost

Gradient Boosting 알고리즘 기반, 병렬 처리를 사용해 학습과 예측이 빠른 모델

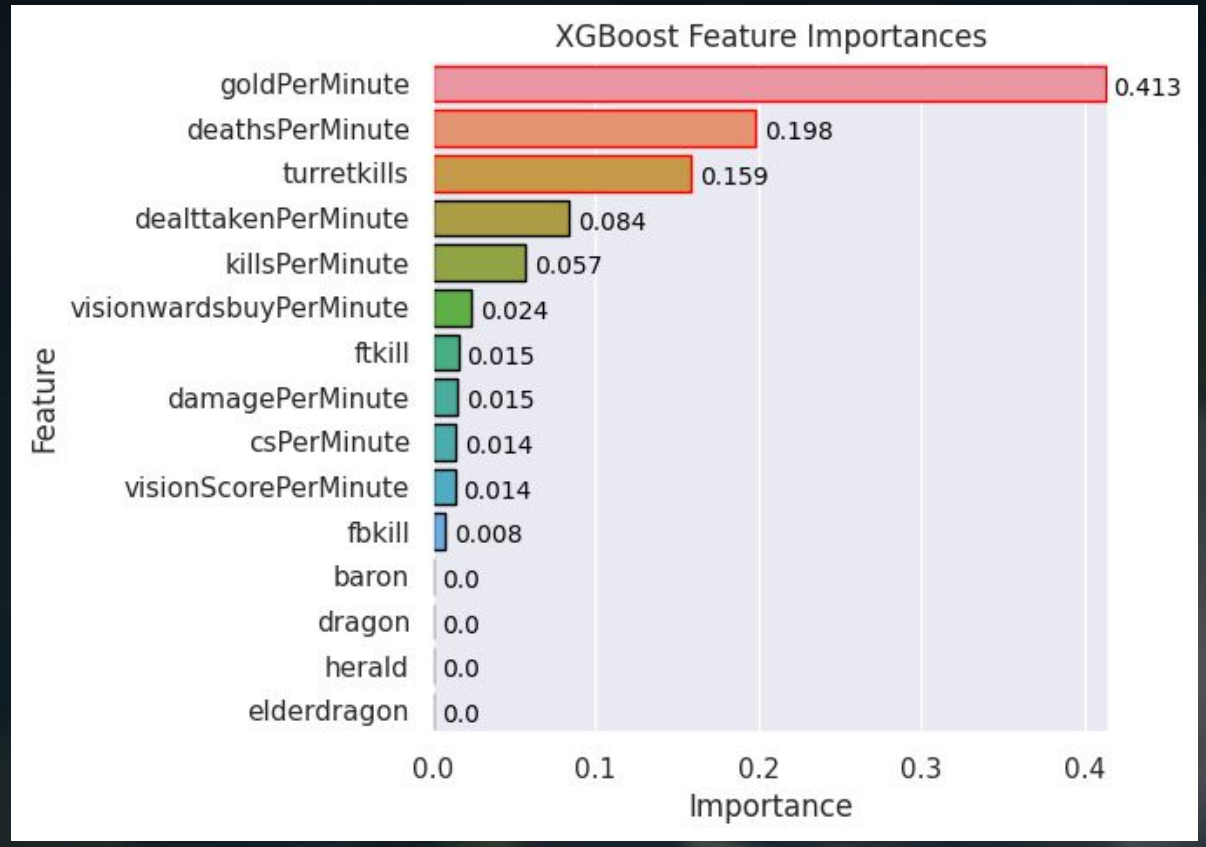
LightGBM

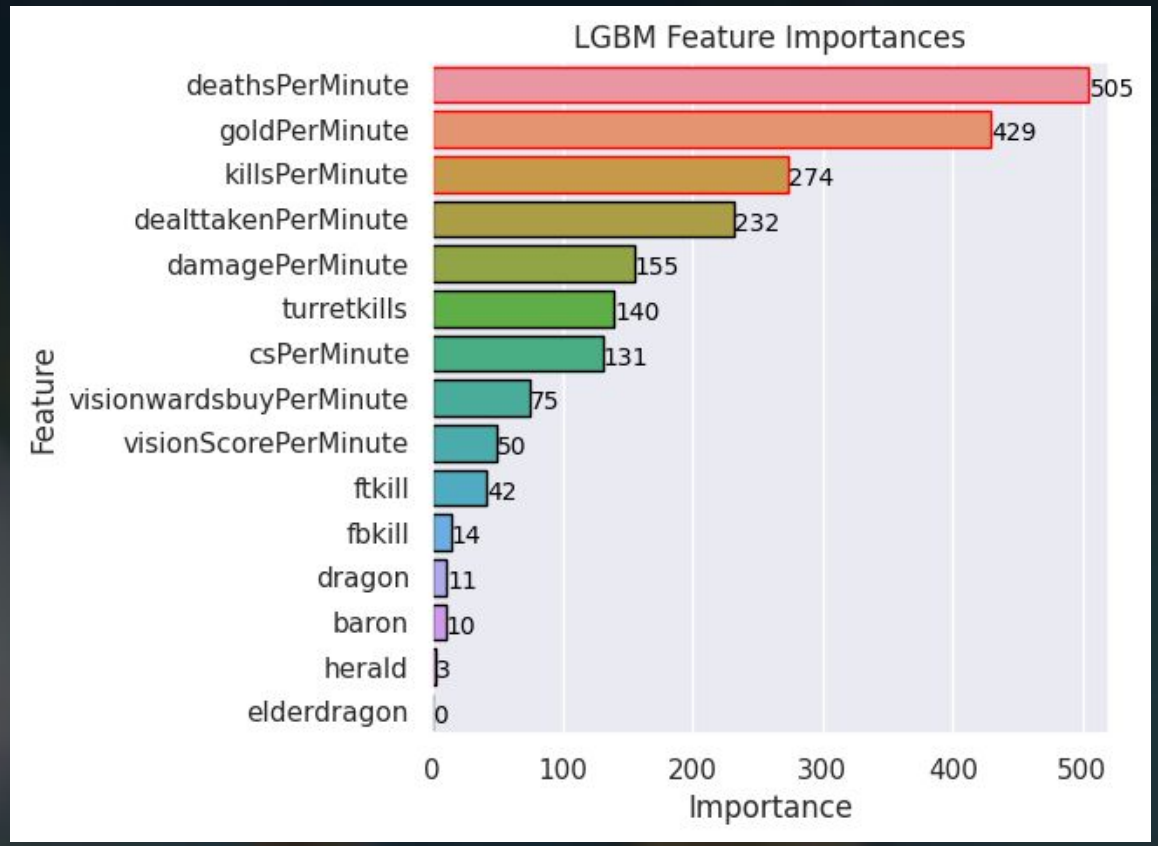
XGBoost보다 빠르고 적은 메모리를 사용한 모델

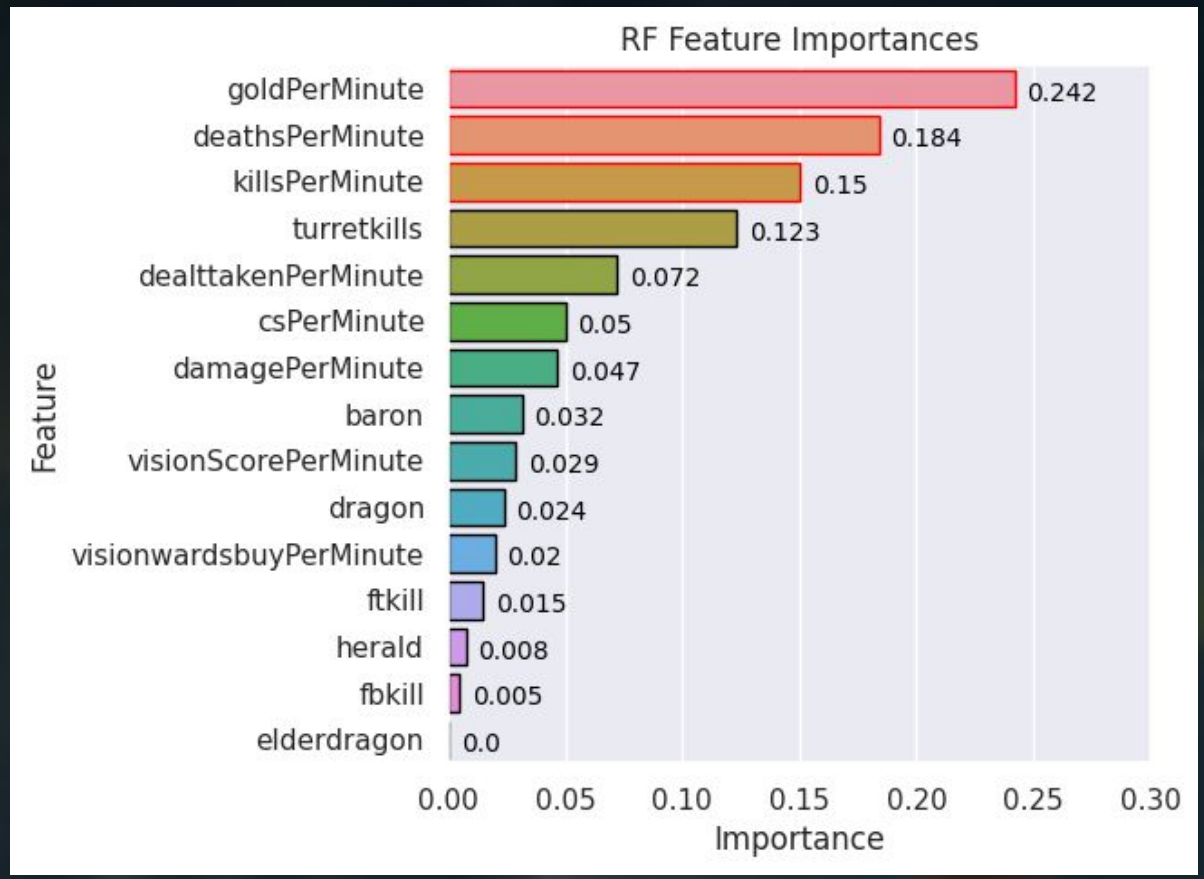
LogisticRegression

분류 문제를 해결하기 위한 알고리즘

```
X_train, X_test, Y_train, Y_test = train_test_split(df, y, test_size = 0.3)
```


XGBoost Feature Importances
goldPerMinute 0.413
deathsPerMinute 0.198
turretkills 0.159
dealttakenPerMinute 0.084
killsPerMinute 0.057
visionwardsbuyPerMinute 0.024
ftkill 0.015
damagePerMinute 0.015
csPerMinute 0.014
visionScorePerMinute 0.014
fbkill 0.008
baron 0.0
dragon 0.0
herald 0.0
elderdragon 0.0
Feature
0.0 0.1 0.2 0.3 0.4
Importance


LGBM Feature Importances
deathsPerMinute 505
goldPerMinute 429
killsPerMinute 274
dealttakenPerMinute 232
damagePerMinute 155
turretkills 140
csPerMinute 131
visionwardsbuyPerMinute 75
visionScorePerMinute 50
ftkill 42
fbkill 14
dragon 11
baron 10
herald 3
elderdragon 0
Feature
0 100 200 300 400 500
Importance

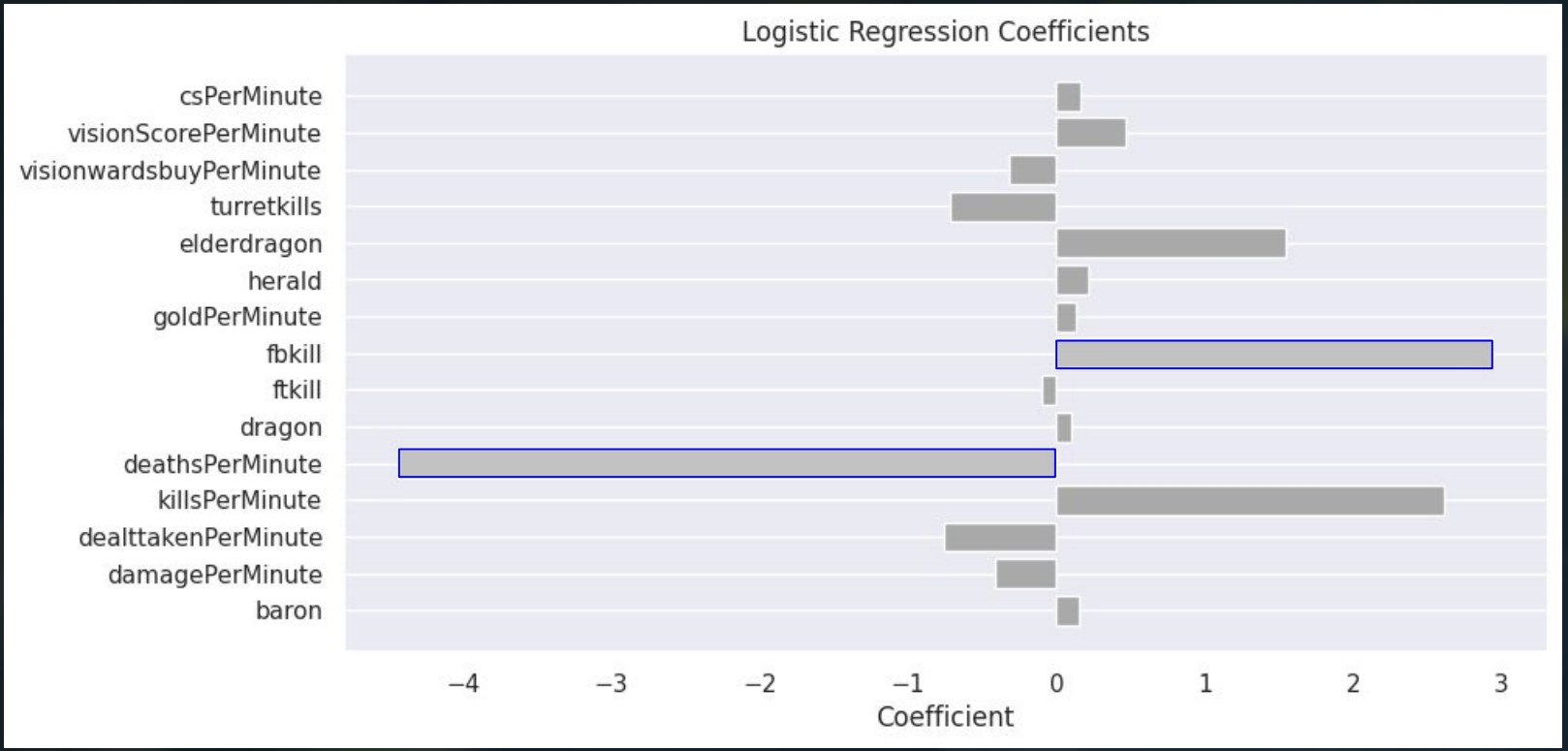

RF Feature Importances
goldPerMinute 0.242
deathsPerMinute 0.184
killsPerMinute 0.15
turretkills 0.123
dealttakenPerMinute 0.072
csPerMinute 0.05
damagePerMinute 0.047
baron 0.032
visionScorePerMinute 0.029
dragon 0.024
visionwardsbuyPerMinute 0.02
ftkill 0.015
herald 0.008
fbkill 0.005
elderdragon 0.0
Feature
0.00 0.05 0.10 0.15 0.20 0.25 0.30
Importance

LogisticRegression

y = 0.15004914 * baron - 0.41135821 * damagePerMinute - 0.75511225 * dealttakenPerMinute + 2.61877185 * killsPerMinute - 4.43497745 * deathsPerMinute + 0.09751238 * dragon - 0.09541447 * ftkill + 2.94225111 * fbkill + 0.13146507 * goldPerMinute + 0.21439989 * herald + 1.54957357 * elderdragon - 0.71523348 * turretkills - 0.31454591 * visionwardsbuyPerMinute + 0.47106488 * visionScorePerMinute + 0.16063479 * csPerMinute - 0.09013465

LogisticRegression

y = 0.15004914 * baron - 0.41135821 * damagePerMinute - 0.75511225 * dealttakenPerMinute + 2.61877185 * killsPerMinute - 4.43497745 * deathsPerMinute + 0.09751238 * dragon - 0.09541447 * ftkill + 2.94225111 * fbkill + 0.13146507 * goldPerMinute + 0.21439989 * herald + 1.54957357 * elderdragon - 0.71523348 * turretkills - 0.31454591 * visionwardsbuyPerMinute + 0.47106488 * visionScorePerMinute + 0.16063479 * csPerMinute - 0.09013465



“ deathsPerMinute ”

“ fbkill ”

RandomForest	XGBoost	LightGBM	LogisticRegression
n_estimators = 70	n_estimators = 100 learning_rate = 0.1 max_depth = 3	n_estimators = 70	Default

accuracy_score(Y_test, y_pred)

RandomForest	XGBoost	LightGBM	LogisticRegression
94.87 %	95.51 %	95.29 %	94.23 %

실제 게임 데이터셋

게임 찾기

LightGBM

No	예측값 / 실제값	예측한 패 확률 / 승 확률
1	승 / 패	63 % / 37 %
2	패 / 패	93 % / 7 %
3	승 / 승	15 % / 85 %
4	패 / 패	7 % / 93 %
5	패 / 패	93 % / 7 %
6	승 / 승	7 % / 93 %
7	패 / 패	87 % / 13 %
8	패 / 승	61 % / 39 %
9	패 / 패	59 % / 41 %
10	패 / 패	77 % / 23 %

Test Accuracy : 80 %

1. 더 많은 데이터로 학습
2. 실시간 승부 예측
3. ai 프로젝트

Q & A

453 강민수 온라인

491 정영화 자리 비움

Q & A