实验报告

——数据分析与特征提取

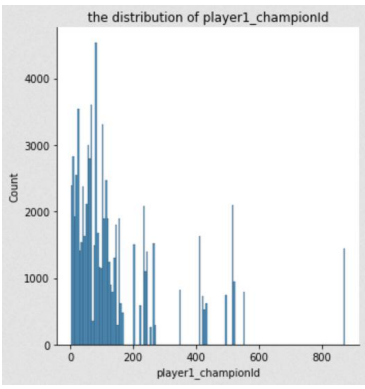
PB19020499　　桂栋南

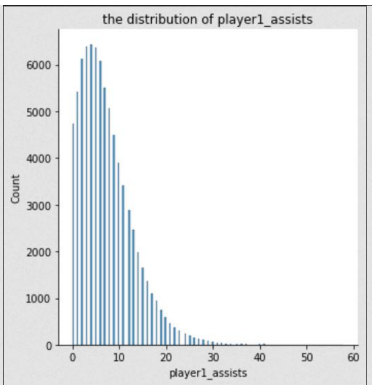
目录

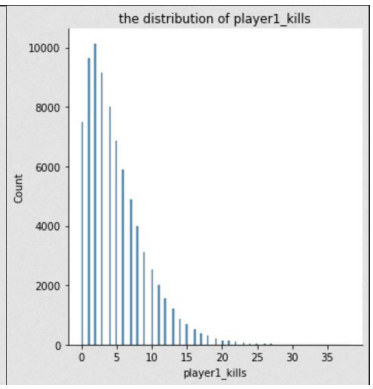
1. 数据分析

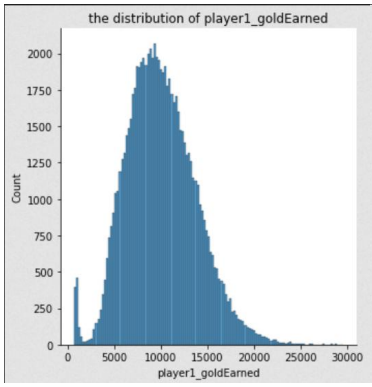
1.1. 单个特征的分布

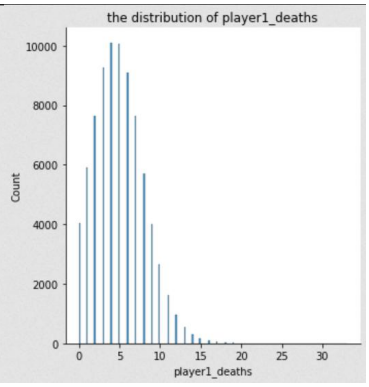
- player_goldEarned 的分布（以 player1 为例）
- player_kills 的分布（以 player1 为例）
- player_deaths 的分布（以 player1 为例）
- player_assists 的分布（以 player1 为例）
- player_championId 的分布（以 player1 为例）





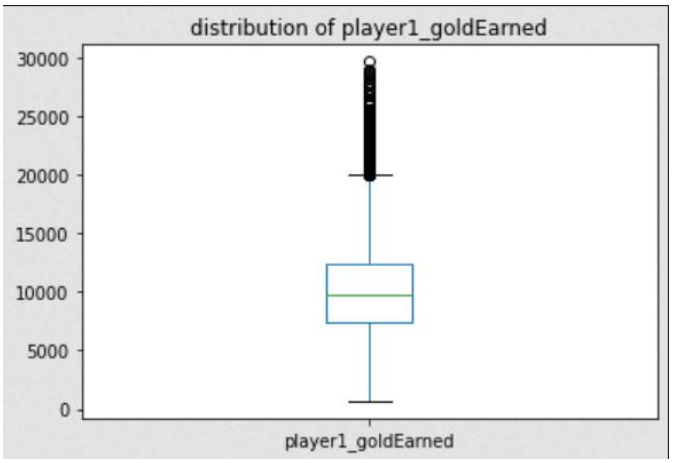






1.2. 异常值分析

- 最后并没有采用异常值处理后的数据进行分析
- 以 player1_goldEarned 为例
    - 箱型图分析
    - 正态拟合分析



离群值较多，不易分析，采用正态拟合尝试

- 3*sigma 离群值舍弃

0	0	420	11	13
1	1	420	11	13
2	2	420	11	13
3	3	420	11	13
4	4	420	11	13
...	...	...	...	...
79995	79995	420	11	13
79996	79996	420	11	13
79997	79997	420	11	13
79998	79998	420	11	13
79999	79999	420	11	13

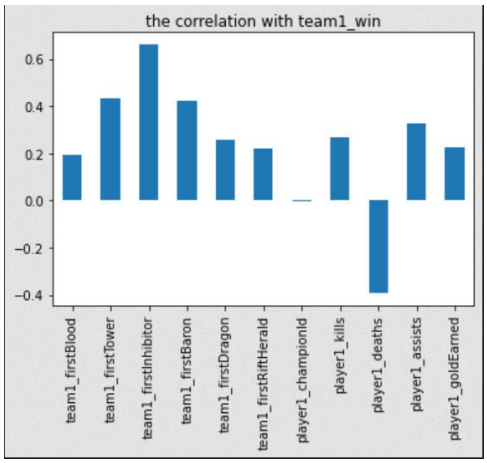
79632 rows × 81 columns

剩余 79632，大致删除 300+条数据。

1.3. 特征间的关系

1.3.1. 与胜率之间的相关性(以 player1 为例)

- Pearson 相关性
  - 可以看出 team1_firstInhibitor 与胜率相关性最大



1.3.2. player 属性内部的相关性（以 player1 为例）

可以看出 player 内部 kills 与 goldEarned 的相关性最高（与游戏尝试相符）

	player1_championId	player1_kills	player1_deaths	player1_assists	player1_goldEarned
player1_championId	NaN	-0.037501	0.018510	0.034845	-0.053181
player1_kills	-0.037501	NaN	0.123403	0.189374	0.779798
player1_deaths	0.018510	0.123403	NaN	0.197157	0.288500
player1_assists	0.034845	0.189374	0.197157	NaN	0.383657
player1_goldEarned	-0.053181	0.779798	0.288500	0.383657	NaN

1.3.3. 不同 player 之间的相关性

以 goldEarned 为例

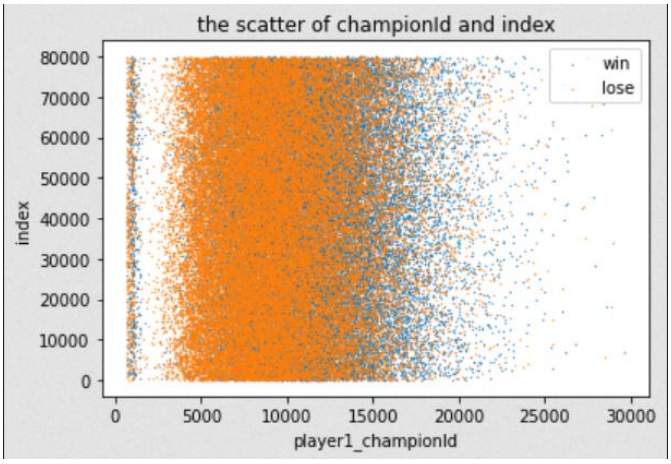
- 竟然发现不同 player 的经济是正相关的，不分队伍，震惊！
- 这或许可以从游戏时长的角度来解释

	player1_goldEarned	player2_goldEarned	player3_goldEarned	player4_goldEarned	player5_goldEarned	player6_goldEarned
player1_goldEarned	NaN	0.613010	0.622110	0.623688	0.626577	0.565356
player2_goldEarned	0.613010	NaN	0.616243	0.622493	0.626253	0.561433
player3_goldEarned	0.622110	0.616243	NaN	0.618477	0.626184	0.564935
player4_goldEarned	0.623688	0.622493	0.618477	NaN	0.617865	0.562033
player5_goldEarned	0.626577	0.626253	0.626184	0.617865	NaN	0.565958
player6_goldEarned	0.565356	0.561433	0.564935	0.562033	0.565958	NaN
player7_goldEarned	0.561295	0.560658	0.558466	0.558811	0.562763	0.614277
player8_goldEarned	0.563508	0.560356	0.557230	0.560590	0.563330	0.621661
player9_goldEarned	0.560430	0.557590	0.558118	0.557870	0.561365	0.617099
player10_goldEarned	0.561242	0.561196	0.561624	0.563684	0.565403	0.623349

1.4. 特征与预测目标

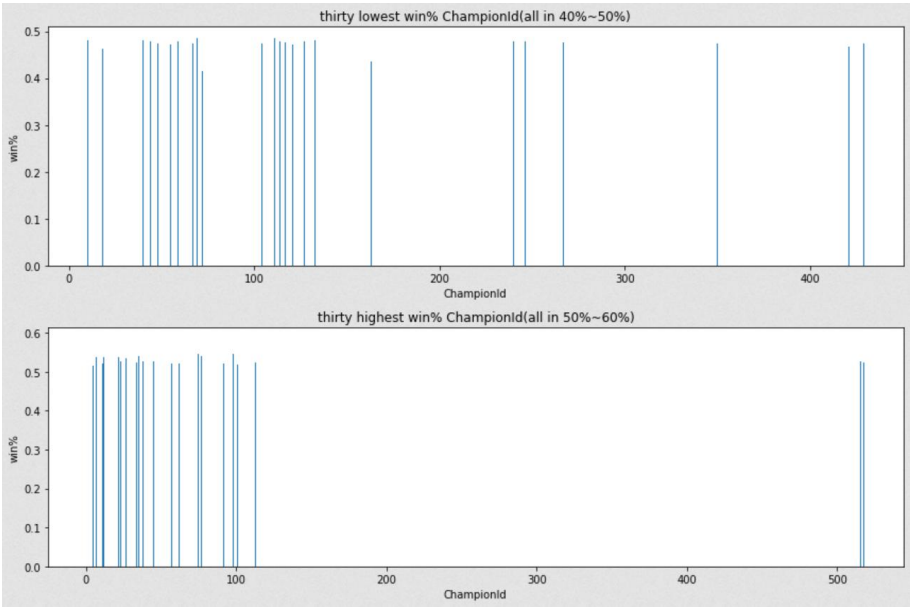
1.4.1. ChampionID 对胜率的影响

大致分布的散点图（仅 player1）

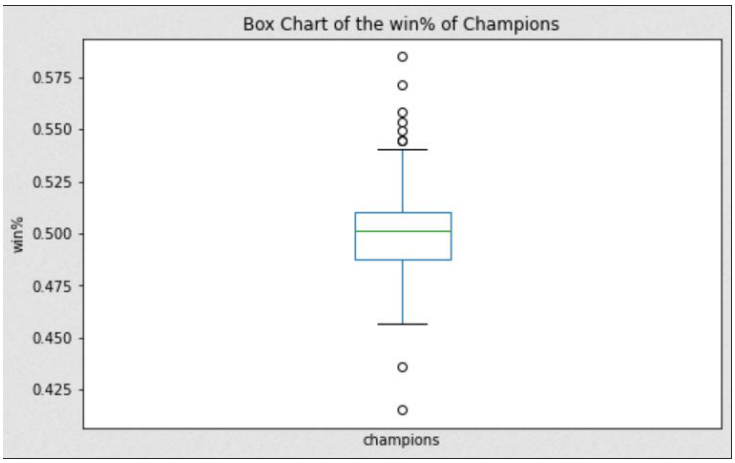


各个英雄胜率的统计图

- 直方图
  - 胜率前 30 和后 30 位的 ID 分布



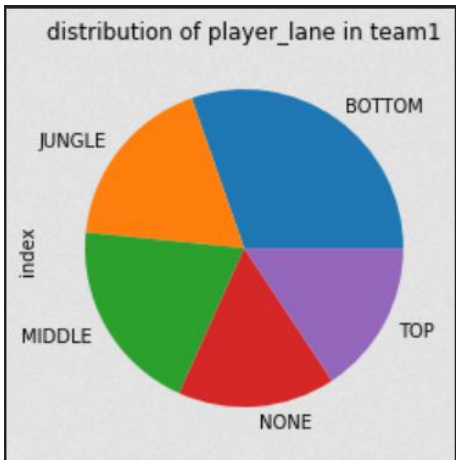
- 箱图
  - 不同英雄胜率的总体分布



1.4.2. player_lane(player_role) 对胜率的影响

先大致了解 player_lane 的分布情况

- 仅对 team1 进行了初步分析

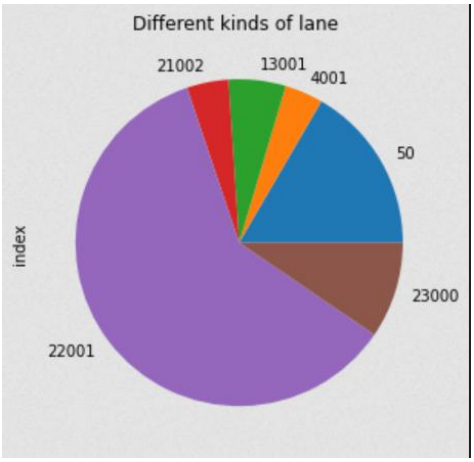


player_lane 对胜率的影响

- 采用 map 方法获取不同 lane 组合

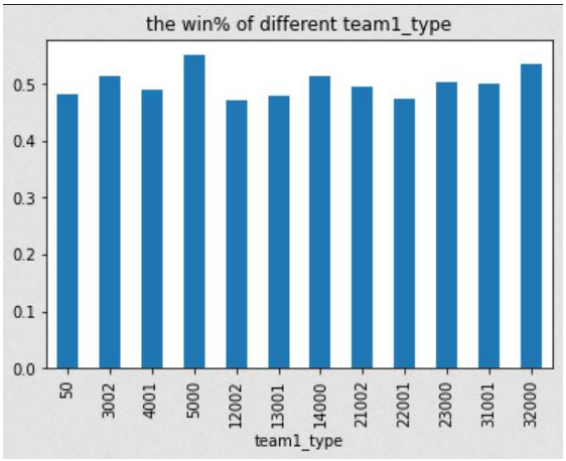
不同组合的饼图

- 其中第一位表示 BOTTOM 的数目，而后表示 JUNGLE,MIDDLE,NONE,TOP 的数目
- 0 略去不表



不同 lane 的组合方式的胜率直方图

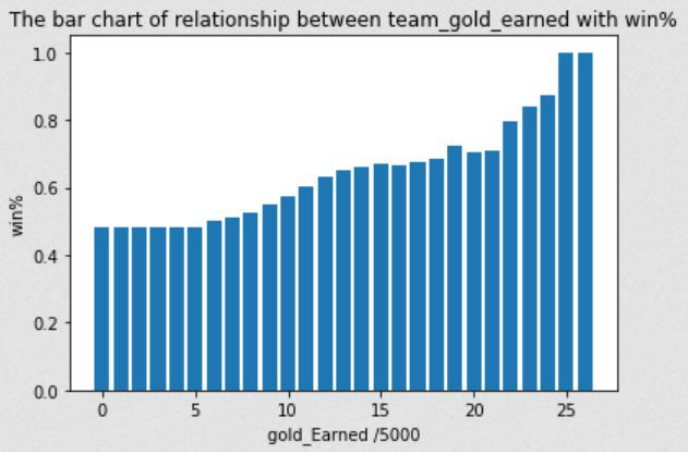
- 神奇的发现，可能五个打野的胜率高一点！



1.4.3. 考察总经济对胜率的影响

画出不同经济与胜率之间的关系图

- 看出明显的正相关关系
- 当经济大于 100000 时，几乎必赢



2. 特征提取

2.1. 特征变换

2.1.1. str -> int

将 player10 的 str 数据转换为 int 格式
获得 player10 的各项数据之间的相关系数

- int 化的 player10_role 和 player10_lane 与其余值的相关性都不大

	player10_championId	player10_kills	player10_deaths	player10_assists	player10_goldEarned	player10_role	player10_lane
player10_championId	1.000000	-0.035441	0.013209	0.031406	-0.055653	-0.093295	-0.101081
player10_kills	-0.035441	1.000000	0.123002	0.183538	0.777997	0.246381	0.262983
player10_deaths	0.013209	0.123002	1.000000	0.204596	0.291811	0.120216	0.170965
player10_assists	0.031406	0.183538	0.204596	1.000000	0.376449	0.047259	0.059902
player10_goldEarned	-0.055653	0.777997	0.291811	0.376449	1.000000	0.371361	0.412811
player10_role	-0.093295	0.246381	0.120216	0.047259	0.371361	1.000000	0.391563
player10_lane	-0.101081	0.262983	0.170965	0.059902	0.412811	0.391563	1.000000

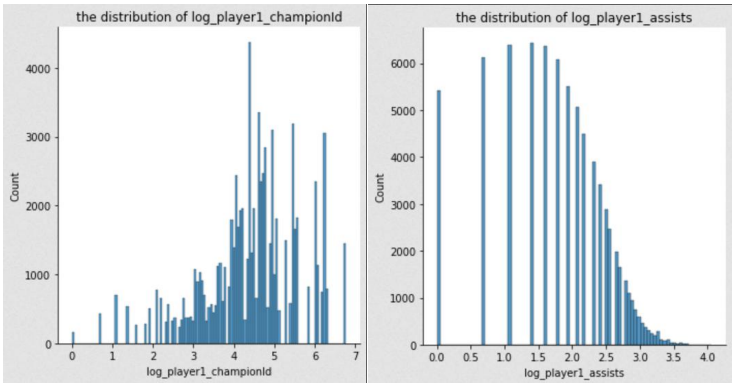
下面获取 player1~5 之间 str 值的相关性

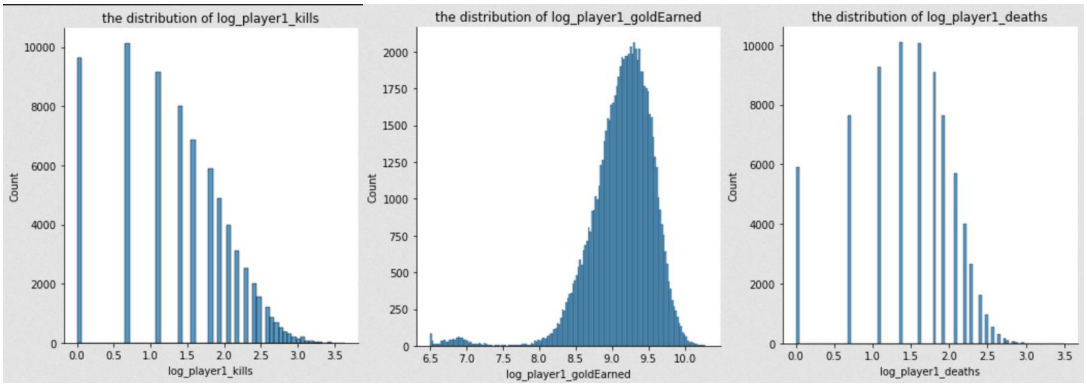
- 可以看出不同 player 间的 lane 和 role 的相关性也不高
- player1_role 和 player1_lane 间的相关性最高

	player1_role	player1_lane	player2_role	player2_lane	player3_role	player3_lane	player4_role	player4_lane	player5_role	player5_lane
player1_role	NaN	0.615789	0.179718	0.217623	0.137106	0.175612	0.138272	0.167722	0.151462	0.275260
player1_lane	0.615789	NaN	0.236930	0.467375	0.245108	0.460998	0.201481	0.444554	0.196661	0.516389
player2_role	0.179718	0.236930	NaN	0.268557	0.059390	0.101348	0.114742	0.066996	0.132915	0.227174
player2_lane	0.217623	0.467375	0.268557	NaN	0.213119	0.394880	0.138080	0.339152	0.162821	0.559824
player3_role	0.137106	0.245108	0.059390	0.213119	NaN	0.183238	0.139727	0.198872	0.053458	0.199259
player3_lane	0.175612	0.460998	0.101348	0.394880	0.183238	NaN	0.107149	0.292356	0.141423	0.523685
player4_role	0.138272	0.201481	0.114742	0.138080	0.139727	0.107149	NaN	0.657605	0.156823	0.249596
player4_lane	0.167722	0.444554	0.066996	0.339152	0.198872	0.292356	0.657605	NaN	0.144328	0.499641
player5_role	0.151462	0.196661	0.132915	0.162821	0.053458	0.141423	0.156823	0.144328	NaN	0.162148
player5_lane	0.275260	0.516389	0.227174	0.559824	0.199259	0.523685	0.249596	0.499641	0.162148	NaN

2.1.2. 取 log

以 player1 的各个数字特征为例
没有看见明显的分布趋势





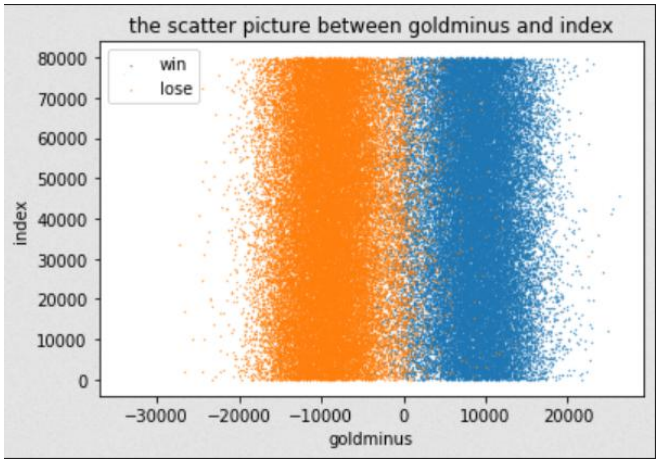
2.2. 组合特征

2.2.1. 两队的经济差

建立新的一列——经济差
画图查看大致趋势

- 可以明显看到分类聚集的现象

- 分类面大致为 goldminus=0

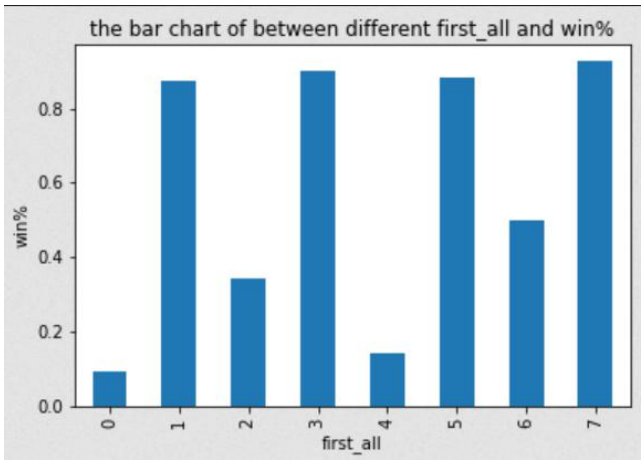


调用 knn 算法简单模拟

- 看出仅通过两队经济差来判断胜负，就有 97%的准确率！

2.2.2. 尝试组合 firstblood, firsttower, firstinhibitor

尝试组合的原因是因为这三者与 team1_win 的相关性相对最高



8 种 first_all 代表 firstblood, firsttower, firstinhibitor 的所有组合方式
是三种组合的二进制转十进制表示，如 5 代表 101 表明 firstblood 取 1，firsttower 取 0，firstinhibiter 取 1
最大值取值处分别为 001, 011, 101, 111 代表 firstinhibiter 对 team1_win 的影响最大
三者的组合特征与胜率的相关性小于单纯的 firstinhibiter 相关性

- 这个组合特征取的无效！
- sad! sad! sad!

	team1_win	team1_firstInhibitor	first_all
team1_win	1.000000	0.660634	0.438605
team1_firstInhibitor	0.660634	1.000000	0.433086
first_all	0.438605	0.433086	1.000000

2.3. 特征聚集

2.3.1. 利用 PCA 实现数据降维

- 降维对象是 team1 除了 team1_win 的所有对象

	pca	win
pca	1.000000	0.633679
win	0.633679	1.000000

降维可能比较成功。

- 尝试对 team1 的 kills,deaths,assists 进行降维

	pca_kda	win
pca_kda	1.000000	0.431259
win	0.431259	1.000000

失败了，合并后的聚集特征明显造成了相关性降低。

2.3.2. 利用 featureagglomeration 实现特征聚集

- 失败了！不用看了！

	agglo_gold	win
agglo_gold	1.00000	-0.00781
win	-0.00781	1.00000

两队的经济特征聚集到一维与胜负不相关。

可能的原因是游戏时长的随机性导致了经济特征的随机性。