PUBG Placement Analysis

Jingwen Nan, Ruoxin Shi, Yiqun (Crystal) Xu

01

Introduction

Industry Background &
Problem Statement & Data Profile

Industry Background

Nowadays, with the rapid development of technology and the economy, the video game industry is thriving and prospering. Video game, being a major entertainment approach, is pervasive and reaches all types of social backgrounds and age groups.

3.2 billion

Total number of gamers reached over 3.2 billion.

8h 27mins

The average time spent was 8 hours 27 minutes a week.

$84

US gamers annually spent an average of $84 on console purchase.

1084.1 million

eSports scene generated $1,084.1 million revenue.

Problem Statement

Goal of Analysis

- Predict the final placement for each player/team
- Classify different types of players, summarize their types and patterns, and analyze the final placements
- Analyze different game strategies and their win ratio
- Identify zombie players and cheaters

Business Value

- Game company can adjust the game setting based on the analysis to improve the game balance
- Derive the user portrait/behavior patterns from the data to help design the game and operation activities
- Identify cheaters to form a benign game environment
- Improve user experiences, user engagement and user retention

Data Profile

Our dataset contains 59 columns, 4446966 rows with no missing value.
The target of prediction is **percentile winning placement**, where 1 corresponds to 1st place, and 0 corresponds to last place in the match.

Identifier

- Player ID
- Match ID
- Group ID

Match

- Duration
- Type
- # Participants
- Worst placement

Teams

- #Teammate revive
- # assists

Item

- Weapons Acquired
- #Boosts
- #heals

Kills

- Headshot kills
- Rank based on # kills
- Kill point
- Kill streak
- # Kills
- Road kills
- Team kills
- Longest kills
- # Vehicle Destroyed
- # knocked
- Total damage dealt

Distance

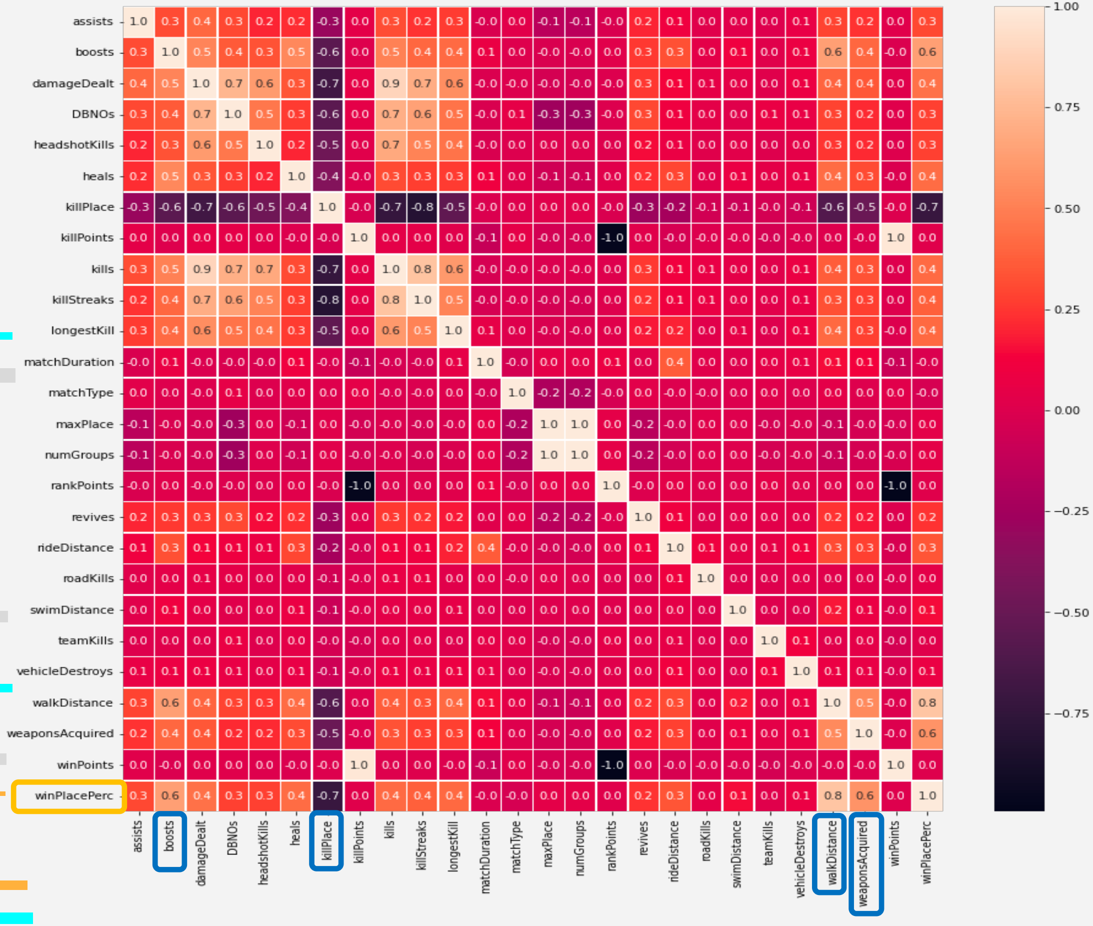
- Ride Distance
- Swim Distance
- Walk Distance

02

Exploratory Data Analysis

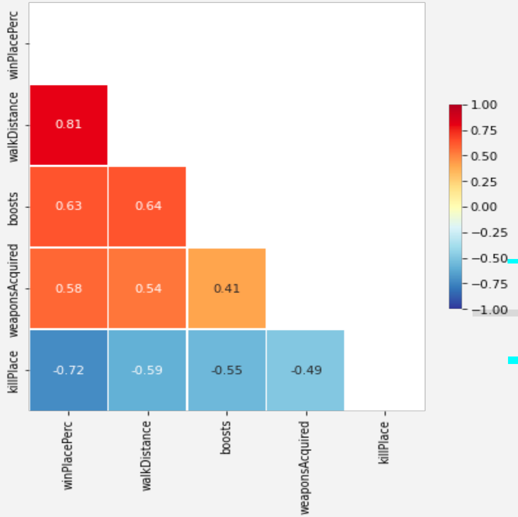
Explore Feature Distribution &
Correlation & Clustering

Correlations Exploration

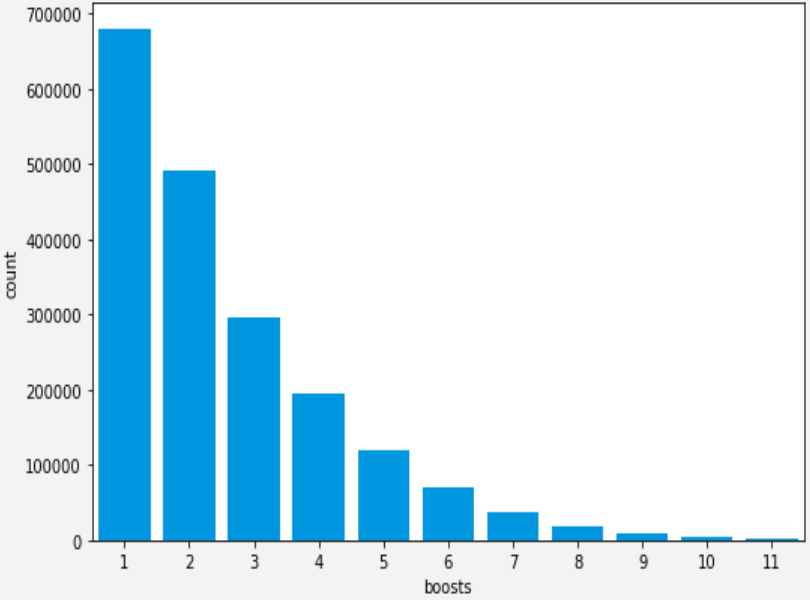


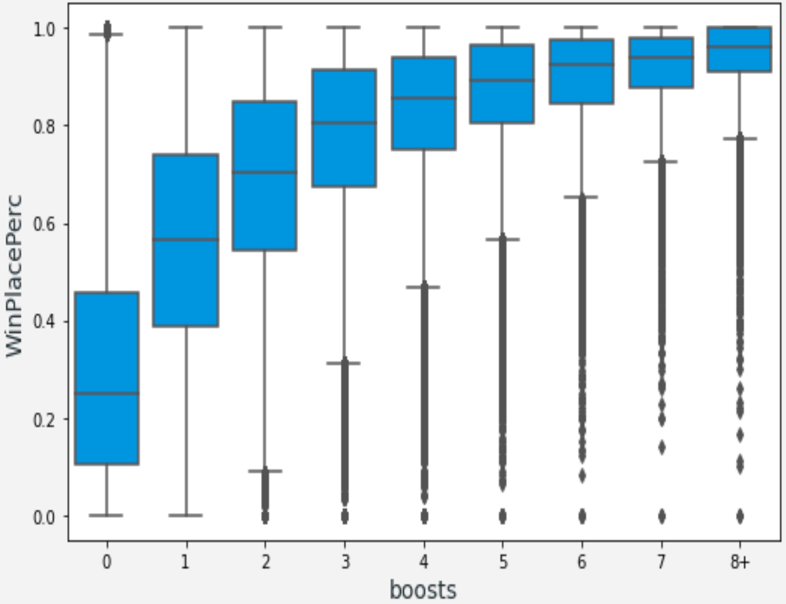
The target variable is highly correlated with these features:

- Boosts
- KillPlace
- WalkDistance
- WeaponAcquired



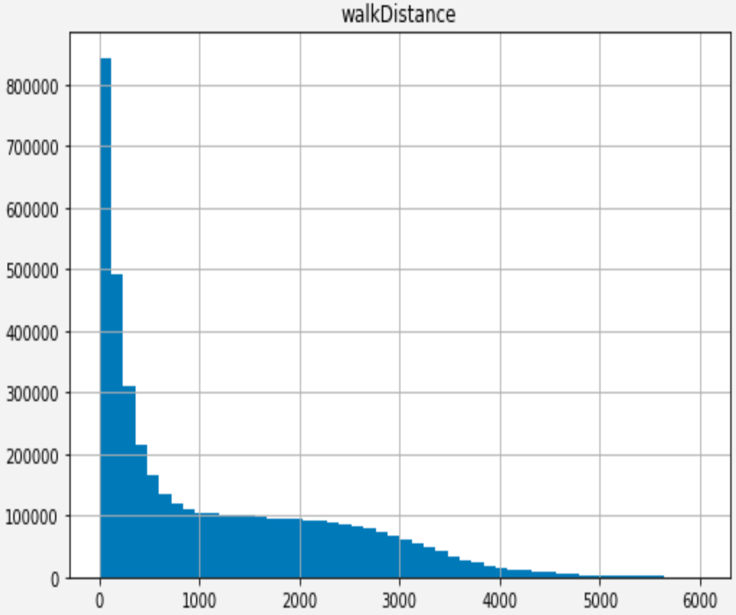
Boosts Variable Exploration

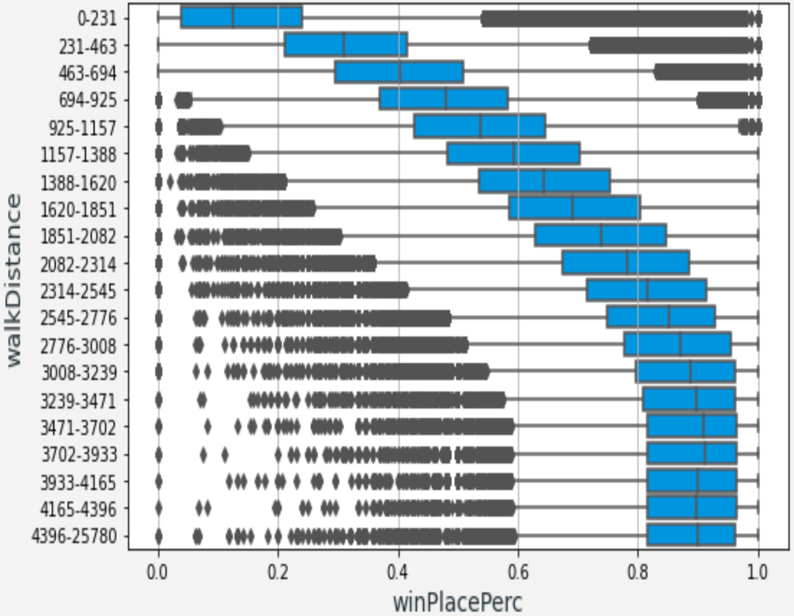




- Number of boosts is in the range of 1 to 11
- Players with 1 boosts are most common while player with 11 boosts are the rarest
- Boosts have a positive impact on percentile winning placement.

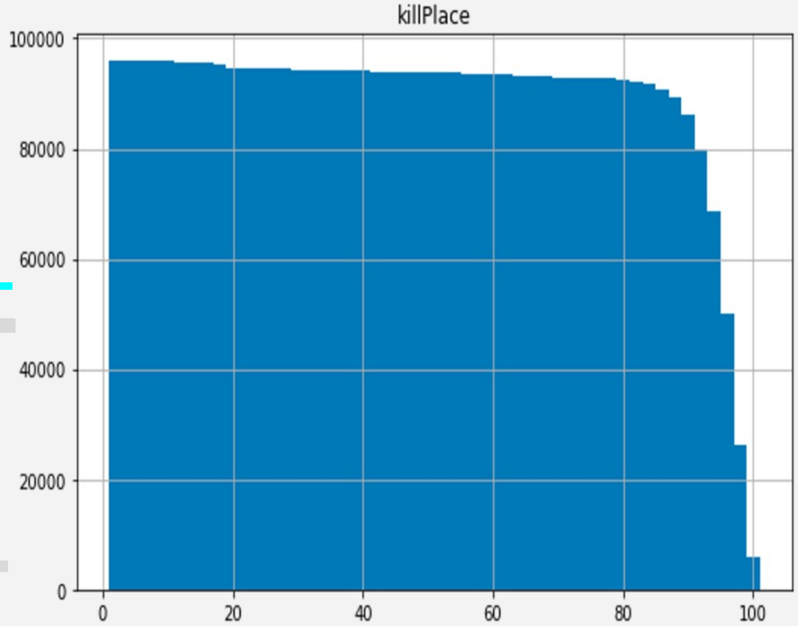
WalkDistance Variable Exploration

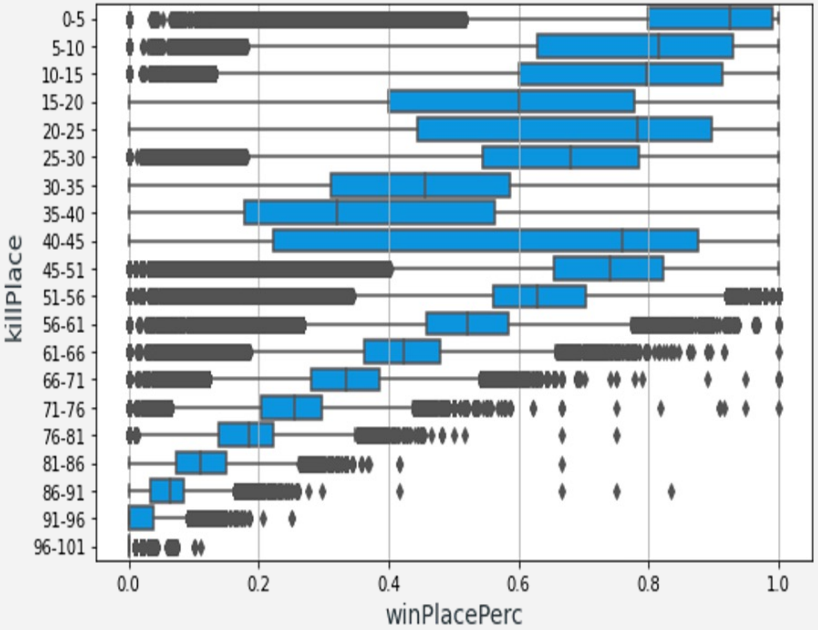




- The plot drops sharply when walk distance<1000 and the trend is flattened when walk distance>= 1000
- When walk distance<3000, the variable has a great positive impact on percentile winning placement.
- When number of weapons>=3000, the influence is limited.

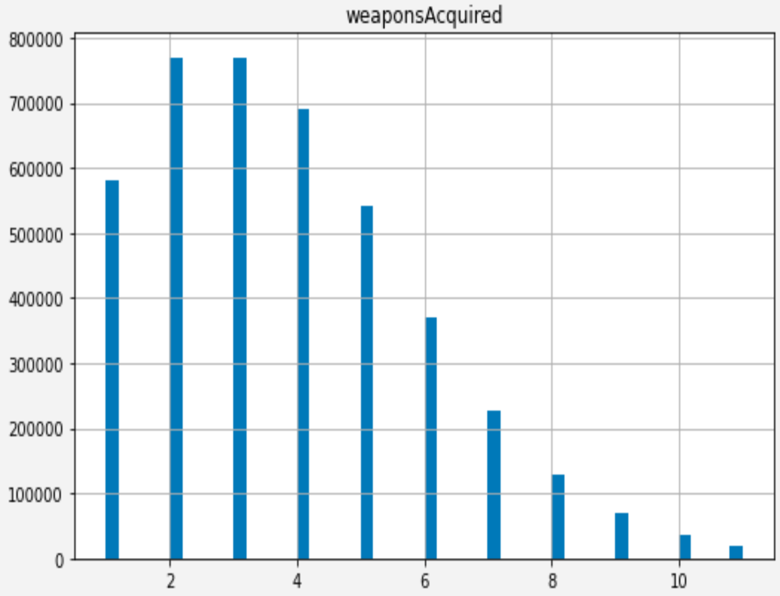
KillPlace Variable Exploration

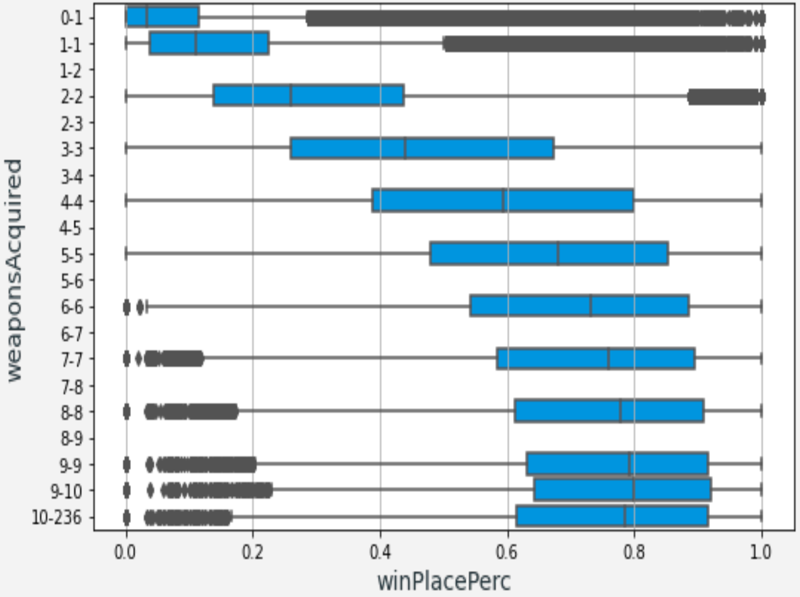




- Only few players' ranking based on kills in the dataset are in the range of 90 to 100.
- The influence of variable on response is not direct when killPlace<50.
- When killPlace>50, the variable has a negative effect on percentile winning placement.

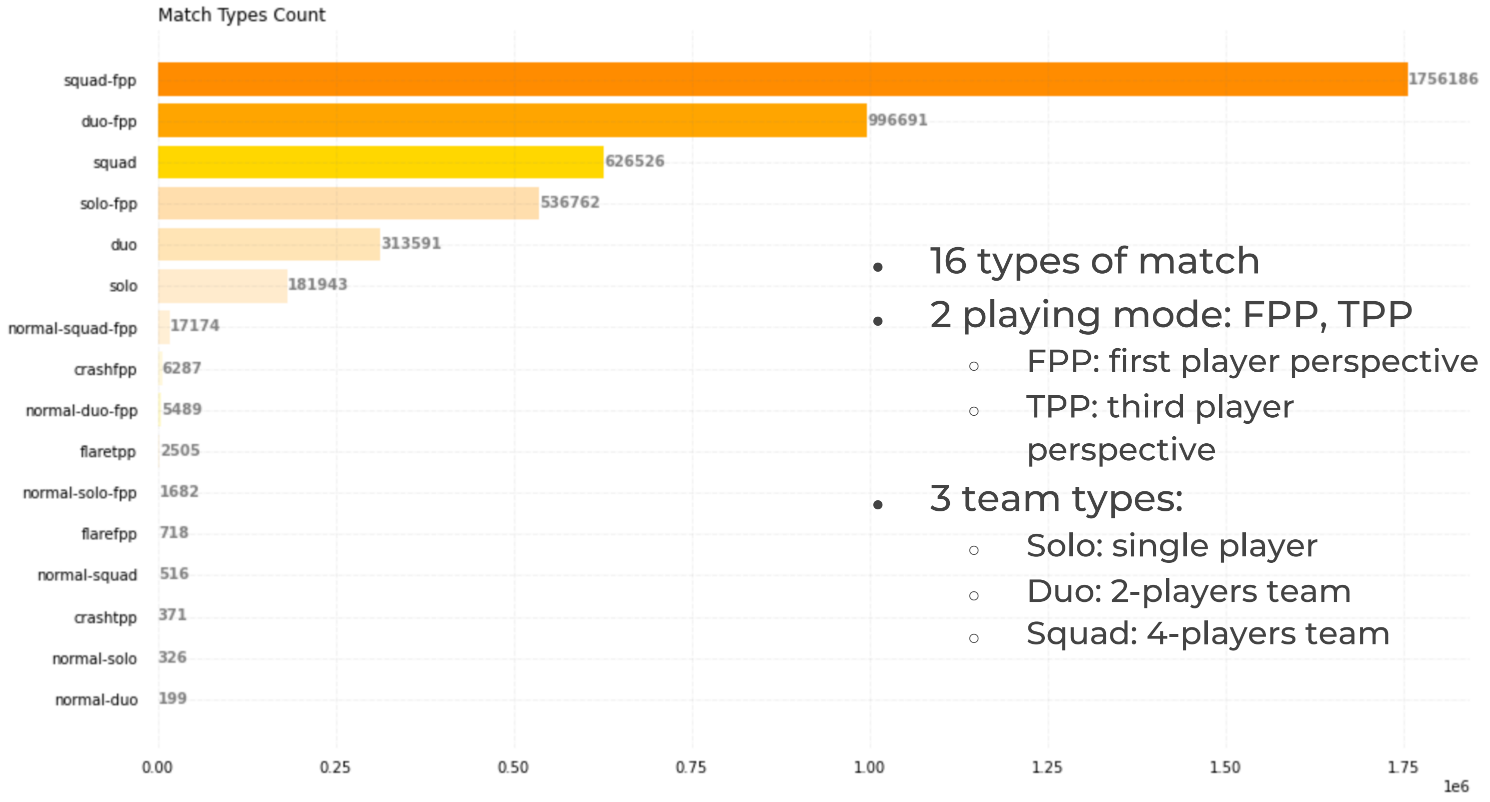
WeaponsAcquired variable Exploration





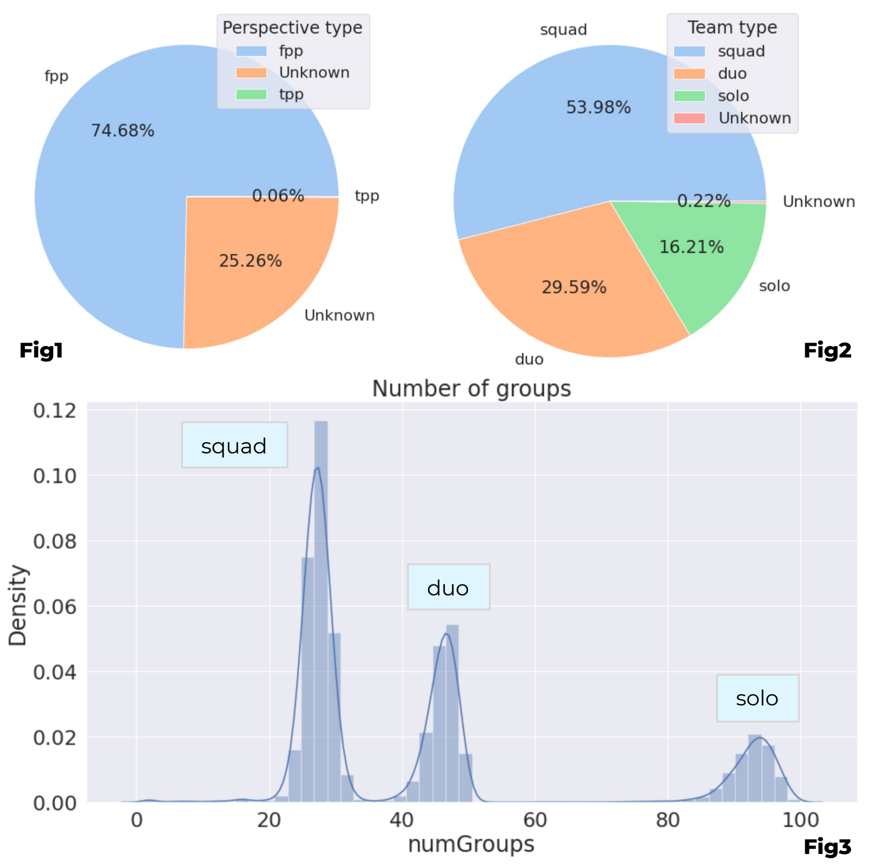
- Most players acquire 2-4 weapons.
- When number of weapons<6, the variable has a great positive impact on percentile winning placement.
- When number of weapons>=6, the variable has the limited effect on percentile winning placement

Match Type Exploration



- 16 types of match
- 2 playing mode: FPP, TPP
  - FPP: first player perspective
  - TPP: third player perspective
- 3 team types:
  - Solo: single player
  - Duo: 2-players team
  - Squad: 4-players team

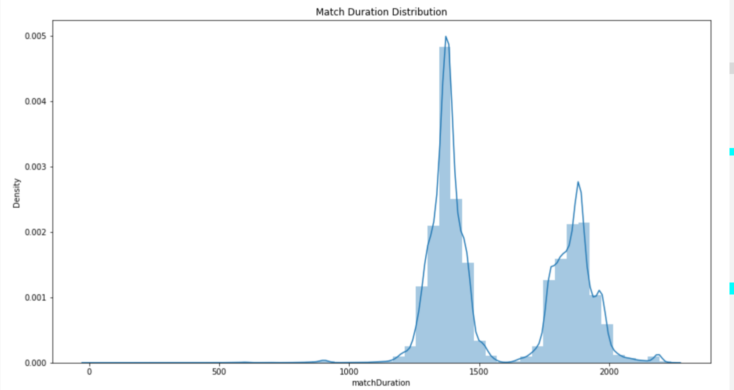
Match Type Exploration

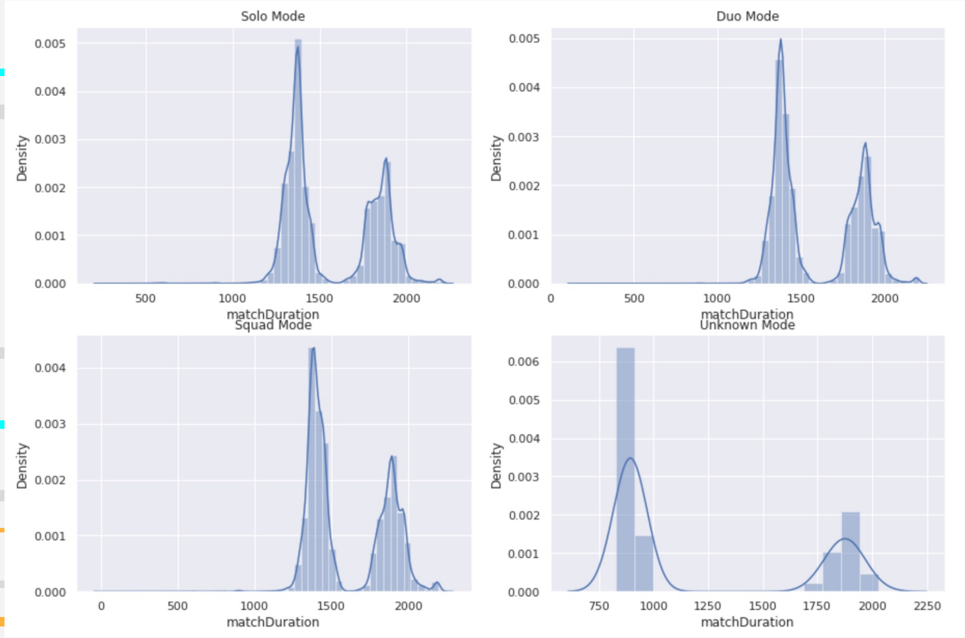


- **Fig1** - Based on playing perspective
- **Fig2** - Based on # of players in team: solo, duo, squad
- **Fig3** - Matches the number of groups distribution (3 peaks correspond to 3 match types)

Match Duration Exploration

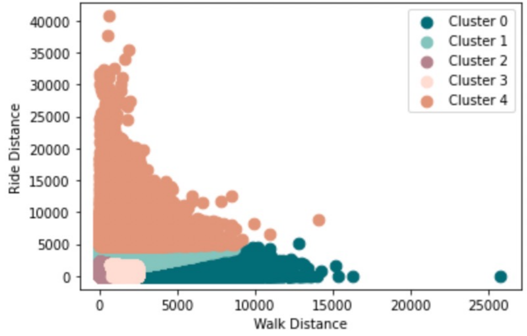
2 peaks exist in the overall distribution: around 1300 secs and 1800 secs.

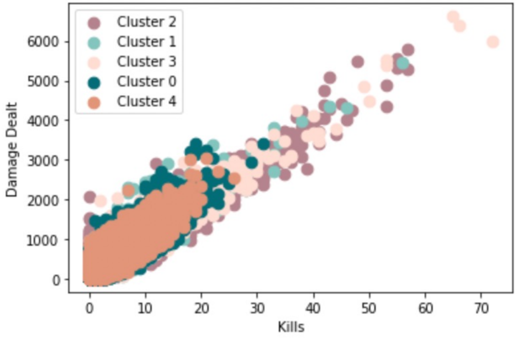


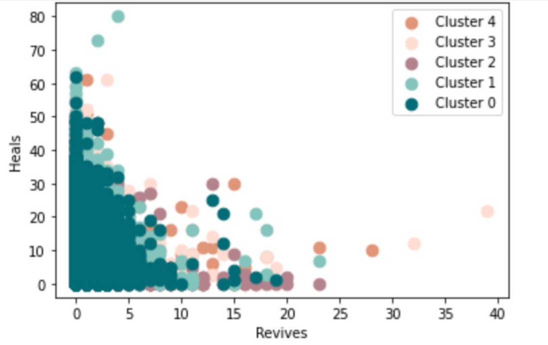


- Similar peaks exist in the distributions for different team types, meaning that the match duration might not be influenced by team size.
- Possible cause for these peaks is the ***map size***

Player Cluster







Cluster	kills	damageDealt	walkDistance	rideDistance	vehicleDestroys
0	2.032864	253.323024	3251.052267	304.006898	0.012707
1	1.367797	189.597173	2059.703222	3048.744268	0.031434
2	0.477928	78.272302	263.586637	29.958470	0.000972
3	1.151374	156.963793	1671.314409	236.309895	0.006793
4	1.382194	195.810986	2036.421858	6527.038467	0.044403

Cluster	swimDistance	heals	boosts	assists	DBNOs	revives	winPlacePerc
0	13.489243	2.989113	2.967558	0.558847	1.261451	0.348829	0.853376
1	7.999683	3.095863	2.334343	0.383871	0.939999	0.288382	0.712025
2	0.601315	0.405545	0.224557	0.106526	0.413604	0.074322	0.253975
3	7.183004	1.747592	1.463233	0.272267	0.752860	0.211265	0.663015
4	8.094267	3.841192	2.823281	0.419283	0.979767	0.309779	0.773293

- **Cluster 0 -** High kills, Prefer walking, Mid moving distance, Save and assist most teammates
- **Cluster 1 -** Mid kills, Prefer Driving, Long moving distance, Use most healing and boosting items
- **Cluster 2 -** low kills, Short moving distance, Die quickly
- **Cluster 3 -** Mid kills, Prefer walking, Mid moving distance, Use less items, lower damage made comparing to cluster 1
- **Cluster 4 – Mid kills, Driver, Super long moving distance, Using more healing items**

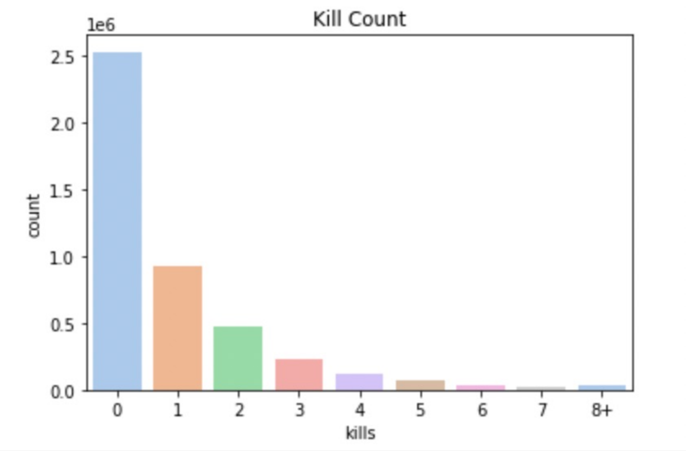
Player Types

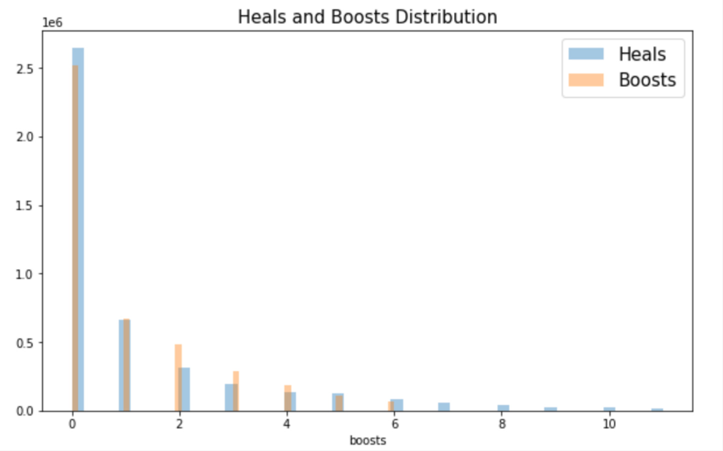
Killer

- The average kills is 0.93
- 99% of people have 7.0 kills or less
- The most kills ever recorded is 60

Healer

- Person uses average 1.2 heal items
- 99% of people use 11.0 or less
- The most used is 59
- Average person uses 1 boost items
- 99% of people use 7.0 or less
- The most used is 18.





Player Types

Walker

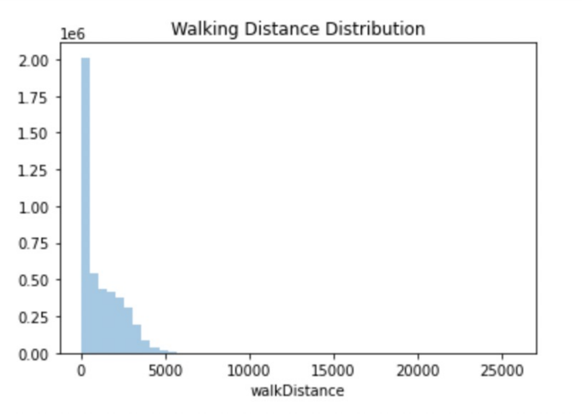
- The average person walks is 1055m
- 99% of people walked less than 4138m
- The longest walking distance is 17300
- 2% players walked 0 meters, who might be killed moving

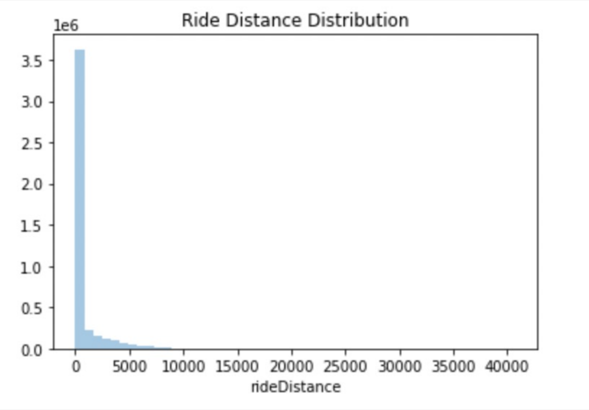
Driver

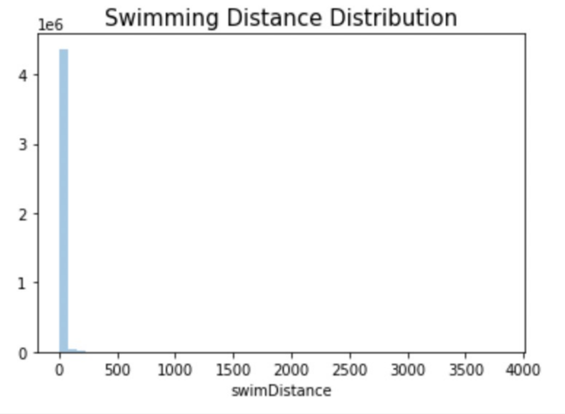
- The average person drives 423.9m
- 99% of people drive 6133m or less
- The longest driving distance is 48390m.

Swimmer

- The average person swims for 4.1m
- 99% of people swim 116m or less
- The longest swimming distance is 5286m.







03

Feature Engineering

Remove Outliers & Create
new features

Remove Outliers

- Kills: --> possible CHEATERS
    - Kills_without_moving: calculate the total distance for each player (adding up the walkDistance, swimDistance and rideDistance), remove the records with '0' total distance and > 1 kills.
    - Kills: remove records with kills greater than 35
    - Longest_kills: remove records with longest killing distance > 1000

- Travelling:

Remove records anomalous with walking, swimming and riding distance as well as the total distance separately.

- Weapons:

Remove records with anomalous total number of weapons acquired. (For example, a player is highly possibly a cheater if acquiring more than 40 weapons in one single game.)

- Heals:

Remove records with anomalous total number of healing items used. (For example, a player is highly possibly a cheater if using more than 40 healing items in one single game.)

Create New Features

- Standardize the match type into 4 main categories: **Solo, Duo, Squad, Other**
- Combine the number of healing and boosting items. -> **health_items**
- Calculate the headshot ratio
- -> **headhshot_perc**
- Add the elements that indicate teamwork(assists & revives) -> **team_work**
- calculates the average length of a kill streak
- -> **killStreak_len**
- Calculate the total hits (DBNOs + kills + teamKills)
- -> **totalHits**

Data Preprocessing

- Use function defined in https://www.kaggle.com/gemartin/load-data-reduce-memory-usage to reduce data memory.
- Select the numerical columns and categorical columns.
  - Perform standard scaling on numerical columns
  - Perform OneHotEncoder on categorical columns
- Train-test split the data (80-20)

04

Methodology

Models & Model Selection & Deployment

Models

01

Linear Regression

The baseline model

02

LightGBM

A Great algorithm to deal with large amount of data with less memory

03

XGBoost

A useful way to explore features importance

04

Neural Network

A Series of algorithms endeavors to recognize underlying relationships in a set of data

Neural Network

Model 1

Sequential 3-layer

Test loss: 0.0064

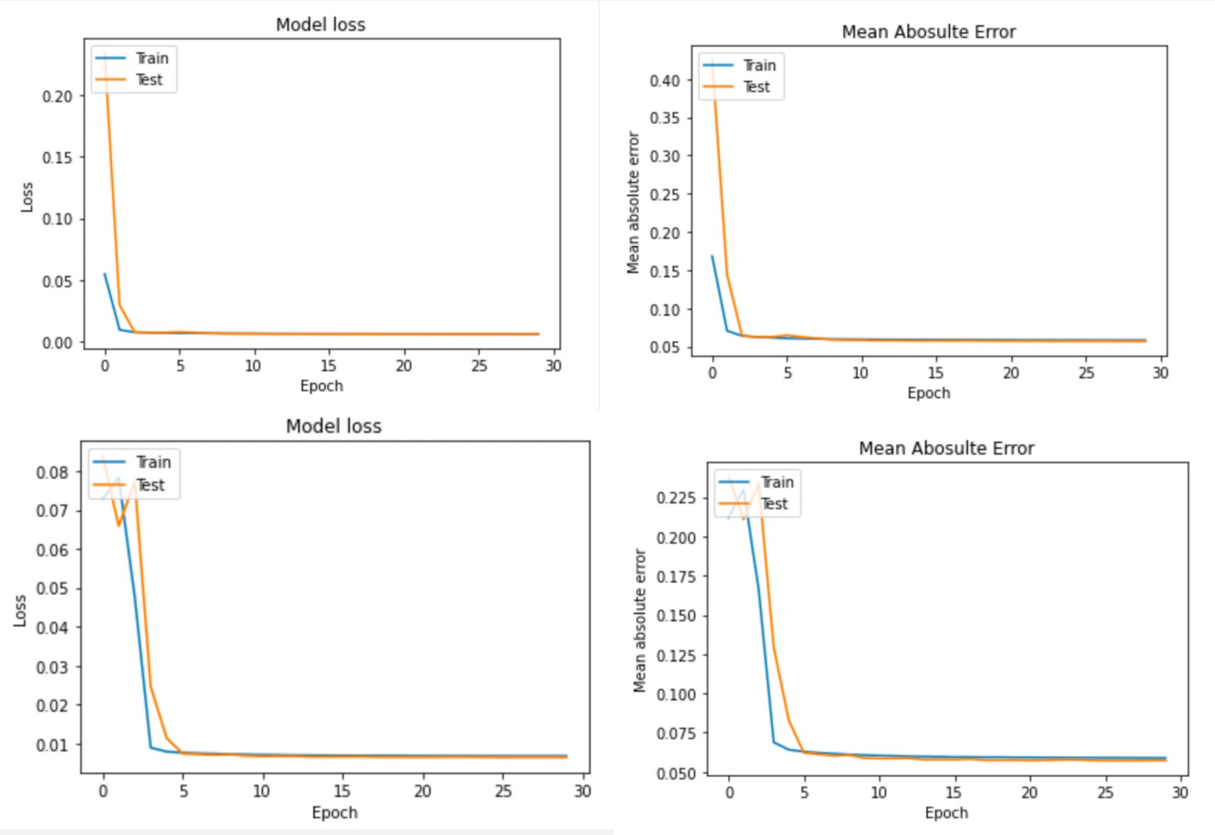
Test MAE: 0.0570

Model 2

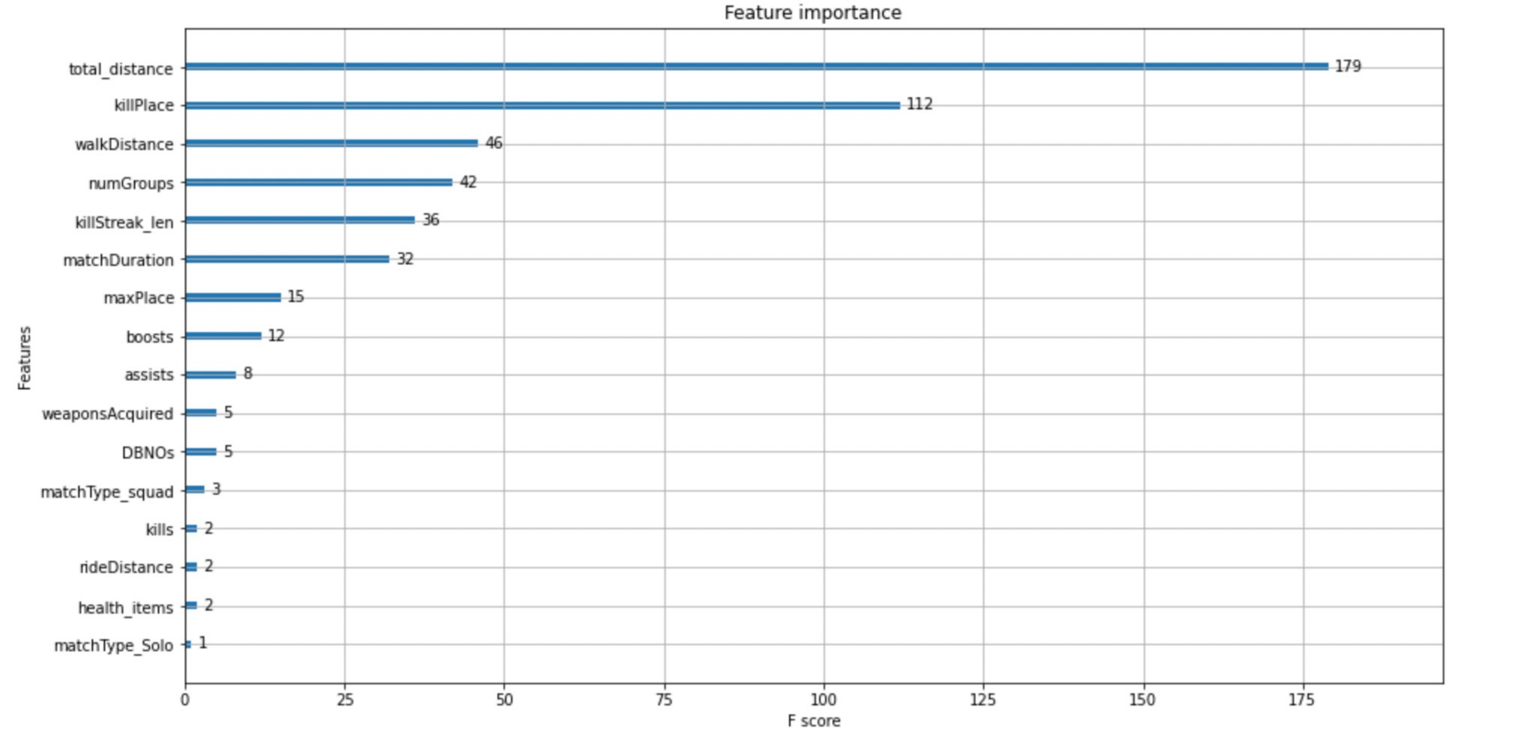
Sequential 4-layer

Test loss: 0.0064

Test MAE: 0.0573



Features Importance result from XGBoost



Linear Regression

- Train MAE: 0.08645
- Test MAE: 0.08662
- Train RMSE: 0.11866
- Test RMSE: 0.11896

Neural Network

- Test loss: 0.0064
- Train loss: 0.0063
- Test MAE: 0.0570
- Train MAE: 0.0568

XGboost

- Train MAE: 0.07489
- Test MAE: 0.07492
- Train RMSE: 0.10764
- Test RMSE: 0.10767

Light gbm

- Train MAE: 0.05944
- Test MAE: 0.05958
- Train RMSE: 0.08263
- Test RMSE: 0.082874

05

Conclusion

Findings & Conclusion &
Next Steps

Findings & Conclusion

- The match type (team size) does not have great correlation to the match duration. A more important factor might be the map size, which is not mentioned in our data.
- The ranking of # of enemies killed, the total traveling distance and the number of weapons acquired are the features of top importance to the target variable – winPlacePerc
- There are many cheaters and robot players in the records (unreasonable kills, travelling distance, etc)
- Currently, based on the MAE errors, we found that the **neural network** model has the best performance. In the future, after trying different hyperparameters and fine-tuning XGBoost and LightGBM, we might get better results.

Lessons Learned & Recommendations

Lessons Learned

- Learned how to reduce the size of memory while keeping the data in same, which help us when dealing with such a huge dataset.
- Eliminate the outlier before fitting models. In our dataset, there are a lot of robot players and cheaters. How to identify and eliminate them are important for making unbiased prediction.

Next Steps

- Fine tune Xgboost and LightGBM
- Try neural network with more layers and other structures
- Deployment plan for big data platforms
- Add more team data and focus on team performance

THANKS!