PUBG: A Guide to Free Chicken Dinner

Yang Li(yangli97@stanford.edu), Xin Lu(luxin@stanford.edu), Wenxin Wei(wxwei@stanford.edu)
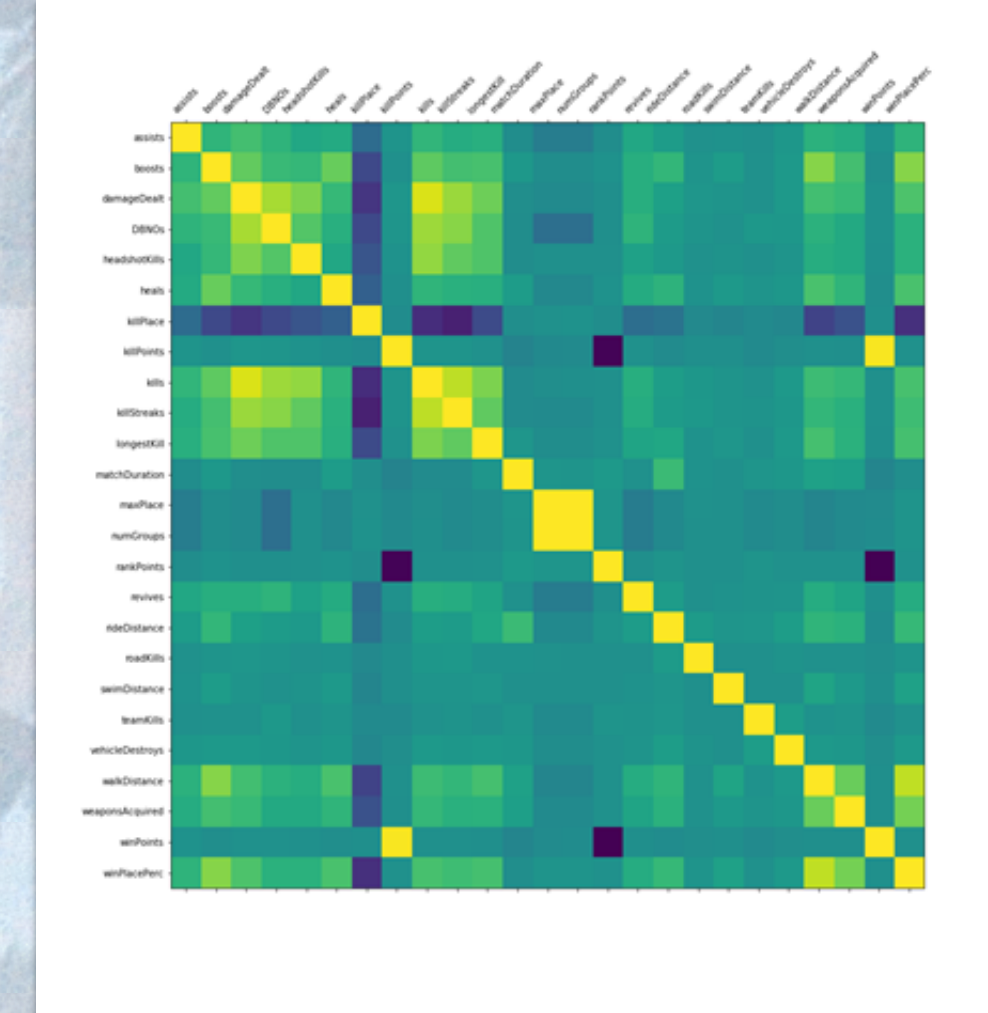
Advised by Professor Andrew Ng

Introduction

PUBG(Playerunknown's Battlegrounds) is a multiplayer online battle arena video game that has become popular in the past year. The game features the last man standing deathmatch mode, in which one hundred players fight against each other in a battle royale. Given players' statistics within a game, we build ridge regression, lasso regression, lightGBM model, and random forest model to predict their final ranks. After predicting final ranks, we perform an additional step to classify game strategies used by top players.

Features

We applied a few feature engineer methods to process the data: 1) Added group-statistic data, e.g, mean/sum/max/min of each feature, and the corresponding ranking, to each group as new features, since the final score is same for all members in each group. 2) Normalized feature values based on the amount of players joined in match, and also normalize by mean and variance. 3) Converted feature 'matchType' (16 different match types in total) from a numeric value to a one-hot vector. 4) To better capture information hidden behind group of features, we added combined features, e.g.,'totalDistance' = 'rideDistance' + 'walkDistance' + 'swimDistance' 5) Calculated feature importance and kept only important features:
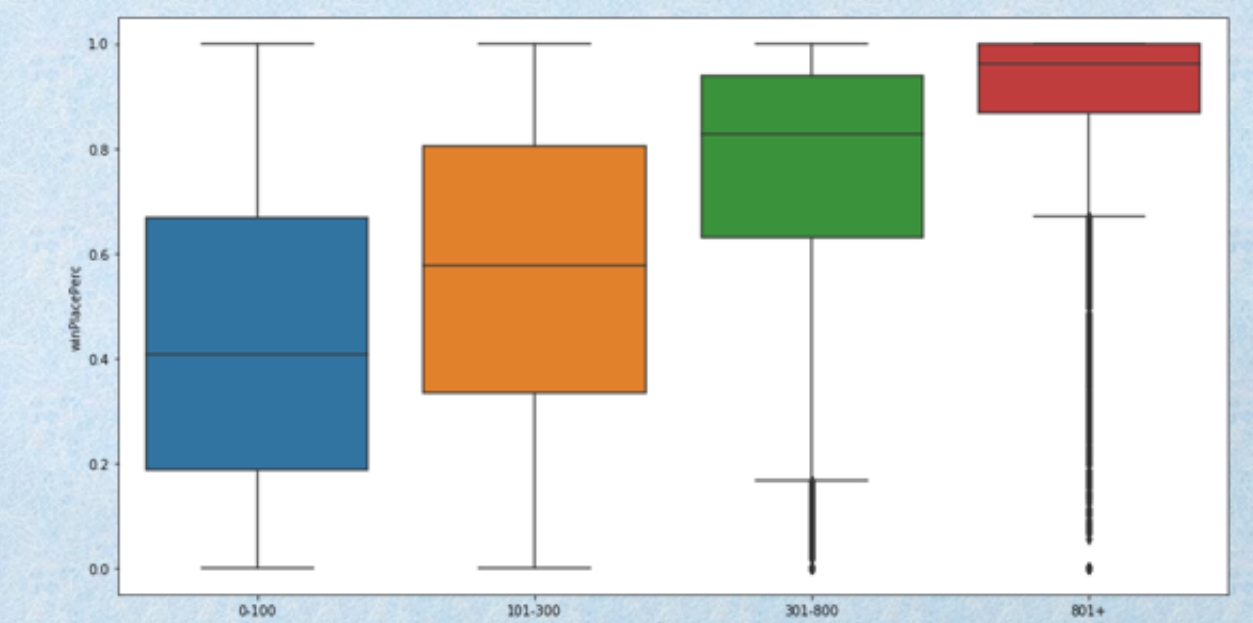

Data

Our data comes from Kaggle competition, which contains anonymized PUBG game stats, formatted so that each row contains one player's post-game stats. The columns are features that influences the players' finishing placement, such as number of enemy players killed, total distance traveled on foot/in vehicle/by swimming, number of weapons picked up, etc. We perform fully-covered data cleaning, which excludes NaNs, outliers, and cheaters among the players.

Models

Regression models: we tried linear regression (SGD regressor), ridge regression, lasso regression and elasticnet regression. Ridge regression has the best out-of-box result with the help of aforementioned feature engineering, likely because it handles multicollinearity data better.

Tree models: we tried random forest and light-GBM (light gradient boosting machine). Light-GBM results in quite good MAE score, which shows significant improvement on linear regression models.

Result

In terms of predicting final ranks, we have achieved minimum MAE (Mean Absolute Error) of 0.0204, and ranks #57 on Kaggle leaderboard as of Dec 10.

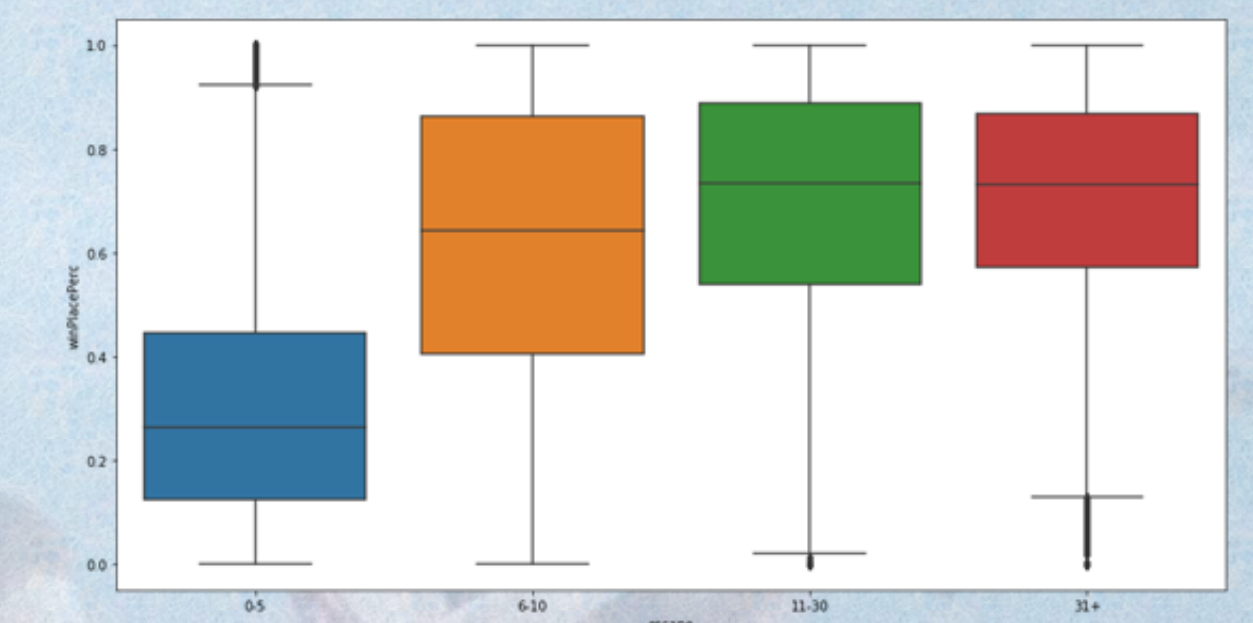| MAE on 20% validation set	Linear Regression	Ridge Regression	Light-GBM
Raw features	0.09000	0.08989	0.05654
Raw + mean	0.05736	0.05736	0.04158
Raw + mean+sum+max-min	0.04845	0.04845	0.02896
Everything above +match_mean+size	0.04825	0.04825	0.02755
Everything above + cross validation	0.04812	0.04810	**0.0204**

Discussion

Thanks to the effective feature engineering and appropriate model selection, we got some decent results in predicting final ranks. Moreover, we have observed some interesting gaming strategies which we will give out to PUBG players as Free Chicken Dinner!
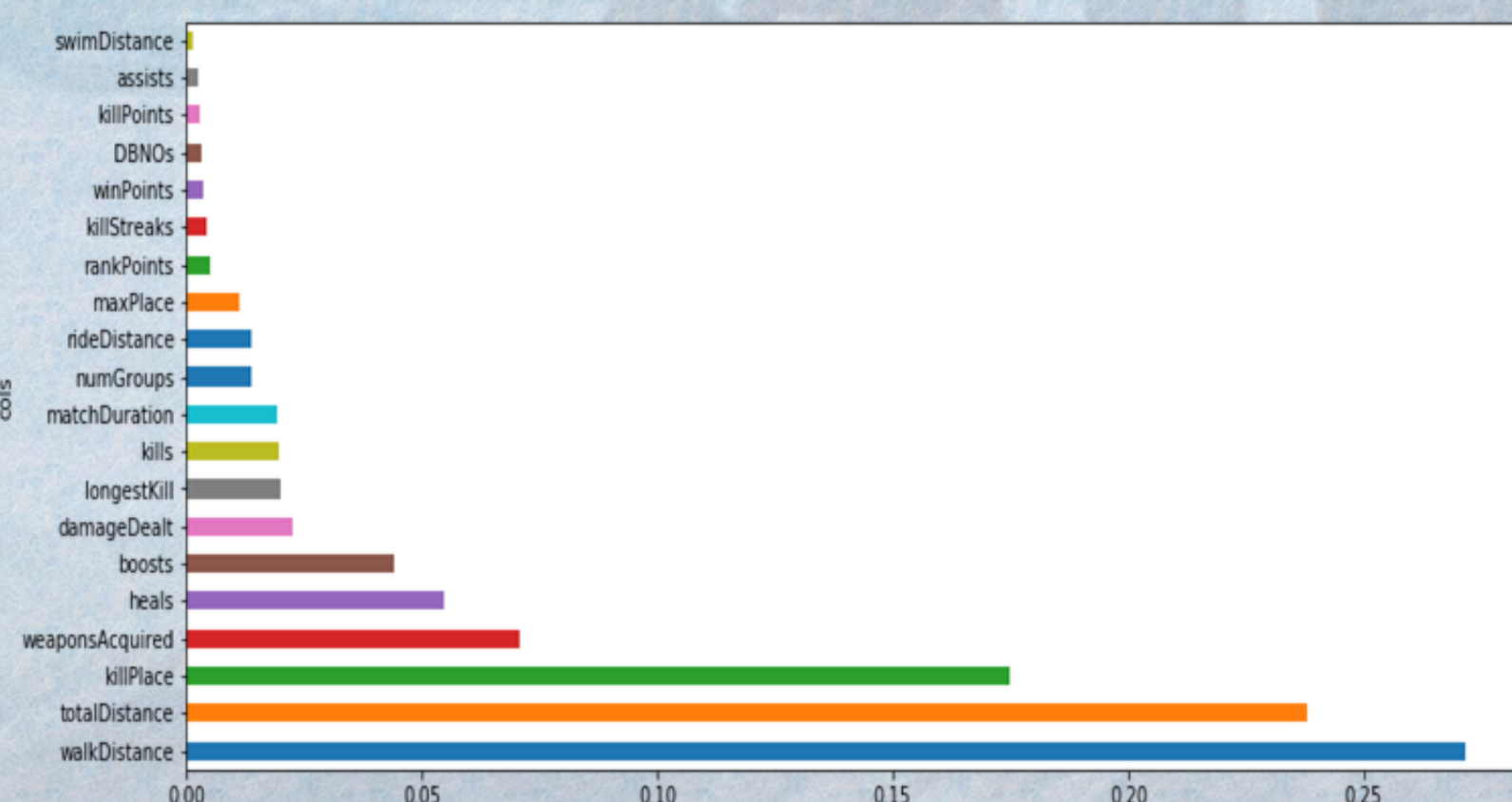
Game prefer fighters

Don't stay in one place

Future

We will train the AI to play PUBG and win. With the results we have arrived at, we will train the bots to maximize the exponentially decayed sum of future rewards with data produced entirely from self-play, that is, the bots are playing against themselves to "learn" to play by the strategies that maximize their chances of winning. By building AI's that succeed in complex and dynamic games like PUBG, we believe it can bring humanity closer to the ultimate goal of making AGI's that function in the messiness of the real world.