Group Imposters
Xu Han `xh852`, Jason Wang `jw7383`, Chloe Zheng `cz1300`

Data Analysis and Prediction on Among Us Dataset

Introduction

Among Us is the most popular online multiplayer social deduction game in the world. In September 2020 alone, the game was downloaded over 100 million times with a peak of 3.8 million concurrent players. The rules of Among Us are simple. Each player is designated with a hidden role of either "crewmate" or "imposter." If crewmates want to win the game, they must work together to vote all imposters (usually 2) out of the game or complete all their tasks. For the imposters to win, they must avoid getting voted out and kill all crewmates before the crewmates finish their tasks.

Among Us's popularity and our own interest in the game motivated us to ask and answer the following questions: which factors influence the win rate for crewmates? For imposters? Can we use players' performance when they are crewmates to predict the outcome when they are imposters?

The original dataset comes from Kaggle https://www.kaggle.com/ruchi798/among-us-dataset. The dataset contains 29 .csv files where each file consists of a unique user's data. Each file has 13 columns and X rows such that X is the number of games the user played, so each row consists of the user's stats from one game – X ranges from [33,100]. The following is the dictionary for the dataset (column features):

1. `Game Completed Date`: the date and time a game was completed
2. `Team`: role designated, imposter or crewmate
3. `Outcome`: game result, win or lose
4. `Task Completed`: the number of tasks completed
5. `All Tasks Completed`: whether all tasks were completed, yes or no
6. `Murdered`: if the player was murdered, yes or no
7. `Imposter Kills`: the number of players the user killed if they were an imposter
8. `Game Length`: the duration of one game, in minutes and seconds
9. `Ejected`: whether the player was ejected, yes or no
10. `Sabotages Fixed`: the number of sabotages fixed
11. `Time to complete all tasks`: time used to complete all tasks, in minutes and seconds
12. `Rank Change`: rank change after each game
13. `Region/Game Code`: the location and game code of user

To prepare the dataset, we combined all user data into one Dataframe. This consisted of 2227 rows, and 14 columns after adding the following feature:

14. `User`: user id, ranges from [0, 28] for 29 unique users

We also converted all string values from columns 3-13 into numerical values if necessary. For instance, win/loss and yes/no were replaced with 1 or 0, and minutes/seconds were converted to total seconds. Refer to `Among_Us_Prediction.ipynb` for the data preprocessing and data engineering code.

In the next three sections, we will clarify our questions more specifically and explain how we used different methods, including hypothesis testing, logistic regression and linear regression, to approach each question.

Group Imposters
Xu Han `xh852`, Jason Wang `jw7383`, Chloe Zheng `cz1300`

Hypothesis Testing

We wanted to first delve into analysis that could provide some strategic insight into the game. There are two main objectives a crewmate can focus on to win the game - either completing their tasks or finding and voting out imposters. Therefore, the first question we wanted to answer was whether completing more tasks as a crewmate increases a player's chance of winning, since that is the conventional knowledge which we were interested in assessing.

The method we used to answer this question was via a hypothesis test, specifically a one-sided independent samples t-test. This test assumes that the data was sampled independently from two populations, so to control for the fact that multiple observations existed for a single user, we averaged each user's number of tasks completed so we could classify users as either players who complete less tasks on average or players who complete more tasks on average. The threshold we used was the average number of tasks completed over all users, which was approximately 5.22 tasks. Our test is one-sided because we are only trying to answer whether completing more tasks increased the player's chance of winning. Finally, because we represented wins and losses as Bernoulli random variables where 1 represents a win and 0 represents a loss, the mean of each user's outcomes simply represents the proportion of wins to the total number of games, which we call win rate.

Our null hypothesis was that there was no difference between the win rate of users who complete less tasks on average and users who complete more tasks on average. Our independent samples t-test yielded a t-statistic of -1.679 and a p-value of 0.052. With a significance level of 0.05, we fail to reject the null hypothesis and cannot conclude that completing more tasks increases win rate. In **Fig 1**, we plot the t-distribution represented by the red line. Note that our t-statistic, represented by the black dotted line, yields p-value, represented by the blue area under the curve, that is not less than our chosen significance level.

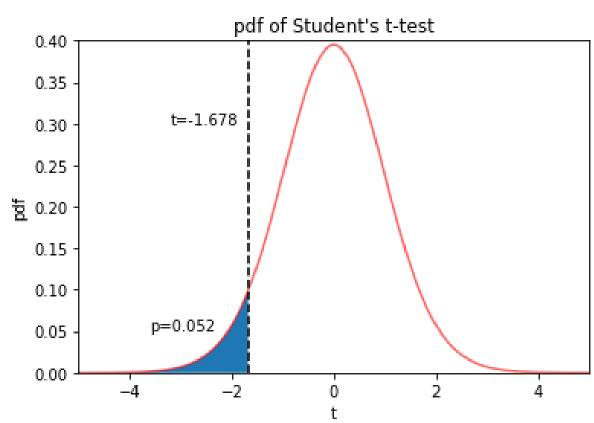

Fig 1. Plot of the t-distribution and our t-statistic and corresponding p-value.

Although we failed to reject the null hypothesis, the result is still interesting as we could not confirm what is considered to be conventional strategy in Among Us, which is to complete more tasks as a crewmate to win. Though we require more data to test, our results could be implying that a better strategy for crewmates is to focus more on finding and voting out imposters than completing their tasks, but unfortunately without more data and additional tests, there is nothing we can currently infer about the effect of completed tasks.

Group Imposters
Xu Han `xh852`, Jason Wang `jw7383`, Chloe Zheng `cz1300`

Logistic Regression

Given the results of our previous hypothesis, a crewmate's ability to finish more tasks might not be a significant predictor of winning or losing. This leads to the question, out of all the features provided, which are the best predictors for a win or loss? What is the best strategy for a crewmate?

Since this question is only regarding the crewmates, we extracted the crewmate data and used the "Outcome" column as our expected output (y) and the following columns as our input data (X):

1. `Task completed`
2. `All tasks completed`
3. `Murdered`
4. `Game length`
5. `Ejected`
6. `Sabotages fixed`

We implemented a 75/25 split to create a training and testing dataset. We also ensure that for each unique user, their data is only in the training or testing dataset. This way when our model is cross-validated, it will get a test set with new and unique users.

Our model is a binary classifier, so we calculated the AUC score to evaluate the model. Our model's resulting AUC score is .61. This indicates that overall, our features are poor predictors for a win or loss, although our model still predicts the outcome better than chance. Based on **Fig 2**, we can verify that the Tasks Completed, along with Game Length, Sabotages Fixed, and Region are not good predictors of outcome. However, whether a crewmate is Murdered or Ejected are more reasonable predictors for game outcome. **Fig 2** also indicates that being ejected is a better predictor than being murdered. Intuitively, this makes sense because if a crewmate is murdered, they can still help the other crewmates complete tasks and win the game. On the other hand, if they are ejected, then the crewmates lose a team member, they will have to re-evaluate who they think is the imposter, and the imposter will have less crewmates to murder. Therefore, being ejected has a bigger effect on the game.

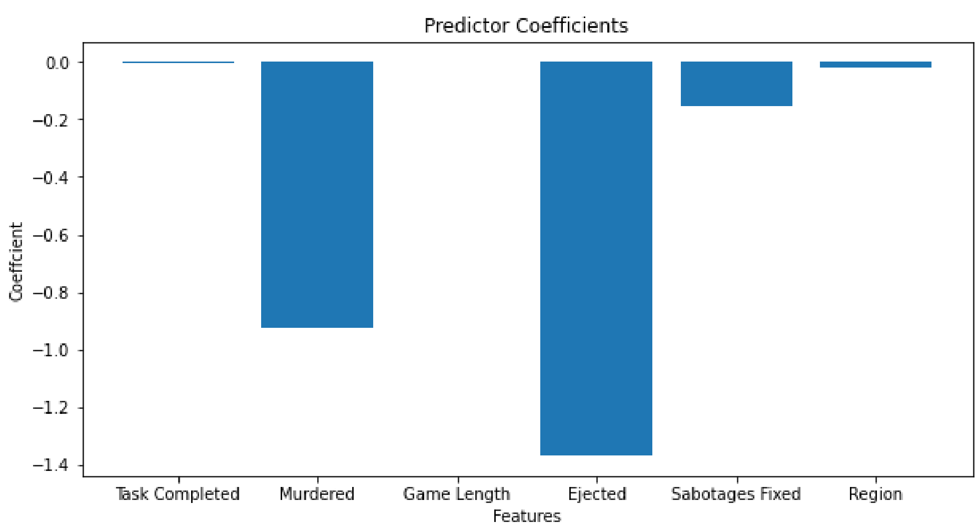

Fig 2. Bar graph of predictors and their coefficients of the Logistic Regression model

Group Imposters
Xu Han `xh852`, Jason Wang `jw7383`, Chloe Zheng `cz1300`

Multiple Linear Regression with Regularization

As each player can be designated as a crewmate or an imposter in this game, we wondered if players' performance when they are crewmates can predict their performance when they are imposters. We created a new dataset, `users`, which consists of predictors from grouping the original dataset by users. The following is the dictionary of `users`:

1. `avg_winrate`: average win rate as crewmates
2. `avg_task_comp`: averaged number of tasks completed
3. `num_time_crew`: the number of games the user plays as a crewmate
4. `avg_murdered_rate`: averaged murdered rate
5. `avg_ejected_rate`: averaged ejected rate
6. `avg_sabotaged_fixed`: averaged number of fixed sabotaged
7. `avg_game_length`: averaged duration as a crewmate

We found out every user has played as an imposter at least once; therefore, we created a target variable (`outcome`) imposter win rate.

Since variables in this dataset are continuous, we chose multiple linear regression, more specifically, ridge regression and lasso regression to solve this problem. To determine which regression fits our dataset the best, we split users into train and test sets with a `test_size` of 0.2, fitted each model with different values of parameters and evaluated models by `mean_squared_error`. We tested 6 different alphas on ridge regression and visualized mean squared error on both train and test set. From **Fig 3** we notice that when $\alpha = 100$, the minimum error approx. equals to 0.0236. Similarly, we ran 4 lambdas on lasso regression and the error reached the lowest value at $\lambda = 0.1$ and it equals 0.0237 on **Fig 4**.

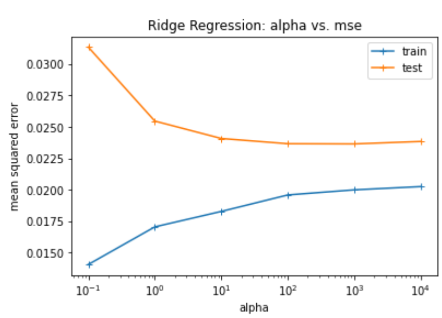

Fig. 3. mean square error on ridge regression

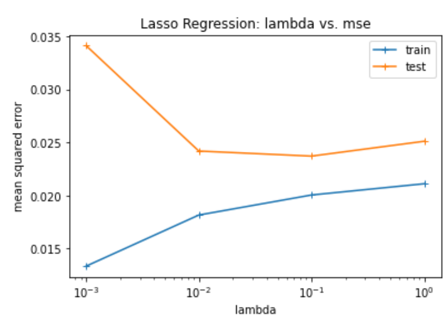

Fig. 4. mean square error on lasso regression

The minimum mean square errors on both models are approximately the same, but we noticed that the optimal parameter for ridge regression is very high, meaning that our models may be underfitting the data. We wanted to see if a player's performance as crewmates can predict his or her performance as an imposter, but both models indicate that there is not a strong correlation between the two given our current features. Our mean squared error of 0.0237 means that on average, the difference between the actual win rate and our prediction is the square root of that, or 0.14. On the other hand, we found that it would have been better to use the average imposter win rate as our predictor over this model – the mean squared error was lower at 0.0196.

Group Imposters
Xu Han `xh852`, Jason Wang `jw7383`, Chloe Zheng `cz1300`

Conclusion

Our three questions were all formulated around analyzing which gameplay aspects were the most important in predicting a win or a loss. Our first question asked whether completing tasks as a crewmate was actually the best strategy to win, but surprisingly, we did not find a significant p-value that supported this theory. To better understand why this was the case, in question 2, we used logistic regression to see which features best predicted the game's outcome through the feature's weight. The results coordinated with our hypothesis test results in that the number of tasks completed matters little. Instead, it is more important to not get murdered and especially ejected. Intuitively, it is more important for crewmates to stay alive so that they can gather clues on who the imposter is (since it is much easier to die when a crewmate is doing a task). It also confirms the idea that the worst thing for crewmates to do is eject a crewmate, so crewmates should not eject people unless they are confident that they are ejecting an imposter. Finally, our regression model showed that a player's crewmate performance is not a good predictor for their performance as an imposter. Moreover, it would be sufficient to use the mean imposter win rate to predict each user's imposter performance.

Unfortunately, our dataset included several limitations that affected the power of our analysis. Both our AUC score in question 2 and MSE in question 3 indicated that our model predicted our target better than chance, but they were both poor models. We think that this is attributed to two main factors - not enough features and not enough rows. When we were doing our analysis, we found that we were missing important features such as rank (instead of just rank change), number of sabotages called and which sabotages were called (as an imposter), how often a crewmate correctly voted to eject an imposter, how fast a crewmate was completing their tasks, and others. More features, as long as they were uncorrelated with each other, would have increased our prediction accuracy. Additionally, the lack of data was mostly a problem in question 3, when we grouped by user to predict a user's imposter win rate based on their average crewmate performance. Because we only had a total of 29 users, we had a severe lack of enough data, and as a result, it is very likely that our lasso regression model is highly dependent on the `train_test_split` and is very sensitive to noise. Because our data was obtained from Kaggle and no other Among Us data is publicly available, we were unable to obtain more data.

We assumed that the data was obtained randomly and independently. However, no documentation was provided on Kaggle that explained how the data was obtained, so it could be very likely that the data was not random. This would impact our results in the hypothesis testing, as the t-test assumes random sampling. Although our findings are introductory and limited due to the dataset, we are interested in further research which could involve an actual experiment where we could control which features we wanted to measure and obtain enough data for more accurate results.

All in all, we wanted to work with the Among Us dataset as a similar project to AlphaGo. Among Us was a cultural phenomenon among millennials and Gen Z across the globe in 2020. It was compelling to figure out how to use game data to produce the best strategies for the game, and although our project was extremely basic compared to AlphaGo, our project ended up extracting meaningful insights that could potentially help Among Us players and perhaps even AI in the future.