Kaylin Dee
Kamron Uther

Exploratory Analysis in League of Legends Kaggle Data using Python

Context and Description of the Data

The data explores statistics of League of Legends competitive matches between 2015-2017. The matches include the NALCS, EULCS, LCK, LMS, and CBLoL leagues as well as the World Championship and Mid-Season Invitational tournaments. The data came from in game stats pulled from competitive matches consisting of 7 separate data sets that all contain competitive data from 2014-2018. We will be focusing on the general LeagueofLegends, the Gold, and the Structures data set for this analysis. LoL dataset includes 7620 observations and 76 variables all based on game stats from the tournament teams. We chose to focus on 2017 because it is the most recent, complete year. We also only want to focus on Season rows, so our new total is 2572 observations.

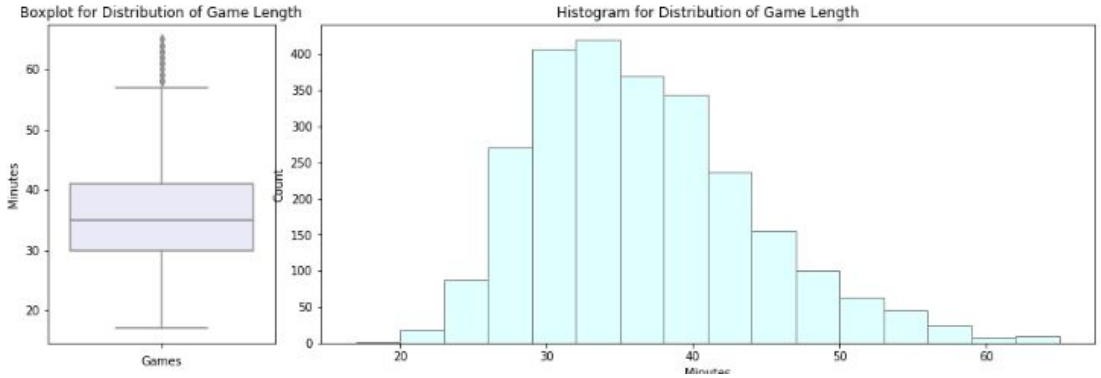
League of Legends is a fast-paced, competitive online game that blends the speed and intensity of an RTS with RPG elements. Two teams of powerful champions, each with a unique design and playstyle, battle head-to-head across multiple battlefields and game modes. Different champions suit different roles and strategies so it is best to find ones that mesh well with your playstyle and teams strategy. With an ever-expanding roster of champions, frequent updates and a thriving tournament scene, *League of Legends* offers endless replayability for players of every skill level. In LoL there are three roads that connect your base to the enemy's. These roads are called lanes, and they'll serve as the means of engaging the enemy team. To win a game you'll have to push down your lane into the enemy base and destroy the nexus at the middle of the other team’s base. Once a team has destroyed the enemy nexus, the game is over and that team is declared victorious.

Chuck Ephron collected the data for analysis of the game for the same reason that data informs other professional sports throughout the world, data can show trends and best-practices in LoL as well. There are quite a few player, league, and team statistics that Chuck captured in his data. A strategy used in tournament games is to ban champions that the best player on each team are very good with. This takes away the best aspects of the playstyle of a team and causes the best player to adjust based on the other teams strategy. Another strategy is to take down a teams structures and fortifications such as towers. Turrets are powerful defensive structures that defend each lane.

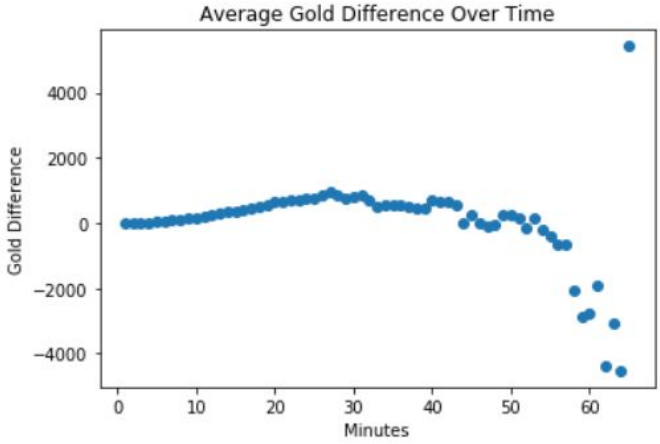
Exploratory Analysis

We decided to focus our analysis on 2017 Season data because it was the most recent year that was complete. First, we chose to display the amount of wins for the Blue and Red assigned teams. From the table, we can see that the blue side has won 340 more games than the red.

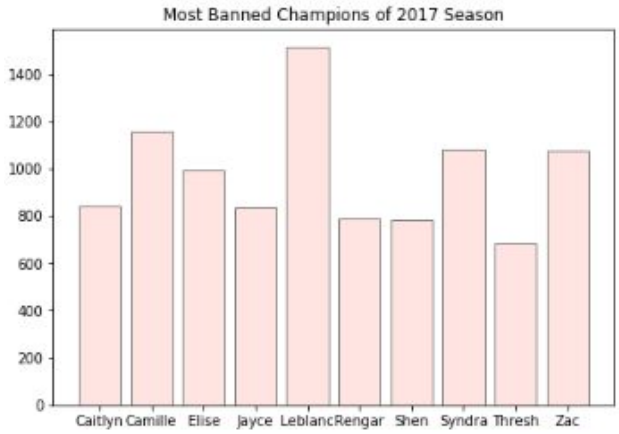
From the summary statistics of the Game Length variable, a game can go from 17 minutes to 78 minutes. However we decided to remove games over 65 minutes because there were only ten games that were over three standard deviations. From the boxplot and histogram created, we can observe number of games and game length. Most competitive games during 2017 Season only lasted between 30-40 minutes long. The distribution is slightly skewed left, indicating that there are a few games where teams were having a harder time competing for the win against one another. The teams put up a good fight!

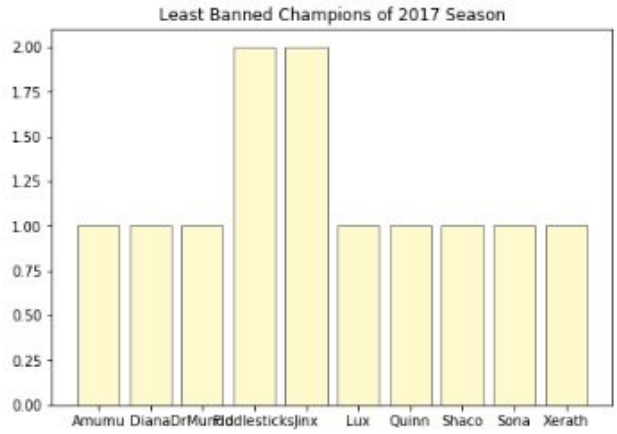


The original Gold dataset includes 99060 observations and 97 variables but then we were able to subset it to 7620 with 97 variables based on the 'goldtype' that was equal to 'golddiff' which allows us to only use the gold difference. For further subsetting, we merged the new gold data set with the 2017 Season data to keep the analysis consistent. We then took the Average Gold Difference for 2017 competitive games for each minute. The scatter plot showed that after the 50 minute mark the average gold difference over time started to diverge more drastically as the game length grew.



We then analyzed the data to find the most and least banned champions within the 2017 LoL Season Games. The data for the blue bans and red bans variable had to be separated out of their ‘pseudo-lists’ and were then combined to find the total champions banned for every game in 2017. This showed that Leblanc, Camille, Syndra, Zac, and Elsie were among the top 5 most banned LoL Champions in the tournament while DrMundo, Xerath, Amumu, Lux, and Quinn were among the many least banned Champions. These results can help gamers strategize by knowing the commonly used heros that pros ban and adjust their play style by mastering other champions. Surprisingly, this indicated that every champion during 2017 was banned at least once! There were none that were left out of the big screen ban picks.





Data Modeling

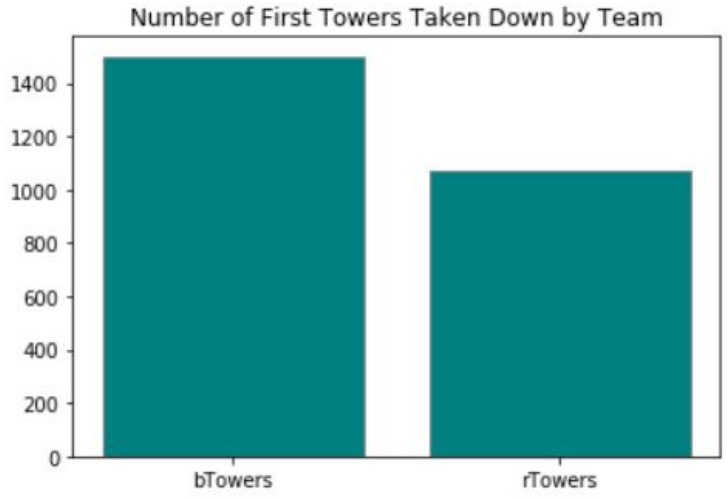
Our research question:

Can we accurately predict if a team won or lost based on when the first tower was taken down and which one it was?

We used the structures data set in the League of Legends Kaggle Data zip file and merged it with the general LoL set, which was eventually subset to only include 2017 Season data again. The data set on structures was grouped by the Address. We kept the 'Time' variable and found the minimum, which indicates when the first tower is taken down.

For some quick exploratory analysis on this new data set, we observed that the shortest time to take down a tower was about 4 minutes. The average time for the winning blue team to take down a tower is about 12.54 minutes, while the average time for the winning red team to take down a tower is 12.61 minutes. They're basically equal.

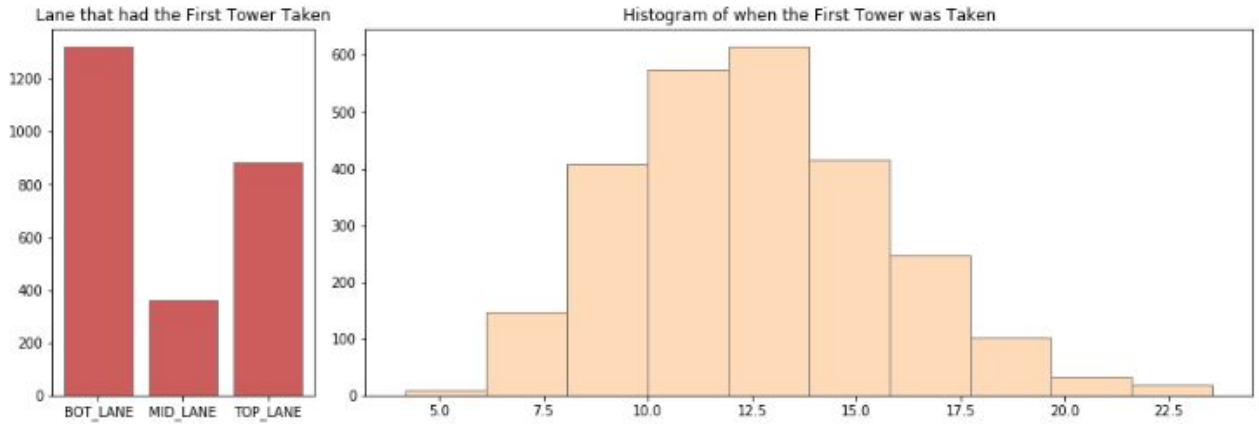
In addition, we have created a table with a corresponding barplot that demonstrates that the blue team tends to take down more of the first towers. Coincidentally, to the right we can see that blue teams win more games. There seems to be some trend that if a team wins more, chances are they will have taken down the first tower more.



Year	bResult	rResult
2017	1456	1117

For the barplot below with Lanes, we observed the count of towers that were the first to be taken down in the data set. It seems that Bottom Lane towers are the most frequent in being the first to go in all 2017 competitive games recorded. Meanwhile, the Mid Lane tower seems to be the least.

The histogram beside the Lanes, depicted the amount of time taken for the first tower to be taken down. 10-15 minutes appears to be a common time the first structure falls, but for a few games it took over 20 minutes.



In constructing the logistic model we focused on using the variables 'Time', 'Team', and 'bResult'. The 'Time' variable one of our features and is the minute where the first tower was destroyed, while 'Team' is whether the blue or red side was the team that took down the objective. This way, we can factor in the concept of whether the team that claimed the first tower had a higher chance of winning. Our response variable is 'bResult' is whether the blue team won or lost. We only need to focus on one of them because it implies the other team's result already. In applying the Logistic Regression on the full data set, our coefficients are:

	Intercept	Team[rTowers]	Time
(Output) Log-Odds	0.5140865	-1.15471327	-0.01118849
(Odds) Exponentiate	1.672	0.21987	0.98887

In order to interpret the coefficients, we need to take the exponent because they are log-odds right now. The exponentiate of the time coefficient is 0.99. We have to subtract from 1 because it is less than 1 so 1 -0.98887 = 0.01113. As time gets larger by every minute, the odds of winning as the blue team decreases by 0.01113. Similar to the process before, if the red team is the one who takes down the first tower, the odds of the blue team winning decreases by 0.7802. For the intercept, we say that at the beginning of the game, a blue team is considered to have a 67.2% odds of winning.

After fitting the model and calculating the score, we observe a 68.51% accuracy in predicting win or lose in the blue team. This is pretty decent! For the Model Evaluation using Train, Test, and Split methods, we found our accuracy to be 68.61%, which is about the same as the accuracy when applying the model across all of the data. We used a 30% of the data to be a testing set. Our Confusion Matrix is:

	0	1
0	263	167
1	145	419

We correctly predicted the blue team lost about 64% of the time, and we correctly predicted the blue team won about 72% of the time. In contrast, we incorrectly predicted a win around 41% of the time, and incorrectly predicted a loss around 25% of the time. For the Model Evaluation using Cross-Validation using 10 folds, our average accuracy is still about 68.52%. This is consistent with the other models created. In conclusion, throughout all of these models and evaluations we result in around at 68% accuracy.

Slides: https://docs.google.com/presentation/d/11iXGQxsS7n4X3IorESqvjpJ8jqbeq2JzpLbg3G4XHo4/edit?usp=sharing
GitHub: https://github.com/kaylindee/131_project
Video: https://youtu.be/fc8-8ocxS5Y