Predicting Rank and Boosting in StarCraft 2

Samantha Gunnerson

Introduction

In the large Esports landscape, StarCraft 2 is one of the earliest competitors, and one of the most persistent. StarCraft 2 was released in 2010, though it's first tournament was during its beta testing, spurred on by the popularity of the original game. Up to now, it has hosted 1970 players in 5569 tournaments, awarding over 31 million dollars in prize money, making it the 5th largest Esport of all time. For that reason, player development is a must for many of the teams around the world, from the Korean giants attempting to maintain their dominance, to foreign underdogs trying to unseat the giants.

All players are a part of the league system run by the parent company of StarCraft 2, Blizzard Entertainment. The league system has 7 levels that get more competitive and smaller in size as a player climbs through the level. This data set was taken during the Wings of Liberty expansion of the Starcraft 2 game, so the leagues were divided with the bottom 4 leagues (Bronze, Silver, Gold and Platinum) each containing 20% of players, 18% of players at the next league (Diamond), 2% of players in the Master ranking, and 1000 players worldwide in the top league level at Grandmaster. Professional players are then sorted differently, though they are usually pulled from the Master and Grandmaster rank. League status is usually determined by the success that a player is having in the game, i.e. a player must hit a certain percentage of wins in a certain period of time in order to move up the league rankings (or certain percentage of losses to go the opposite direction).

This analysis was motivated by a need to understand where players fall in the league system of StarCraft 2, as well as a need to use that placement as a way to root out cheating. By understanding where players fall, developers can figure out players that are misplaced, either accidentally on Blizzard's end, or intentionally on the players end.

The Data Set

The data being used was acquired by Mark Blair, Joe Thompson, Andrew Henrey and Bill Chen from Simon Fraser University on September 20, 2013. It was aggregated using screen movements into screen-fixations based on the 2000 paper from Salvucci & Goldeberg. It has been made publically available via the UCI Machine Learning Repository. The data contains just over 3000 observations on 35 columns, or features of the data.

Using the Blair et. all data set on StarCraft 2, the following data wrangling steps were completed. First, the data was imported and checked for cleanliness. Second, the data was inspected for null values, and those null values were dealt with. Finally, the data was inspected for outliers across all of the columns.

When the data was initially imported, it was already fairly clean. The only steps necessary to have fully clean data was to convert the columns of age, hours per week and total hours from objects to floats. Once this was completed, every column was either of the type float, or integer.

Additionally, the data was inspected for null values. While most columns had no null values, the age, hours per week and total hours columns did have null values. Of the 3395 observations in the data set, age had 55 null values, hours per week had 56, and total ours had 57. When inspected, these sets of null values lined up almost perfectly, with 55 overlaps, and only 57 total rows with null values. Upon further investigation, it was found that these null values represent all of the professional players, as well as 2 extra players without this information. The two extra players were dropped, and the 55 professionals were kept, with an understanding of not having the factors of age, hours per week, and total hours available for the professional section of the data set.

Finally, the data was inspected for outliers. There was one outlier found in the total hours column, that appeared to be from a mistake in the initial input of the data. While some of the other columns of the data set had values that may have been close to being outliers, they are considered for removal after further analysis, as when the data is subsetted, they have fallen within an appropriate range. In most cases, they fell within acceptable ranges upon subsetting and were not removed.

A Note of Vocabulary

As much of this data set and analysis depends on a base understanding of gaming terminology, especially those terms used for features in the data, as well as important terms about gameplay, understanding those terms is important to interpreting the results. These terms are referenced below for clarity.

APM is an acronym for actions per minute, a statistic that StarCraft 2 computes for every single game, and that is reported to the player in every single game.

PACs is an acronym for perception-action cycles. This is a neuroscience term defined as a circular interaction between an organism and its environment during a sensory-guided sequence of behavior towards a goal. Or, perhaps more helpfully, it is the cycle of perception - prediction - action and outcome. In this dataset, a PAC is measured as a fixation with at least one action. So, when a player fixes on a part of the screen, and then acts, this is considered a single PAC.

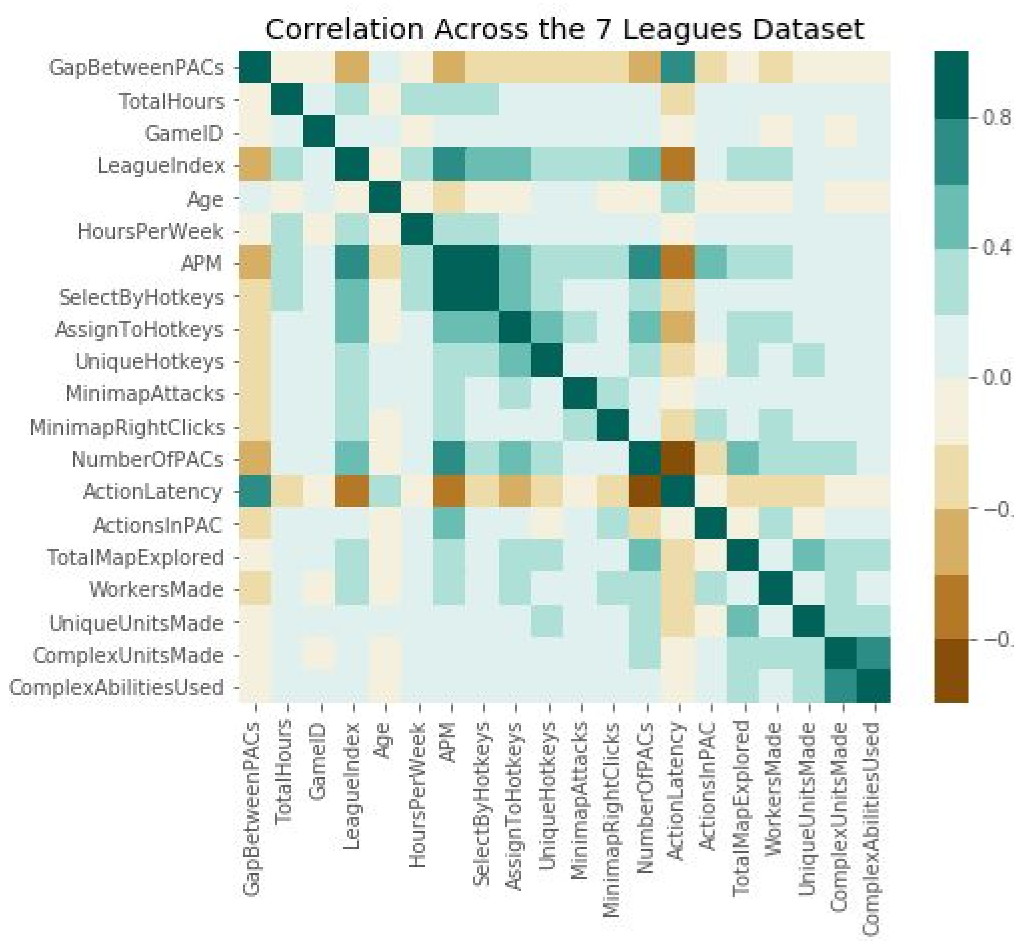
Hotkeys are used in a similar way to hotkeys used in general computer use, though in the face of a real time strategy game (RTS) like StarCraft 2, the assignment of hotkeys becomes vital to strategy and speed of a play.

Action Latency is defined as the mean time between the onset of a PAC to the first action in the PAC. So, it is a measure of how quickly a player reacts to a stimulus.

Initial Findings

Through further investigation of the StarCraft 2 data set, a few trends stick out across both the professional and non-professional data set, which were separated for ease of comparison, as well as investigated together.

General Trends



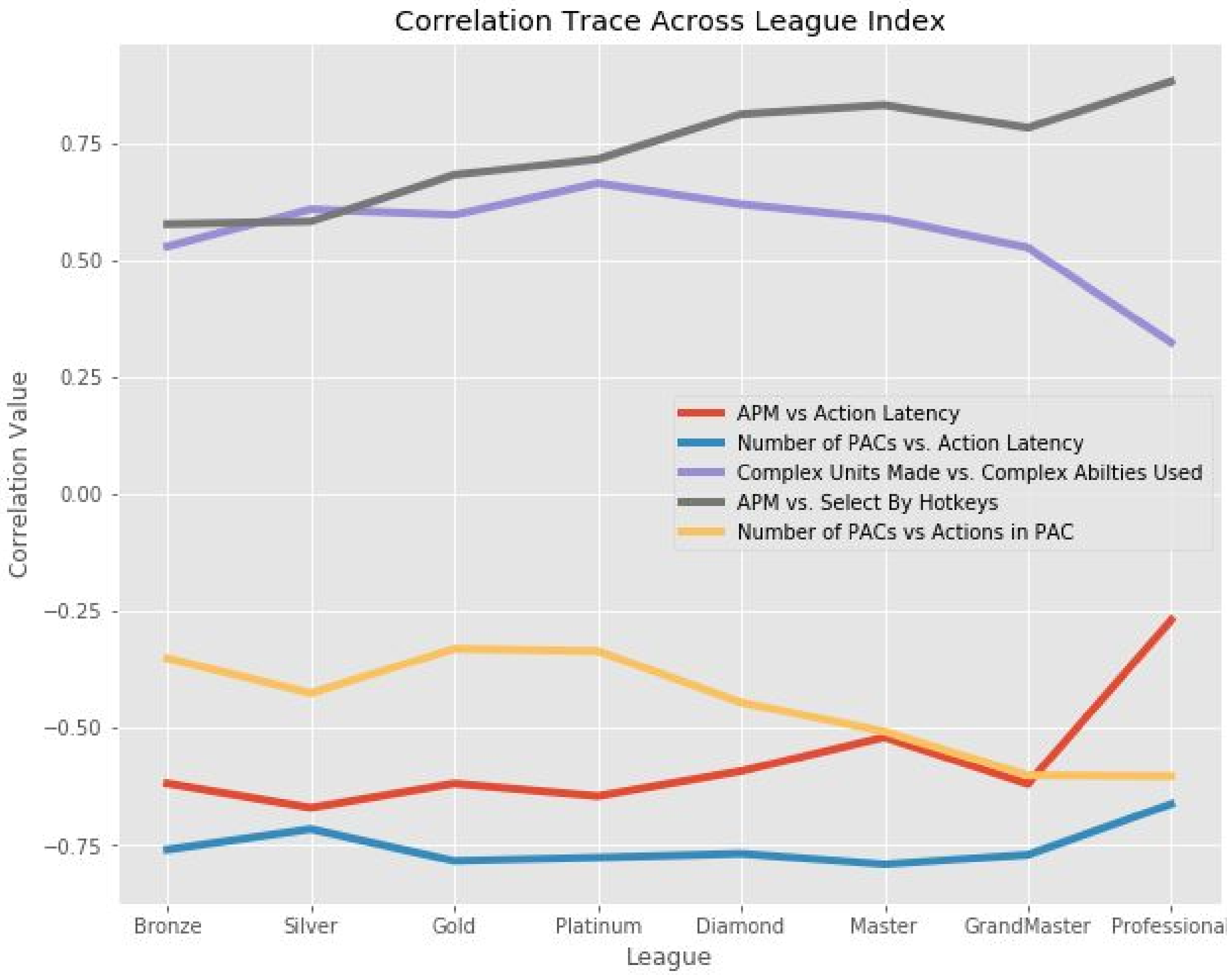
Note that the above correlation graph does not include the professional section of the data set.

In general in the data, there are two very strong correlations, and many minor ones, some of which are investigated further later. The two major ones, as seen above, occur between APM vs. select by hotkeys and number of PACs vs. action latency. Both make some sense in terms of gameplay.

In the case of the first, it makes sense that if you can use selection via hotkeys, rather than using a mouse to move around the map, you can use more actions per minute. This can be seen across many video games, where understanding of hotkeys allows for faster gameplay and faster response time. In StarCraft 2, hotkeys are seen as even more important. In fact, they are so important that there are hotkey layouts designed by players to make the use of hotkeys easier, and faster. These layouts are often designed to make common hotkey combinations easier for players. Hotkeys are a key part of the game, and are used from the lowest levels.

The other relationship between number of PACs and action latency also makes sense from a game standpoint. You would hope as a player that the more PACs you can perform, the less time you spend dormant. Or, stated another way, the faster you move on to a new task, the more tasks you can complete. So, for a game where reacting to new information as fast as possible will often determine the effectiveness of the reaction, being fast is incredibly important.

Trends Across Leagues



From above, we can see the trends of some selected correlations across the leagues, or levels of play.

The first relationship, at the top of the graph in gray, represents the correlation between APM and select by Hotkeys. This indicates that at higher levels of play, the number of actions per minute increases along with the use of hotkeys for selecting units. The correlation follows the idea that higher level players can perform more actions more quickly, usually through the use of hotkeys.

The second relationship, shown in purple, shows a more interesting and less obvious trend. In lower league players, there is a correlation between the number of complex units made and the number of complex abilities used. But, by the time a player reaches a high level of play, the correlation has reduced in significance. So, players who are just starting out will use complex units and complex abilities in conjunction at a higher rate than more senior players will.

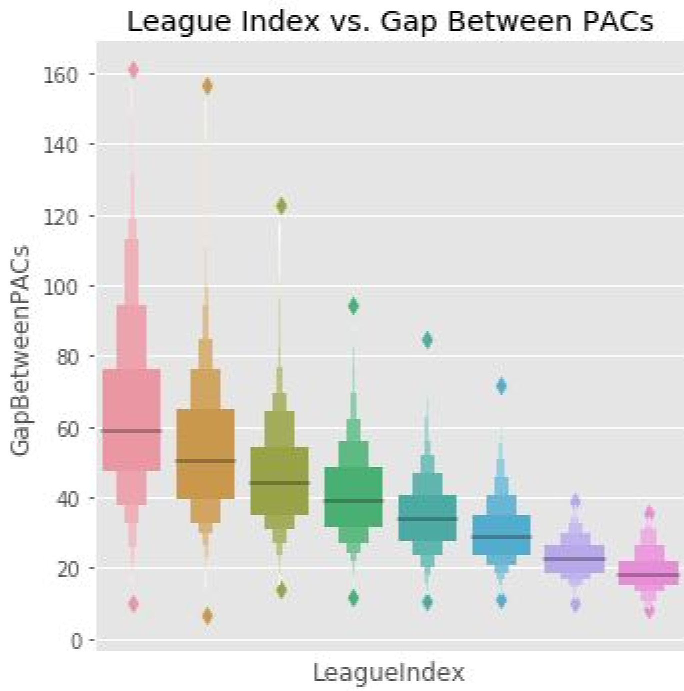
The third relationship, in gold, is between the number of PACs a player has and the number of actions in a PAC. This inverse relationship suggested by the negative correlation suggests that players who have more and more PACs will have less and less time between those PACs. This makes sense, as most players at a high level can perform a lot more actions than low level players, as seen in top level tournament play, where the action in StarCraft 2 can move so fast that an untrained eye can barely keep up.

The fourth relationship, in red, represents the correlation between APM and action latency. As you travel up the leagues, this negative correlation decreases in significance. This is nearly the opposite of the trend in the gold relationship. In StarCraft 2, where players need to reduce their down time between actions in order to keep up with the play, the reverse relationship to APM makes sense. In the final professional league, this relationship is slightly less, suggesting that the professional players will not necessarily have less time between actions when they have more actions. This may be due to them using fewer actions more efficiently, as professionals must have very specific actions as part of their strategy in order to win against equally talented players.

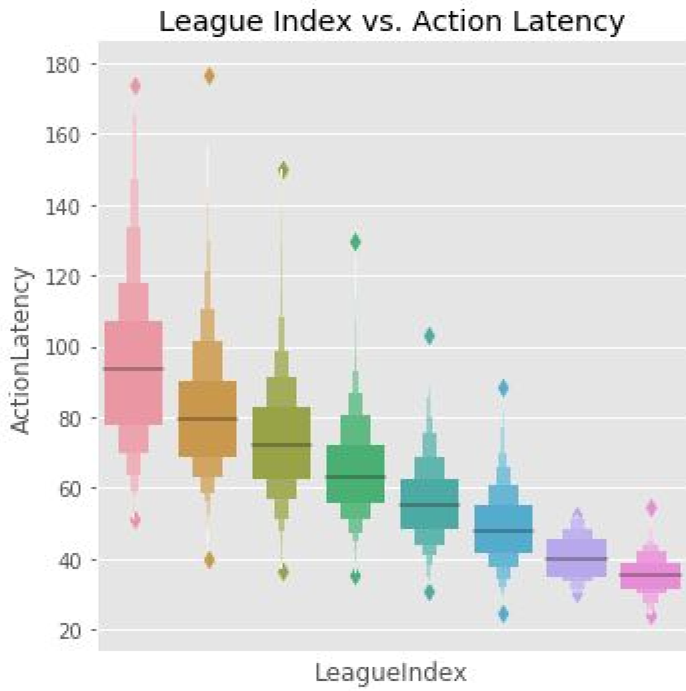
Finally, in a relationship that is mostly consistent across all levels of play, the blue line represents the correlation between the number of PACs and action latency. Since action latency is a measure of the space between PACs, it makes sense that as the number of PACs increases, the action latency would decrease. In real game terms, the faster you can interact with units, the more units you can interact with. This speed component is what allows top players to react to their opponents, as well as plan their own strategies simultaneously.

Trends within League Index

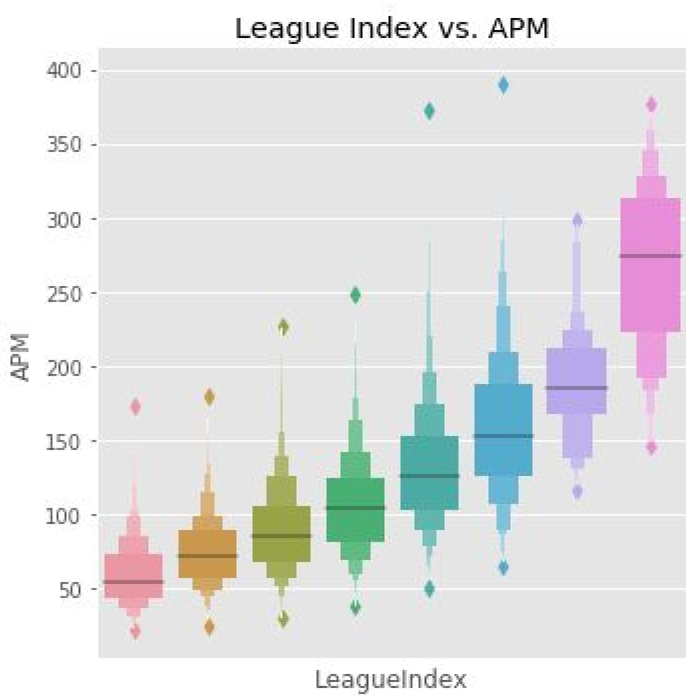
Within the league index, there are often trends that occur as you go up the leagues. These trends vary from feature to feature. Note that the following graph have the following order for leagues along the x-axis: Bronze, Silver, Gold, Platinum, Diamond, Master, Grandmaster and optionally Professional. This is also the order that the league structure is built on, with bronze at the bottom and professionals at the top.



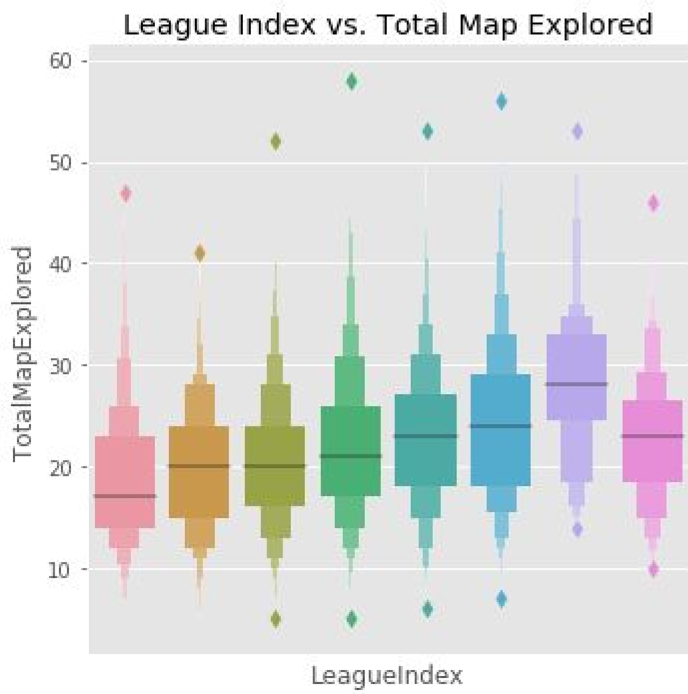
For example, when the gap between PACs is disaggregated by league index, the median of the gap between PACs per each league decreases somewhat as you go up the leagues. And, more interestingly, the spread, or variance, at the higher leagues of play is significantly smaller than the variance at lower leagues. This means that while there are players at the lower levels with very small gaps between their PACs, at the higher levels you are far more likely to have small gaps. In terms of the game, in order to be at the higher, more competitive levels, players must be faster in their actions, whether those actions be movements, selection, or interacting with the map.



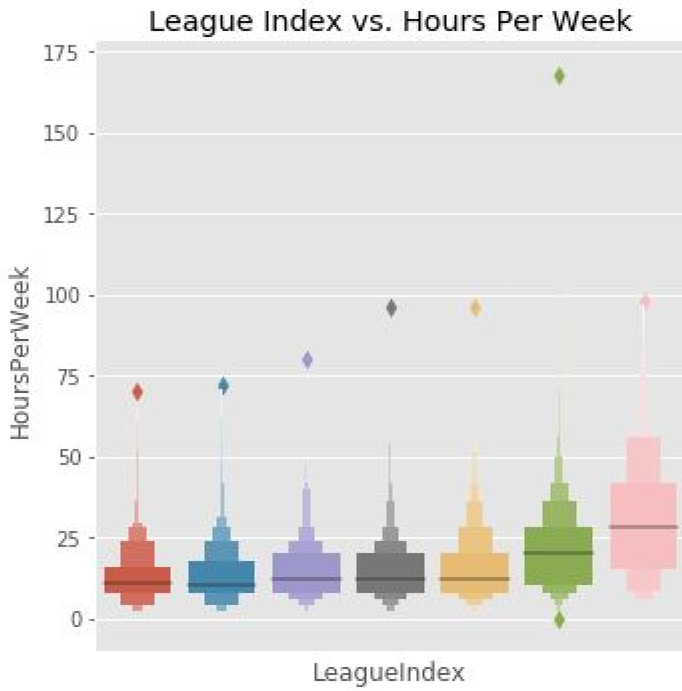
This same trend continues with action latency across the league index. It also follows the same logic for players. If you can get to a new action quickly, you can react faster, and make decisions faster.



Next, when APM is broken down by league index, the opposite trend to the previous two is seen. The median of the leagues increases significantly across the leagues, all the way up to the pros. And, as you increase in league, the variance in the APM increases significantly. In game, this may represent players who use each action recorded by the game much more importantly, versus a player who bumps up their APM by just randomly moving around the map. It may also represent the differences required to be effective across the multiple races of the game.



Total map explored has its own interesting trend as well. In this case, there is a marked difference between the trend through the ranked levels and the values of the professionals. While in general, players explore more of the map as they get to higher levels, the professionals show a drop from the grand masters, who seem to be exploring more than anyone else. Besides the professionals, this trend makes sense. The more of the StarCraft map you can see, the more likely you are to be able to see what your opponent is doing, and the more likely you will be able to predict what they are doing. This is key for the strategy component of the game.



Hours per week follows an interesting trend, where the medians are fairly similar across the lower levels of play, but as you get up to the grand master league, there is a big uptick in time spent in the game. Note that the below graph does not include the professional league, as there was no data on their time spent with the game. This upwards trend makes sense, as it takes time to not only be an expert in the game, but to keep your skills sharp. It also allows time for strategy building, to create, test or tweak new or old strategies.

Statistical Data Analysis

Using One Way ANOVA and Tukey's test, the relationship between the league and the various features of the data was quantified more thoroughly. These features can be sorted in 3 categories based on how easily leagues can be separated using these variables. First, there are features that have a significant (at an alpha of 0.05) difference between all or almost all of the possible league means via a Tukey's test. Second are the variables who have differences in means in a subset of leagues. Finally are the ones who struggle to tell leagues apart across the map, and are thus likely to be less helpful for modelling in the future.

Significant	Regional	Poor
APM	Workers Made	Unique Hotkeys
Action Latency	Select By Hotkeys	Minimap Right Clicks
Assign To Hotkeys	Minimap Attacks	Actions in PAC
Number of PACs	Total Hours	Total Map Explore
Gap Between PACs	Hours Per Week	Unique Units Made
		Complex Units Made
		Complex Abilities Used
		Age

In the first category, the features of APM, action latency, assign to hotkeys, gap between pacs and number of PACs had differences between all or almost all of the leagues in the Tukey test. Specifically, APM showed a difference between the means of all groups, which indicates that the positive trend seen earlier is significant towards indicating the league of a player. Action latency has a marked difference in all paired means besides the grandmaster to professional pair, as does number of PACs. Assign to hotkeys shows no difference between bronze and silver, but statistically significant differences at all other leagues. Finally, gap between PACs shows a difference between all of the leagues except on the transition between grandmasters and professionals.

In the second category, there are 6 features. Firstly, workers made can differentiate between lower leagues, from bronze to platinum. Secondly select by hotkeys and minimap attacks can differentiate between middle leagues (platinum to masters and platinum-grand masters respectively). Finally, when the data set does not contain the professionals, total hours and hours per week can differentiate between the upper levels (diamond to grandmasters and platinum to grandmasters respectively).

In the third category are features that are all across the map. Some would only be helpful if leagues were skipped, for example pairing bronze with gold. Others struggle across almost all of the features tested. These features are unique hotkeys, minimap right clicks, actions in PAC, total map explored, unique units made, complex units made, complex abilities used and age.

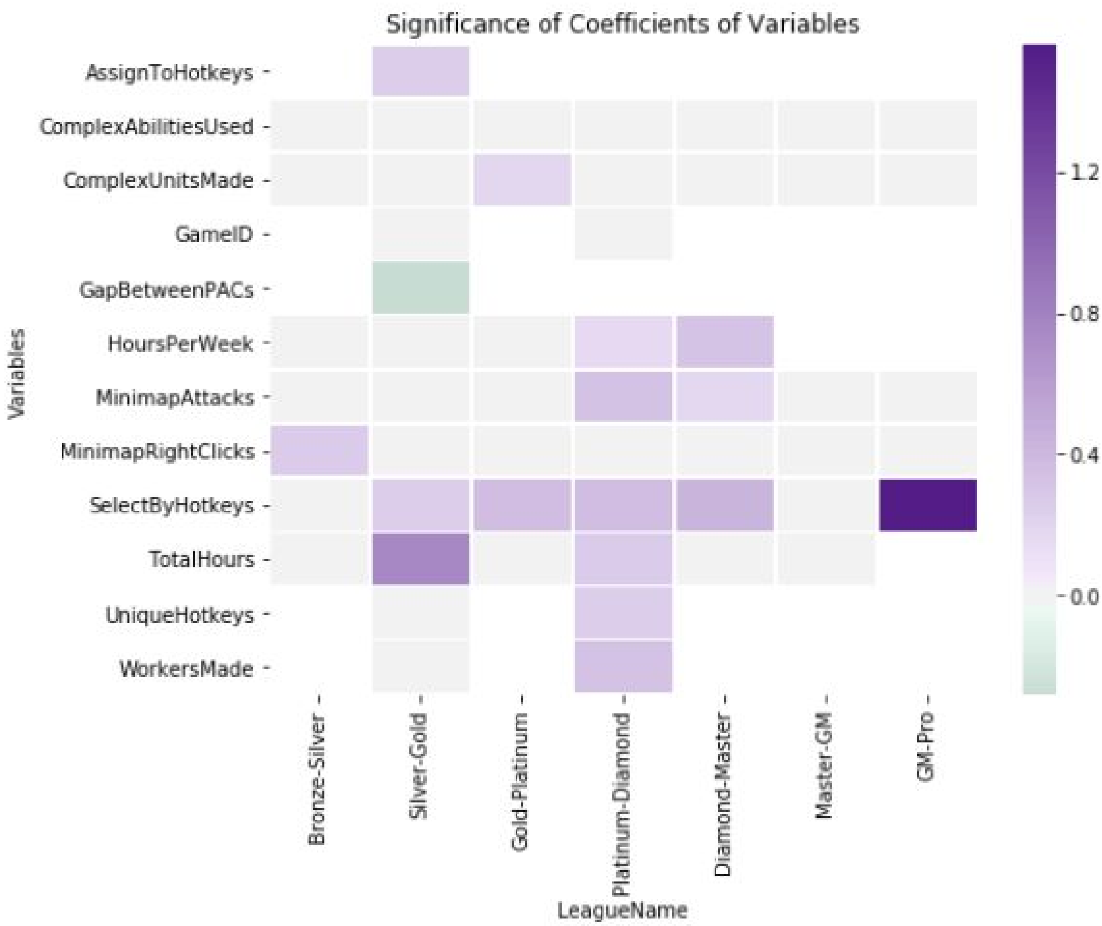
In Depth Analysis

In the final stages of analysis for this data, machine learning was used to look at league to league transitions as well as looking at the business case of boosting, a form of cheating common in many video games.

League to League Transitions

For each transition between leagues, the VIF of each of the features was computed to look for variables with multicollinearity. Then, these VIF values were minimized by dropping the greatest value and recomputing until all the values were under a threshold value. The initial threshold was a VIF value of 5, and then another round was done to select for variables under a threshold VIF value of 2.5. These variables were then selected, and logistic regression was run on each of the league transitions using the appropriate subset of variables.

Finally, all of the league transition models were thresholded to choose the best model and the best threshold for that model. So, at each VIF level, the predicted probabilities of the logistic regression on that league were compared to a threshold value to find the threshold at which the model predicted the transition most accurately. Then, whichever model had better predictive power between a VIF of 2.5 or a VIF of 5 was kept for analysis.



In the above graph, the results of all of the league transitions are shown. The white spaces represent variables that were not included in the particular league model. Grey values represent coefficients are not significant, i.e are not statistically different from zero. Finally, the colored spaces represent positive coefficients (in purple) and negative coefficients (in green). Notice that only one of the coefficients across all of the variables is negative. So, in the case of gap between PACs in the silver to gold transition, those in the gold league have a smaller gap. This sign difference is due to the fact that better players should have a shorter gap between actions, so there should be a negative relationship, as the value gets smaller in higher leagues as higher level players take more actions. All of the other variables have a positive relationship if any, which makes sense in how they are defined.

Some variables, like assign to hotkeys, and complex units made, are only significant in one of the league transitions. Some, like total hours and hours per week, are significant in two. And select by hotkeys is significant in 5 of the 7 transitions. Another interesting trend is that in the middle of the league transitions, there are more significant variables than at either end of the spectrum, where the models have relatively few variables with significant values. Some more of the trends are discussed more in depth below.

The Story of the Platinum-Diamond Transition

The platinum diamond transition had the logistic model with the most coefficients that were statistically significant. This means that out of the 10 variables that were maintained through the VIF dropping process, 6 have coefficient values in the model that are statistically different from 0, in that a confidence interval of 95% around these values does not contain zero. So, for these variables, the coefficient value can be trusted because of low multicollinearity, and it is also significant to the model.

These variables are hours per week, minimap attacks, total hours, unique hotkeys, workers made and select by hotkeys (see select by hotkeys section later). In fact, all 6 variables have a positive effect, so in all six cases, diamond tier players have higher values than lower platinum tier players. Hours per week and total hours make sense. A higher level player must practice more. The other 4 show a possible ideology shift between platinum and diamond.

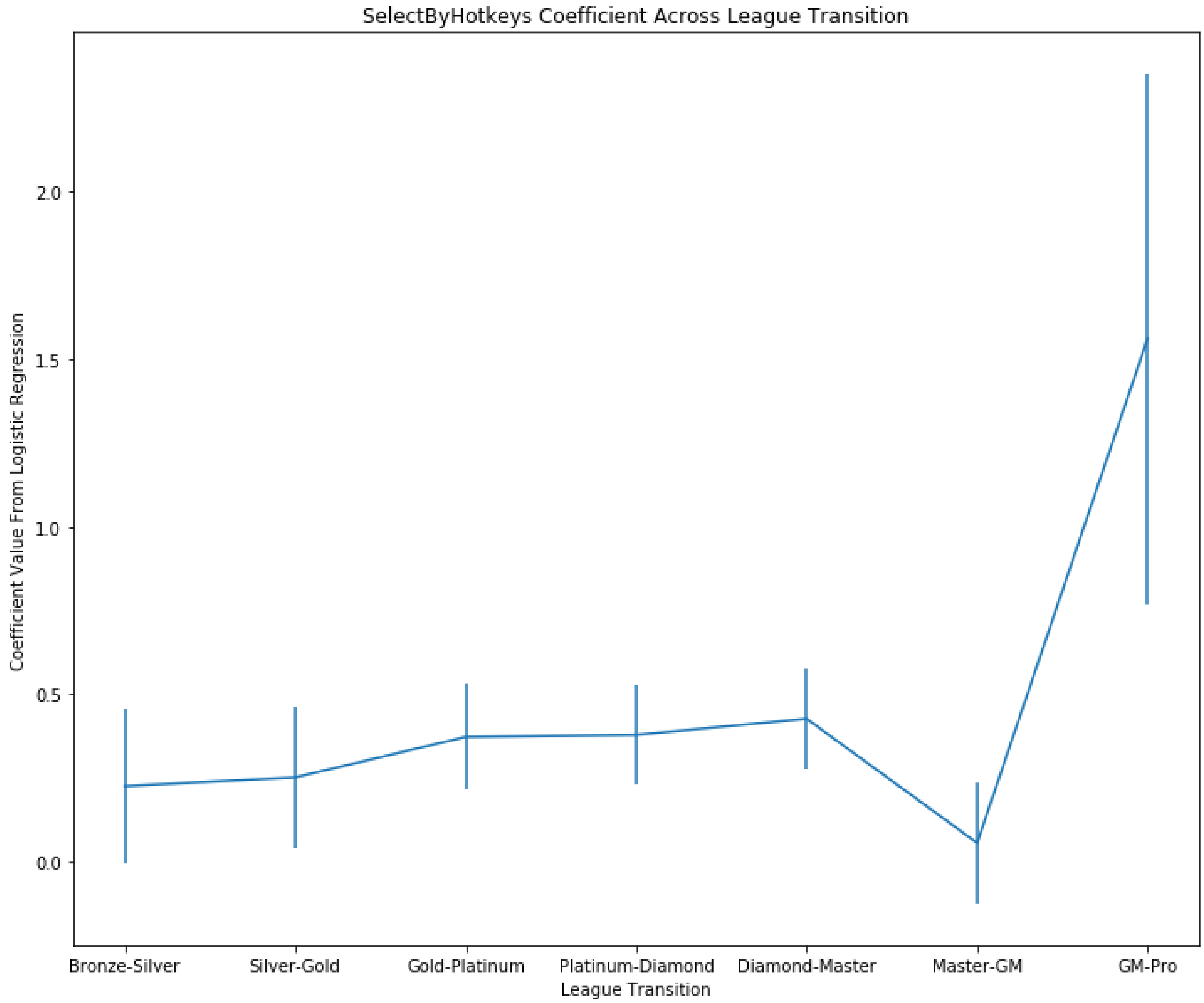
In many real time strategy video games, there is a concept of micro and macro play. These play styles focus on different core mechanics of the game, and are both needed to be a successful player. In StarCraft 2, micro is the management of single units or groups of units by over ruling the AI, in order to reach a specific objective. It is used to keep as many units alive as possible. Micro is similar to real-world micro-managing, where one manager focuses on changing the process done by a single employee. Macro on the other hand, is the management of constant resource farming and maintaining high production across the board. It is used to create the largest possible army. Macro is similar to real world managers who focus on only the numbers of the group as a whole and getting the most profit for the company in that fashion.

In the case of the 4 other significant variables of the platinum to diamond transition, 3 of the 4 point to better macro gameplay by the diamond players. This is most obvious with workers made, as having more workers allows for a player to collect more resources and thus build a bigger army. Unique hotkeys allow players to move around the map faster, making sure that all of their production units and buildings are producing at full capacity. More unique hotkeys also means that a player is doing many different tasks, which may also indicate a better macro understanding of the game and what tasks need to be done when. Finally, minimap attacks thrive on allowing a player to let the AI take control while they work on another task. By using the minimap to attack, units can be controlled remotely without having to divert attention from production tasks that the player needs to focus on.

So, the platinum to diamond transition represents a possible change in the ideology of the players as they move to higher leagues, as macro seems to be a much larger consideration than micro.

Select By Hotkeys: Micro Strategy and the Importance of Specific Control

While in the platinum to diamond transition, macro game play is newly emphasized, micro gameplay is important across all levels of play.



For example, to select a specific scout, a player without hotkeys must first find that that unit, then use the minimap to navigate to it, and then finally select the unit. A player who has defined a control group with the unit must only press the associated number key. And, they can tap that number twice in order to view the unit, or only once to select the unit remotely. This allows a player to choose between fully giving their attention to a specific unit, or commanding it from afar, an option that the player without hotkeys would not have.

This addition of micro play allows a player to better adjust to an opponent's move, and adjust faster. Both would help a player win, and help them move up the league system.

Finally, the pros show significantly higher rates of selection by hotkeys than the grand masters. This may suggest that the ideology at the professional level is different than that at the grand master level, with more benefit to be gained by focusing on solid micro strategy. This may also be attributed to the fact that higher level players must have solid macro understanding in order to make their way to high level play, so any gains in micro can allow a player to beat out an equivalently skilled macro player.

So, selection by hotkeys showcases one of the key components of any real time strategy game: ability to react appropriately to the situation at hand. The player who is able to control their units quickly, efficiently and effectively is the most likely to win.

Boosting: Random Forests and the Hunt for Cheaters

Background

In the world of video games, there are often levels of play. And to get to higher levels, a player must hone their skills, and spend more time playing the game. But what if a player has finite time, and a lot of spare cash? They may choose to partake in boosting (sometimes called elo boosting), one of the many ways to cheat league systems in games. The basis of boosting is this: a player pays someone from a much higher league to play on their account until they reach a certain level. Its history dates back to early MMO-RPGs, such as World of Warcraft, where players would pay each other for in game currency. In popular games like Overwatch, players can make hundreds of dollars selling their services. A simple Google search will reveal high level players willing to play for you, with records of success, and pricing available within one click. And for players who are not contracted at the professional level, it is one of the only ways to make money, and by far the most lucrative.

Boosting has disadvantages on a game level. It causes poor game experience for low level players, and is a high risk solution to the problem of moving up leagues. Most major games will ban players on either side of the boosting equation: those who pay to boost, and those performing the service. Boosting is also time consuming for the players who boost, making its hourly wage far lower than expected. And, in the case of players in South Korea, one of the biggest e-sports hubs in the world, boosting is criminalized, with up to an $18,000 fine, and a two year suspended prison sentence.

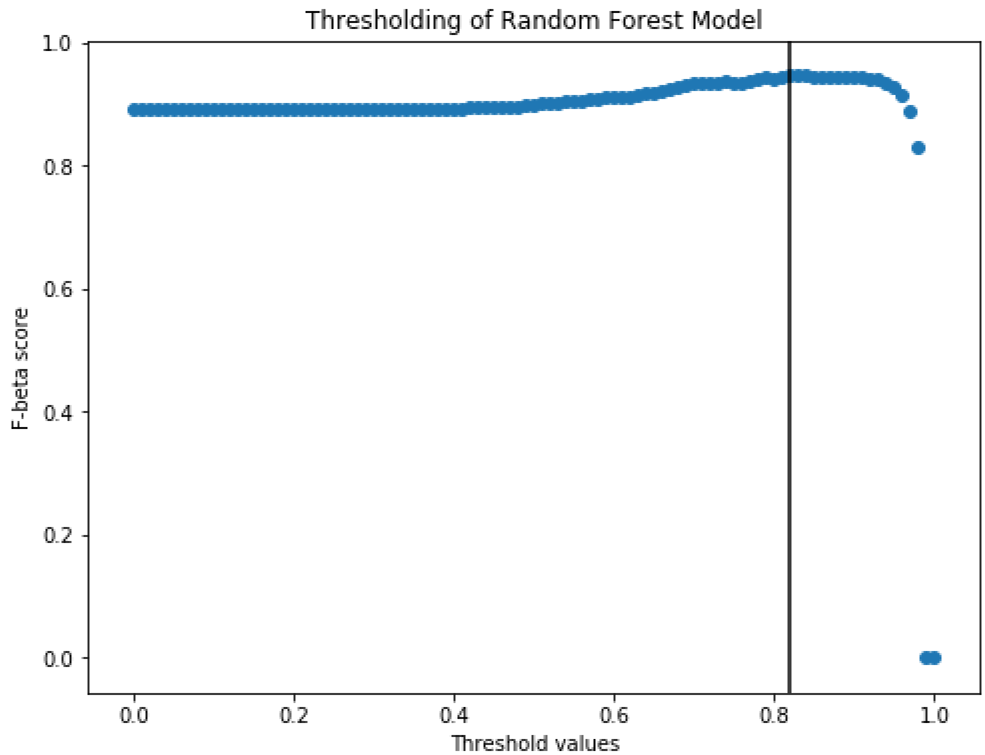
While some games have begun cracking down on boosting, some have taken a more pragmatic approach. In the case of many games, a customer who is boosting is still a customer, and still an opportunity to make money. In these cases, the incentive to catch players who are cheating is low, even when players in game feel the effects of those cheaters. Games like StarCraft 2, which is free to play at this point, have little incentive to spend a large budget on catching boosters. Thus, having a machine learning solution allows them to use very few resources while still attempting to find and punish those who are boosting.

Application in the data

To look for players being boosted, a comparison was made between silver league players and all of those in higher leagues. In theory, players who the algorithm mis-labels as higher league than their actual silver level may be boosting. This assumes that higher level players will still act like higher level players when they play on a lower level account. Since this may not actually happen, the model can only suggest potential boosters, and may miss boosters who are changing their play style significantly when the work on a lower level account.

Method	F-Beta Score (Beta=3)
Random Forest	0.9465
Logistic Regression	0.9385
KNN Means	0.7510

In order to model the data appropriately, 3 different possible methods were tested and tuned. First, a random forest classifier was fit to the data, and then initially tuned via a grid search on the number of estimators and the depth of the estimators (the optimal of each was 6 and 180 respectively). Then, the resulting predicted probabilities of that model were thresholded to find the best threshold, and the value of the f-beta score to determine the best threshold (see below for a more thorough explanation). This would end up being the prefered model. Second, a logistic regression was fit to the data. The data was first assessed for multicollinearity by calculating the VIF values and then removing variables to reduce the VIF, in the same method used earlier. Then, the predicted probabilities were thresholded, finding the best f-beta score of 0.9385 at a threshold of 0.45. Finally, a KMeans model was fit to the unlabelled data. Since the number of labels was known, the KNN model was fit to 2 clusters. In this case, due to the data shape, KNN Means produced a nearly even split in the data, thus having the poorest f-beta score of 0.7510. So, the random forest model had the best performance, and was chosen as the final model.



An f-beta test was used to assess the effectiveness of the models. A beta value of 3 was chosen, which favors recall. In the case of this data, favoring recall favors finding all of the boosters in the data set. As the whole goal of this model is to find boosters, favoring recall makes sense, as the consequences of lower precision are much smaller, as they would only require checking on a possibly larger pool of boosters. Then, as shown above, the random forest model was thresholded across its predicted probabilities in order to find the case when the f-beta score (where beta was 3) was the highest.

		Prediction	
		Silver	Non-Silver
Actual	Silver	79	16
	Non-Silver	165	547

The above confusion matrix is created by the model on the test set. While there are still many errors, there are relatively few identified boosters, which was the goal of the f-beta testing. Note again that the assumption of boosters existing in the dataset possibly untrue, and that there is no evidence as to how higher level players play when they are boosting an account.

So, an accurate random forest model of the data such as the one found above allows for StarCraft 2 developers to easily detect and analyze potential boosters. This identification would then be sent on to a fraud department, who could investigate the potential boosters, and tag them in the data set. Finally, the model could be run again on the full data set, with confirmed boosters tagged, in order to find other or new potential boosters.

Conclusion

StarCraft 2 is a complex game, where there are a lot of factors that affect gameplay, and the players experience. And while metrics can be used to determine characteristics of top players, the use of machine learning allows developers to inspect the game not only for commonalities among players, but also for potential cheating.

Many metrics show trends across the different leagues. These trends are often intuitive to those who understand StarCraft 2 well, and can be explained by many of the default norms of the game.

The analysis also shows that ideological differences in strategy, as well as skill in using those strategies, can be seen in the metrics in game. Players can been seen to get better at their overall understanding of the game, as they use more types of hotkeys more often. They also improve in their immediate reactions to situations, as they select groups and command them to specific tasks more often in the higher leagues.

Developers can also find cheaters by comparing silver league players against those in higher leagues in order to identify people who are possibly boosting their accounts, and to crack down on it. By having machine learning identify possible cheaters, man power can be more appropriately used on people who are obviously cheating, rather than being spent on finding any of the cheaters, especially in games with large ecosystems, like StarCraft 2.