Ethan Cruz
Projects in ML, RPI
November 2022

Progress Report

For this project, I started with gaining understanding of how to gather data from the League of Legends API. This involved getting connected to the LoL Developer's community, which was oddly large and accepting. I got in touch with a data scientist who went by 'Griffin'. He was doing a similar project as me, using data from games to predict the winner/loser during champion selection.

There are some differences from his project and mine, but in general, they are shockingly similar! I don't want to just, take his project, so I'm going to be making some major changes as I see fit.

As his project goes, he collects around 14 million games a patch and adjusts a massive linear regression function daily. A patch is a time frame of about a week where the in game characters' abilities remain unchanged. After that week, riot (the game company) usually adjusts certain characters' stats so they can be more competitive in the game. This means that each week, the data is no longer IID since some fundamental aspect of the game (characters' power) has been changed. From there, the data HAS to be recollected.

This means, at the start of every week, you go back down to 0 games analyzed and your hoeffding bound is literally at maximum! The confidence in the in-training error is zero. Because so many games are played, though, you can be quite confident in the training error near the end of the week. It also should be noted, the week starts and ends on wednesdays since that's when the patches go live.

In my project I proposed a model of 325 variables since there are 162 champions (aka characters) in league and there are two teams which a champion can be on. Griffin, on the other hand, uses a model of a whopping 14k+ variables!!!! This is based on the idea that not only can a chapion be on a certain team, but that it can posses a certain ROLE on that team. For example, a jungler on blue team is different than a top laner on blue team. This extra layer of detail adds way more possible variables.

Because of Griffins' massive number of variables, he has to collect TONS of games daily just to remain accurate. I, a humble undergrad, don't have access to League's advanced API and cannot collect so much data. Therefore, 325 is a compromise that not only keeps me unique from Griffin, but improves my chances of being accurate.

Although our number of variables are different, there is still a lot I learned from Griffin. One of the more important things I learned is the idea of Sparse matricies! Each vector in my matrix is of size 325. But since there are only ever 10 champions played during a game, 315 of those values are 0, while 10 are 1. This means the matrix has insane percentage sparsity. Griffin mentioned to me that sparse matrix algebra is much more efficient to compute, so I might decide to use sparse matrix work in linear regression if computation becomes lengthy.

Griffin also described his interest in using neural networks, but unfortunately his area of expertise remains in statistics, so he hasn't gotten around to doing NN learning yet. I believe it's my duty to attempt neural network learning on the data! Therefore, once I gather my data, I won't just optimize my linear regression model, but experiment in SVMs, Neural Networks, and perhaps some other models (I'm thinking k-Nearest Neighbors).

Aside from the theoretical and community outreach, I have been testing the League API and it's much, much more frustrating than I originally thought. Just to collect game data, you need to make multiple API calls and I only get 100 requests every 2 minutes. If I had a public website with use for players, I could get up to 30,000 requests every 10 minutes. Perhaps setting up a website might be next on my list, although I'd need to get a friend to help me on this, so maybe during my free time in the future...

That's all so far, I'm very happy with the amount of progress I've made and it's all thanks to the incredible Riot Developer community. Alone, it was tough to get even a couple calls into the server, but there are UIs and python libraries dedicated to fixing issues with the Riot Servers API which accelerated my progress.