**LoL Esports Win Prediction**

1. **Introduction**

LoL Esports has been one of my favorite things to watch over the last 4 years. Deep Learning has also piqued my interest recently, that is how this project was born. I wanted to analyze pro matches to see if there was some kind of pattern to understand the game on another level.

Firstly, I wanted to analyze the draft, checking if the composition of the draft truly mattered when it came to the probability of winning said game. I rapidly realized that only using basic stats wasn’t going to allow me to find any pattern to discriminate between a winning and a loosing team. The results will be shown in a future section.

After the first approach, with the data I had I noticed it would be easier to predict the winning chances a team had when already inside a match, focusing on in-game stats and not only character stats. This approach is yet to be finished but it seems more logical to analyze the matches this way with the data I have as you will see.

1. **Data collection**

The first step for any Deep Learning project is having a dataset to analyze. I signed up to the Riot Games API, thinking there I would find stats for the pro matches, or at least for the Tier 1 leagues. To my surprise there was no API available for esports games. You had access to all the matchmaking games from all the regions, which could be useful for other projects, but it didn’t seem particularly useful for what I wanted to do.

After searching for a bit, I stumbled upon Oracles Elixir web page. They have csv files for every competitive season dating back to the 2014 season. In these files they have every match from every major and minor league, around 10k matches per file. Inside every file you have stats for each player in each match like ‘Damage Dealt’, ‘Dragon killed’, ‘total kills’ and a lot more stats. You can access these files from here:  
<https://oracleselixir.com/tools/downloads>  
I downloaded the file from 2023 as that competitive season had ended so the data wouldn’t change, and also downloaded the 2024 one to try the project on an ongoing season.

For the first approach, where I analyze the draft, I had to think about which stats mattered more when designing the draft. The first stat that came to mind was the damage split of the team or each champion’s damage split, but on the file we only had access to the percentage damage each champion dealt in the match they played. To calculate the damage split from each champion I coded a script which accessed [www.leagueofgraphs.com](http://www.leagueofgraphs.com) website. In this website you can check the stats for each champion separately, even checking their damage split. I got the damage split from each champion in all matches above master and saved it in a dictionary. After that I calculated the average percentage damage for each champion in all the matches played, so that we could multiply the percentage damage by the damage split to see, on average, the percentual damage split for each champion.

Another stat that could be useful when analyzing the draft would be how tanky a team would be. Taking the ‘damage mitigated’ stat we could define the tankiness of a team.

Out of all the stats in the files these were the only ones I thought were useful when designing a draft. However, this is a very lax approach, as we are not considering things like how a champion scales, meaning if it is better in the early game or late game, or the average range of the team. However, as a first approach I thought these stats would give us a rough idea of where we are starting from.

1. **Draft Analysis**

Now that we have some stats to analyze the drafts, I will explain the process I followed.