Tom Burns
CS 314
FINAL Project
ReadMe

I started this project with a somewhat open-ended and rather ambitious scope of what I would do for this project and where I could take it. I wanted to do a project that dealt with data science involving Dota 2, a popular MOBA game that rivals league of legends in terms of player base and audience sizes. Recently over the past year and a half or two, AI projects have come onto the scene and begun to do work in complex games. OpenAI, founded by Elon Musk, was the first company to do so and began to make AI agents that would play Dota 2 and learn how to improve. This project began in early 2016, and by August of last year, the project's AI models were playing at the game's largest tournament and beating professional players. In the past couple months, the company released a challenge where their AIs could be played against in a 5v5 setting by any group of players that felt they were up to the challenge. OpenAI's bots sported a win rate of about 99.4%. Not only is artificial intelligence progress happening in Dota, but Google's Deepmind is following in OpenAI's footsteps and taking up the game of Starcraft 2 and have managed to beat professional players in early 2019. This area of artificial intelligence research greatly interests me, and to do work and create a project using data science in games really appealed to me and was what served as my motivation for this final project. I wanted to use the imperative programming skills that we learned about python (which was lucky because that was the language used for OpenDota's API) and combine that with a more data science focused language like R and analyze some of my own data that I have created by playing Dota 2.

Starting from my original proposal, I realized that Valve's Dota 2 API, despite being open source and rather easy to use, did not provide anywhere near the amount of stats that I was looking for to analyze and utilize in a data science themed project. I was able to find an API by a site that deals with Dota data by parsing the matches that occur on Valve's servers as they happen and are made available to the public and to developers that are looking for data on Dota 2. The API I used was from a website called OpenDota. Their API page has documentation for how to use the data they scrape from public and professional matches and how it can be used for personal use. I wanted to pull my own match data through python, then be able to analyze the data I found in R so that whatever I found could be visualized and the tools available in R could provide more insight into the data.

My steam handle is Pobbish/Pob. Two examples of what my profile looks like for Dota 2 match data are DotaBuff and OpenDota. You can also see what kind of information is usually kept track of if you choose any random game or look at any of the tabs under my profile.

https://www.dotabuff.com/players/131711228

https://www.opendota.com/players/131711228

```
curl https://api.opendota.com/api/heroes > heroes.json
```

```
curl https://api.opendota.com/api/players/131711228/matches > my_matches.json
```

I first had to pull the information about the characters in the game into its own information file that would help with codifying and organizing the data from my own matches. To do this I used the above commands in the terminal in my project directory. The second link that is given is all of the match data from my account that I then stored in a json file. The json contained 4441 matches I had played since the start of my Dota 2 career in June of 2013. With over 5000 hours spent playing the game I figured that my match history would serve as a decent sample data set to analyze. Each match in the json contains a list of different match attributes relating only to my own play. I would have to parse the matches using their API for more detailed match information.

In the first python file, pullmatches.py, I use the my_matches json to find the surface level information about the matches that the site has about each of the games I have played and then pull the more detailed information about each game using OpenDota's API. I store this information in its own file. In order to make the API calls, I registered my own key with the site, since without one there is a rate limit as well as a call cap. I wanted to analyze information from all of the Dota 2 matches that I had ever played and in order to do that I would have to have an uncapped rate. Without the rate pulling data from the matches would take an obscene amount of time. I would have had to use a sleep(x) call in order to stay under the limit and the sleep would have to be called for each request call that is made using their API. The rate they charged me for this though was not that expensive for just retrieving the data through one run of my program. If you want to verify that it works, please use the rate limited version and uncomment the sleep line. I only had to retrieve the data by running the pullmatches.py program once. After that the data from the API calls from OpenDota's site were available locally on my machine.

Now with all of my match data available to me, I was able to analyze the data and make conclusions about the games I had played. I intended to do this with R, but with the way the data was formatted (being in a json file) I ran into difficulties. I found a few different ways to read json files into R for them to be analyzed, but the problem I then faced after I was able to do this was that the information was maintained as a list. This single list was actually a list of lists, and the way indexing data works in R meant that I could not even tabulate the data properly when it was in this form. I tried a few different ways of dealing with this, such as flatting the list or converting a multidimensional list into a data frame (which would be the easiest way of dealing with the data I want to use for this project). I could not find a way to create my own data frame from the json file without doing manual work, and there is no way that I could perform that many calculations to create the data frame by hand for over 4000 individual lists. Since I still wanted to try working with R again, I downloaded the data set I had found previously on Kaggle(https://www.kaggle.com/devinanzelmo/dota-2-matches - the files are way too large to include in submission file). I did a few simple things with that set but didn't end up coming back to it because I found manipulating my own data that I had pulled to be much more interesting. I tried a few more times to get my own match data working from the text file, including writing it to a json file and reading it into R as such but the problem persisted. I wanted to get the data working in R first and then figure out how I could incorporate my R program with my python program by either using a Jupyter notebook or rpy2 and most likely utilize numpy as well to assist with integration of data calculations. I don't have enough experience with R and json files specifically to figure out how to handle fixing the problems I faced so that I could have visual plots and calculations about my data. :(

Since I couldn't do much with the data in R, I decided to come back to python and try to analyze some of the matches there. I started by attempting to find my own win rate over the 4441 matches I had consolidated data from. I again started facing problems with json files and the way the data was indexed from what OpenDota's API call returned.

https://docs.opendota.com/#tag/matches%2Fpaths%2F~1matches~1%7Bmatch_id%7D%2Fget

Here you can see what is returned for each match, and what I stored in the file containing ALL of the match data (the detailed version) using the IDs from the surface level match data available on their site. I attempted to find which games containing my steam ID but I couldn't figure out the methodology for indexing as it kept returning an error. I feel like the API docs for OpenDota could be better written to explain exactly what is being returned and its type within Python. Instead I have to start with the API docs, figure out what I want to be using, and then search online for other people's answers to how it is actually utilized. I was able to analyze the heroes picked in all of my matches and calculate the pick rate of all heroes across my matches. I also counted my most played heroes against the matches that I had pulled to check if the processing was working properly. For some reason though, only 678 matches were being properly processed. I was never able to figure out why this wasn't working correctly, but I think that this could be because of how OpenDota stores match data. Since I have matches dating back to mid-2013, if they haven't maintained the same syntax for their storage of match data, the way I attempt to access it in my program now might throw errors if even one part of it is off.

Since I am going to be attending a research program over the summer in artificial intelligence and data science, I figured that this was a good project to start with. I plan to continue working on this project throughout the summer as well since this is a career path that I would like to pursue. If you have any comments on my project, pointers of where I could take this project next, or if you know any potential solutions to the problems I faced I would greatly appreciate any feedback. I know it was a lot to read but thanks for sticking it out. :)