William Ness & Derek Thomas
CSCI 4622 Final Project Progress Update

Predicting the Outcome of FIRST Robotics Competition Matches

1. This project is about using machine learning models to predict the outcome of FIRST Robotics Competition (FRC) matches. FRC each year is a game played by individual teams who come to competitions bringing robots to compete in that game, but matches are not 1v1, they are 3v3, and so each match contains six unique teams with six unique robots. Each competition has between 50-100 teams and the first 1-2 days is spent doing random 3v3 matches where 3 random teams play 3 other random teams. The last day is playoffs, with more 3v3 matches. Each match the teams try to score points, and the alliance with the highest score wins.
2. Each match for each year (the game changes each year) can be found on thebluealliance.com, who also provides a very easy to use API that we used to pull data from. Each match has many datapoints (~30) about what robots did during the matches that can be used as features for model training. There are around 15,000 total matches total for data to be used.
3. Yes, we have done some exploratory data analysis. The first thing that is important in data exploration and getting ready for model training in this scenario is to extract the features from the matches and aggregate them by team. You want to do this because for example, you could train your model on matches with teams (team1, team2, team3) vs (team4, team5, team6) and (team7, team8, team9) vs (team10, team11, team12), but there needs to be a way to predict a match with teams that have never faced each other before ex: (team1, team2, team3) vs (team10, team11, team12). Our initial plan for implementing this is to average each match datapoint for each match a team has, and storing those. Then to prep the data for model training/prediction, given any arbitrary match with six teams, you can simply append the lists of each teams averaged features to each other, and then the model's job would be to classify those features as a redWin = false (0) or a redWin = true (1). The training matches already have labels we can grab from the API. This figure is an example feature set that might help visualize what I'm trying to explain: (f1 = feature 1, f2 = feature 2)

bot1_f1	bot1_f2	bot2_f1	bot2_f2	bot3_f1	bot3_f2	bot4_f1	bot4_f2	bot5_f1	bot5_f2	bot6_f1	bot6_f2	red Win
8	12	6	3	2	8	1	8	4.3	2	3	9	0