Machine Learning Engineering Nanodegree

Capstone Proposal

Sam Strohkorb
October 10th, 2018

Domain Background

This project is based on the FIRST Robotics Competition (FRC) challenge give in 2016, FIRST Stronghold. FIRST is an organization that promotes STEM (Science Technology Engineering and Mathematics) to secondary education students around the world. I am using the data from FIRST Stronghold because I participated in this event on FRC Team #2410, the Metal Mustangs (http://mmr2410.com/). While I was on the Metal Mustangs, I did statistical analysis work to try to predict what teams would win a match. With the ability to predict who will win a match, if your team is slated to win, then you can relax, but if you are slated to lose, then you have time to adjust your strategy to try not to lose. I was partially successful on this venture, for I was only using excel and VBA at the time.

Now working on the capstone project for this course, I realized that I could use machine learning to predict what teams would win a match. Even though I have moved on from the Metal Mustangs, I still want to provide them and other FRC teams with the tools to be able to predict future outcomes. This is the culture of FIRST, to always be contributing, no matter how you are involved, as a student, parent, or mentor. But this could have not been possible without a white paper published on chiefdelphi.com, the most popular FRC forum. User 'wgardner' posted a great paper on FIRST statistics (link: https://www.chiefdelphi.com/forums/showthread.php?threadid=137451), which gave me great ideas on how to expand the dimensionality of the data without adding useless dimensions.

Problem Statement

I want to be able to predict what alliance (a grouping of three teams) will win a match for the FIRST competition FIRST Stronghold. FIRST organizes many regional and district competitions during the 6 week competition season in the spring of every year. Every competition has matches played, were two alliances of three teams each play to win points and ranking points. With every match, there is data collected on individual robots, as well as alliances as a whole. The ability of a model used to predict who will win a match can be easily compared with the actual outcomes (so, a supervised learning problem). This is very applicable to the future, for the same models can be used for future FIRST competitions, with only tweaks to the data input side.

Datasets and Inputs

The data for this project is publically accessible through the www.thebluealliance.com API (documentation link: https://www.thebluealliance.com/apidocs). I will use the API v3 for this project. There is a lot of data that is available through this API, but I will only be using a few of them. First, I will use '/events/{year}' to get all of the event tags (or event keys) for that year. This will allow me to use other API calls to get the data I need. Second, I will use '/event/{event_key}/rankings' to get all of the ranking data for a given event. This will be necessary to see get a few general statistics about the teams. Third, I will use '/event/{event_key}/matches' to get the match by match data for a given event. This will allow me to calculated specific statistics for each team at the event. Lastly, I will use '/event/{event_key}/predictions' to compare my models with the predictions FIRST has for the events.

The data obtained through this API is in a JSON format. This then can be converted to a dictionary within python by using the json module. There are number of data columns that contain more than one non-numerical value, so these columns will need to be one-hot encoded.

Solution Statement

A supervised learning algorithm using the match by match data with the match outcome as the target is the solution to this problem. It is easy to measure the performance of the model with a testing set and can be easily used on other problems as well.

Benchmark Model

The benchmark model for this project is the prediction that FIRST has for their competitions. FIRST releases predictions match by match for competitions for teams to use. I do not know how FIRST predicts the outcome of a match, but it is a dataset available. I will be able to compare the performance of my model to the performance of FIRST's model.

Evaluation Metrics

There are three evaluation metrics that I will use for this project, accuracy, fbeta score with beta = 0.5, and precision. The accuracy metric will let me know how accurate my model is compared to the actual outcomes in my training set. The fbeta score will give me a good indicator of the weighted average of recall and precision. The precision score will give me an idea of how precise the model is, because I care more about precision than recall.

Project Design

The first part of this project will be data acquisition and processing. I will need to use The Blue Alliance's API to get the data into python. There are few statistics that are recorded per team at a regional. Most of the recorded statistics are for a given alliance (a group of three teams) instead of each individual team. Take the score an alliance gets from a match for an example. FIRST doesn't record who scores, just that someone scored. Then for each team, they will have the total score the alliance got as their score. This problematic if you want to know who scored the points in a given match. This is where OPR and a number of other statistics that are referenced in the white paper in the Domain Background section. In short, OPR uses linear algebra to calculate a prediction of how many points and individual team scores per match. It is not perfect, but it is good enough. I also plan to calculate the standard deviation of the few statistics that are recorded per team per match and I plan to calculate the average of every statistic and see how far each team is away from the average.

After the data processing, I may do some unsupervised analysis. This can contain dimensionality reduction, clustering, or anything that helps organize the data in an unsupervised way. This could impact the type and use of supervised learning model.

After that, I will need to develop the supervised learning model. I will need to evaluate different kinds of supervised learning models and choose the best one for the project, by looking at the evaluation metrics I specified and training time.

After training the model with a training set of data, I will test it with a testing set of data. Then I can compare the model I've made to the model that FIRST has and do analysis of that. Additionally, if I notice any factors that are more predominant than others, I could remove them from the data before training and see how the model responds.

After the model training, if there is further data analysis to be done on the model or the data, it will occur here. Depending on the model type, it might be possible to extrude the factors the model used. This might be a point of comparison useful for analysis of why certain alliances win and why others lose.