T20 Cricket Predictor

Cricket as a sport is very technical and quite complicated, which is the reason why its not the most popular sport worldwide. It is a team sport and is a sport where the momentum of the team can sway towards either side in short amount of time making the sport really tough to predict.T20 cricket is a shortest format of the game which will be the format we will be using for our project. It is extremely exciting considering the fact that there are multiple parameters (features) that affect the output of the game. Some factors being the pitch(turf) conditions, the initial toss of the coin before the game to decide if the team is going to bowl or bat. The format of the game is such that the team batting first sets a score and the team batting second have to score 1 more run to win.

We will be applying supervised learning techniques to predict the result of the game given several features available after the completion of the 1st innings to decide if the team batting first is going to win the game or not. The inputs to our neural networks will likely include the number of runs gained, and wickets lost by the team that bowled in team 1. We'll be setting this up as a classification problem, using whether the team that bowls in the first inning wins the game as our target variable.

The data set we will be using has data for 6000 t20 cricket matches with the result. We found this dataset on Kaggle and the link is:

https://www.kaggle.com/imrankhan17/t20matches#t20_matches.csv

From this data set we will use features that are available after the first inning to predict the outcome of the game.. This data set appears to be large enough for us to clean, train, validate and test. We're planning to experiment with the numbers of layers, transfer functions, and numbers of nodes per layer in a multilayer perceptron network, and to experiment with more traditional classification models as well (ie. SVM, Random Forest, etc..) Our goal is to identify the best performing model, using cross-validation and multiple model performance metrics to strike a balance between the models accuracy and overfitting.

We're planning to use the Keras framework for implementing our neural network experiments, numpy and pandas for general data preprocessing, and sklearn for more traditional classification modeling techniques. Through this project we'll be relying heavily on the course materials, the Neural Network Design text book, and sklearn documentation. Any additional resources we find useful along the way, will certainly be mentioned in the final write-up.

Project Milestones Table:

Date	**Description**
7/25/19	*Group Proposal Due*
7/27/19	Data Ingestion Working Initial EDA Complete
8/03/19	Neural Network Experiments Started Traditional Models Completed
8/10/19	Neural Network Experiments Completed
8/17/19	Group Presentation Completed Group Final Report Completed
8/20/19	*Group Presentation*
8/20/19	*Group Individual Report*