Machine Learning Engineer Nanodegree

Capstone Proposal

Adam Bulow
March 1, 2019

Proposal

Domain Background

In this project, we will attempt to build a model that uses an MLB hitter's career statistics to predict whether or not he is a National Baseball Hall of Famer. The process for a player being inducted to the National Baseball Hall of Fame (the Hall) is a very complicated one, but at a high level it is as follows:

- All players who played for ten seasons or more are eligible to be inducted starting five years after their final game.
- Eligible players appear on a ballot, which is handed out to a committee composed of all Baseball Writers Association of America (BBWAA) members who have been an active member for at least ten years.
- If a player receives a vote on 75% of ballots or more, he is inducted into the Hall.

Notably, the criteria on which the committee votes is ill-defined, and highly subjective. According to the BBWAA Election Rules (https://baseballhall.org/hall-of-famers/rules/bbwaa-rules-for-election), the only voting criteria given is "Voting shall be based upon the player's record, playing ability, integrity, sportsmanship, character, and contributions to the team(s) on which the player played."

With such limited criteria, there are constant arguments and discussions amongst fans, voters and baseball men and women as to whether or not certain players belong in the Hall.

While integrity, sportsmanship and character are nearly impossible to ascertain from looking at box scores, it seems that a player's record, playing ability and contributions (which are by far the most crucial categories) can be analyzed quantitatively.

Problem Statement

The goal of this project is to see if it is possible to build a model that will take player career statistics as inputs and predict which players are Hall of Famers.

For simplicity, we will restrict our analysis for this project to hitters (that is, we will exclude players who were primarily pitchers). We can accomplish this by training a model on players who's Hall of Fame fate has already been decided, with their career statistics as the features and Hall induction status as the labels. The model will be a classification model that takes as input a player's career statistics and returns a simple Yes/No (1/0) as predicted Hall of Fame induction status. Since we will have labels for all of the players, it will be straightforward to calculate the accuracy of our model.

Datasets and Inputs

In order to accomplish this, we will utilize several publicly available datasets:

1: Fangraphs' career hitting data:
https://www.fangraphs.com/leaders.aspx

This dataset will contain the full career statistics of all MLB players. This will contain baseball statistics such as PA, HR, OBP, etc. These will form the majority of the features we will feed into our model.

2: Lahman's Baseball Database
http://www.seanlahman.com/baseball-archive/statistics/

This dataset contains more information about each player, such as which position they primarily played, which years they played in and how many All-Star game appearances each player had. These will also be features for our model.

3: Baseball Reference's Hall of Fame Inductees Table
https://www.baseball-reference.com/awards/hof.shtml

This is a dataset containing all the players who have been inducted into the Hall. All players in this table will receive a label of 'Inducted' or 1, while all other players will receive a label of 'Not Inducted' or 0.

4: MLB Rosetta
https://github.com/geoffharcourt/mlb_rosetta

This table is simply a mapping between player id's across these different data sources (in addition to other baseball sites). We will use this table as a reference so that we can join information all of our datasets together.

Solution Statement

The solution to the problem will be creating a classifier that takes each player's career batting statistics as features and outputs a predicted Hall of Fame status: Yes/No (1/0). To do this, we will use the datasets listed above to creating a dataset that we can use to build our model. We will first start by filtering down to only Hall eligible players who have already been determined to be in the

Hall or not be in the Hall. Then we will split this data into training, testing and cross-validation sets. We will then attempt to train several different types of classification models on this data, tuning hyperparameters as necessary, to achieve a model that has a high accuracy as well as F1-score.

Benchmark Model

As a benchmark model, we can compare our results to a simple/naive model that classifies every player as not a hall of famer (0). Since the bar for induction is very high, and only roughly 15% of eligible players are ultimately inducted, this model would already have an accuracy of around 85%. However, this model would have a Precision of 100%, a Recall of 0%, and thus a an F1-score of 0, since it simply classifies all players as negative.

Evaluation Metrics

Two evaluation metrics we can use to evaluate our model are accuracy and F1-score. This will allow us to compare our classification results to our simple benchmark model. Ultimately, what we want to know is how often our model makes the correct Hall of Fame induction prediction, and accuracy will let us know that. The F1-score will also be useful to balance the Precision and Recall of our model, and will distinguish our model as hopefully being far better than our simple benchmark model.

Project Design

In order to complete this project we will first start by constructing a dataset which we will use to train, validate and test our classification model. To do this, we will take the Fangraphs career hitting database, which will provide the players and their career hitting statistics, which will be the main features for the model. We will then join in information from the Lahman database to get each player's years they played in as well as number of All Star appearances, which will also be features we can use for our model. We will then filter to only Hall eligible players who played their last game well enough into the past such that their Hall of Fame fate has already been decided. Finally, we will use the Baseball Reference Hall of Fame Inductees table to create labels for the players in our dataset, with all players in this table receiving a label of 1, and all other players receiving a label of 0.

Once we have this dataset, we can conduct some exploratory analysis of this dataset, including finding out what percentage of these players are in the Hall of Fame. Since we are employing a simple benchmark model of classifying each player as negative, this number will give us 1 - accuracy for our benchmark model. We can also try to identify some trends in the data, looking for things

such as “All players with over 3000 hits are in the Hall of Fame” or “Every player in the Hall of Fame have at least 100 HR”.

After this exploratory analysis, we can begin to construct our model. We will first split our data into a training set, a cross-validation set, and a testing set. We will then try using several different classification algorithms, tuning hyperparameters, and try to come up with the best model we can that maximizes accuracy. The algorithms we will consider are 1) Decision Trees/Random Forests, 2) Logistic Regression, 3) Support Vector Machines, and 4) Neural Networks. Once we have constructed models using each of these algorithms, we will weigh accuracy, F1-score, training time, and interpretability when selecting which model is the best for solving this problem. We will then conduct and in-depth analysis of our selected model, detailing its performance, evaluating its strengths and weaknesses, and attempting to reason about why it gets predictions for certain players right and others wrong if there are any identifiable patterns.