Machine Learning Engineer Nanodegree

Andrew Doyle
August 13, 2017
Capstone Proposal

Domain Background

For my Capstone project I will be applying several different unsupervised learning algorithms to Major League Baseball data, specifically clustering algorithms, in order to properly classify each pitch into its type (i.e. Four-Seam Fastball, Curveball, etc.). There are several reasons I chose the domain of baseball for my final project. First and foremost, I am a big fan of baseball, particularly the analytical and quantitative side of the game. It was this interest that lead me to the field of machine learning in the first place, and even helped me learn some machine learning and statistical software (Marchi & Albert, 2013). Statistical analysis is in the DNA of the game of baseball, as game data has been collected for as long as the game has been around. Because of the vast amount of data, the field lends itself very well toward machine learning.

Secondly, with the advent of the Statcast system, recently collected data can offer insights into the game that were previous unavailable. Per the website Baseball Savant (https://baseballsavant.mlb.com/about), Statcast

> is a state-of-the-art tracking technology, capable of measuring previously unquantifiable aspects of the game. Set up in all 30 Major League ballparks, Statcast collects data using a series of high-resolution optical cameras along with radar equipment. The technology precisely tracks the location and movements of the ball and every player on the field, resulting in an unparalleled amount of information covering everything from the pitcher to the batter to baserunners and defensive players.

Finally, the datasets for MLB data are both large and free to use by anyone, making it very conducive toward this type of analysis. Machine learning has been applied to baseball data both publicly among the baseball community for pleasure and privately among professional teams with the goal of gaining a competitive advantage. An example of using this field of research on a Statcast dataset would be using a Random Forest classifier to predict whether a batted ball will be a hit or an out (http://www.hardballtimes.com/using-statcast-data-to-predict-hits/).

Problem Statement

The problem I will be attempting to solve is “can unsupervised Machine Learning algorithms correctly classify pitches thrown by Major League Pitchers into their proper pitch type?” Using Statcast data, consisting of perceived velocity, spin rate, horizontal movement, and vertical movement I will apply unsupervised clustering algorithms in my effort to accomplish this. Statcast tracks the above features for every pitch that is thrown in major league baseball, as well as the result of each pitch and numerous other metrics. The metric I will be attempting to identify is the resulting label of the pitch type.

Datasets and Inputs

Statcast tracks many different features of each pitch thrown in a major league baseball game, with all data being free to use and publicly available via www.BaseballSavant.com. The impetus for the project came from watching the game - pitch recognition is a vital aspect of being a successful hitter. This recognition is based on some combination of the velocity of the pitch, the spin direction and rate, and horizontal and vertical movement (known as the break).

Solution Statement

The feature I will be trying to identify is pitch type, which is identified along with each other feature that Statcast tracks.

In order to accomplish this I will use several different clustering algorithms on the pitch data. Clustering "is the task of grouping a set of objects in such a way that objects in the same group (called a cluster) are more similar (in some sense or another) to each other than to those in other groups (clusters). It is a main task of exploratory data mining, and a common technique for statistical data analysis" (https://en.wikipedia.org/wiki/Cluster_analysis). I've decided to use clustering as a way of labeling each pitch thrown because the data lends itself well to this sort of analysis, namely that these large datasets exist with labels attached to each record that contains multiple features for the clustering algorithms to use.

Specifically, the two clustering algorithms I will use are K-Means and a Gaussian Mixture model. I will review each individually in detail.

Per the scikit-learn site (http://scikit-learn.org/stable/modules/clustering.html#k-means), the K-Means clustering algorithm

> clusters data by trying to separate samples in n groups of equal variance, minimizing a criterion known as the inertia or within-cluster sum-of-squares. This algorithm requires the number of clusters to be specified. It scales well to large number of samples and has been used across a large range of application areas in many different fields.
>
> The k-means algorithm divides a set of N samples X into K disjoint clusters C, each described by the mean μ_j of the samples in the cluster. The means are commonly called the cluster "centroids"; note that they are not, in general, points from X, although they live in the same space. The K-means algorithm aims to choose centroids that minimise the inertia, or within-cluster sum of squared criterion:

$$\sum_{i=0}^{n}\min_{\mu_j \in C}(||x_j - \mu_i||^2)$$

A Gaussian Mixture model

> is a probabilistic model that assumes all the data points are generated from a mixture of a finite number of Gaussian distributions with unknown parameters. One can think of mixture models

as generalizing k-means clustering to incorporate information about the covariance structure of the data as well as the centers of the latent Gaussians.

Benchmark Model

In this case, the benchmark model is the actual pitch type. The results are recorded for every pitch thrown, so I can easily separate the data set into the features and the variable I am attempting to predict. I can then use that variable as a method for evaluating the classification made by the algorithm.

In the researching this project, the only other possible benchmark was a blog post conducting the same sort of cluster analysis on pitch data (https://baseballwithr.wordpress.com/2015/02/22/pitch-classification-with-k-means-clustering/). This analysis would not be identical however, as the author uses different software and a different dataset. It can be used as a reference however when conducting the analysis, even if the work done is slightly different.

Evaluation Metrics

Because the label of each pitch is already known, I can use the pitch type as a way to evaluate the accuracy of each cluster created.

Project Design

The initial step in the process is to download the data set. The data set will consist of a single major league pitcher and every pitch that pitcher have thrown in the 2016 and 2017 seasons, along with all of the features measured by Statcast. The reason for using a single pitcher is because no two pitchers are exactly alike. The curveball of one pitcher may be significantly different, in terms of velocity and movement, from that of another pitcher. The pitch repertoire of a single pitcher will be much more consistent.

Once the dataset is complete the data needs to be cleaned and preprocessed. This consists of extracting only those features which are relevant to the clustering algorithm. Those features are pitch type (fastball, curveball, etc.), which is the label that will be used to evaluate each sluter, release speed (in miles-per-hour), the horizontal and vertical movement of the pitch, and the spin rate. Once the relevant features have been extracted the values need to be converted to a format that can be used by the algorithm. This is composed of 2 steps.

The first step is to view how each feature is distributed. If any particular feature is heavily skewed or includes outliers those will need to be addressed. The second step is to identify whether principal component analysis (PCA) is needed in order to perform the analysis. PCA is used to reduce the dimensionality of the dataset by weighing each feature by its variance.

Once the data has been preprocessed, the clusters can then be created. After the clusters have been created, the labels created by the cluster can be compared to the actual label of each pitch to view the accuracy of the labeling. Any pitches which are mislabeled will be inspected further for the purposes of edification.