Final_Draft

Mike McGunagle

April 28, 2016

Throw Them Heat Ricky! Guessing the next pitch in baseball.

Abstract:

Through the use of machine learning, we can improve the guessing of what the next pitch a pitcher will throw in the game of baseball. This study is concerned with a select number of pitchers who pitched in both the 2014 and 2015 seasons, and threw at a high level (leaders in ERA or wins, such as Madison Bumgarner, and Corey Kluber). By using advanced algorithms, with statsical analsysis we can see if an improvement was made using machine learning techniques. Using such techniques as random forests, k-nearest neighbor (kNN), and Naïve Bayes, we can see if any of these algorithms can improve pitch guessing.

##Introduction: ###Previous work: Hitting is a fundamental of baseball. If a batter knows what movement the ball will have coming at him, he will be more likely to make contact with the ball, and potentially get on base, or even score a run. By understanding which pitch is thrown, and when, a lot of the guess is taken out of the game. In 2016 Major League Baseball will start allowing iPads into dugouts for the first time (Chew, 2016). With the increase in technology, teams will have the opportunity to preload stats and other information, that will be at their fingertips. Imagine if a team had a matrix preloaded that stated what a pitcher would do on a 2-2 count, in the second inning, with a runner on second, left handed bat at the plate, a tied game, with one out. The team would have an advantage to know a curveball might be coming their way. Currently batters position themselves into hitters counts, and pitchers count. We know that when there are more balls then strikes, (a hitter's count) the pitcher is more likely to throw a fastball, while when its reverse the count is more strikes than balls (say 0-2), the hitter is more likely to see something with more movement, such as a curve ball or slider.

There has been a fair amount of work done on this question, but not too many actual research papers. There are a lot of blogs about using advanced analytics to capture what is happening in baseball, but little scientific research into the actual guessing of the next pitch. In March 2012, Gartheeban and some colleagues proposed an idea that Support Vector Machine (SVM) learning could increase the ability to guess what the next pitch over a naïve Bayes model. Gartheeban's model increased pitcher prediction from 59.5% in naïve, to 70% in the SVM model (Gartheeban, 2012). In March 2012 Attarian presented a paper on pitch sequencing using k Nearest Neighbor (kNN) using Manhattan distance. He was able to improve his results by 4% over naïve Bayes. In a March 2015 paper for the journal of sports, Brock used Linear Regression Analysis and SVM to predict what a pitchers pitch would be in a certain situation. This study analyzed pitcher predictability in three general pitch count situations: (1) when the batter is ahead (more balls than strikes); (2) behind (more strikes than balls); and (3) the pitch count is even. The data were partitioned accordingly, and three predictive models for each of the four pitches were developed and evaluated. (Brock, 2015).

Based on the previous studies, this paper will look at several factors that affect a pitcher's pitch selection. This paper will look at count, inning, batter handedness (batting right or left), previous play outcome, and previous pitch. I will use random forests and k-NN and compare those results with a naïve Bayes model for pitch prediction.

Methods:

Data Extraction

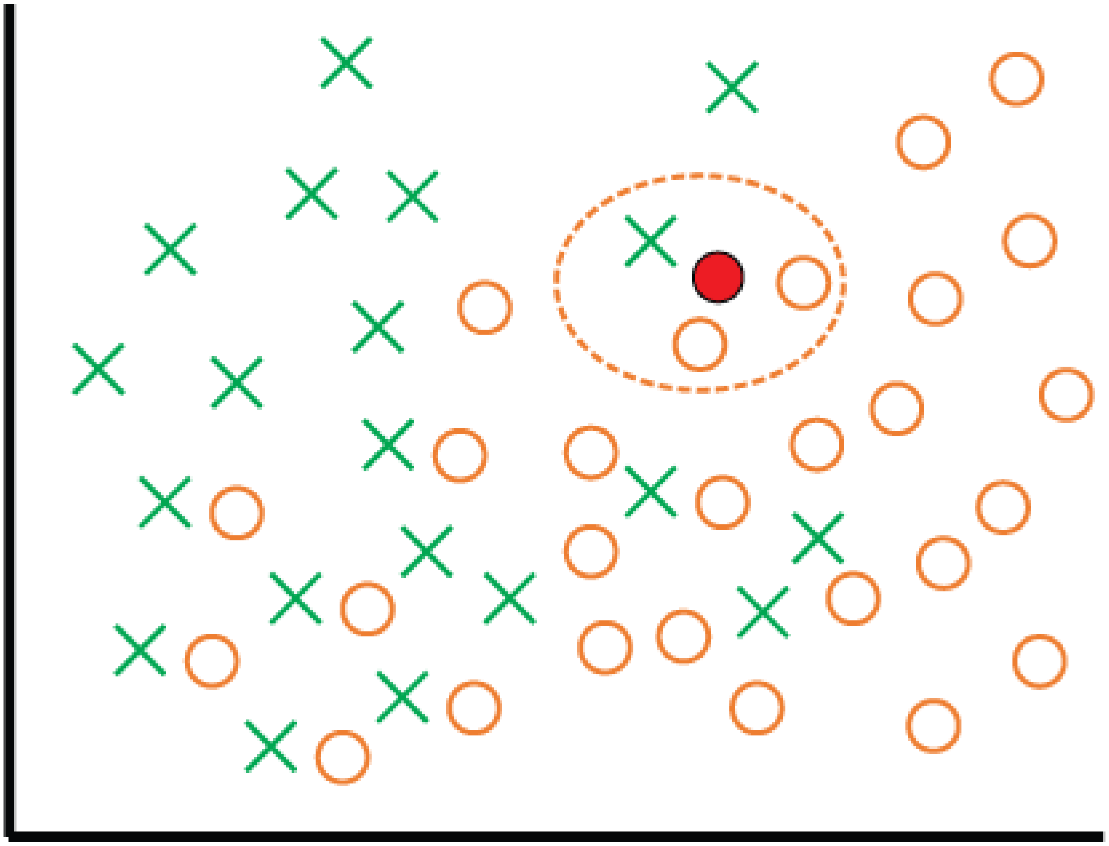
In order to procure my data, I used R programming software, and the pitchRx package. I was able to download the data for Major League Baseball gameday server, known as pitchF/x, which was established in

2008. I downloaded all data for 2014 and 2015. This data contained for tables, two of which I'll be using to query my data on. The pitches data, includes pitcher name, handedness, pitch start speed, angle of pitch, starting location of pitch, ending location of pitch, gamedaylink (primary key with event num), pitcher number, and a few other things as well. The other set of data included information about the batter. This information included (but was not limited to) batter name, batter handedness, batting against pitcher, at bat description, balls, strikes, and outs. The data was downloaded written into weekly csv's (my computer didn't have the memory to take the entire season at once), then each week was appended to the previous week. The separate tables were then joined on their primary keys (gameday, num).

Algorithms

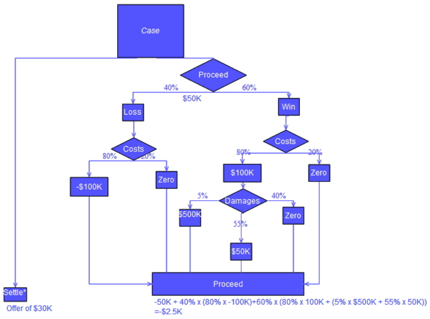
kNN

k-Nearest Neighbor, is an algorithm, that plots out points on a x/y axis. As you pick a point, the algorithm picks a preselected amount of nearest points, to that event, and codes it to the majority of its closest neighbors. After some research I found the package knncat, that can handle categorical data very well.



Random Forests

Random Forests is a decision tree approach to a categorical situation. If it's sunny, do we play outside, but only if it's over 65 degrees, and only if it is not too windy.



Decision trees are good at using categories to split up the data, and decide if one thing happens, then there is a likely chance another might happen.

Naive Bayes

Naïve Bayes, is the standard Bayesian algorithm that the other algorithms I mentioned are loosely based off. Bayes is a more general way at looking at things. Naive Bayes is a simple technique for constructing classifiers: models that assign class labels to problem instances, represented as vectors of feature values, where the class labels are drawn from some finite set. It is not a single algorithm for training such classifiers, but a family of algorithms based on a common principle: all naive Bayes classifiers assume that the value of a particular feature is independent of the value of any other feature, given the class variable. For example, a fruit may be considered to be an apple if it is red, round, and about 10 cm in diameter. A naive Bayes classifier considers each of these features to contribute independently to the probability that this fruit is an apple, regardless of any possible correlations between the color, roundness and diameter features. -Naïve Bayes, Wikipedia April 12, 2016

Results:

Results for Corey Kluber and Madison Bumgarner on knncat

```
#Corey Kluber pitch prediction table
kluber_knncat$table;
```

```
##           Reference
## Prediction  SI  SL  CU  CH  FF  IN
##         SI   0   0   0   0   0   0
##         SL  18  60  45   0  48  94
##         CU   0   0   0   0   0   0
##         CH   0   0   0  10   0   0
##         FF  92 318 459   0 838 520
##         IN  35  85  94   0  93 191
```

```
#Madison Bumgarner's pitch prediction table
MadBum_knncat$table
```

```
##           Reference
## Prediction  FF  SL  CU  CH  FT  IN
##         FF   0   2   1   0   0   3
##         SL   5  93  47  11   0  74
##         CU  16 160 438  81   0 330
##         CH   1  16  57  12   0  48
##         FT   0   0   0   0   4   0
##         IN  37 259 688 116   0 501
```

```
knn_cat_result
```

```
##          Corey Kluber Madison Bumgarner
## Accuracy 3.663333e-01         0.3493333
## P-Value  2.009540e-06         1.0000000
```

Results for Corey Kluber Random Forest test

```
#Corey Kluber pitch prediction table

kluber_RF$table
```

```
##           Reference
## Prediction  SI  SL  CU  CH  FF  IN
##         SI   9  10   0   0  99  27
##         SL   3 110   3   0 254  93
##         CU   4  60   4   0 420 110
##         CH   0   0   0   9   1   0
##         FF   3  68   5   0 760 143
##         IN   4  93   2   0 481 225
```

```
#Madison Bumgarner pitch prediction
MadBum_RF$table
```

```
##           Reference
## Prediction  FF  SL  CU  CH  FT  IN
##         FF   2   3   7   1   0  46
##         SL  13 100 119  34   0 264
##         CU  29  38 417 121   0 626
##         CH   8  10  68  14   0 120
##         FT   0   0   0   0   4   0
##         IN  37  78 222  54   0 565
```

```
RF_result
```

```
##          Corey Kluber Madison Bumgarner
## Accuracy    0.3723333         0.3673333
## P-Value     1.0000000         1.0000000
```

Results for Naïve Bayes

```
#Corey Kluber pitch prediction table

kluber_nb$table
```

```
##           Reference
## Prediction  SI  SL  CU  CH  FF  IN
##         SI   4   9   2   0  46  45
##         SL  12  53   9   2 199 113
##         CU   0   0   0   0   8   4
##         CH   1   1   0   0   3   0
##         FF  79 319  44   3 959 507
##         IN  21  88   8   3 304 150
```

```
#Madison Bumgarner pitch prediction
MadBum_nb$table
```

```
##           Reference
## Prediction  FF  SL  CU  CH  FT  IN
##         FF  19  29  51  28   0  64
##         SL  11  17  39  31   1  52
##         CU  43  81 152 118   2 201
##         CH  11  30  63  43   0  77
##         FT   0   1   0   2   0   0
##         IN 142 251 469 305   4 661
```

```
nb_result
```

```
##          Corey Kluber Madison Bumgarner
## Accuracy    0.3891856         0.2975317
## P-Value     1.0000000         1.0000000
```

Conclusion:

As we look at the confusion matrices produced for each pitcher we see that we have some mixed results. The random forest evaluation gave us the best results (Kluber 37.5%, Bumgarner 37%), however the P-values were not significant. The k-nearest neighbors model was slightly less successful at predicting pitches for Corey Kluber, but it was significant, although it was worse at both levels of accuracy and P-value for Madison Bumgarner. The naïve bayes model had a slight improvement of accuracy for Corey Kluber (38.9% to 37.5% for random forest), but was less accurate for Madison Bumgarner (29.8%), than either of his two other algorithms.

This study shows only a weak correlation between the factors of pitch type and inning, outs, count, batter stance, and previous pitch. Future studies should include more dependent variables, such as runners on base, runners in scoring position, day or night games, catcher-pitcher combo, hitter's batting average, and hitter's slugging percentage.

References

1. Chew, Jonathon, (2016). "Apple Just Struck a Huge Deal with Major League Baseball" March 30, 2016, fortune.com, http://fortune.com/2016/03/30/mlb-apple-ipad-pro/

2. Gartheeban G, et.al, (2012). "Predicting the Next Pitch", MIT Sloan Sports Analytics Conference 2012 http://theebgar.net/wp-content/uploads/2011/04/Predict-the-next-pitch.pdf

3. Attarian, A, (2013). "A Comparison of Feature Selection and Classification Algorithms in Identifying Baseball Pitches", Proceedings of the International MultiConference of Engineers and Computer Scientists 2013 Vol I,, March 2013 http://www.iaeng.org/publication/IMECS2013/IMECS2013_pp263-268.pdf

4. Bock, Joel R.(2015) "Pitch Sequence Complexity and Long-Term Pitcher Performance" Journal of Sports, March 2, 2015

5. Anonymous, "Naïve Bayes" Wikipedia https://en.wikipedia.org/wiki/Naive_Bayes_classifier April 11, 2016.

Appendix

Pitch Type Definitions FA = fastball FF = four-seam fastball FT = two-seam fastball FC = fastball (cutter) FS / SI / SF = fastball (sinker, split-fingered) SL = slider CH = changeup CB / CU = curveball KC = knuckle-curve KN = knuckleball EP = eephus UN / XX = unidentified PO / FO = pitch out