A Dynamic Feature Selection Based LDA Approach to Baseball Pitch Prediction

Chapter · November 2015

DOI: 10.1007/978-3-319-25660-3_11

Phuong Hoang
CareerBuilder

Hien T. Tran
North Carolina State University

A Dynamic Feature Selection Based LDA Approach to Baseball Pitch Prediction

Phuong Hoang[1(✉)], Michael Hamilton[2], Joseph Murray[3], Corey Stafford[2],
and Hien Tran[1]

[1] North Carolina State University, Raleigh, NC 27695, USA
{phoang,tran}@ncsu.edu
[2] Columbia University, New York, NY 10027, USA
jmurray@zerofox.com, {mh346,css2165}@columbia.edu
[3] ZeroFOX, Baltimore, MD 21230, USA

Abstract. Baseball, which is one of the most popular sports in the world, has a uniquely discrete gameplay structure. This stop-and-go style of play creates a natural ability for fans and observers to record information about the game in progress, resulting in a wealth of data that is available for analysis. Major League Baseball (MLB), the professional baseball league in the US and Canada, uses a system known as PITCHf/x to record information about every individual pitch that is thrown in league play. We extend the classification to pitch prediction (fastball or nonfastball) by restricting our analysis to pre-pitch features. By performing significant feature analysis and introducing a novel approach for feature selection, moderate improvement over published results is achieved.

Keywords: Machine learning · Hypothesis testing · Feature selection · Pitch prediction · PITCHf/x · MLB · LDA · ROC

1 Introduction

One area of statistical analysis of baseball that has gained attention in the last decade is pitch analysis. Studying pitch performance allows baseball teams to develop more successful pitching routines and batting strategies. To aid this study, baseball pitch data produced by the PITCHf/x system is now widely available for both public and private use. PITCHf/x is a pitch tracking system that allows measurements to be recorded and associated with every pitch thrown in Major League Beaseball (MLB) games. The system, which was installed in every MLB stadium circa 2006, records useful information for every pitch thrown in a game such as the initial velocity, plate velocity, release point, spin angle, spin rate, and pitch type (e.g., fastball, curveball, changeup, knuckleball, etc.). The pitch type is a value reported by PITCHf/x using a proprietary classification algorithm. Because of the large number of MLB games (2430) in a season and the high number of pitches thrown in a game (an average of 146 pitches), PITCHf/x

X.-L. Li et al. (Eds.): PAKDD 2015, LNCS 9441, pp. 125–137, 2015.
DOI: 10.1007/978-3-319-25660-3_11

system provides a rich data set on which to train and evaluate methodologies for pitch classification and prediction. The pitch analysis can either be performed using the measurements provided by PITCHf/x in their raw forms or using features derived from the raw data. Because each of the recorded pitches has a pre-assigned pitch classification provided with measurement data, a comparison between the proprietary PITCHf/x classification algorithm, which is assumed to generally represent truth, and other classification methods is possible. For example, in [2,3], several classification algorithms including support vector machine (SVM) and Bayesian classifiers were used to classify pitch types based on features derived from PITCHf/x data. The authors evaluated classification algorithms both on accuracy, as compared to the truth classes provided by PITCHf/x, and speed. In addition, linear discrimination analysis and principal component analysis were used to evaluate feature dimension reduction useful for classification. The pitch classification was evaluated using a set of pitchers' data from the 2011 MLB regular season. Another important area of ongoing research is pitch prediction, which could have significant real-world applications and potentially provides MLB managers with the statistical competitive edge to make crucial decisions during the game. One example of research on this topic is the work by [7], who use a linear support vector machine (SVM) to perform binary (*fastball* vs. *nonfastball*) classification on pitches of unknown type. The SVM is trained on PITCHf/x data from pitches thrown in 2008 and tested on data from 2009. Across all pitchers, an average prediction accuracy of roughly 70 percent is obtained, though pitcher-specific accuracies vary.

In this paper, we provide a machine learning approach to pitch predictions, using linear discriminant analysis (LDA) to predict binary pitch types (*fastball* vs. *nonfastball*). A novel and distinct feature of our approach is the introduction of an adaptive strategy to features selection to mimic portions of pitchers' cognition. This allows the machine learning algorithm to select different sets of features (depending on different situations) to train the classifiers. Features that are used contain not only original features but also hybrid features (that are created to better resemble the way of *data processing* by pitchers). Finally, cross validation is implemented to detect and avoid overfitting in our predictions. Overall, the prediction accuracy has been significantly improved by approximately 8 % from results published in [7]. A report of our initial effort in this study can be found in [8].

It is noted that the proposed methodology can be applied for pitch prediction of various pitch types (other than fastball or nonfastball). However, for other pitch types (curveball, slider, knuckleball, etc.) the data set is not sufficiently large to carry out the analysis. Hence, in this paper, we only study pitch prediction for fastball versus nonfastball.

This paper is organized as follows. Section 2 describes pitch data that are used in this study. In Sect. 3, we present our adaptive approach to feature selection and dimension reduction. Pitch prediction results are presented in Sect. 5 and conclusion is presented in Sect. 6.

2 Pitch Data

In this study, we used pitch data created by Sportvision's PITCHf/x pitch tracking system. We conduct our analysis over the period of 5 seasons from 2008 to 2012 with over 3.5 millions observations (pitches); each contains about 50 features (quantitative and qualitative). We only use 18 features from the raw data (see Table 1), and create additional features that we believed to be more relevant to pitch prediction. The reason is simply that classification uses post-pitch information about a pitch to determine which type it is, whereas prediction uses pre-pitch information to classify its type. We may use features like pitch speed and curve angle of that pitch to determine whether or not it was a fastball. These features are not available pre-pitch; in that case we use information about prior results from the same scenario to judge which pitch can be expected. Some created features are: the percentage of fastballs thrown in the previous inning,

Table 1. Description of original attributes selected for pitch prediction

No	Variable	Description
1	*atbat_num*	number of pitchers recorded against a specific batter
2	*outs*	number of outs during an at-bat
3	*batter*	batter's unique identification number
4	*pitcher*	pitcher's unique identification number
5	*stand*	dominant hand of batter; left/right
6	*p_throws*	pitching hand of pitcher; left/right
7	*des*	out come of one pitch from pitcher's perspective; ball/strike/foul/in-play, etc.
8	*event*	outcome of at-bat from batter's perspective; ground-out/double/single/walk, etc.
9	*pitch_type*	classification of pitch type; FF= Four-seam Fastball, SL = Slider, etc.,
10	*sv_id*	date/time stamp of the pitch; YYMMDD_ hhmmss
11	*start_speed*	pitch speed, miles per hour
12	*px*	horizontal distance of the pitch from the home plate
13	*pz*	vertical distance, of the pitch from the home plate
14	*on_first*	binary column; display 1 if runner on first, 0 otherwise
15	*on_second*	binary column; display 1 if runner on second, 0 otherwise
16	*on_third*	binary column; display 1 if runner on third, 0 otherwise.
17	*type_confidence*	likelihood of the pitch type being correctly classified
18	*ball_strike*	display either ball or strike

the velocity of the previous pitch, strike result percentage of previous pitch, and current game count (score). For a full list of features that were used in our study, see Appendix.

3 Adaptive Feature Selection

A key difference between our approach and former research of [7] is the feature selection methodology. Rather than using one static set of optimal features (for example, [7]), an adaptive set of features is used for each pitcher/count pair. This allows the algorithm to adapt to achieve the best prediction performance result as possible on each pitcher/count pair of data.

In baseball there are a number of factors that influence the pitcher's decision (consciously or unconsciously). For example, one pitcher may not like to throw curveballs during the daytime because the increased visibility makes them easier to spot; however, another pitcher may not make his pitching decisions based on the time of the game. In order to maximize accuracy of a prediction model, one must try to accommodate each of these factors. For example, a pitcher may have particularly good control of a certain pitch and thus favors that pitch, but how can one create a feature to represent its favorability? One could, for example, create a feature that measures the pitcher's success with a pitch since the beginning of the season, or the previous game, or even the previous batter faced. Which features would best capture the true effect of his preference for that pitch? The answer is that each of these approaches may be best in different situations, so they all must be considered for best accuracy. Pitchers have different dominant pitches, strategies and experiences; in order to maximize accuracy our model must be adaptable to various pitching situations.

Of course, simply adding many features to our model is not necessarily the best choice because one might run into issues with curse of dimensionality. In addition, some features might not be relevant to the pitch prediction. Our approach is to change the problem of predicting a pitch into predicting a pitch for each given pitcher in a given count. Count, which gives the number of balls and strikes in a at bat situation, has a significant effect on the pitcher/batter relationship. For example, study by [10] showed that average slugging percentage (a weighted measure of the on-base frequency of a batter) is significantly lower in counts that favor the pitcher; however, for neutral counts or counts that favor the batter, there is no significant difference in average slugging percentage. In addition, [7] concluded that pitcher are much more predictable when there are more balls than strikes. These studies showed that count is an important factor in making pitch prediction. In order to maximize accuracy in pitch prediction, in our study we took an additional step by choosing (for each pitcher/count pair) a most relevant pool of features from the entire available set. This allows us to maintain our adaptive strategy while controlling dimensionality.

3.1 Implementation

As discussed above, our feature selection algorithm is adaptive; that is, finding a good set of features for each pitcher/count situation. The implementation of this adaptive strategy mainly consists of the following 3 steps.

1. First, we select a subset of features (18) from the raw data and create additional features (59) from the data that are deemed more relevant to pitch prediction. This set of 76 features is further divided into 6 groups of similar features. The number of features from each group varies from as small as 6 to 22 features (see the full list in the Appendix).
2. Second, we then compute the receiver operating characteristic (ROC) curve for each group of features, then select the most useful features for pitch prediction. In practice, selecting only the best feature provides worse prediction than selecting the best two or three features. Hence, at this stage, the size of each group is reduced from 6–22 features to 1–10 features.
3. Third, we remove all redundant features from our final feature set. From our grouping, features are taken based on their relative strength. There is the possibility that a group of features might not have good predictive power. In those instances, we want to prune them from our feature set before we begin to predict. The resulting set of features is pruned by conducting hypothesis testing to measure significance of each feature at the $\alpha = .01$ level.

The above 3 steps in our adaptive feature selection approach is summarized and depicted in Fig. 1.

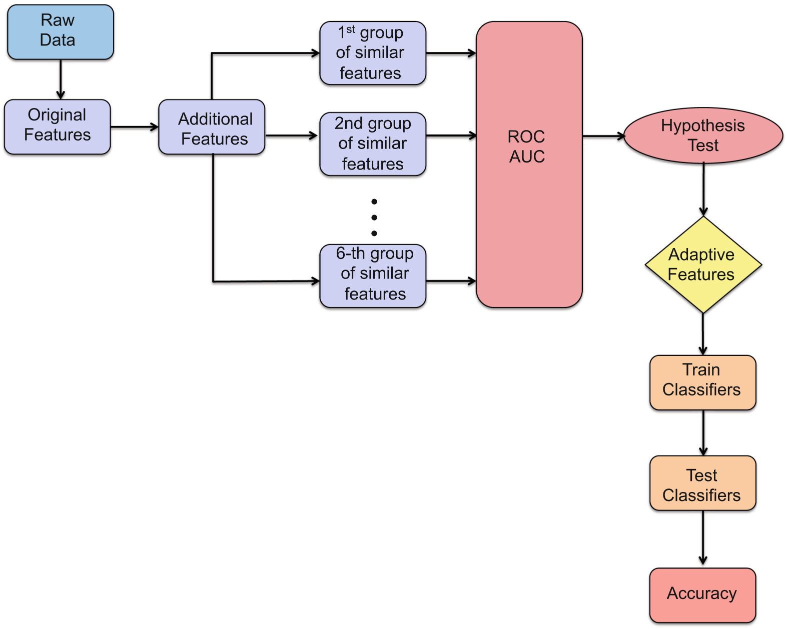

Fig. 1. Schematic diagram of the proposed adaptive features selection.

3.2 ROC Curves

Receiver operating characteristic (ROC) curves are two-dimensional graphs that are commonly used to visualize and select classifiers based on their performance [6]. They have been used in many applications including signal detection theory [5] and diagnostic tools in clinical medicine [11,12]. It is noted that a common method to determine the performance of classifiers is to calculate the area under the ROC curve, often denoted by AUC [4]. An example of a ROC curve using data from PITCHf/x pitch tracking system is depicted in Fig. 2.

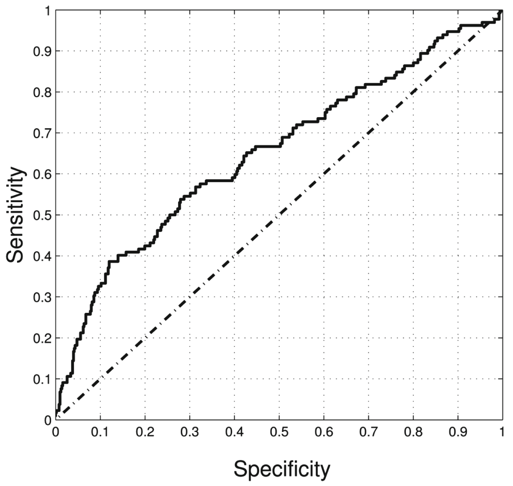

Fig. 2. ROC curve for a specific feature selected for pitch prediction

In this study, ROC curves are used for each individual feature in order to measure how useful a feature is for the pitch prediction. We calculate this by measuring the area between the single feature ROC curve and the diagonal line represents the strategy of random guessing. This value of area quantifies how much better the feature is at distinguishing the two classes, compared to random guessing. Because the area under the diagonal line from (0,0) to (1,1) is 0.5, these area values are in the range of [0, 0.5), where a value of 0 represents no improvement over random guessing and 0.5 would represent perfect distinction between both classes.

3.3 Hypothesis Testing

The ability of a feature to distinguish between two classes can be verified by using a hypothesis test. Given any feature f, we compare μ_1 and μ_2, the mean values of f in Class 1 (*fastballs*) and Class 2 (*nonfastballs*), respectively. Then we consider

$$H_0 : \mu_1 = \mu_2, \tag{1}$$

$$H_A : \mu_1 \neq \mu_2, \tag{2}$$

and conduct a hypothesis test using the student's t distribution. We compare the p-value of the test against a significance level of $\alpha = .01$. When the p-value is less than α, we reject the null hypothesis and conclude that the studied feature means are different for each class, meaning that the feature is significant in separating the classes. In that sense, this test allows us to remove features which have insignificant separation power.

4 Linear Discriminant Analysis

Classification is the process of taking an unlabeled data observation and using some rule or decision-making process to assign a label to it. In this study, we use classification for pitch prediction (fastball or nonfastball) using pre-pitch features. There are several classifications one can use to accomplish this task, we selected Linear Discriminant Analysis (LDA) [9] for this binary classification study.

The Linear Discriminant Analysis (LDA) classifier assumes that the observations within each class k are generated from a Gaussian (or normal) distribution with a class-specific mean vector μ_k's and a common variance σ_k^2's. Estimates for these parameters are substituted into the Bayes classifier results in LDA.

Assume that we only have one predictor (or feature) in a K-class classification problem. Then Bayes' theorem states that

$$\Pr(Y = k|X = x) = \frac{\pi_k f_k(x)}{\sum_{l=1}^{K} \pi_l f_l(x)} \tag{3}$$

where

1. π_k represent the prior probability that a randomly chosen observation is associated with the k-th class,
2. $f_k(X) \equiv \Pr(X = x|Y = k)$ denotes the density function of X for an observation that comes from k-th class
3. We will use the abbreviation $p_k(X) = \Pr(Y = k|X)$ is referred as the posterior probability that an observation $X = x$ belongs to the k-th class.
4. If we can compute all the terms for (3), we would then easily classify an observation to the class for which $p_k(X)$ is largest.

Since $f_k(x)$ is Gaussian, the normal density takes the form

$$f_k(x) = \frac{1}{\sqrt{2\pi}\sigma_k} \exp\left(-\frac{1}{2\sigma_k^2}\left(x - \mu_k\right)^2\right). \tag{4}$$

Substitute (4) into (3) under assumption that $\sigma_k^2 \equiv \sigma^2$, taking the log and rearranging the terms, we find that this is equivalent to

$$\delta_k(x) = x \cdot \frac{\mu_k}{\sigma^2} - \frac{\mu_k^2}{2\sigma_k^2} + \log(\pi_k). \tag{5}$$

The LDA method approximates the Bayes classifier by substitute the following estimates for π_k, μ_k and σ^2 into (5)

$$\hat{\pi}_k = \frac{n_k}{n}$$
$$\hat{\mu}_k = \frac{1}{n_k} \sum_{i:y_i=k} x_i$$
$$\hat{\sigma}^2 = \frac{1}{n-K} \sum_{k=1}^{K} \sum_{i:y_i=k} (x_i - \hat{\mu}_k)^2,$$

where n is the total number of training observations and n_k is the number of training observations in k-th class. After the LDA procedure, (5) becomes

$$\hat{\delta}_k(x) = x \cdot \frac{\hat{\mu}_k}{\hat{\sigma}^2} - \frac{\hat{\mu}_k^2}{2\hat{\sigma}_k^2} + \log(\hat{\pi}_k). \tag{6}$$

The LDA classifier can be extended to multiple predictors. To do this, we assume that $X = (X_1, X_2, ..., X_p)$ is drawn from a multivariate Gaussian distribution with a class-specific mean vector and common covariance matrix.

The formulas for estimating the unknown parameters π_k, μ_k, and Σ are similar to the one dimensional case. To assign an observation $X = x$, LDA assigns the class label for which discrimination function $\hat{\delta}_k(x)$ is largest. The word *linear* in the classifier's name comes from the fact that these discrimination functions are linear functions of x.

5 Results

To form a baseline for our prediction results, we compare our prediction model against the naive guess model. The naive guess simply return the most frequent pitch type thrown by each pitcher, calculated from the training set, see [7]. The improvement factor I is calculated as follow

$$I = \frac{A_1 - A_0}{A_0} \times 100, \tag{7}$$

where A_0 and A_1 denotes the accuracies of naive guess and our model, respectively.

In order to avoid possible overfitting issue, we applied cross validation, a common strategy for model validation and selection [1]. In specific, the repeated subsampling cross validation was used in our baseball pitch prediction as follows: in each pitcher-count training set with more than 15 pitches in the training set we split the training set randomly in half (e.g., if there are 20 pitches, we randomly pick ten of them. The remaining ten becomes the test set.) We then predict with this training set and compare to the actual class membership. We do this ten times for each pitcher-count training set and take the average accuracy of the ten tests.

We conduct prediction for all pitchers who had at least 750 pitches in both 2008 and 2009. After performing feature selection (see Sect. 3) on each data subset, each classifier is trained on each subset of data from 2008 and tested on each subset of data from 2009. The average classification accuracy for each classifier is computed for test points with a type confidence of at least 80 %.

Table 2. Baseball pitch prediction results comparison. Note that percentage improvement is calculated on a per-pitcher basis rather than overall average

Author	Training	Testing	Classifier	Accuracy	Cross Validation	Improvement
J.Guttag	2008	2009	SVM	70.00	.	18.00
This study	2008	2009	LDA	77.97	77.21	21.00
This study	2011	2012	LDA	76.08	75.20	24.82
This study	2010 and 2011	2012	LDA	76.27	75.54	24.07

Table 2 depicts the average accuracy among all pitches in 2008–2009 season, our model attained 78 %. Compared to the naive model's natural prediction accuracy, our model on average performs 21 % improvement. In previous work, the average prediction accuracy of 2008–2009 season is 70 % with 18 % improvement over naive guessing [7]. It should be noted that previous work only use SVM classifier and considers 359 pitchers who threw at least 300 pitches in both 2008 and 2009 seasons.

With cross validation implemented, the prediction accuracy slightly decreased by less than 1 %. In addition, applying this model on new data set from 2010, 2011 and 2012 season, the performance remain stable within ±2 % of the original results. This serves as an important confirmation that our results are reliable and our methods would perform well when applied to a newly introduced data set.

Table 3. Prediction results by Type Confidence (TC) levels (for a description of TC see Table 1).

TC (%)	50	60	70	80	90	95	99
Test size	355,755	344,300	332,238	312,574	196,853	24,412	7,150
Accuracy	77.19	77.17	77.13	77.13	77.97	83.19	81.76

As shown in Table 3, higher (better) type confidence cut-off thresholds reduces the sizes of testing sets. In fact, majority of test points have 80 % or higher type confidence. There is not a significant reduction in test sizes from 50 % level (355,755) to 80 % level (312,574), hence prediction performances from all methods remain stable throughout these intervals. Only when the cut-off threshold of type confidence are raised to 90 % level, we can notice the reduction in test sizes and the increase in average prediction accuracies among all methods. We obtain even higher average prediction accuracy, 83 % accurated at 95 % level.

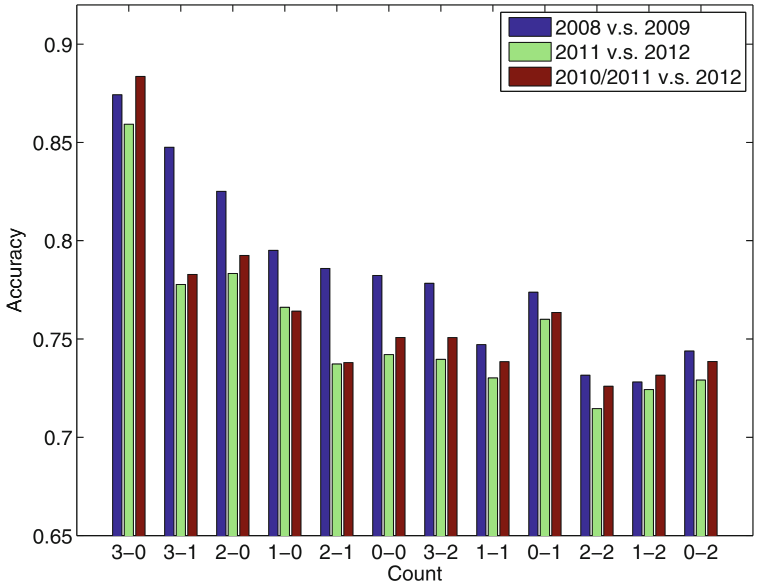

Fig. 3. Prediction accuracy by count.

However, there is only 24, 412 pitches at this level, less than 7 % of all pitches from original test set of about 360, 000 pitches, at 0 % level. Hence, we choose 80 % level to be the most reasonable choice to cut-off, which contains more than 50 % of the original test points. Notice that even a low type confidence level of 50 %, this model still outperform formal study [7], results in 77 % accurate.

Figure 3 depicts average prediction accuracy per each count situation. Accuracy is significant higher in batter-favored counts and approximately equal in neutral and pitcher-favored counts. Prediction is best at 3-0 count (89 %) and worst at 1–2 and 2-2 counts (73 %).

Table 4. Pitch prediction results for selected pitchers in 2008–2009 season

Pitcher	Training Size	Testing Size	Prediction Accuracy	Naive Guess	Improvement
Park	1309	1178	72.40	52.60	37.62
Rivera	797	850	93.51	89.63	0.04
Wakefield	2110	1573	100.00	100.00	0.00
Weathers	943	813	77.76	35.55	118.72
Vaquez	2412	2721	73.05	51.20	42.68
Fuentes	919	798	80.15	71.05	12.81
Meche	2821	1822	74.83	50.77	47.39
Madson	975	958	81.85	25.56	220.23

We randomly selected 8 pitchers from 2008 and 2009 seasons to examine in details. Table 4 illustrates average prediction accuracy per pitcher in comparison with Naive guess. For pitchers with only one dominant pitch type such as Wakefield with knucle-ball (nonfast) or Rivera with cutter-ball (fastball), Naive Guess is sufficient to reach almost perfect prediction. However, for pitchers having various pitch types at their disposals such as Weathers or Madson, our model adapt much better with accuracy improvements of 100 % to 200 %. Overall, our model outperforms naive guessing significantly.

6 Conclusion

Originally our scheme developed from consideration of the factors that affect pitching decisions. For example, the pitcher/batter handedness matchup is often mentioned by sports experts as an effect ([7,10]), and it was originally included in our model. However, it was discovered that implementing segmentation of data based on handedness has essentially no effect on the prediction results. Thus, handedness is no longer implemented as a further splitting criterion of the model, but this component remains a considered feature. In general, unnecessary data segmentations have negative impact solely because it reduce the size of training and testing data for classifiers to work with.

Most notable is our method of feature selection which widely varies the set of features used in each situation. Features that yield strong prediction in some situations fail to provide any benefit in others. In fact, it is interesting to note that in the 2008 v.s. 2009 prediction scheme, every feature is used in at least one situation and no feature is used in every situation.

It is also interesting to note that the LDA classification algorithm of this model is supported by our feature selection technique. In fact, as a Bayesian classifier, LDA relies on a feature independence assumption, which is realistically not satisfied. However, our model survives this assumption because although the features within each of the 6 groups are highly dependent across groups, the features which are ultimately chosen are highly independent. The model represents a significant improvement over simple guessing. It is a useful tool for batting coaches, batters, and others who wish to understand the potential pitching implications of a given game scenario. For example, batters could theoretically use this model to increase their batting average, assuming that knowledge about a pitch's type makes it easier to hit. The model, for example, is especially useful in certain intense game scenarios and achieves accuracy as high as 90 percent. It is in these game environments that batters can most effectively use this model to translate knowledge into hits.

Acknowledgements. Part of this research was supported by NSA grant H98230-12-1-0299 and NSF grants DMS-1063010 and DMS-0943855. The authors would like to thank Lori Layne, David Padget , Brian Lewis and Jessica Gronsbell, all from MIT LL, for their helpful advices and constructive inputs and Mr. Tom Tippett, Director of Baseball Information Services for the Boston Red Sox, for meeting with us to discuss this research and providing us with useful feedback, and Minh Nhat Phan for helping us scrape the PITCH f/x data.

A Appendix

From the original 18 features given in Table 1, we generated a total of 76 features and arranged them into 6 groups as follows:

Group 1

1. Inning
2. Time (day/afternoon/night)
3. Number of outs
4. Last at bat events

5-7. Pitcher v.s. batter specific: fastball or nonfastball on previous pitch/ lifetime percentage of fastballs/ previous pitch's events

8. Numeric score of previous at bat event

9-11. Player on first base/ second base/ third base (true/false)

12. Number of base runners
13. Weighted base score

Group 2

1-3. Percentage of fastball thrown in previous inning/game/at-bat

4. Lifetime percentage of fastballs thrown to a specific batter over all at bats

5-8. Percentage of fastballs over previous 5/10 /15/ 20 pitches

9-10. Previous pitch in specific count: pitch type/ fastball or nonfastball

11-12. Previous 2 or 3 pitches in specific count: fastball/nonfastball combo

13-14. Previous pitch: pitch type/ fastball or nonfastball

15. Previous 2 pitches: fastball/nonfastball combo
16. Player on first base (true/false)

17-18. Percentage of fastballs over previous 10/15 pitches thrown to a specific batter

19-21. Previous 5 /10 /15 pitches in specific count: percentage of fastballs

Group 3

1. Previous pitch: velocity

2-3. Previous 2 pitches/ 3 pitches: velocity average

4. Previous pitch in specific count: velocity

5-6. Previous 2 pitches/ 3 pitches in specific count: velocity average

Group 4

1-2. Previous pitch: horizontal/ vertical position

3-4. Previous 2 pitches: horizontal/ vertical position average

5-6. Previous 3 pitches: horizontal/vertical position average

7. Previous pitch: zone (Cartesian quadrant)

8-9. Previous 2 pitches/3 pitches: zone (Cartesian quadrant) average

10-11. Previous pitch in specific count: horizontal/vertical position

12-13. Previous 2 pitches in specific count: horizontal/vertical position average

14-15. Previous 3 pitches in specific count: horizontal/vertical position average

16. Previous pitch in specific count: zone (Cartesian quadrant)

17-18. Previous 2 or 3 pitches in specific count: zone (Cartesian quadrant) average

Group 5

1. SRP[1] of fastball thrown in the previous inning
2. SRP of fastball thrown in the previous game

3-5. SRP of fastball thrown in the previous 5 pitches/ 10 pitches/ 15 pitches.

6. SRP of fastball thrown in previous 5 pitches thrown to a specific batter

7-8. SRP of nonfastball thrown in the previous inning/ previous game

9-11. SRP of nonfastball thrown in the previous 5 pitches/ 10 pitches/ 15 pitches

12. SRP of nonfastball thrown in previous 5 pitches thrown to a specific batter

Group 6

1. Previous pitch: ball or strike (boolean)

2-3. Previous 2 pitches/ 3 pitches: ball/strike combo

4. Previous pitch in specific count: ball or strike

5-6. Previous 2 pitches/ 3 pitches in specific count: ball/strike combo

References

1. Arlot, S.: A survey of cross-validation procedures for model selection. Stat. Surv. **4**, 40–79 (2010)
2. Attarian, A., Danis, G., Gronsbell, J., Iervolina, G., Layne, L., Padgett, D., Tran, H.: Baseball pitch classification: a Bayesian method and dimension reduction investigation. IAENG Transactions on Engineering Sciences, pp. 392–399 (2014)
3. Attarian, A., Danis, G., Gronsbell, J., Iervolino, G., Tran, H.: A comparison of feature selection and classification algorithms in identifying baseball pitches. In: International MultiConference of Engineers and Computer Scientists 2013. Lecture Notes in Engineering and Computer Science, pp. 263–268. Newswood Limited (2013)
4. Bradley, A.: The use of the area under the ROC curve in the evaluation of machine learning algorithms. Pattern Recogn. **30**(7), 1145–1159 (1997)
5. Egan, J.: Signal detection theory and ROC analysis. Cognition and Perception. Academic Press, New York (1975)
6. Fawcett, T.: An introduction to ROC analysis. Pattern Recogn. Lett. **27**(8), 861–874 (2006)
7. Ganeshapillai, G., Guttag, J.: Predicting the next pitch. In: MIT Sloan Sports Analytics Conference (2012)
8. Hamilton, M., Hoang, P., Layne, L., Murray, J., Padget, D., Stafford, C., Tran, H.: Applying machine learning techniques to baseball pitch prediction. In: 3rd International Conference on Pattern Recognition Applications and Methods, pp. 520–527. SciTePress (2014)
9. Hastie, T., Tibshirani, R.: The Elements of Statistical Learning. Springer, New York (2009)
10. Hopkins, T., Magel, R.: Slugging percentage in differing baseball counts. J. Quant. Anal. Sports **4**(2), 1136 (2008)
11. Swets, J., Dawes, R., Monahan, J.: Better decisions through science. Sci. Am. **283**, 82–87 (2000)
12. Zweig, M.H., Campbell, G.: Receiver-operating characteristic ROC plots: a fundamental evaluation tool in clinical medicine. Clin. Chem. **39**(4), 561–577 (1993)

[1] Strike-result percentage (SRP): a metric we created that measures the percentage of strikes from all pitches in the given situation.