MLB 試合データを用いた失点予測と継投計画の最適化

境田 晃大† 清 雄一† 田原 康之† 大須賀 昭彦†

†電気通信大学 情報理工学域 〒182-8585 東京都調布市調布ヶ丘 1-5-1

E-mail: †sakaida.koudai@ohsuga.lab.uec.ac.jp, {seiuny, tahara, ohsuga}@uec.ac.jp

あらまし：現在、野球界において「データ野球」というものが非常に浸透している。しかし、従来のデータ解析は統計的手法が用いられることが多く、機械学習が用いられる例は多くない。詳細なデータを得られるようになり、かつ、そのデータが公開されるようになったため、データ収集とこれを用いた機械学習が容易になった。野球においては、投手交代のタイミングが試合の勝敗を大きく左右する。そこで本研究では、1球ごとの失点数ラベル付きデータに、外部からの気候情報などを加えたデータセットを構築し、その1球での失点予測を行うことで、継投計画の最適化を図るシステムを提案する。元データを用いた5値分類よりも高い精度を示すことができた。

キーワード：失点予測、データ分析、投手交代支援、最適化

1 はじめに

現在、野球界において「データ野球」が非常に浸透している。1970 年代にセイバーメトリクスに基づいたデータ分析が提唱され、2000 年代にはデータ野球による貧乏球団の快進撃によって、データの重要性が注目されるようになった。日本プロ野球においても、当時ヤクルトスワローズの監督であった野村克也が「ID 野球」を提唱し、優勝争いから無縁であったチームを常勝チームに変えていった。近年においては、軍事技術の転用によるトラックマンなどの精密機器を用いて、回転数や回転軸、球種といった詳細なデータを得られるようになった。今ではこのようなデータが Baseball-Savant[1]や Baseball-Reference[2]、FanGraphs-Baseball[3]などで公開されており、誰であっても分析が可能になっている。各球団が試合において目指すところは「勝つ」ことであり、そのためには相手よりも得点する、または可能な限り失点を抑えることが必要になる。データ野球が広まる前までは、投手交代について監督の経験や勘に基づいて判断されてきた。これは非常に曖昧であり、適切な判断であったのかは結果からでしか確認することができない。データ野球が普及してからも、基本的に統計手法が使われることが多く、機械学習による研究は多くない。そこで、本研究ではフォーカスの比較的当たっていない継投計画について、pybaseball ライブラリから得られる 1 球ごとのデータを用いた失点数の予測と、投手交代の意思決定までを行うことを目的とする。

2 関連研究

機械学習による野球分析は比較的新しい研究分野[4]であり、既存の研究としては、おおよそ 3 つに分類することができる。1 つ目は野球の試合結果の予測である。Valero[5]はセイバーメトリクス統計を含むデータセットを用いて、MLB のレギュラーシーズンの試合結果（勝敗）を予測するために、分類および回帰ベースのデータマイニング手法で予測を行っている。結果としては、SVM が各チーム平均約 60%の予測精度で予測が可能になった。2 つ目としては選手関連の予測で、選手パフォーマンスやシーズン打率、年俸などを予測する論文が複数ある。Ishi[6]は、プロセスは優れているが結果が出ていない選手に焦点を当てており、クラスタリングアルゴリズムを使用して、成功した選手とプロセスが似ている選手を特定している。また、Ganeshapillai ら[7]は、正則化された線形回帰を使用して投手固有の予測モデルを構築している。この予測モデルは、投手交代のタイミングを決める際に役に立つ。このモデルが投手交代させたが監督が続投させた試合では、60%の確率でパフォーマンスが低下したことが明らかになった。3 つ目は、球種や球速、最適投球ゾーンなどの投手に関する研究である。Woodham ら[8]は試合中のある特定の時点で、先発投手を交代するかどうかを予測している。Woodham らの知る限りでは、先発投手をいつ交代させるのかについての監督の意思決定をモデル化したのはこの研究が初めてであるそうだ。先発投手が交代した最後の投球に対して 1 のラベル(その他：0)が付与され、Light-GBM や DNN、

ランダムフォレストなど複数の分類アルゴリズムで比較している。結果として、ランダムフォレストを用いた際に Accuracy が最も高くなり（96.96%）、Gradient-Boost を用いた際に F-score が最も高くなった（53.54%）。

本研究では、Woodham らの研究とは異なり、失点という部分に焦点を当てて投手交代の意思決定支援を行う。

3 提案手法

全体の流れとしては、まずデータの前処理を行う。そして、前処理済みのデータを用いて、失点予測モデルを構築する。最後に、作成した学習済みモデルを用いて、最も失点を抑えてくれる投手への交代を支援する。

3.1 データセット

一般に公開されている 1 球ごとのデータセットがないため、データセットの構築から始めた。データ収集のために用いた Python ライブラリは pybaseball で、MLB の主要なデータサイトから取得したデータを扱うことができる。これを用いて、2011 年から 2021 年の 10 年間での 1 球ごとのデータ(約 700 万球分)を取得した。ラベルの割り当てについては、投手が失点した 1 球に対して、失点数をラベリングしている。例えば、失点数 2 であるとき、データには 2 のラベルが割り当てられている。

3.2 データの前処理

・特徴量作成：

不足していると考えられる特徴量に関して、精度改善のため特徴量の作成を行う。加えて、Visual-Crossing Whether and API[9]から得られる試合日の気候データを追加している。また、複数の特徴量を組み合わせて新たに特徴量を作成を行う。具体的には、各試合に対する投手名、打者名の TF-IDF 値を算出し、特徴量として加えている。

・欠損値・外れ値の処理：

今回取得したデータセットには、情報が計測されなかった投球や敬遠など、失点を予測する際に結果にネガティブな影響を与えてしまうデータがある。本研究では、このようなデータは削除している。

・エンコーディング

機械学習におけるカテゴリ変数は、数値化する必要がある。そこで、投手名・打者名・球種に関してはカウントエンコーディングを、ホームチーム名、アウェイチーム名に関しては、ラベルエンコーディングをすることで数値化を図っている。

3.3 不均衡データへの対処

失点予測をする際、1 球ごとのデータには 0 ～ 4 までの 5 つのクラスが存在し、表 1 からも明らかなように、失点数 0 とその他ではクラスデータに強い偏りがある。偏りが存在するまま機械学習を行うと、失点予測ができていないにも関わらず、高い精度を出してしまう。そこで、この問題を対処するために Synthetic Minority Over-sampling Technique (SMOTE)[10]を適用した。これはオーバーサンプリング手法の 1 つで、少数派データを単にコピーするのではなく、検出した近傍のデータとの内挿を使ってデータを増やしている。新たに増やすデータ x_{new} は、ある少数派データ x_i とその任意の数の近傍データからランダムな 1 つのデータ $\hat{x}_i$ が選ばれ、その差と $p \in [0,1]$ との乗算を加えて作成される。式 (1)にこれを示す。

$$x_{new} = x_i + (\hat{x}_i - x_i) * p \ldots (1)$$

表 1 ：ラベルごとのデータの割合

ラベル	0	1	2	3	4
割合 (%)	97.7	1.8	0.4	0.096	0.016

3.4 モデル

前処理済みのデータセットにおける、LightGBM を用いた 5 値分類と失点数 1,2 の数が少ない、かつ、失点パターンが多いクラスを個別に 2 クラス分類し、5 値分類による結果に対し、予測値を上書きするモデルを提案する。これを図 1 に示す。

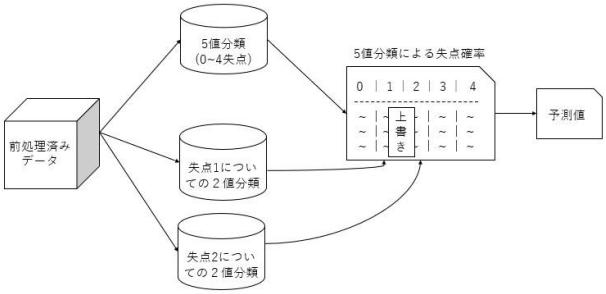

図 1 ：5 値分類と 2 値分類の組み合わせモデル

5 値分類の際にはオーバーサンプリングを、2 値分類の際はアンダーサンプリングを用いている。

3.5 継投計画最適化システム

ピンチ時における継投計画を行う。データセットをチーム毎に分割し、投手・球種・投球ゾーンの全組み合わせから、機械学習モデルを用いて失点 0 になる確率が最も高いものを探索する。モデル図は以下の図 2 に示す。

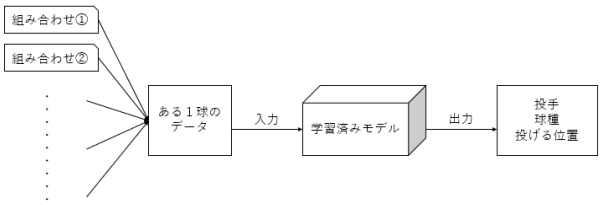

図 2:継投計画最適化システム図

4 実験と評価

本研究ではチームごとにモデルを構築し、チームごとにパラメータ設定、評価を行う。

ベースライン手法としては、前処理を行う前のデータセットに対する LightGBM を用いた 5 値分類を用いる。各手法において 5 分割交差検証により、モデルの汎化性の改善を行っている。評価指標としては、 各クラスにおける precision 値、f-score、macro_f-score を用いる。例として、ボストン・レッドソックスの投手に対して、ベースライン手法、提案手法を用いた結果が以下の表 2,3,4 のようになった。

表 2：ベースライン手法

ラベル	precision	f-score
0	0.98	0.99
1	0.15	0.01
2	0.00	0.00
3	0.00	0.00
4	0.00	0.00

表 3：前処理済みデータによる 5 値分類モデル

	precision	f-score
0	0.98	0.99
1	0.09	0.00
2	0.00	0.00
3	0.06	0.07
4	0,03	0.05

表 4：前処理済みデータによる組み合わせモデル

	precision	f-score
0	0.98	0.99
1	0.14	0.18
2	0.05	0.02
3	0.14	0.03
4	0,00	0.00

そして、f-score の macro average 値を表 5 に示す。

表 5 ：各モデルの macro f-score

モデル	macro average
ベースライン	0.20
前処理済み 5 値分類	0.22
前処理済み組み合わせモデル	0.24

各モデルにおける、特徴量の重要度上位 5 つは以下の表 6 のようになる。そして、表 7 にその特徴量の簡易的な説明を示す。

表 6:各モデルにおける重要度

ベースライン(特徴量 20 個)	前処理済み 5 値分類(特徴量 29 個)
Zone(15%)	Zone(11%)
Pitcher_name(14%)	Pitcher_name(8.8%)
Game_ID(7.7%)	Batter_name(6.3%)
Batter_name(6.6%)	uvindex(4.1%)
On_2b(6.0%)	On_bases(3.9%)

表 7:特徴量の説明

Zone	投球場所
Pitcher_name	投手名
Game_ID	試合ごとに付与される固有の ID
Batter_name	打者名
uvindex	紫外線量
On_2b	2 塁上にランナーがいるか
On_bases	塁上のランナーの数

継投計画最適化システムによる出力結果は表 8 のようになる。実際の投球シーンと提案手法を用いたシーンを比較している。

表 8：最適化モデルによるピンチ時の継投計画結果

	実際の失点数	投球ゾーン	投手名	球種
実際のシーン	4	12	Sale, Chris	4-Seam
提案のシーン		6	Ort, Kaleb	Fast-Ball

この投手に交代し、指定の球種、ゾーンに投げることで失点 0 になる確率は最大で 89.9%になる。

5 考察

データの前処理に関して、特に TF-IDF を用いた特徴量作成は工夫した点でもあり、精度向上にも役立ったと考える。特に試合に対する投手の TF-IDF 値は、先発投手に有効であるとわかる。なぜならば、1 度登板すると数日は投げないため IDF 値が高くなり、登板し、調子が良ければ 1 試合あたり 100 球近くなげるため、TF 値が大きくなる。つまり、調子の良い先発投手の値が大きくなる。実際に、特徴量の重要度も 7 位に入ってきている。

予測モデルに関して、特に 1 失点に関しては失点するパターンが非常に多いと考える。例えば、ランナーなしでホームランによる 1 失点や、ランナー 1 塁での 2 ベースヒットによる 1 失点などがある。逆にランナー 1 塁での 2 ベースヒットであっても、1 塁ランナーの走る速度によっては得点されないケースも存在する。2 失点に関しても同様のことが言える。5 値分類に関しては、SMOTE アルゴリズムを使用したオーバーサンプリングを適用したが、このモデルでの 1～2 失点ラベルの水増しがより学習を混乱させてしまったのではないかと考える。そこで、1～2 失点ラベルに関しては、個別にアンダーサンプリングを用いることとなった。無駄にデータを増やすことよりも、質の高いデータによって、量が少なくても精度の改善（macro f-score 4%増）が可能であることがわかる。また、失点確率に閾値を設けることでより精度が増すのではないかと考える。失点 0 の予測確率は常に比較的高い値を示しており、他の失点確率が投球の前後で相対的に増加していてもその影響が小さい。Recall 値の増加になる可能性もあるが、各失点確率にある閾値を設けることで、確率の相対的な変化を汲み取ることができるのではないかと考える。

6 まとめ

本研究では、pybaseball から得られる投球データに対し、精度向上のために複数の前処理を行った。加えて、失点予測モデルを構築し、このモデルを用いて継投計画最適化システムを作成した。精度としてはベースライン手法と比べ、提案手法は macro_f-score 4%の向上が見られた。今後の展望としては、より多くのデータ収集をし、精度の向上を目指す。また、シミュレーションなどを用いた継投計画最適化システムの評価を行っていきたい。

参 考 文 献

[1] Baseball-Savant, https://baseballsavant.mlb.com/

[2] Basevall-Reference," https://www.baseball-reference.com/"

[3] Fan-Graph baseball, "https://www.fangraphs.com/"

[4] K. Koseler and M. Stephan, "Machine learning applications in baseball: A systematic literature review", *Applied Artificial Intelligence*, vol. 31, no. 9-10, pp. 745-763, 2017.

[5] C. S. Valero, "Predicting win-loss outcomes in mlb regular season games: A comparative study using data mining methods", *International Journal of Computer Science in Sport*, vol. 15, no. 2, pp. 91-112, 2016.

[6] T. Ishii, "Using machine learning algorithms to identify undervalued baseball players", *Technical Report: Stanford University*, 2016.

[7] G. Ganeshapillai and J. Guttag, "A data-driven method for in-game decision making in mlb", *Sloan Sports Analytics Conference*, 2014.

[8] M. Woodham, J. Hawkins, A. Singh and S. Chakraborty, "When to Pull Starting Pitchers in Major League Baseball? A Data Mining Approach," *2019 18th IEEE International Conference On Machine Learning And Applications (ICMLA)*, Boca Raton, FL, USA, 2019, pp. 426-431, doi: 10.1109/ICMLA.2019.00080.

[9] Visual Crossing Wether and API, "https://www.visualcrossing.com/"

[10] Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: synthetic minority over-sampling technique. Journal of artificial intelligence research, 16, 321-357.