Multimodal Machine Learning for Major League Baseball Playoff Prediction

Aliaa Saad Yaseen[1*], Ali Fadhil Marhoon[2], and Sarmad Asaad Saleem[1]
E-mail: aliaa.yaseen@uobasrah.edu.iq, ali.marhoon@uobasrah.edu.iq, sarmad.saleem@uobasrah.edu.iq
[1]College of Computer Science and Information Technology, University of Basrah, Iraq
[2]College of Engineering, University of Basrah, Iraq

Keywords: machine learning, prediction, sport, baseball

Received: December 11, 2021

The introduction on sabermetrics has changed the way Major League Baseball (MLB) teams valued their players. Since then, new baseball stats have been made to make various predictions for MLB teams. This domain contains an immense amount of data on baseball players, teams, and scores. Using various supervised machine learning algorithms, we plan to see how well we can accurately predict which teams will make it to the playoff for year 2019. For this research, we have gathered data from the last 20 years. The features that we will utilize for our machine learning algorithm include Runs, Batting Average, Homeruns, Strikeouts, Innings Pitched, Earned Runs, and Earned Runs average. We decided to use a Logistic Regression model and a Support Vector Classifier (SVC) as the two machine learning algorithms for our features. After running our tests, our models showed that our trained algorithms were only able to predict accurately 77% of the teams correctly. Of those 77% accurately predicted, 59% was recalled correctly. This led to our overall projected model being only 60% accurate. The projected model was only able to correctly predict 6 out of 10 teams that made the 2019 playoffs. We believe that we can improve upon our findings by using other machine learning algorithms or including more features that can increase the overall accuracy of our training model.

Povzetek: Za Major League Baseball so z različnimi algoritmi strojnega učenja skušali na osnovi statističnih podatkov napovedati, katere ekipe se bodo uvrstile v zaključne tekme.

1 Introduction

In 2004 Michael Lewis published a New York Times bestseller, Moneyball: The Art of Winning an Unfair Game, which was made into a movie in 2011 featuring Brad Pitt and Jonah Hill [14]. The book is about a General manager Billy Beane of the Oakland Athletics, who used sabermetrics [9] to measure undervalued players. Due to budget constraints he figured out a more strategic way to measure players. Under Billy Beane, the Oakland A's won 20 consecutive games in 2002. The A's was the first team in American League baseball in over 100+ years to win that many games consecutively. After his world breaking record, he was offered $12.5 million from the Boston Red Sox to become their General manager (GM) which he declined. Billy Beane's tactic has changed the way baseball teams value their players [2].

The use of statistical analysis in baseball has always been used since the early days. General managers and baseball scouts used to measure players based on traditional batting and pitching statistics such as batting averages (BA), at bats, home runs, strikeouts etc. The popularity of sabermetrics has brought better measurements to light whether from a batter's statistics to a pitcher. We'll use sabermetrics to help us collect data from each team's stats from 1998 to 2018. Although there is data, we can gather from 50+ years of baseball, we're just going to evaluate the last 20 years [11]. Making predictions in baseball is relatively tough and it can never be perfect. "Years ago, sabermetrician Tom Tango researched the amount of talent and luck that go into team winning percentages and found that chance explains one-third of the difference between two teams' record.". [16] Which makes it hard to predict how many wins a team will make in a season. With each team playing 162 games, there's a lot of factors that can affect predictions on any given day.

There are tons of baseball fans and diehard fans. Like football, there's fantasy baseball where you build a team and accumulate points based on your player's up-to-date stats [4]. Fans alone will make bets on which teams will win or not. "Teams at all levels are improving their ability to evaluate players, make decisions on personnel and game plan by taking a fresh look at data. The field continues to grow and change, with the integration of video analysis, defensive statistics and health analytics the advancement of baseball through data manipulation has no end in sight.". [11] Whether someone is a baseball fan or not or breaking into the analytics part of it, this paper takes

[*] Corresponding author

an interesting measure of how stats contributes to the world of baseball.

2 Related works

There are many researchers presents papers in the field of Predicting Major League Baseball. Brandon Tolbert et al. [23] uses Data Mining technique to Predict Major League Baseball Championship Winners. They will attempt to develop classifiers using Support Vector Machines (SVM's) to predict the winners on multi- levels. 42 teams were labeled as World Series champions and 42 teams were labeled as World Series losers. The classifier that produced the highest accuracy of 77.1 was the Gaussian Kernel RBF SVM.

Justine Jones et al. [24] presented a Sports Illustrated (SI) Predictions pre-season forecasting accuracy for four major North American sport championships over the last 30 years. While results varied across leagues. SI was generally more successful at predicting divisional winners compared to conference and league champions. The study had some limitations, such that, 30 years of data, presents a longitudinal data source but provided a relatively small number of data points, some of it was missing because of league lockouts, missing pages in the magazine, or SI may not make a prediction that year. Finally, the authors were unable to determine how SI predictions were/are made. Soto Valero, C. [25] employs sabermetrics statistics with the purpose of assessing the predictive capabilities of four data mining methods (classification and regression based) for predicting outcomes (win or loss) in MLB regular season games.

The model approach uses only past data when making a prediction, corresponding to ten years of publicly available data. The obtained results prove that the classification predictive scheme forecasts game outcomes better than the regression scheme. Among the four data mining methods used, SVMs give the best predictive results with a mean of about 60% prediction accuracy for each team. Chia-Hao Chang [26] used the Markov process method and the runner advancement model to estimate the expected runs in an MLB match for the teams based on the batting lineup and the pitcher. The source of data was the 70 MLB matches with most batter versus pitcher matchup stats in 2018.

During the theoretical analysis for these matches in this article, when they restore the very moment of betting, where the outcomes were unknown, and the total return for the 70 matches according to the prediction probability models of NIP, and NIP–NBD are 23.89 and 22.69, respectively, converted to return on investment (ROI) as 34.13% and 32.41%. Ting-Chun Yu and et al. [27] utilized both of the GA and Support vector machine (SVM) for the prediction purposes; it can avoid the over-fitting, local minima and help to do the classification. They collect all the baseball team's batting, pitching and fielding records for the period 1995-2016 to establish the classification model. Finally, they compare between the performance of GA-SVM with SVM and C4.5 methods. The comparison result has higher accuracy of 92.34, than the C4. 5 and traditional SVM.

Table 1: Summary of the related works results.

Ref. No.	year	data	Technique	Accuracy
[23]	2016	42 TEAMS	Data Mining-SVM	77.1%
[24]	2021	30 YEARS	Sports Illustrated	unable to determine
[25]	2016	10 previous years	Data mining-SVM	60 %
[26]	2021	70 MLB matches 2018	Markov process	34.13 %
[27]	2017	GA-SVM	GA-SVM	92.34

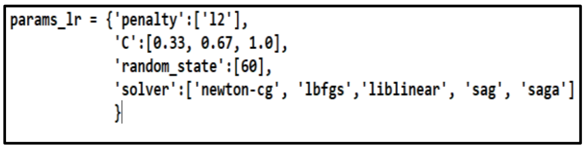
From all of the above, one can summarize the results obtained from previous research, as shown in the table 1.

3 Methodology

For our project we're going to gather each MLB teams end of year statistics of important batting and pitching averages to try and predict if they will make it to the playoffs. We'll build our model to collect data for the last 20 years up until 2018 and test the results with what the outcome was for 2019's data. We believe that given enough data almost anything can be calculated or predicted. By doing this research we aim to see how we can take numerous amounts of information and data and create a model that can predict future or recent news/data. We combine this concept (known as Predictive Modeling) with machine learning techniques to better enhance and "streamline" our models [14]. By incorporating machine learning, we hope to be able to create an algorithm that can predict the future performance of baseball teams if given a year. This research is aimed to entice and attract those who see the benefit of combining machine learning concepts with real world data (baseball data). This research is meant to attract sports fans, data analysists, marketers, and etc.

This research was motivated on the team's overall interests in combining baseball data (a sport that is widely watched and globally played) with knowledge of various machine learning techniques (namely, Supervised Learning techniques). With the development of this research, the team's overall goal is to be able to create a working algorithm that can predict what baseball teams will be able to make it to the 2019 playoffs. We hope to create a model that can achieved more than 50% accuracy in prediction.

Our organization in terms of how the research is structured (organized) is that there will be two Supervised Machine Learning techniques that we will incorporate (Logistic Regression and Support Vector Classifier). We will train and test these models using various baseball data and create our models, to improve upon these two models, we will then optimize them using a Grid Search algorithm with the hope that it improves the accuracy of our previous models. The idea of using baseball data to create predictable models is not a new concept. Various groups have proposed and implemented their version of such models.



```
params_lr = {'penalty':['l2'],
             'C':[0.33, 0.67, 1.0],
             'random_state':[60],
             'solver':['newton-cg', 'lbfgs','liblinear', 'sag', 'saga']
             }
```

Figure 1: Hyperparameters for Logistic Regression prior to Grid-Search.

Some groups have used other machine learning concepts such as Random Forests and Gradient Boosting to create their models [12]. However, the most common machine learning algorithm that all of these groups have used are Logistic Regression models. This is because when it comes to baseball (or any other sport), you can only have a team that wins or lose. This makes using a Logistic Regression model the most ideal model to incorporate [5].

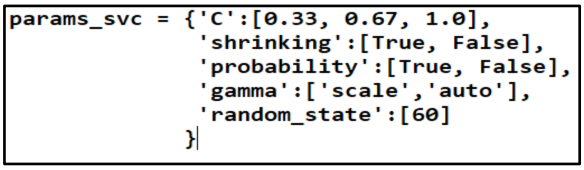
We decided to utilize a Logistic Regression model in our research because of this reasoning. We decided to also include the use of a Support Vector Classifier (SVC) because as our research deals with using labeled training data (supervised learning), we want to find the separation in our classes and create a hyperplane. Thus, the use of an SVC gives us the capability of doing both classification and regression.

Dealing with a collection of data to draw some conclusion from observed values we'll use a few classification models. Given one or more inputs the classification models will try to predict the value of one or more outcomes. The models we'll test are Logistic Regression and Support Vector Classifier (SVC). We chose to focus on these two models, because the concept of our project deals with whether the Atlanta Braves or another team is going to make it to playoffs or the world series and that in itself can be consider a type of classification project. Either a team makes it to playoffs ([1]) or not ([0]). Since you only have two possible outcomes (makes it to playoffs or not), we decided to incorporate a Logistic Regression. We can also utilize an SVC by adding a hyperplane to divide the data. With the predictions being made based on where the point sits relative to the hyperplane.

In order to optimize our hyper parameters, we will incorporate a Grid Search (a function within the SciKit-learn library). Grid-searching is the process of scanning the data to configure optimal parameters for a given model [13]. We will use it cross-validate our models and refit it with our training and testing dataset. The algorithm will build a model for each parameter combination possible. It iterates through every parameter and stores a model for each combination.

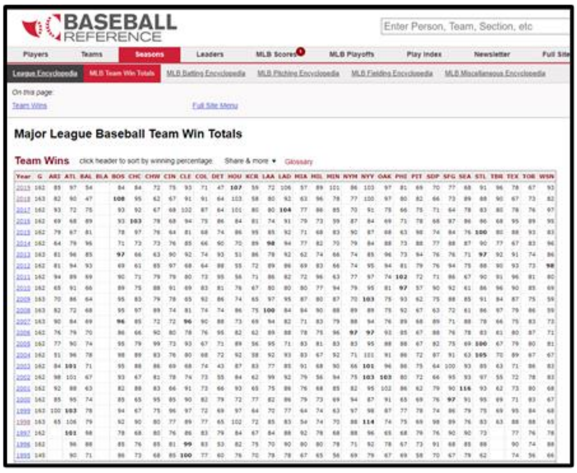
Before we import our models into the Grid Search function, we will begin by defining the hyperparameters for both models. For the logistic regression model, we will first start by creating the regularization for the penalty space. We specify that we want to utilize an "L1" regularization to improve the generalization performance for any new unseen data. We set our regularization hyperparameter space to be around "[0.33, 0.67, 1.0]" (i.e. we want the sequence to start at value ~0.33 ending at value ~0.67, while generating "1" sample for each iteration. We set the random state to "60" for the random number generator. For the "solver" parameters, we will use "'newton-cg', 'lbfgs', 'liblinear', 'sag', and 'sage'" as they work better for larger datasets and can handle L2 regularization (see Fig. 1).

Similar to the logistic regression model, we will also set the parameters for the support vector classifier as well. We start by specifying the regularization parameter, similar to the logistic regression model, we will use the same values ("[0.33, 0.67, 1.0]"), we set the heuristics to enable "shrinking", our "probability" to 'True' for the model to use probability estimates, along with having the "gamma" parameter set to 'scale' and 'auto', with a random state set at "60" (see Fig. 2).



```
params_svc = {'C':[0.33, 0.67, 1.0],
              'shrinking':[True, False],
              'probability':[True, False],
              'gamma':['scale','auto'],
              'random_state':[60]
             }
```

Figure 2: Hyperparameters for Support Vector Classifier (svc) prior to Grid-Search.



BASEBALL REFERENCE

Enter Person, Team, Section, etc

Players Teams Seasons Leaders MLB Scores MLB Playoffs Play Index Newsletter Full Site

League Encyclopedia MLB Team Win Totals MLB Batting Encyclopedia MLB Pitching Encyclopedia MLB Fielding Encyclopedia MLB Miscellaneous Encyclopedia

On this page:

Team Wins Full Site Menu

Major League Baseball Team Win Totals

Team Wins click header to sort by winning percentage Share & more ▾ Glossary

Year G ARI ATL BAL BLA BOS CHC CHW CIN CLE COL DET HOU KCR LAA LAD MIA MIL MIN NYM NYY OAK PHI PIT SDP SFG SEA STL TBR TEX TOR WSN

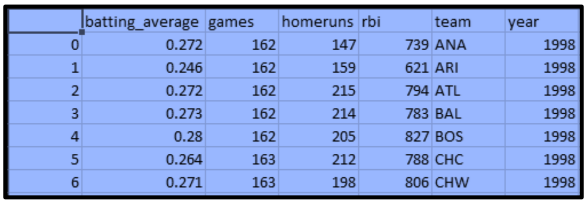
Figure 3: Raw Baseball dataset sorted based on year [Baseball-reference.com].

Once these parameters are set, we will incorporate them into the Grid Search function. We will also utilize a Scaler (MinMaxScaler) library to scale each team down to either they did "really well this [insert year]", indicated by the scale of "1" or "they did average or not so well", indicated by a scale of "0". For this project we will be utilizing several python libraries to analyze the gathered baseball data. We will be using a "Pandas" to store our data as data-frames and "NumPy" to develop our models when used in combination to "SciKit-learn" to develop our regression and SVC computations.

To gather data, we used baseball-reference.com (Fig. 3) which is the complete source for current and historical baseball players, teams, scores, and leaders. We took data over the last 20 years from 1998 - 2018 of influential batting and pitching stat averages per team. The data we will be collecting are as follows:

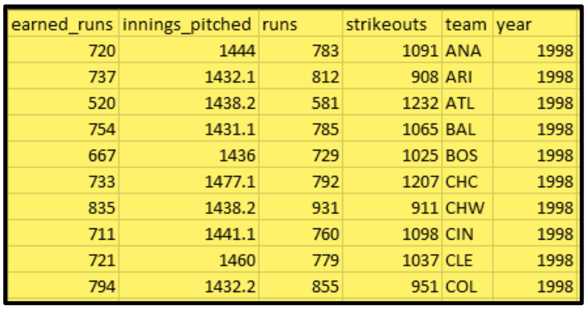
Hitters/Batters**:**

R: Runs



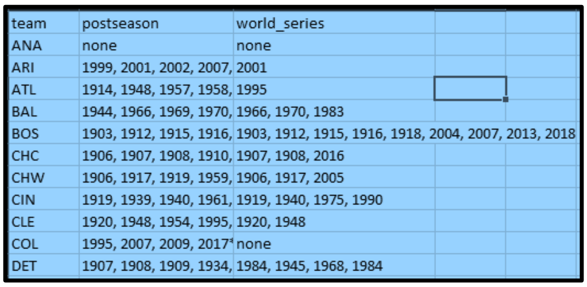
	batting_average	games	homeruns	rbi	team	year
0	0.272	162	147	739	ANA	1998
1	0.246	162	159	621	ARI	1998
2	0.272	162	215	794	ATL	1998
3	0.273	162	214	783	BAL	1998
4	0.28	162	205	827	BOS	1998
5	0.264	163	212	788	CHC	1998
6	0.271	163	198	806	CHW	1998

Figure 4: Batting Stats for each MLB team from 1998 to 2018, organized into an Excel spreadsheet.



earned_runs	innings_pitched	runs	strikeouts	team	year
720	1444	783	1091	ANA	1998
737	1432.1	812	908	ARI	1998
520	1438.2	581	1232	ATL	1998
754	1431.1	785	1065	BAL	1998
667	1436	729	1025	BOS	1998
733	1477.1	792	1207	CHC	1998
835	1438.2	931	911	CHW	1998
711	1441.1	760	1098	CIN	1998
721	1460	779	1037	CLE	1998
794	1432.2	855	951	COL	1998

Figure 2: Pitching Stats for each MLB team from 1998 to 2018, organized into an Excel spreadsheet.



team	postseason	world_series		
ANA	none	none		
ARI	1999, 2001, 2002, 2007,	2001		
ATL	1914, 1948, 1957, 1958,	1995		
BAL	1944, 1966, 1969, 1970,	1966, 1970, 1983		
BOS	1903, 1912, 1915, 1916,	1903, 1912, 1915, 1916, 1918, 2004, 2007, 2013, 2018		
CHC	1906, 1907, 1908, 1910,	1907, 1908, 2016		
CHW	1906, 1917, 1919, 1959,	1906, 1917, 2005		
CIN	1919, 1939, 1940, 1961,	1919, 1940, 1975, 1990		
CLE	1920, 1948, 1954, 1995,	1920, 1948		
COL	1995, 2007, 2009, 2017	none		
DET	1907, 1908, 1909, 1934,	1984, 1945, 1968, 1984		

Figure 6: The playoffs data created using information gathered from Wikipedia [12].

BA: Batting Average

HR: Homeruns

SO: Strikeouts

Pitchers**:**

SO: Strikeouts

IP: Innings pitched

HR: Homeruns

ER: Earned runs

ERA: Earned runs average

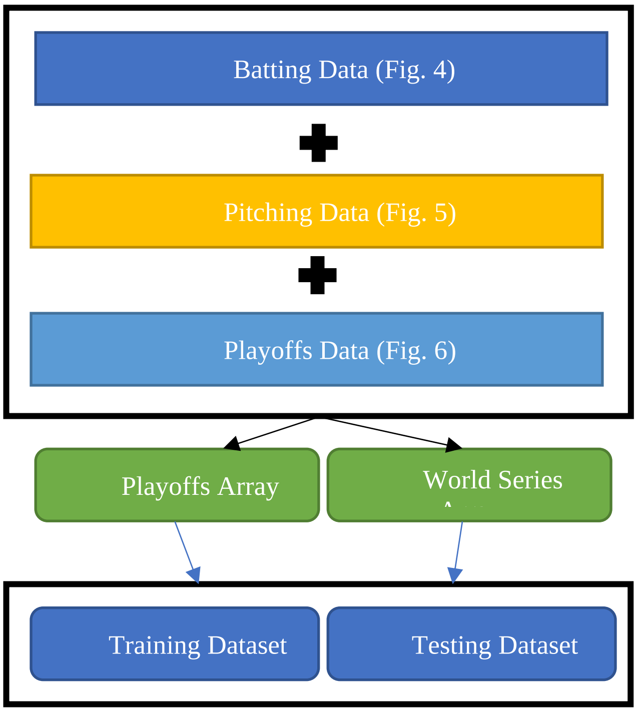


Figure 1: Methodology for obtaining the testing and training datasets.

The following stats from baseball-reference.com will be imported into two separates excel files, one for pitching and one for batting. For the teams that made it to the playoffs we created a separate excel file from 1998 – 2019 (referenced from Wikipedia). See Fig. 4, 5, 6.

After gathering the data, we will combine the pitching and batting datasets into one large dataset. It is during this stage that we will also sort out our playoff dataset.

In our excel file for playoffs and world series, we will organize the teams that made it to playoffs from teams that didn't make it to playoffs between the years of 1998 and 2018, this eliminates the other teams that didn't make it to playoffs from our dataset. We will further sort out our data by organizing the teams that won the world series for a particular year. In total, for each year every team will be label "1" if they made it to the playoffs and/or world series, and "0" if they didn't make it to either one or both. We'll create two set of arrays (playoffs and world_series) that will store the teams that made it to the world series and playoffs.

In baseball, a single game usually consists of nine innings, but there are situations where a game can more or less than nine. This difference can skew our dataset. In order to combat this, we will be Standardizing our data so that each data entry is adjusted to the difference in number of innings/games.

We will also take the time to scale down our data, to account for each new season (year), in each season a team may do extremely well or extremely average or worse. By scaling each team down to either "1", the team did above average or "0", the team did average to below average based on the maximum or minimum of each feature column, we can gauge each team's performance. We will create a new data-frame for these new scaled datasets before testing. Once our new data-frame is made, we will begin splitting up our data into two sets, training, and testing, with an 80% training and a 20% testing. A simplistic diagram explaining our method is provided below (see Fig. 7).

4 Experimental results

To evaluate how successful our models were, we compare the results with different types of metrics. These metrics include the Precision and Recall, where if both the Precision and Recall are high it can be concluded that our output is very accurate, a low Precision or Recall can mean low accuracy or high false negatives within our output. We will also utilize an F1-Score, which is the average of our Precision and Recall, and the score it gives us tells us if our test is accurate or not based on how closed to “1” it is. We will also generate an AUCROC (Area Under Curve Receiving Operating Characteristic) Score, which will help us understand how well our algorithm is able to distinguish between false positives and true positives, based on how close our score is “1” for each model. For this project we will do a total of four tests to gather our conclusion. We will do two tests using the Logistic Regression (Fig. 8) and SVC models (Fig. 9) using the scaled and standardize dataset, and two additional tests, where we will run each of the model through the Grid Search algorithm (Fig. 10, 11).

When testing our datasets using an SVC and Logistic Regression Model, we found that there was no difference in ROC AUC score when using either the SVC or the Logistic Regression Models. The same can be said when running both models again into the Grid Search algorithm. For all four tests, we were given a ROC Score of .76, meaning our model was only able to distinguish ~76% of the data as false positives and true positives. Below show the results we’ve obtained for each model, with the first two being where we tested both models without running them through the Grid Search.

We can see from the figures above that for each test, of the teams that made it to the playoffs, ~77% of the teams predicted were accurate (Precision), vice versa for predicting the teams that didn’t make it. Of the teams that were predicted correctly (of making playoffs), ~59% of them were correctly identified (recalled). From the results above, we can create an AUCROC graph (*fig.* 12) that reflects our figures.

From the graph, we can see that both SVC and Logistic Regression models have a very similar curve, likewise, they’re also very close to our true positive rate. This means our test model, was for the most part mostly accurate. From this we then ran our predictions (using the

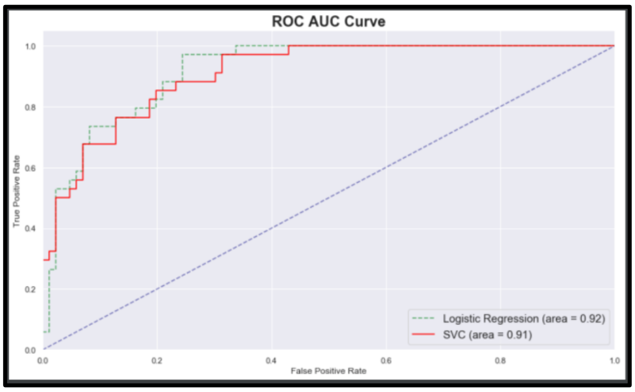
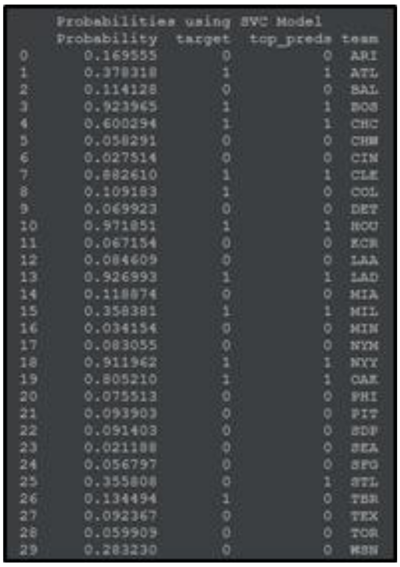


Figure 3: ROC AUC Curve.



```
      Probabilities using SVC Model
      Probability  target  top_preds team
0        0.169555       0          0  ARI
1        0.378318       1          1  ATL
2        0.114128       0          0  BAL
3        0.923965       1          1  BOS
4        0.600294       1          1  CHC
5        0.058291       0          0  CHW
6        0.027514       0          0  CIN
7        0.882610       1          1  CLE
8        0.109183       1          0  COL
9        0.069923       0          0  DET
10       0.971851       1          1  HOU
11       0.067154       0          0  KCR
12       0.084609       0          0  LAA
13       0.926993       1          1  LAD
14       0.118874       0          0  MIA
15       0.358381       1          1  MIL
16       0.034154       0          0  MIN
17       0.083055       0          0  NYM
18       0.911962       1          1  NYY
19       0.805210       1          1  OAK
20       0.075513       0          0  PHI
21       0.093903       0          0  PIT
22       0.091403       0          0  SDP
23       0.021188       0          0  SEA
24       0.056797       0          0  SFG
25       0.355808       0          1  STL
26       0.134494       1          0  TBR
27       0.092367       0          0  TEX
28       0.059909       0          0  TOR
29       0.283230       0          0  WSN
```

Figure 4: Probability for each Team.

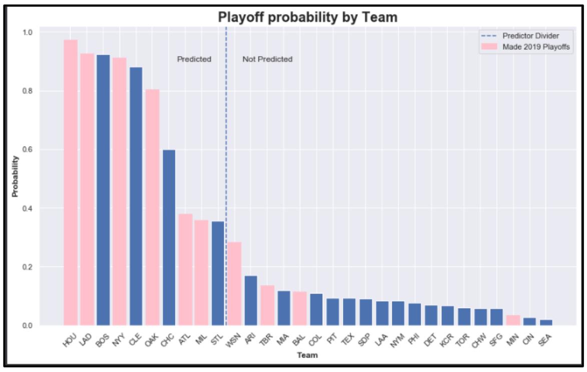
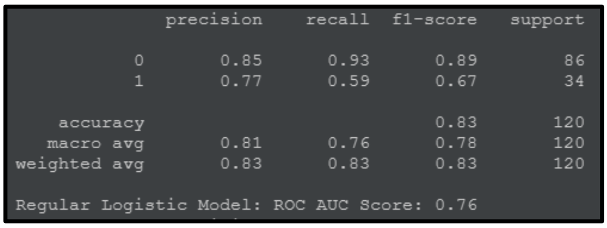


Figure 14: Playoff Probability by Team.



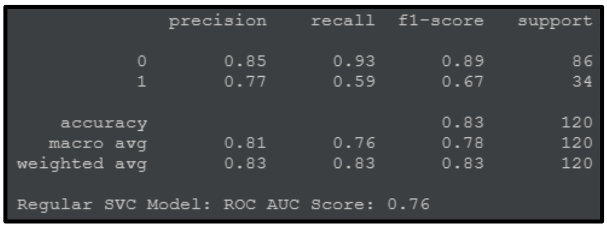
```
              precision    recall  f1-score   support

           0       0.85      0.93      0.89        86
           1       0.77      0.59      0.67        34

    accuracy                           0.83       120
   macro avg       0.81      0.76      0.78       120
weighted avg       0.83      0.83      0.83       120

Regular Logistic Model: ROC AUC Score: 0.76
```

Figure 5: Standard Logistic Regression Model.



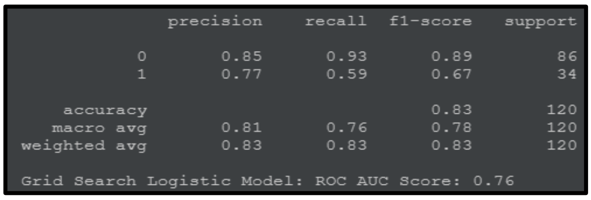
```
              precision    recall  f1-score   support

           0       0.85      0.93      0.89        86
           1       0.77      0.59      0.67        34

    accuracy                           0.83       120
   macro avg       0.81      0.76      0.78       120
weighted avg       0.83      0.83      0.83       120

Regular SVC Model: ROC AUC Score: 0.76
```

Figure 6: Standard SVC Model.



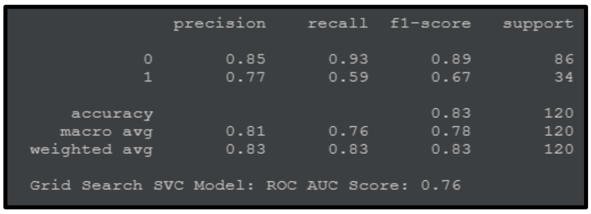
```
              precision    recall  f1-score   support

           0       0.85      0.93      0.89        86
           1       0.77      0.59      0.67        34

    accuracy                           0.83       120
   macro avg       0.81      0.76      0.78       120
weighted avg       0.83      0.83      0.83       120

Grid Search Logistic Model: ROC AUC Score: 0.76
```

Figure 7: Grid Search Logistic Regression Model.



```
              precision    recall  f1-score   support

           0       0.85      0.93      0.89        86
           1       0.77      0.59      0.67        34

    accuracy                           0.83       120
   macro avg       0.81      0.76      0.78       120
weighted avg       0.83      0.83      0.83       120

Grid Search SVC Model: ROC AUC Score: 0.76
```

Figure 8: Grid Search SVC Model.

SVC model) to see what probability each team has of getting to playoffs (see *Fig. 13*).

From Fig. 13, we can see what the probability of each team getting to the playoffs are. Fig. 14 gives us a better idea of who the algorithm thinks will make into the playoffs (based on Fig 13).

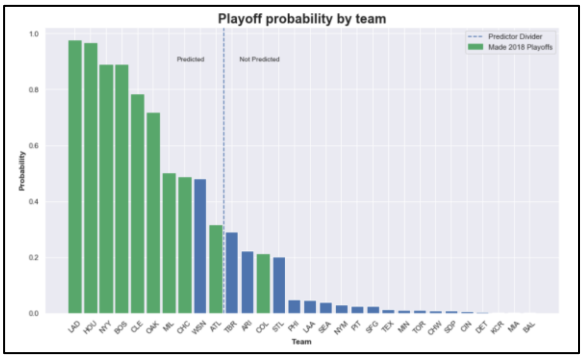


Figure 15: Jordan Bean's Playoff Probability by Team.

In Jordan Bean's research he provided 48 years of data, from 1969 – 2017 to make MLB playoff predictions for 2018 [7]. Bean measured baseball data using 10+ of pitching and batting stats whereas our model we only measured 8 total. He trained five different classification models including Logistic Regression, Random Forest, KNeighbors Classifier, Support Vector classifier (SVC) and an XGBoost Classifier. For his model prediction, based on his tuned grid search models, the best results came from the XGBoost Classifier, which had a precision of 1.0 meaning that all predictions made the playoffs and a recall of 0.80 which means that the model correctly identified 8 out of the 10 teams that made it to the playoffs. From all of his models the one that performed the best was the SVC model. Using the SVC model he created a data frame and plotted the playoff probabilities by team (Figure 15).

Bean's results correctly predicted 9 out of 10 teams that made it to the playoffs in 2018. The only mistake made was predicting Washington would make it to the playoffs and Colorado would not.

5 Introduction

The original goal of this project was to be able to predict which teams would make it to the playoffs, from our testing we can see that teams that had a probability higher than ~.3 were selected by our algorithm. When we take fig. 6 and convert it into a graph (fig. 7) we can see that the top 10 predicted teams are as followed (teams to the left of the "Predictor Divider"). From here, we can see that the algorithm was only able to predict ~60% of the teams correctly (teams in blue did not make playoffs) which is comparable with that results obtained by Soto [25] which used the nearest technique we used.

While the other results of [23, 24, 26, 27] get results better than our because of additional optimization techniques used which in turn increase the latency. We find it interesting that for all four of our tests, regardless of which model we used nor if we optimized them or not returned with an exact output for each. This may be caused by very low variation in our dataset. We hypothesized that at least 20 years' worth of data will give us a close enough result as Jordan Bean's research due to the amount of games played per season and the many factors that can change a single game. Comparing both results it is clear that using more data would've given us a more accurate prediction.

References

[1] "2019 MLB Team Statistics," 16 March 2020. [Online]. Available: https://www.baseball-reference.com/leagues/MLB/2019.shtml. [Accessed 17 March 2020].

[2] Adams, Mark. "The Man Behind Moneyball: The Billy Beane Story: Domo." Connecting Your Data, Systems & People, Domo, 24 Feb. 2015, www.domo.com/blog/the-man-behind-moneyball-the-billy-beane-story/.

[3] "A Guide to Sabermetric Research," [Online]. Available: https://sabr.org/sabermetrics.

[4] Blackburn, Ghoji. "What Is Fantasy Baseball? How Do I Play It?" Fake Teams, Fake Teams, 16 Mar. 2017,

www.faketeams.com/2017/3/16/14942064/what-is-fantasy-baseball.

[5] D. Prasetio and D. Harlili, "Predicting football match results with logistic regression," 2016 International Conference On Advanced Informatics: Concepts, Theory And Application (ICAICTA), George Town, 2016, pp. 1-5. https://doi.org/10.1109/icaicta.2016.7803111

[6] J. Bean, "Modeling MLB's 2018 Playoff Teams," 9 October 2018. [Online]. Available: https://towardsdatascience.com/modeling-mlbs-2018-playoff-teams-b3c67481edb2. [Accessed 17 March 2020].

[7] J. Bean, "Modeling MLB's 2018 Playoff Teams," 9 October 2018. [Online]. Available: https://towardsdatascience.com/modeling-mlbs-2018-playoff-teams-b3c67481edb2. [Accessed 17 March 2020].

[8] J. Dutcher, "Book Review: Moneyball: The Art of Winning an Unfair Game," 28 March 2014. [Online]. Available: https://datascience.berkeley.edu/moneyball-book-review/. https://doi.org/10.5860/choice.41-4733

[9] J. Silverman, "How Sabermetrics Works," 21 January 2009. [Online]. Available: https://entertainment.howstuffworks.com/sabermetrics.htm.

[10] K. Fuchs, "Machine Learning: Classification Models," 28 March 2017. [Online]. Available: https://medium.com/fuzz/machine-learning-classification-models-3040f71e2529. [Accessed 17 March 2020].

[11] Lashbrook, Lynn. "Why Baseball Analytics Matters and How You Can Make It into a Career.", SportsManagementWorldwide, 20 Jan. 2017, www.sportsmanagementworldwide.com/content/why-baseball-analytics-matters-and-how-you-can-make-it-career.

[12] "List of Major League Baseball Postseason Teams." Wikipedia, Wikimedia Foundation, 1 Nov. 2019, en.wikipedia.org/wiki/List_of_Major_League_Baseball_postseason_teams.

[13] Lutins, Evan. "Grid Searching in Machine Learning: Quick Explanation and Python Implementation." Medium, Medium, 5 Sept. 2017, medium.com/@elutins/grid-searching-in-machine-learning-quick-explanation-and-python-implementation-550552200596.

[14] "Major League Baseball Team Win Totals." Baseball, Baseball-Reference, www.baseball-reference.com/leagues/MLB/.

[15] Micahmelling@gmail.com. "Using Machine Learning to Predict Baseball Hall of Famers." Baseball Data Science, 27 Sept. 2017, www.baseballdatascience.com/using-machine-learning-to-predict-baseball-hall-of-famers/.

[16] "Moneyball." Moneyball (2011), IMDb.com, 23 Sept. 2011, www.imdb.com/title/tt1210166/.

[17] N. Paine, "The Imperfect Pursuit of a Perfect Baseball Forecast," 27 March 2014. [Online]. Available: https://fivethirtyeight.com/features/the-imperfect-pursuit-of-a-perfect-baseball-forecast/.

[18] Pharr, Roger D. "Predicting MLB Game Outcomes with Machine Learning." Medium, Towards Data Science, 3 Aug. 2019, towardsdatascience.com/predicting-mlb-game-outcomes-with-machine-learning-594eac9484e9.

[19] Raschka, Sebastian. "Predictive Modeling, Supervised Machine Learning, and Pattern Classification." Dr. Sebastian Raschka, 25 Aug. 2014, sebastianraschka.com/Articles/2014_intro_supervised_learning.html

[20] R. Ribeiro, "Houston Astros Strive for Balance Between Quantitative and Qualitative Data Analytics," 3 July 2014. [Online]. Available: https://biztechmagazine.com/article/2014/07/houston-astros-strive-balance-between-quantitative-and-qualitative-data-analytics.

[21] S. Banerjee, "Linear Regression: Moneyball - Part 1," 15 April 2018. [Online]. Available: https://towardsdatascience.com/linear-regression-moneyball-part-1-b93b3b9f5b53.

[22] S. Banerjee, "towardsdatascience," 1 June 2018. [Online]. Available: https://towardsdatascience.com/linear-regression-moneyball-part-2-175a9dc72e89.

[23] Brandon Tolbert, Theodore Trafalis " Predicting Major League Baseball Championship Winners through Data Mining" 2016. Thens Journal of Sports - Volume 3, Issue 4– Pages 239-252 https://doi.org/10.30958/ajspo.3.4.1

[24] Jones, J.; Johnston, K.; Farah, L.; Baker, J. 2021"Predicting Seasonal Performance in Professional Sport: A 30-Year Analysis of Sports Illustrated Predictions". Sports 2021, 9, 63. https://doi.org/10.3390/ sports9120163.

[25] Soto Valero, C." Predicting Win-Loss outcomes in MLBRegular season games – A comparative study using data mining methods" 2016, International Journal of Computer Science in Sport Volume 15, Issue 2. https://doi.org/10.1515/ijcss-2016-0007

[26] Chia-Hao Chang, "Construction of a Predictive Model for MLB Matches", 2021. Forecasting 2021, 3, 102–111. https://doi.org/10.3390/forecast3010007

[27] Ting-Chun Yu and Jui-Chung Hung," Forecasting MLB Playoff Teams Using GA-SVM", 2017. IEEE-ICASI 2017. https://doi.org/10.1109/icasi.2017.7988450.

[28] Al, Noor M. Al-Moosawi M., and Raidah Salim Khudeyer. "ResNet-34/DR: A Residual Convolutional Neural Network for the Diagnosis of Diabetic Retinopathy." Informatica 45.7 (2021). https://doi.org/10.31449/inf.v45i7.3774.

[29] Raheem, Sabreen Fawzi, and Maytham Alabbas. "Dynamic Artificial Bee Colony Algorithm with Hybrid Initialization Method." Informatica 45.6 (2021). https://doi.org/10.31449/inf.v45i6.3652.

[30] Saddam, Saba Abdual Wahid. "Wind Sounds Classification Using Different Audio Feature

Extraction Techniques." Informatica 45.7 (2022). https://doi.org/10.31449/inf.v45i7.3739.

[31] Ampomah, Ernest Kwame, et al. "Stock Market Prediction with Gaussian Naïve Bayes Machine Learning Algorithm." Informatica 45.2 (2021). https://doi.org/10.31449/inf.v45i2.3407.