Predicting situation-specific OPS in MLB

Christian Chase*

March 2022

Abstract

On-base percentage plus slugging percentage (OPS) in baseball is classified as the percentage of times a player gets on base plus his slugging percentage (total bases per at-bat). In order to quantify the predicted OPS for any given at-bat, we first use hierarchical clustering to cluster individual hitters – and separately pitchers – before running a hierarchical clustering system to cluster batters and hitters into specific groups based on their categorical similarities. Then, utilizing these clusters, we will model a separate multinomial regression for each pair of hitter and pitcher clusters in order to model the probabilities of various outcomes in an at-bat dependent on hitter/pitcher matchups. We will use each individual interaction of clusters in order to see whether or not we can predict OPS for certain types of players at any given moment. The data used was taken from the modern StatCast Era (2015-2019) and we will attempt to provide an easy-to-use model and application for a more effective formula for MLB teams to utilize when pinch-hitting, selecting relief pitchers, and signing Free Agents in the off season.

1 Introduction

For years, one of the most common metrics for measuring baseball hitters' success was *batting average* (i.e., number of hits divided by the number of at-bats). With the increased popularity of more advanced performance metrics – specifically, after the publication of Moneyball – it has been showed that *on-base percentage* (percentage of at-bats where the hitter ends up on base) is more highly correlated with runs scored and wins than is batting average. However, on-base percentage alone does not speak to a players' hitting power: for this, *slugging percentage* (total number of bases a player records per at-bat) has become most popular. Together, *on-base percentage* and *slugging percentage* speak to a players' ability to

*Advised by Aaron J. Molstad

	J.T. Realmuto	Salvador Perez	Yasmani Grandal
BA	0.263	0.273	0.240
OPS	0.782	0.859	0.940

Table 1: Batting average (BA) and on-base plug slugging percentage (OPS) for three of the top catchers in the MLB.

get on base, and speaks to how many bases they get during an at bat. A natural combined statistic, *on-base plus slugging*, abbreviated OPS, is simply the sum of on-base percentage and slugging percentage.

Both on-base and slugging percentage have their own merits, but when combined together tell a more convincing story about each player within the game. Traditionally, power hitters have the highest slugging percentage due to their ability to hit more extra base hits (home runs, triples, doubles). On-base percentage is not as cut and dry, leaving a wide variety of hitters to have the highest OBP ranging from hitters who hit for a high average (i.e., Ichiro Suzuki) to hitters who walk a lot (i.e., Yasmani Grandal). In order to demonstrate the effectiveness of OPS as a statistic, we can show how OPS compares to Average in the case of 2021 MLB catchers.

See Table 1. Arguably the three best catchers in baseball, Realmuto, Perez, and Grandal highlight the importance of OPS in quantifying offensive prowess. On paper, Grandal gets a hit in a measly 24 percent of his at-bats, trailing both other players by a significant margin; however, his OPS is 0.940, leading this elite group by a wide margin. Grandal has walked a staggering 87 times in the 2021 season, getting on base 42 percent of the time, providing a hidden value that average can't show. Salvador Perez walks significantly less (only 28 times in 2021), but his OPS was bolstered by 48 home runs which leads to a high Slugging percentage (Realmuto - 17, Grandal - 23). Obviously, these three players have their own strengths and weakness on defensive end of the game, but offensively, OPS shows a more complete story.

Due to its versatility, OPS has become a common statistic for measuring a hitter's success on the field. In addition, a pitcher's OPS against – that is, the OPS of the hitters they face – is a useful statistics for quantifying their performance. In this project, we build a user-friendly application to predict a player's situation-specific OPS (or OPS against). By doing so, the intended goal is to determine which hitters/pitchers are best in any given situation. After accounting for a multitude of situational variables such as inning, outs, score, runners on base, and the characteristics of the opposing pitcher, we are able to model

the probability of every possible outcome of an at-bat. From there, based on the estimated outcome probabilities, a predicted OPS can be obtained straightforwardly from the formulas provided in a later section. Such an application is meant to guide in-game decision making and the construction of lineups.

The remainder of this article will go into the methods, analysis, and results regarding situational OPS.

2 Methods

2.1 Overview

Given the randomness of an individual MLB at-bat, to predict OPS with reasonable accuracy requires data from a large number of at-bats. Thankfully, there is an enormous amount of at-bat-level data available through StatCast (MLB, 2021). To characterize the situation in a particular at-bat, we used inning, score differential, number of outs, and players on base. Our initial goal was to characterize how a particular hitter would fair against a particular pitcher in a given situation. Unfortunately, to estimate hitter-pitcher combination-specific effects is unrealistic: many hitters do not face a particular pitcher during the course of a season, and even if they do, it is often for only a handful of at-bats. For this reason, we focused our attention on first clustering hitters and pitchers, separately into groups of players who have similar characteristics. To increase the reliability of our model, we include only hitters who had at least 100 at-bats, and pitchers who faced at least 20 hitters during the 2017-2019 seasons. Once clusters were well-defined, we then fit a multinomial logistic regression model (Agresti, 2018) with cluster-combination specific effects. From the estimated probabilities obtained from the multinomial logistic regression model, it is straightforward to predict OPS for a given situation.

In the remainder of this section, we will describe each step of this model-building process in more detail.

2.2 Data

We obtained the data used for both clustering and model fitting from StatCast. As mentioned, we used data available from the years 2017-2019 covering three major league seasons (2017, 2018, 2019). This three year span provides an ample amount of data that is consistent (StatCast data from 2015-2016 are generally regarded as less accurate, and 2020 was a

shortened 60 game season due to COVID-19).

We obtain mutiple dataset: data on the overall statistics of both hitters and pitchers over 2017-2019 in order to perform clustering, and data from every major league at-bat during this window.

2.3 Clustering

First, we clustered both hitters and pitchers separately. As mentioned, the goal is to cluster hitters according to shared characteristics. That is, each cluster to represent a group of hitters who will fare approximately equally in all situations.

In order to construct these clusters, we used numerous hitting characteristics that describe a hitter's tendencies: launch angle launch speed, spray chart statistics, and how they hit the ball (ground ball, line drive, fly ball). For launch angle and speed, we used 25th, 50th, and 75th percentiles for both. For spray chart statistics, we used percentage of balls hit to left, right, and center field. For how they hit the ball, we used percentage ground balls, percentage line drives, and percentage fly balls. Finally, to speak to a hitters' plate discipline, we also used walk percentage and strikeout percentage. If we knew a hitter's walk and strikeout percentage, along with all the aforementioned statistics on their balls in play, we would have a strong sense of the hitter's overall performance. Finally, as it is accepted that left-handed and right-handed hitters tend to fare differently against, e.g., the average left-handed pitcher, we created clusters separately for both left-handed and right-handed hitters.

We constructed clusters in R using hierarchical clustering based on Ward's minimum variance method. For the sake of simplicity, we created five clusters for both left-handed and right-handed hitters. We display the clustering dendograms in Figure 1. Interestingly, these clusters are highly interpretable. The clusters picked up on which players were pitchers, and therefore struggled at the plate solely based on their tendencies. In some cases, an National League starting pitcher would meet the cutoffs we designated for hitters (> 100 at-bats), and our approach tended to cluster these 'non-hitters' together based on their tendencies at the plate (as we would hope). Other clusters tended to consist of players often regarded as having similar traits (e.g., "power hitters").

To cluster pitchers, we repeated a similar process. Clustering both left-handed and right-handed pitchers separately, we clustered based on 10th, 30th, 50th, 70th, and 90th percentiles of release speed and spin rate; the three-dimensional coordinates (x, y, and z planes) of release position; walk and strikeout percentages; as well as their opponents' 25th, 50th, and 75th percentiles launch angle and lauch speed; and their oppenents' fly ball, line

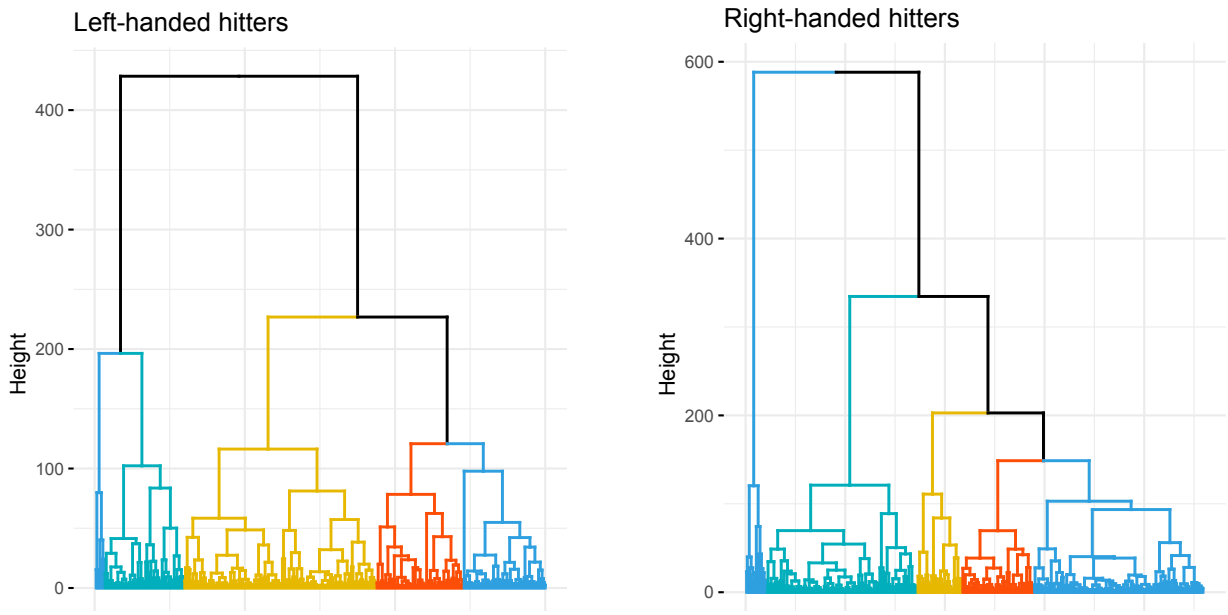

Figure 1: Cluster dendograms for both left-handed and right-handed hitters.

drive, and ground-ball percentage.

As before, we used hierarchical clustering based on Ward's minimum distance criteria. Also as will hitters, pitcher clusters were easily interpretable. For example, focusing on left-handed pitchers, one cluster included pitchers like Rich Hill, Clayton Kershaw, Aroldis Chapman; whereas another included CC Sabathia, Madison Bumgarner, and Jon Lester. These clusters tell us that these pitchers have similarities in their tendencies on the mound. Hill, Kershaw, and Chapman all have power breaking balls they use to their advantage, while Bumgarner, Sabathia, and Lester give up hard contact a small percentage of the time with lots of diversity in their pitches.

2.4 Modeling at-bat outcomes

With clusters in hand, our goal was fit a model to answer the question: "Given the same situation and the pitcher's/hitter's tendancies, what is the probability of each outcome of this at-bat?" To address this question, we fit a multinomial logistic regression model separately for each hitter and pitcher cluster pair. The response in our model, Y is a categorical random variable

$$Y \in \{FO, GO, LO, HBP, SF, 1B, 2B, 3B, HR, SO, W\}$$

FO	GO	LO	HBP	SF
Fly out	Ground out	Line out	Hit by pitch	Sacrifice fly

1B	2B	3B	HR	SO	W
Single	Double	Triple	Home run	Strikeout	Walk

Table 2: Outcomes in out multinomial logistic regression model

where the outcomes in the set of possible outcomes $\mathcal{O}$ are described in Table 2. Given that the hitter belongs to cluster j and the pitcher belonged to cluster k, we fit the multinomial logistic regression model which assumes

$$P_{j,k}(Y = 1 \mid x) = \frac{1}{1 + \sum_{s=2}^{11} \exp\left(\tau_s + \alpha_s^{(j)} + \eta_s^{(k)} + x^\top \beta_s\right)}$$
$$P_{j,k}(Y = i \mid x) = \frac{\exp\left(\tau_i + \alpha_i^{(j)} + \eta_i^{(k)} + x^\top \beta_i\right)}{1 + \sum_{s=2}^{11} \exp\left(\tau_s + \alpha_s^{(j)} + \eta_s^{(k)} + x^\top \beta_s\right)}, \quad i = 2, \ldots, 11$$

for all $(j, k) \in \{1, \ldots, 10\} \times \{1 \ldots, 10\}$, where x is a vector consisting of the game situation covariates and $\eta_i^{(1)} = \alpha_i^{(1)} = 0$, for all i. As one can see, our model assumes that the situation specific effects – encoded by the β_i – are identical across the combinations of clusters. The intercepts $\alpha_i^{(j)}$ and $\eta_i^{(k)}$ encode the cluster-specific effects. For the game specific covariates x, these include indicators for runners on 1st base, 2nd base, 3rd base and all their interactions, the number of outs in the inning, the inning, and the score differential at the time.

We fit a multinomial logistic regression model to hundreds of thousands of at-bats from the StatCast data. To do so, we used the `multinom` function in R and increased the number of iterations to 2000. We display some of the estimated regression coefficients in Figure 2. Here, we see for example, that in hitter cluster 2 the odds of strikeout versus double (which is the reference category) is higher than in cluster 1 (since $\hat{\eta}_{SO}^{(2)} > 0$).

When we compared to the model which included an interaction between clusters via likelihood ratio test, we saw no significant improvement in model fit.

2.5 Predicting OPS

Finally, with the fitted model, we are able to straightforwardly estimate OPS for a given situation. As mentioned, OPS is simply on-base percentage plus slugging percentage. To

	(Intercept)	hitter_cluster2	hitter_cluster3	hitter_cluster4	hitter_cluster5	hitter_cluster6
fly_out	1.17109863635829	0.0154705458348936	0.178412479636879	0.0047447125569949	0.0094011830936457	0.035395632272318
ground_out	1.56148974502389	0.367340331404697	-0.224636523039201	0.998235925142424	-0.11149452862321	0.0174599119275321
hit_by_pitch	-1.82537404594689	0.00812332828012421	0.0431980629792655	0.793600076627143	0.170333777698491	0.0260571303050378
home_run	-0.43619205804958	-0.378483047939352	0.343584126041943	-0.929804611041909	0.490679725146805	0.058541105025085
line_out	0.396701272154296	0.166039653357289	-0.117609306081052	0.242693861605382	-0.194665204007808	0.00266508940708885
sac_fly	-15.184190562676	0.0730920563325573	-0.0377377732254843	0.0492992592850876	-0.181587745056326	-0.0374514294886308
single	1.15597113782997	0.304800204577417	-0.248420694214354	0.473801915382146	-0.170028748439567	0.0127802333932599
strikeout	1.14597136529873	0.388880540422703	0.26889672527193	1.88715015360169	0.489760693497069	0.0481335180771517
triple	-2.38719705306269	0.555202349702678	-0.395000881705368	-0.671892109053837	-0.121816385873279	-0.306186282330409
walk	0.337035227581247	0.163932169546739	0.275558346744636	0.00495767479535204	0.317329960660163	-0.275450388325484

Figure 2: Estimated intercepts τ_i and hitter-cluster effects, $\hat{\eta}_i^{(2)}, \ldots, \hat{\eta}_i^{(6)}$.

compute on-base percentage, the standard formula is on-base percentage (OBP) is

$$OBP = \frac{\#H + \#BB + \#HBP}{\#AB + \#BB + \#HBP + \#SF}$$

where $\#H$ is the number of hits (1B, 2B, 3B, or HR), #BB is the number of walks, #HBP is number of times getting hit by pitch, #AB is the total number of at-bats, and #SF is the number of sacrifice flys. Given probabilities of each of these events in a particular at bat, we can estimate this quantity for a particular pitcher and hitter cluster combination (j, k) using

$$\widehat{OBP}_{(j,k)}(x) = \frac{P_{j,k}(Y \in \{HBP, 1B, 2B, 3B, HR, W\} \mid x)}{P_{j,k}(Y \in \{FO, LO, GO, 1B, 2B, 3B, HR, SO\} \mid x) + P_{j,K}(Y \in \{HBP, W, SF\} \mid x)},$$

where together the events $\{FO, LO, GO, 1B, 2B, 3B, HR, SO\}$ constitute an outcome recorded as an at-bat. Then, to compute slugging percentage, the standard formula is

$$SLG = \frac{\#1B + 2 \times \#2B + 3 \times \#3B + 4 \times \#4B}{\#AB}$$

where $\#1B$ is the number of singles, and similarly for doubles, triples, and home runs. Following a similar logic to our estimate of on-base percentage, we have

$$\widehat{SLG}_{(j,k)}(x) = \frac{P_{j,k}(Y = 1B \mid x) + 2P_{j,k}(Y = 2B \mid x) + 3P_{j,k}(Y = 3B \mid x) + 4P_{j,k}(Y = 4B \mid x)}{P_{j,k}(Y \in \{FO, LO, GO, 1B, 2B, 3B, HR, SO\} \mid x)}.$$

Thus, with these quantities, we can then estimate

$$\widehat{OPS}_{(j,k)}(x) = \widehat{OBP}_{(j,k)}(x) + \widehat{SLG}_{(j,k)}(x).$$

3 R Shiny application

To make our fitted model useful to practitioners, we created an R shiny app which can be accessed via `https://christianchase.shinyapps.io/baseball/`. To build the application required loading the fitted model in a way in which one can readily access the estimated probabilities for any combination of the clusters and situation-specific covariates. The interface of the app, displayed in Figure 3, is relatively simple. It allows one to choose the game situation characteristics as aforementioned: how many outs, what innings, score differential, runners on base. Then one can choose which hitter is at the plate and which pitcher is on the mound. When you click the submit button, the predicted OPS of that hitter in that situation will appear below.

To demonstrate how this tool could be used in-game, we focus on two case studies.

4 Case Study: Mike Trout vs Jacob deGrom

For the first example of situational OPS, it makes sense to look at the best hitter and best pitcher in the MLB. Trout and deGrom dominated the league during this time frame, with Trout winning a MVP (Most Valuable Player) and deGrom winning two Cy Young Awards (Best Pitcher Award).

4.1 Start of Game OPS

At the beginning of a game, there are no runners no, inning equals one, and there are no outs. In this case, the predicted OPS is 0.789. This is lower than Trout's career OPS (1.002), but can be explained by the fact that Jacob deGrom is a dominant pitcher, and by the fact that Trout is represented here by his cluster, not him directly.

Next, we will see how having runners on base affects the predicted OPS.

Figure 3: Demonstration of Shiny application (top panel) with hitter Mike Trout, pitcher Jacob deGrom, when there are no outs, the hitter's team is losing by two, in the seventh inning with players on second and third base; (bottom panel) with hitter Billy Hamilton, pitcher Jacob deGrom, a tie game in the first inning with no runners on.

	Empty	1st Only	2nd Only	3rd Only
OPS	.789	.824	.801	.848
	1st and 2nd	2nd and 3rd	1st and 3rd	Full
OPS	.805	.824	.862	.837

Table 3: Predicted OPS under all possible base runner alignments with hitter Mike Trout and pitcher Jacob deGrom.

4.2 Runners Effect

In Tables 3, we display the predicted OPS in the same situation, modified only to include base runners. Tables 4 and 5 confirm the conventional wisdom that players hit better with runners on base. Possible reasons for this include the pitcher being more distracted, pitchers pitching worse from the stretch, and fielders being forced into positions which are not ideal to getting the hitter out. The fact that our model recovers this is reassuring: that more base runners lead to higher OPS will be true for every hitter and pitcher clustering pair.

Next, we assess how inning affects the predicted OPS.

4.3 Inning Effect

The predicted OPS for Mike Trout versus Jacob deGrom with no runners on is provided in Table 4. The inning effect tells an interesting story: namely, the predictions suggest OPS is highest in the 3rd, 4th, and 5th innings, which would be a pitcher's 2nd or even 3rd time through the batting order. It has been shown extensively that a hitter's performance improves the more times they face a particular pitcher. Furthermore, the OPS then drops, most likely as relief pitchers enter the game. It is lowest in the 9th inning, which is likely the inning teams face a dominant closer.

We omitted extra innings from our model because of the recent change in MLB rules where a runner starts on second base.

4.4 Outs Effect

Finally, we assess how the number of outs affects the predicted OPS in 5. The model once again fits along with common baseball knowledge and the run expectancy matrix. The more outs, the less likely that a batter gets on base. Our model again agrees strongly with conventional wisdom.

	Inning 1	Inning 2	Inning 3	Inning 4	Inning 5
OPS	.789	.751	.805	.805	.805
	Inning 6	Inning 7	Inning 8	Inning 9	
OPS	.785	.748	.74	.713	

Table 4: Predicted OPS across nine innings with hitter Mike Trout and pitcher Jacob deGrom.

	0 outs	1 outs	2 outs
OPS	.789	.773	.728

Table 5: Predicted OPS under various numbers of outs with hitter Mike Trout and pitcher Jacob deGrom.

The score differential did not significantly impact the predicted OPS. Within the discussion, we will offer possible explanations for this.

4.5 Billy Hamilton vs Jacob deGrom

In this section, we compare the results from the previous subsections to those with another combination of hitter and pitcher, now focusing on pitcher Jacob deGrom versus Bill Hamilton. This contrasts Mike Trout almost directly, as Billy Hamilton is a completely different hitter. Trout utilizes power and contact whereas Hamilton has no power and lower contact abilities. Hamilton's prowess and value is as a baserunner, not as a batter.

As before, we began by considering the OPS at the start of the game (i.e., no outs, no one on, first inning). As mentioned Billy Hamilton is widely accepted to be a worse hitter than Trout and his cluster's attributes bear that out in terms of predicted OPS: .669. Note that this is more than 0.1 lower than the predicted OPS for Mike Trout versus Jacob deGrom in this scenario.

The predicted OPS with various alignments of base runners – provided in Table 6 again agrees with the well accepted idea that players will hit better with runners on base. Of course, compared to the same table constructed with Mike Trout as hitter, all predicted OPS are lower here. Of course, my model assumes that the all other covariates held fixed, base runners will affect the outcome of an at-bat in an identical way across clusters. Similar results hold for the effects of outs, score differential, etc.

	Empty	1st Only	2nd Only	3rd Only
OPS	.669	.708	.693	.734
	1st and 2nd	2nd and 3rd	1st and 3rd	Full
OPS	.691	.716	.742	.712

Table 6: Predicted OPS under all possible base runner alignments with hitter Billy Hamilton and pitcher Jacob deGrom.

5 Discussion

Our case study confirms that our model largely agrees with conventional wisdom in baseball. For the baserunner effect, it was clear that the OPS increased with runners on base. This makes intuitive sense, as better hitting has been observed in these situations. Most specifically, the highest OPS increases (in comparison with the base number) correlate directly with situations in which there are no force outs: runners on second and third, runner on second, runner on third, etc. Considering OPS is a statistic that involves walks, this increase could be because of a base being open, meaning pitchers are more inclined to walk the hitter in order to create a forceout situation.

For the inning effect, OPS predictions also cohere with existing baseball knowledge. For all combinations, the predicted OPS in the first inning of the game is low. Then it lowers a little more in the second inning, because in the second inning a pitcher will be facing the bottom of the order in most cases, or at least weaker hitters than the first inning. The predicted OPS then spikes in the third inning, most likely as the lineup turns over and hitters get a second look at the starting pitcher. The predicted OPS stays close to the same in the fourth and fifth innings. Then, the predicted OPS starts to drop continually the rest of the game. At this point, relief pitchers come into the game with strong pitches and tendencies, which gives hitters a tough time. The predicted OPS hits its lowest in the 9th inning, unsurprisingly the inning in which teams will bring in their closing pitchers (best relievers).

For the score differential, there is no significant effect on OPS. There are many reasons as to why this could be. To offer the most plausible explanation: players always have a reason to hit, so the score itself doesn't factor into their mindset at the plate. Similarly, pitchers are incentivized to pitch as well as possible in all situations, so score differential should not significantly impact performance.

For the outs effect, hitters get worse on average the more outs there are in an inning.

Our analysis confirms this across all clusters.

5.1 Limitations

There are a number of fundamental issues which we failed to address when creating the application. First and foremost: switch hitters. In baseball, switch hitters tend to have different tendencies on both sides of the plate, so treating them as separate people works in terms of the algorithm. Take for example, Ozzie Albies, the Atlanta Braves second baseman. From the right side of the plate, his average was .250 and from the left side of the plate his average was .340. At present, our application treats all switch hitters as right-handed hitters: an artifact of the way the data were scraped from StatCast. In the future, it would be ideal to be able to select a hitter and their handedness if they are. switch hitter.

Secondly, the data we took from StatCast was only from a set amount of years, which means some of the players – most specifically rookies that debuted after the 2019 season – will not be included in the dataset and thus will lead to an error in our application submission form. This holds true for any players that did not meet our designated qualifications from 2017-2019 and for any players who did not play in this time frame. In order to make the process a more streamlined, we will provide a list of possible combinations to use, ranging from elite players (think Mike Trout and Jacob deGrom) to objectively lower-level players (think Billy Hamilton and Wei-Yin Chen).

Lastly, the predicted OPS that this application provides is a prediction over an infinite number of at-bats and should not be taken as the predicted OPS for a specific player per se. For example, if the application suggests Mike Trout will have an OPS of .850 in these situations, we instead interpret this to mean that players with Mike Trout's tendencies are expected to have an OPS of .850 in this situation. Mike Trout performs better than our expectation, meaning he over performs his cluster, whereas Billy Hamilton performs worse than his cluster.

5.2 Future directions

Of course, an ideal OPS prediction method makes individual level predictions where the effect of inning, outs, etc. are all dependent upon the combination of hitter and pitcher. As mentioned, this is infeasible since a hitter only faces a particular pitcher a handful of times each season. While we felt that clustering was natural, it is clear that this leads us to underpredict OPS for the very top hitters, and overpredict for the worst hitters.

Instead, in a future version of this project, it would be useful to develop a new way to estimate player-specific effects. While in this project we initially attempted this, this led to a model with extremely high-dimensional predictors. Using standard variable selection techniques was not helpful because in general, player specific effects were always removed from the model. This can be understood from the fact that a player-specific effect has a very small contribution to the overall likelihood. Other methods for fitting multinomial logistic regression models in baseball, e.g., (Powers et al., 2018) have used alternative forms of regularization, so this seems a natural approach to consider.

6 Conclusion

Two key conclusions can be made from this project. First teams can use this data to increase the chances they choose the correct pinch-hitter in any given situation later in the game. And second, they can have more data in the process of signing certain players, keeping them, or DFAing (dropping them) in the offseason. By using the application provided, it is clear how teams could use this to choose pinch hitters or relief pitchers based on certain cluster tendencies. As for DFAing players or finding hidden talent to sign, teams can look at over or underperformance within clusters. Mike Trout and Billy Hamilton serve as prime examples: Trout overperforms his tendencies and Hamilton underperforms them.

There is a lot more to do with this project as aforementioned. The first order of business will be to actually create situational statistics for players. Then we hope to expand the clusters to account for each new class of Rookies and Players and auto cluster them into the data so that they can be tested as well. From there, more analysis can occur and more predictive power will be found within the models.

References

Agresti, A. (2018). *Introduction to Categorical Data Analysis*. John Wiley & Sons.

MLB (2021). MLB StatCast Database. https://baseballsavant.mlb.com/statcast_search.

Powers, S., Hastie, T., and Tibshirani, R. (2018). Nuclear penalized multinomial regression with an application to predicting at bat outcomes in baseball. *Statistical modelling*, 18(5-6):388–410.