Predicting Team Outcomes Based on Player Composition

Jeffrey Vartabedian, Charles Williams, and Ryan Peet

Introduction

For decades in professional baseball, teams and their management have often used statistical methods to attempt to construct the optimal roster for both regular season and postseason success. This project is another attempt to do something similar, through the lens of machine learning techniques. Combining player performance metrics with salary and payroll data, we aim to model how the roster constitution of a specific number of player archetypes translates to regular season success.

Our unsupervised task will first explore the natural groupings which exist within teams and between players. We plan to cluster players into groups based on their regular season statistics, including metrics like their salary, source, tenure, position, and wins above replacement. Our supervised task will then use these groupings to predict the number of regular season wins for a MLB team with a specific roster constitution, based upon the clustering of players output by the unsupervised model.

To support these analyses, we will use data from Baseball-Reference for player statistics and Spotrac for salary data, attempt to join the two datasets using a generated unique ID, perform feature selection methods from the performance data, and run an unsupervised clustering algorithm to attempt to create player archetypes for both batters and pitchers. We will then use a supervised learning method to try to predict the number of regular season wins by teams with a specific roster constitution and market size.

Related Work

Over the past few decades, the growing use of analytics in baseball has coincided strongly with the growing disparity in team payrolls. Subsequently, many studies have been conducted to help either understand the financial value of each player and their respective contribution to their team's success. One early study titled: "Estimating the Value of Major League Baseball Players" used regression to analyze team revenue and winning percentage to see if player's salaries corresponded with their marginal revenue product (MRP). The timeframe used in the study overlapped with the 1994 player's strike and the then new collective bargaining agreement that afforded players a larger share of revenues. This led to the finding that where players were often underpaid before, they now boasted salaries that more closely reflected their economic value (Fields, 2001). While these changes led to a more equitable distribution of money in the sport, they also contributed to the growing gap in payroll between large market and small market teams where revenue disparities are much more pronounced. As a result, more recent studies have now aimed to more efficiently evaluate player performance to help teams close the margins created by payroll. A 2012 study by Ojanen, created a model "Linear Runs Estimated" that regressed using runs scored with offensive metrics and found that overpaid players were often older and free agents, while the younger players still under rookie contract rules were often underpaid (Ojanen, 2012). An even more recent study "How Have Advanced Statistics Impacted Salary in Major League Baseball" from 2019, analyzed the sabermetric statistic Wins Above Replacement (WAR) to see if it accurately predicted salaries for free agents.. The study found that while WAR was a good predictor of salaries, contracts were often still inflated due to possible future potential and each year's respective free agent market (Carney et al., 2019). While these studies all have a similar goal of understanding the financial value of players, we believe that through clustering and unsupervised modeling we can expand upon this analysis through a larger set of statistical variables by not forcing the categorization of players simply by salary and WAR. Our hope is that these groupings lead to a more comprehensive view of player valuation. They can then be used in a supervised modeling that evaluates roster compositions of both successful and unsuccessful teams to shed light on any trends in roster building that lead to success.

Data Source(s)

- **Baseball-reference**
    - Format Used: Copy and pasted tabular data in .csv files
    - Variables of Interest:
        - Position Players:
            - Name, Team, Age, Doubles (2B), Triples (3B), Home runs (HR), Strike Outs (SO), Base on Balls or Walks (BB), Intentional Base on Balls or Intentional Walks (IBB), Hit by Pitch (HBP), Batting Average (BA), On-Base + Slugging Percentages (OPS), Stolen Bases (SB), Caught Stealing (CS), Sacrifice Hits (SH), Ground into Double Play (GIDP), Wins Above Replacement (WAR)
        - Pitchers
            - Name, Team, Age, Wins (W), Earned Run Average (ERA), Park Adjusted ERA (ERA+), (Walks + Hits)/Innings Pitched (WHIP), Fielding Independent Pitching (FIP), Games Finished (GF), Complete Games (CG), Shutouts (SHO), Strikeout/Walk Ratio (SO/BB), Intentional Base on Balls or Intentional Walks (IBB), (9 x Walks)/Innings Pitched (BB9), Balks (BK), Wild Pitches (WP), Hit by Pitch (HBP), Wins Above Replacement (WAR)
    - Records Used: 31,497 individual player season records
    - Time Period: 2011-2019, 2021-2025

- **Spotrac**
    - Format Used: Copy and pasted tabular data in .csv files
    - Variables of Interest:
        - Name, Team, Season Salary
    - Records Used: 15,382 player salary records
    - Time Period: 2011-2019, 2021-2025

- **Wikipedia (U.S. Census Bureau)**
    - Format Used: Copy and pasted tabular data in .csv format
    - Variables of Interest:
        - Population in Millions
    - Records Used: 420 records

Data Manipulation

Our feature data manipulation commenced with a series of data cleaning tasks, which included the upload of the collected salary and performance data for all MLB players since 2011. After importing the data, one of the first steps that we took was to rename certain fields within the pitching data to not overlap with their similarly-named batting counterparts, including runs, hits, homeruns, base on balls, strikeouts, and hit-by-pitch.

The next step that we took was to create our own unique ID for each player in both the salary and performance datasets, which would allow us to merge the salary and performance data. The methodology that we used was to take the first letter of the first name, first four letters of the last name, the abbreviation of the team where they were active, as well as the year of the performance and salary data. These values were concatenated to create a unique ID which covered a large majority of the data that we collected, only missing approximately 404 players from both the salary and performance data after the data was merged.

After creating the unique IDs, we decided to remove all batting data from the pitchers as well as all batting data from the batters. In order to accomplish this, we used the player's position provided by Spotrac to define what data would be dropped. One special exception was excluded from this rule: Shohei Ohtani is such an effective batter and pitcher that the batting data for his specific case was not excluded, despite the fact that he is sometimes defined as a starting pitcher within our data.

Because the MLB currently utilizes a 26-man roster for a majority of the season, with a pretty even split per team between pitchers and batters, we decided to reduce the data to include only the top 13 hitters and top 13 batters per team for each year of the data collection. We reduced the data by taking the top 13 pitchers based on the number of innings pitched, as well as taking the top 13 batters based on their number of plate appearances. This resulted in a dataset with 10,921 records, encompassing the top 26 players from each of the 30 MLB teams for each year of data collection, between 2011-2024. This method also served us to avoid overweighting certain teams or certain eras for our clustering model, as MLB rules have changed to reduce the number of active players on a team's roster over the last 15 years.

Feature Selection/Engineering

For our feature selection in preparation for unsupervised clustering, we separated the batting and the pitching data as the performance attributes for pitchers and hitters are very different in nature, and would not lead to accurate clustering if the data was combined. We also removed salary from our data, as salary serves as a major component of our analysis to affect player value, and we wanted to include this metric after our feature selection to ensure that it was a major component within our clustering model. After separating the data, we ran a Predictive Attribute Dependence model on the data to start selecting redundant features and remove them from the data. To accomplish this, we first standardized the columns, then, feature by feature within our data, clustered on all the other columns using a K-means implementation and measured how well the clusters explain the held-out column using an ANOVA R-squared metric.

After running the PAD model and assessing the R-squared metrics, we ranked the numeric features (other than player salary) by the R-squared metric and pruned from the top down until the scores fell below a certain threshold, which we decided would be .65. We decided to use a threshold method to select redundant features rather than an elbow method because of the fact that the redundancy scores generated by the model do not exhibit a clear inflection point that exists in some elbow curves, as well as the fact that an elbow method may have left out features which are redundant in nature but valuable in the context of baseball, like home runs for batters or win/loss record for pitchers. Some of the redundant features that were removed which fell above the .65 threshold included plate appearances, runs batted in, and hits for batters, and games started, innings pitched, strikeouts, and runs for pitchers. Overall, as a result of the PAD, we removed 7 redundant features from the batting data, and 8 redundant features from the pitching data.

The second step of our feature selection process was correlation pruning, in which features which were highly correlated with other features were removed to help prevent multicollinearity and overrepresentation of related features within our model. By calculating pairwise correlation coefficients across our variables, we identified features which exceeded a correlation coefficient threshold of .82 and eliminated a feature from each correlated pair, typically the one that was less interpretable or more redundant based upon the results from the PAD. This ensured that the feature set contributed unique information to the clustering process, improving the stability and accuracy of our unsupervised model. Some of the features that were removed as a result of our correlation pruning include OPS+, rOBA, Rbat+, SLG, and OBP, which are highly correlated with the OPS statistic for batters. For pitchers, we removed the HR9 and SV statistics, which were highly correlated with FIP and GF, respectively. After our correlation pruning, we were left with 15 features for batters and 21 features for pitchers, not including salary.

The third and final feature selection method we used was Principal Component Analysis, which allowed us to reduce the dimensionality of our features set while retaining a vast majority of its variance, as well as construct new features based upon the previous features which capture more meaningful insights about player value. By transforming correlated features into a smaller number of uncorrelated principal components, we were able to capture the underlying structure of player performance in a more interpretable manner. In comparison to our base player performance statistics, these components summarize broad patterns in the data, like overall offensive production, workload and longevity of pitchers, baserunning aggression, and other latent traits which served as the foundation for our unsupervised clustering model.

After running the PCA model on both the batters and the pitching data, we aimed to select a number of principal components for each that explain approximately 90% of the variance in each player performance dataset. From this analysis of variance, we decided to use 9 components for the batting data and 12 components for the pitching data, and performed an analysis of the included PCA metrics to come up with distinct meanings for each of the components, which could be used in our clustering model.

Batter Performance Components (PCA)

Component	**Component Meaning**	**Description**
PC1	Overall Offensive Production	Total batting impact
PC2	Speed & Baserunning Aggression	Steals; small ball
PC3	Contact/On Base Efficiency	Hit and OBP consistency
PC4	Experience/Longevity	Veteran steadiness
PC5	Plate Toughness	Intimidation, respect
PC6	Situational Hitting	Run-scoring situational skill
PC7	Patience and Strikeouts	Plate discipline
PC8	Gap Power and Doubles	Line-drive power
PC9	Productive Baserunning	Basepath efficiency

Pitcher Performance Components (PCA)

Component	**Component Meaning**	**Description**
PC1	Workload/Volume	Innings; reliability

PC2	Run Prevention	Limiting runs/hits
PC3	Command/Control	Strike throwing, walk rate
PC4	Stamina/Complete Games	Endurance
PC5	Bullpen Leverage	Late-game usage
PC6	Strikeouts	Swings and misses
PC7	Veteran Pitching Style	Age-related adaptions
PC8	Mechanical Volatility	Inconsistency, control issues
PC9	Pressure/Composure	Command under duress
PC10	Relief_Effectiveness	Reliever impact
PC11	Overall Performance Value	Balanced performance value
PC12	Sabermetric Efficiency	Modern metrics composite

After performing an analysis of the PCA metrics and assigning meanings to each component, we aimed to add salary back into the data as an additional component in the batting and pitching datasets and weight salary on a different scale than any of the other metrics. Because salary is such an important metric in defining a player's value to any organization, we decided to weight salary differently than the other metrics. After normalization of the salary metric with a standard scaler, we multiplied it by a specific weight in both the batting and pitching datasets so that it would account for around 30% of the total sum of variances of all of the components.

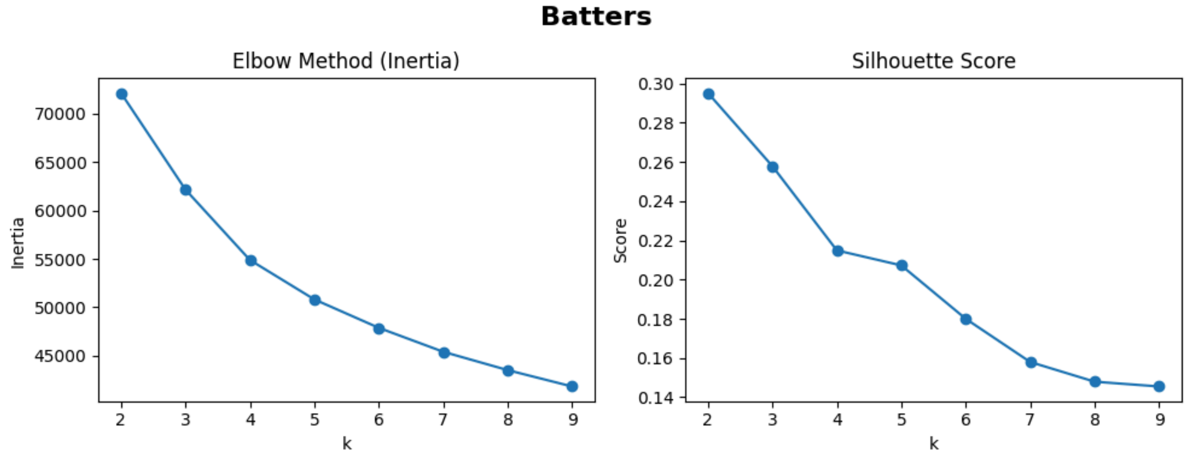
This adjustment allowed salary to have a meaningful influence on the overall clustering structure of our unsupervised model, ensuring that it significantly contributed to how players are grouped within the model, without completely dominating the performance-based components derived from our PCA methodology. By scaling salary to represent approximately 30% of the total variance, we balanced our quantitative performance indicators with economic valuation of the players, enabling our clusters to reflect both on-field performance and investment by the organization itself.

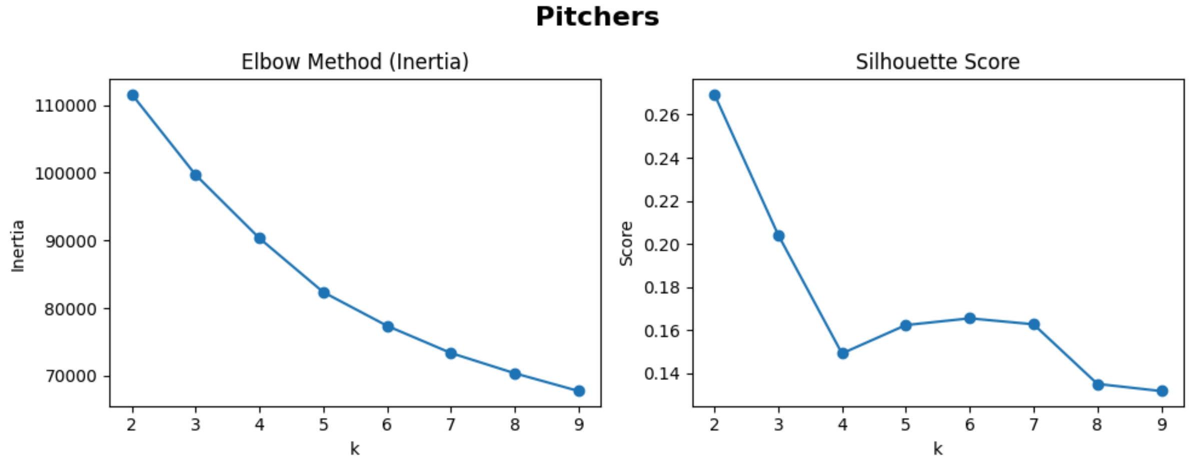
Our approach to feature selection resulted in a refined and balanced dataset for both batters and pitchers, and optimized both datasets for clustering based upon the most meaningful and representative features in our data. It positioned our analysis to generate clusters and reveal data-driven distinctions among players in terms of both performance and market value, forming the backbone of our overall study.

Part A. Unsupervised Learning

Data for the unsupervised model was imported from the feature selection dataset, and required no additional data manipulation in terms of scaling or analysis of components due to the nature of our feature selection method. To cluster on both the pitching data and batting data, we decided to use a K-means implementation for our clustering, as it provides a simple and powerful way to partition players into groups based upon their performance and salary characteristics.

We tested a range of possible cluster counts (from k=2 to k=10) using both the elbow method to assess reduction in inertia and silhouette score to evaluate the separation quality of our clusters. The results of this testing was highly similar between both batters and pitchers, the 'best k' for both our batting and pitching stats centered around k=4 utilizing both the elbow method for inertia and the silhouette score:

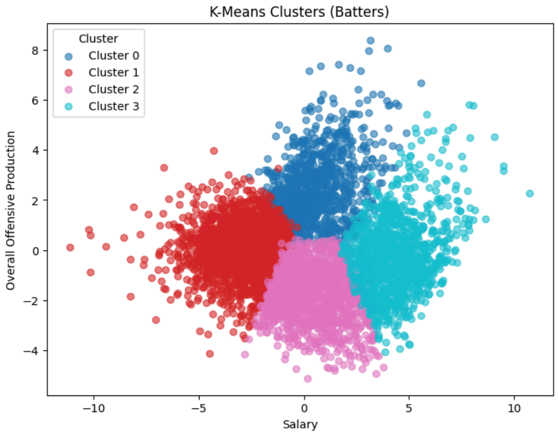


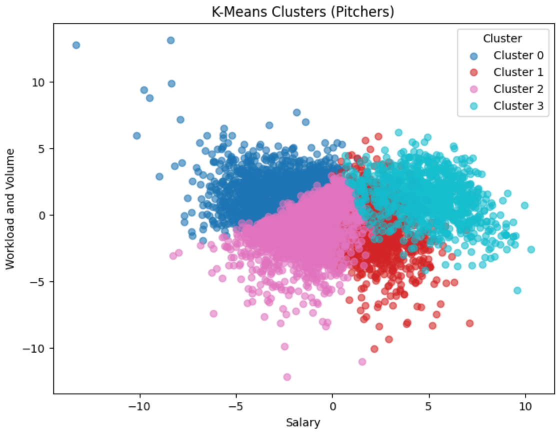


As exemplified above, the inertia curve for both batters and pitchers follows a gradual decline, while the silhouette score forms a noticeable valley with local improvement around k=4. This pattern in our data suggests that four clusters represent the most stable configuration before diminishing returns and noise begin to increase at higher values of K. This also suggests that clustering with more than four groups provides minimal improvement and introduces unnecessary fragmentation within the player archetypes we are trying to construct. Therefore, we used a K-means implementation with k=4 for both batters and pitchers, which also led to increased cohesion between the two datasets in terms of how archetypes were defined, ensuring that both groups could be interpreted in a similar fashion.

For batters, the K-means clustering resulted in 4 distinct groups, with 58% of the variance being accounted for by the top 2 components, which were Salary and Overall Offensive Production. This resulted in groups that appear to be very well-separated when graphed in only 2 dimensions. Through analysis of the mean values of each cluster for each component of our model, we determined that Cluster 0 represented our *Underpaid Performers,* or in other words, players perform to a high level and are relatively underpaid compared to their peers who perform at the same level. Cluster 1 represents the *Overpaid Underperformers* in our data, or players who are paid a high salary but have performance metrics which fall behind their peers. Cluster 3 represents our *Developing/Low Cost Players,* or players who perform at an average level and who are paid an average or below average salary by their teams. Finally, Cluster 4 represents our *Well-Paid Performers,* who are all-star caliber players who are paid highly for their contributions to the team.

For pitchers, the K-means clustering resulted in similar groupings, although the separation between groupings is a little bit harder to visualize in 2-dimensional space due to the fact that less (48%) of the variance was accounted for by the top 2 components in our data, Salary and Workload/Volume. For pitchers, the clustering behaved similarly: Cluster 0 also represents our *Underpaid Performers,* Cluster 1 represents our *Overpaid Underperformers,* Cluster 2 represents our *Developing/Low Cost Players*, and Cluster 3 represents our *Well-Paid Performers:*





In our batting data, our clusters had different populations, which was expected based upon the methodology that we used. 1985 of the players in the dataset were listed as *Developing/Low Cost Players,* 1294 were listed as *Overpaid Underperformers,* 1075 were listed as *Underpaid Performers,* and 1096 were listed as *Well Paid Performers.* In our pitching data, our clusters consisted of 1996 *Developing/Low Cost Players,* 1191 *Overpaid Underperformers,* 1386 *Underpaid Performers,* and 925 *Well-Paid Performers.*

After performing our clustering, we exported a csv containing the normalized PCA components for each player in both the batting and pitching datasets, along with their resulting cluster category to perform further analysis on the effectiveness of our method. When analyzing how the players were clustered, we decided that it would be more effective to analyze use cases rather than perform a statistical analysis of our clusters. This allowed us to interpret each cluster in a more real-world context rather than a purely numeric analysis.

By examining representative examples from each cluster that we were familiar with in a real world context, we were able to better understand the underlying relationships between salary and performance, and to draw conclusions which align closer to how front offices and sports analysts evaluate roster composition in real-world scenarios. The reason that this was the most effective method of evaluation lies in the fact that our unsupervised method simply clustered the four groups for both pitchers and batters based on numeric distances between the points, rather than analyzing players who are alike within the context of baseball itself.

For example, Nolan Arenado was a standout player at the beginning of his career on the Colorado Rockies. Between 2013 and 2016, our model outputs that he was an *Underpaid Performer* for each of the 3 years before he was eligible for arbitration, which is a negotiation between a front office and a player which has the potential to increase a player's salary from a minimal contract to one of higher value. From 2016 to 2019, Nolan Arenado was then grouped as a *Well-Paid Performer* in our clusters as a result of his arbitration as well his ensuing long-term contract which he signed with the Rockies in 2019 before he was traded to the St. Louis Cardinals due to injury concerns.

This example highlights the strength of using real-world context to interpret our unsupervised learning. By tracking Nolan Arenado's career trajectory as well as the labels our model assigned to him on a year-by-year basis, we can see how changes in contract status, arbitration eligibility, and long-term deals influence a player's position within our numeric-based clustering classifications. This version of a player-specific case study provides a tangible validation of our clustering results, showing that the model we developed aligns with how front offices assess player value, contract progression throughout a player's career, and roster efficiency in practice; greater than any statistical or numeric analysis ever could.

Part B. Supervised Learning

First, we imported data from the unsupervised process with each player identified by their cluster. We then aggregated this data so that each team/year combination has a count of the quantity of each of the 8 categories (4 for batters, 4 for pitchers). The total count of players in each of these categories represented the team's roster construction. With this we could assess which categories and combinations led to wins.

Data was also collected about each team to be included in the model to ensure that we solely accounted for the roster composition, not any extrinsic factors. We collected the team city metro population based on the US census standard definition of metro area. We also included the team's total payroll for each year. Since baseball does not have a salary cap, teams can choose to spend significantly more than others. This would meaningfully affect the roster composition since these teams would be able to have more high paid players. Average salaries also increased over time. We normalized by year to account for this. The last team-level data point collected was total wins, which will be our target variable.

We chose to eliminate data from 2020. The COVID year affected how many games people played and how much players were paid in total. Instead of normalizing for the shorter season we chose to eliminate the data to avoid any unobserved confounding variables.

To tune parameters and identify the appropriate model we created a pipeline to test a wide range of parameters:

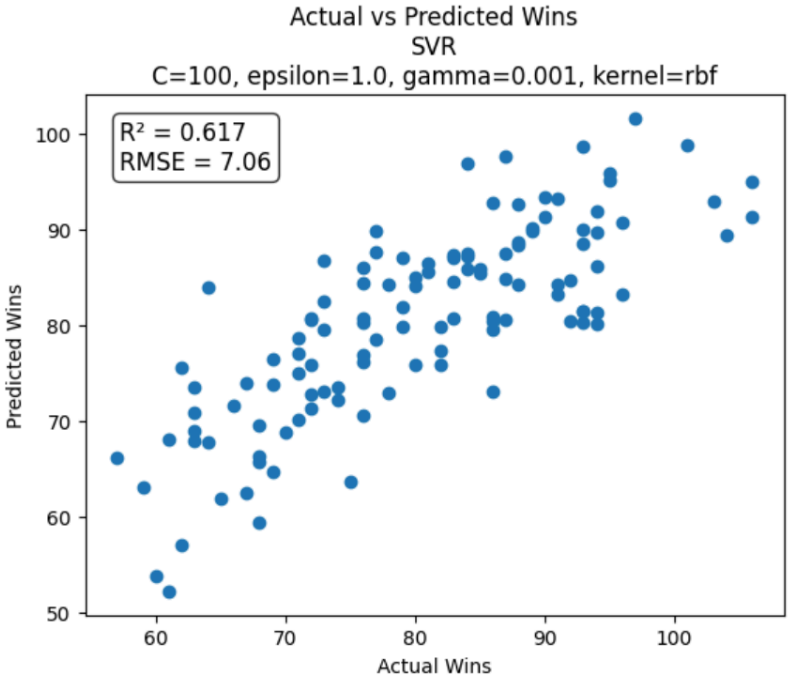
1. Whether or not to use polynomial features
2. Linear Regression:
    1. Ridge, Lasso, ElasticNet Normalization
    2. Alpha's from 0.1 to 100
    3. L1 ratio from 0.1 to 0.9
3. Random Forest Regression:
    a. Number of estimators at 100 or 1000.
    b. Maximum depth of None, 10 or 40.
    c. Minimum samples per split of 2 to 10.
    d. Minimum samples per leaf of 1 or 5.
    e. Maximum features as either square root or log2 of features.
4. Support Vector Regression:
    a. RBF Kernel.
    b. C from 0.1 to 100.
    c. Epsilon from 0.01 to 1
    d. Gamma as "scale", "auto", and values from 0.001 to 0.1.

This collection of models was chosen to make sure that we were able to account for linear and non-linear features as well as potential feature interactions. Regression models will capture simple linear relationships, SVR models will capture any non-linearity, and the Random Forest will account for conditional or interactive feature relationships. The option to use polynomial features was also included to capture potential feature interactions. The parameter grid was selected to capture a wide range of possible parameter settings. If model performance was unsatisfactory or the best parameter is on the edge of the grid we could re-run with an updated parameter grid.

From this grid search, we found that all the models achieve a similar R2 and RMSE value, and that they have only slight drops between training and testing. This indicates that our model is capturing as much signal as it can and that we are not overfitting. Overall, The SVR model was the overall best fit for our use case and we used it for our overall predictive modeling:

Model Name	R2 Train	RMSE Train	R2 Test	RMSE Test
SVR	0.64	7.42	0.62	7.06
Lasso	0.64	7.46	0.61	7.14
ElasticNet	0.64	7.47	0.6	7.18
Ridge	0.64	7.48	0.61	7.17
Linear Regression	0.63	7.53	0.6	7.23
Random Forest	0.61	7.81	0.61	7.16

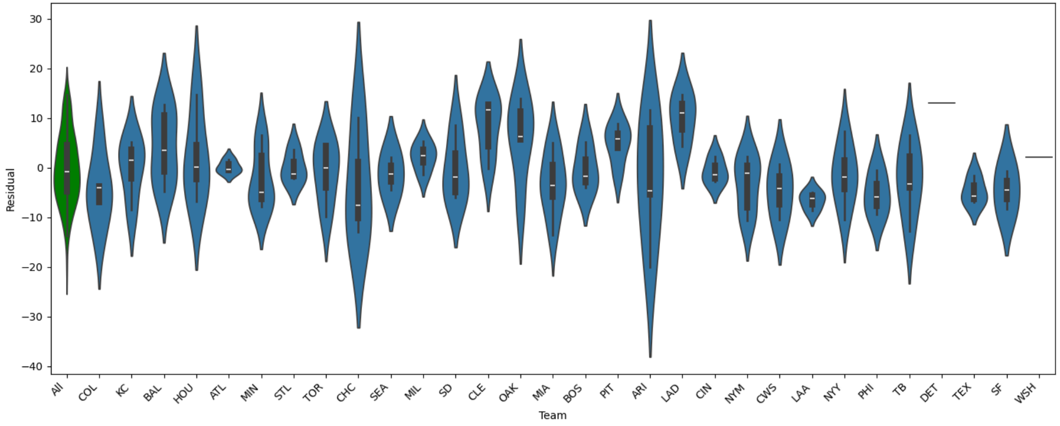
Below is the chart of actual vs. predicted number of wins for our testing data. Included are the parameters of our best SVR model as well as the R2 and RMSE measurements. This output shows that our model is a useful predictor for team wins. There is some room for improvement and additional modeling, however, with an RMSE of 7 when comparing to the 80 average wins for a team each year this is an error of <10% in either direction, which was sufficient for us to determine that our player categories defined in the unsupervised learning phase are meaningful predictors for team success.

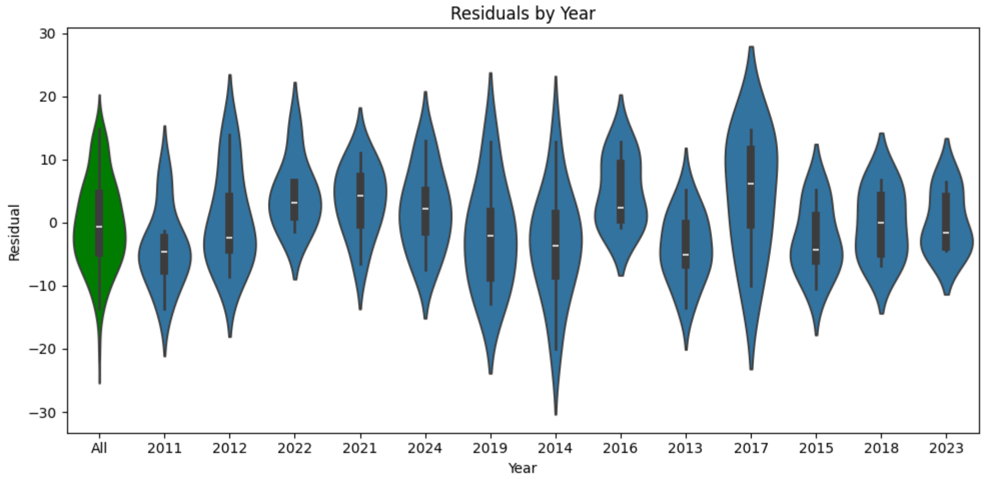


To further assess the quality of our model we explored the failure modes, where the predicted value was most significantly incorrect. We first performed residual analysis. For each indexing feature (team and year) we compared the residuals for each category of that feature. This was to determine if any team or year was predicted significantly better or worse than others. Years showed that there was no significant difference between the distribution of residuals for any given year. Team based residual distributions vary significantly from the overall average, however this is likely not a cause for concern. Based on our train test split ratio of 0.3 there would only be on average 4 data points for each team in the test set. There is expected variance in the distributions that these points create.

In the next step we explored if any of the large sub categories of categorical data had statistically different residuals than the overall distribution. We tested each unique value for each category and removed those that had fewer than 3 members. There were two remaining categories that had significantly different residuals than the rest of the data. When a team had 5 developing lower cost pitchers the model consistently underestimated the teams performance. And when a team had no well paid pitching stars it also consistently underestimated the team’s performance. Based on qualitative baseball understanding, this would seem to be an issue with how the model models, or fails to model, pitcher interactions. The distribution of who pitches when and for how long is a highly variable configuration from team to team and is also highly dependent on what kind and quality of pitchers a team has. It will require further investigation to find a solution to this.

Lastly we examined the teams that model predicted particularly poorly. Overall the predictions were highly consistent. There were only 4 predictions that had 14 wins more or less than our prediction. The only distinct outlier was the 2014 Arizona Diamondbacks. The reasons for this outlier are not perfectly clear, but by examining the 2014 and 2013 ARI teams to the overall averages considering the three most important features (“number of underpaid high performing pitchers”, “developing low cost batters”, and “well paid star batters”). The selection of these features will be discussed further in the next section. The 2013 ARI team had very similar players and performed better and was more accurately predicted. So comparing that team to 2014 should present differences. The first is that the underpaid high performing pitchers on the 2014 team performed significantly worse than they had the previous year and significantly worse than the average for the category (according to our overall metric “workload and volume”). Since there were two of these kinds of players on both the 2013 and 2014 team the model would make similar predictions for both. When looking at the developing low cost batters we see another source of confusion for the model. The number of low cost developing players increased from 2013 to 2014 from 2 to 5. The model considers this a very good thing. However the performance of these players also declined significantly between years. It fell from well above average performance in 2013 to approximately average performance in 2014. Lastly the two well paid stars on the team in both 2013 and 2014 were poor in both years and worse in 2014. While it isn’t clear exactly how cause this highly inaccurate prediction, it is clear that there was information the model couldn’t see in terms of the internal cluster performance that could drive poor results. Since this is a lone outlier we do not feel that the model’s overall performance is in question.





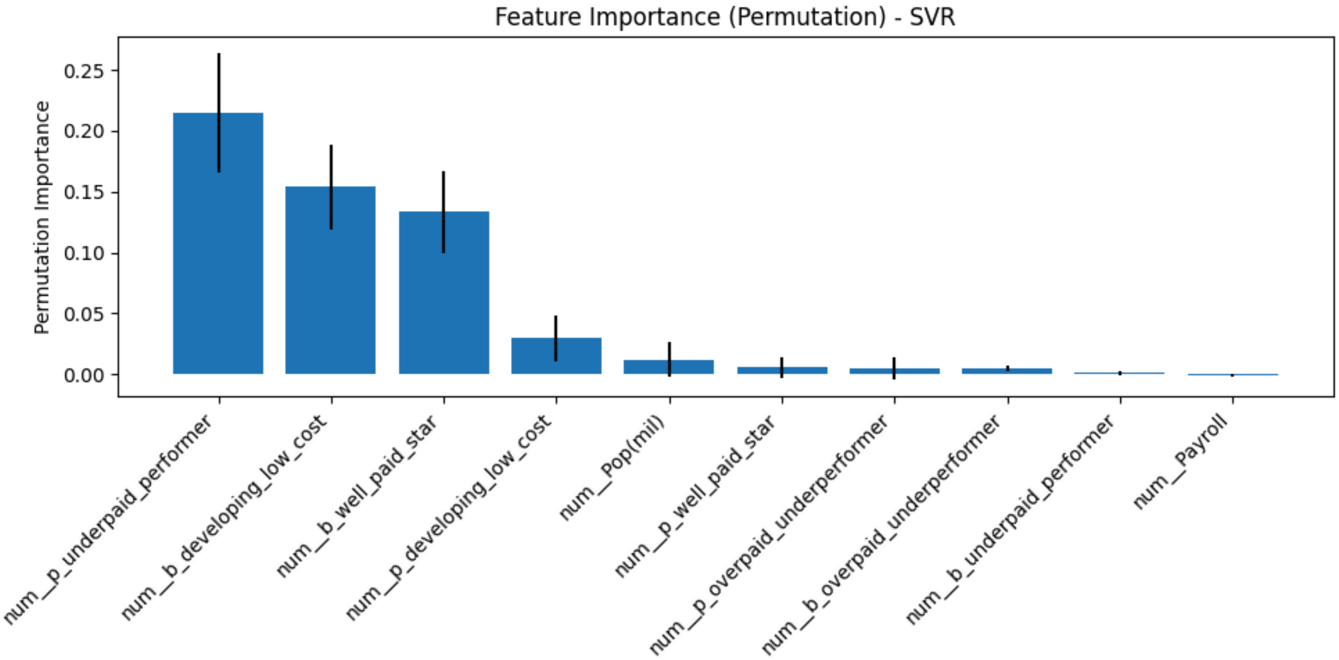
Discussion

Supervised learning

Learnings from the supervised model:

- We used our linear models to identify if any of our player categories were statistically significant. This would show us if a specific player type is central to determining team performance. We computed the standard error for each of the coefficients in our linear regression. With that we computed the z-score for each. Our only feature that was close to significance was “number of low cost developmental batters”. This feature had a 92% confidence, however this was not sufficient to conclude significance. Qualitatively analyzing this feature we can see why this would be an impactful component of team composition. Having many players who don’t cost much while you are developing them is often seen as a winning strategy. This was one of our initial assumptions and while we can’t confirm that this is true, it is clear that assigning importance to this type of player is reasonable.
- We used permutation importance to compute the strength of our SVR features. This showed us which of our player categories are the most predictive and allows us to compare how predictive they are compared to the team features (metro population and total budget). We were surprised that the team features had almost no effect on the prediction. This shows that our player categories are almost solely responsible for the predictive power. This is a better outcome than we had expected. It is possible that the player categories and the payroll data have strong correlation. Since our categories are based significantly on player salary, approximate payroll is encoded by the quantity of each player category on the team. This could explain why the payroll information is not important to the prediction. Underpaid well performing pitchers are the most predictive feature. This makes qualitative sense. Pitcher is a critical position and having multiple pitchers contributing to your team without costing a lot is a good sign. It is notable that having well paid star pitchers is not important to the model. This could suggest that high paid pitchers don’t contribute to wins as much as expected. Low cost developmental batters and to a lesser extent pitchers contribute the next largest portion of the predictive power. This aligns with our intuition and assumptions that good teams have strong developmental pipelines. A large quantity of these players indicates that a team is investing in young talent and not relying on free agent acquisitions. Unlike star pitchers, star batters do contribute significantly to the model. This isn’t easily explained by intuition, but this is an interesting question to investigate further. The chart

below shows the feature importance as well as the standard error of that feature in an error bar. In the feature names “p” indicates pitchers and “b” indicates batters.



Unsupervised Learning

The unsupervised learning portion of our project presented us with both our biggest challenge, as well as, our biggest surprise. In addition, while we were satisfied with the clustering groupings we found, there’s plenty of room to expand upon them with data we were unable to incorporate in this short time frame.

Our biggest challenge we encountered was in our first attempt at implementing our unsupervised model. Following the instructions of the assignment we initially attempted to utilize four different clustering methods we thought would be suitable for our data. Unfortunately, we found that three of the four methods we intended to use (DENCLUE, PROCLUS, KernelKMeans) required libraries that were not compatible with the python version we were running (3.13.7). As a result, we were still able to use our fourth method, KMeans, but we substituted SpectralClustering, AgglomerativeClustering, and DBSCAN in place of the other three methods we had researched more thoroughly. We successfully were able to run a grid search that ranked the methods by their silhouette, calinski, and davies scores, which showed that kmeans_constrained followed by k_means with n_clusters of 3-4 performed the best. However, when we attempted to visualize the individual models with n=3 we ran into issues as we were unable to see any distinction among the clusters we found in a 2 dimensional scatter plot. After attempting to debug the model for a long period of time, we decided to simplify our approach in a new implementation using only KMeans with n=4. We chose KMeans because we felt its combination of high scoring in our grid search and our comfortability with implementing it would give us the best chance to achieve discernable clusters. As our results showed, this did lead to a successful unsupervised model with four distinct clusters that best represented our data.

This provides a perfect segway into our biggest surprise, which was that our initial intuition based purely on baseball knowledge that there would likely be four clusters similar to the ones we found was correct. Our four groupings (Developing/Low Cost Players, Overpaid Underperformers, Underpaid Performers, Well Paid Performers) successfully allowed us to identify what we hoped to achieve.

Now, as discussed in our challenges there are plenty of ways we could expand upon this with more time and resources. The most important of which is to successfully evaluate our intended clustering algorithms by using a compatible version of python that at the time we did not know how to retroactively use. In addition, as we discuss later in our ethical considerations, in order to successfully implement our models we made the decision to omit players beyond the top 26 contributors (13 batters and 13 pitchers) to best align with MLB's 26 man roster. With more time we'd hope to be able to analyze the full depth of every players' contributions, as well as account for the September rule change that allows for a 28 man roster for that month of the season. Lastly, due to the abbreviated 60 game schedule used during the 2020 season to account for the pandemic our initial data was filled with many outliers. We decided the best course of action was to omit the season and not skew the other 13 seasons of data we used. In a further analysis, it would be important to re-incorporate that data as well as expand our analysis to include more seasons worth of data.

Ethical Considerations

There are many ethical considerations that could arise from MLB teams using our model as a basis for roster construction. The way we structured our supervised modeling as an expansion of our findings from the unsupervised modeling makes it difficult to isolate our concerns to only one section, but we try to pinpoint the origins of each concern in regard to both sections of our modeling.

The overarching issue we identified stemming from our clustering in the unsupervised modeling is it reduces a player's worth down to the clusters we identified These four clusters (Developing/Low Cost Players, Overpaid Underperformers, Underpaid Performers, Well Paid Performers) pigeon hole players into categories that potentially could unjustly impact their financial earnings. Specifically, our clusters focus solely on statistical contributions by players and minimize player's leadership roles, impact on team chemistry, and baseball knowledge that we are unable to quantify. In baseball players that embody these characteristics are often referred to as "locker room guys," and teams often emphasize their importance when discussing their impact on their respective teams. By omitting these roles and qualities entirely in our clustering, we don't have a way to incorporate their value and therefore could be potentially mischaracterizing players when we refer to them as "over-paid" or "under-paid."

In our supervised modeling, the previous concern further expanded upon, as well as, acknowledging the potential fallout team executives could face by relying solely upon our data to construct their rosters. In this section we use our player designations from our unsupervised model to try and identify common trends in successful team roster compositions from actual historical mlb rosters. Again, we acknowledge here that our model further pigeon holes players as the ratio of players found on successful teams based on our statistical based player categories could further minimize players' roles as well as potentially overemphasize certain player types. This could further impact the salaries and careers of major league players that our model deems as less impactful toward team success, and as previously mentioned could impact the careers of executives that rely upon our data if used as a standard for team success.

While the financial and career impact of our modeling is the main area of ethical concern we wanted to discuss another important concern is the data limitations faced in the creation of our models. In order to create our models, we utilized the top 26 players (13 pitchers, 13 batters) per team, per season in accordance with the 26 man roster size used in MLB. However, MLB rosters are often in constant flux whether due to player performance or injuries resulting in more players serving significant roles as teams are forced to utilize players in their minor league systems to supplement their roster throughout the season. In addition, MLB expands their rosters in the month of September to allow for teams to help reduce usage of players over the course of a long season. While our models did utilize the players that had the largest contributions to their teams each season, it is important to acknowledge that many players were left unaccounted for by the model. The other important note is that due to the Covid-19 pandemic, the 2020 season was shortened to only 60 games. In our initial modeling we found the data from this season created many outliers in our clustering and to address this issue we omitted the season and therefore the contributions of players participating that season.

Statement of Work

Ryan Peet: Contributed to final cleaning tasks within the data manipulation and preprocessing, performed the feature selection, led the clustering and evaluation of our unsupervised model, and assisted with the maintenance and cleaning of the Github repository

Jeffrey Vartabedian: Contributed to the data manipulation and preprocessing, contributed to the clustering of the unsupervised model, maintained the GitHub repo, and conducted the 1st stand up meeting.

Charlie Williams: Led the data manipulation and preprocessing, developed and evaluated our supervised learning model, and conducted the 2nd stand up meeting.

References

Baseball Reference. (n.d.). *Baseball-Reference.com.* Sports Reference LLC. https://www.baseball-reference.com/

Carney, P., Egan, J., & Schulte, D. (2019). *How Have Advanced Statistics Impacted Salary in Major League Baseball?* Fort Hays State University. Retrieved from Baseball Reference and Spotrac data sources.

Fields, B. (2001). *Estimating the Value of Major League Baseball Players* [Master's thesis, University of Massachusetts Amherst]. ScholarWorks@UMass Amherst.

Ojanen, J. E. (2012). *Modeling Offensive Performance and Salary in Major League Baseball* [Master's thesis, St. Cloud State University]. Repository@StCloudState.

OpenAI. (2025). *ChatGPT [Large language model].* https://chat.openai.com/

Spotrac. (n.d.). *Spotrac — Sports Contracts, Salaries, Cap Tracking.* Spotrac LLC. https://www.spotrac.com/

Wikipedia. (n.d.). *Metropolitan statistical area.* In *Wikipedia.* https://en.wikipedia.org/wiki/Metropolitan_statistical_area