Baseball Salary Prediction

Author: Ryan Helgeson, ID# 315

Email: rhelgeson3@gatech.edu

Date: 2022-04-15

Data Mining & Statistical Learning - ISYE 7406

Abstract

This report explores Major League Baseball (MLB) salaries and the ability to predict a player's next year salary. Data was used from FanGraphs and Cot's Baseball Contracts to create player records that contained performance data and a player's salary for the current year and the player's previous two seasons, if applicable, from 2012-2019. The data was split into separate data sets for batters and pitchers, and a handful of regression models and methods were used on each to try and predict a player's next year salary, with the goal of improving upon using the player's current salary as the estimate. We found that some tree based methods like Random Forest and XGBoost were successful in providing better predictions, but the complexity of MLB contracts made this problem very challenging and hard to improve upon the current player's salary.

Introduction

Major League Baseball (MLB) is a billion dollar industry that millions of people across the United States and the world enjoy. In 2019, over $4 billion dollars went to the players that are under contract for the 30 teams that belong to MLB. Each individual team operates within their own budgetary constraints, which can vary greatly from team to team. In 2019, the team with the highest payroll were the Boston Red Sox at $229 million and the lowest payroll belonged to the Tampa Bay Rays at $64 million. Each team works to maximize the talent on their roster while also trying to stay within the budgets set by their owners. Understanding the salaries of players in the future can help teams better plan for how their money will be allocated and what money they may have available to spend on new players to help improve their team. In this report we will look to try and predict players' next year salaries which can help teams understand how their payroll will look in the future to better aid in planning.

Since we are trying to predict salaries, we will use regression methods and models. We will start with a multiple linear regression to get a baseline and then expand to more sophisticated models like Random Forest and XGBoost. We will use some of the feature importance from these models, as well as our own baseball intuition, to do some feature engineering and selection to try and improve the models. We will then explain the results and conclusions from the model, as well as some lessons learned from our analysis.

Baseball contracts are fairly complicated so we'll quickly overview some of the main stages a player can be in: Rookie, Arbitration, Free Agency. *Note: This will not encompass every detail or possibility of MLB contracts, this is merely a quick overview*

Rookie When a player first makes it up to the major league roster, they are on a rookie contract. At this point the clock starts on their major league service (MLS) time. A rookie contract can last from 6-7 years, and for the first 3 years the team decides what a player's salary is. It often starts at the designated league minimum ($555K in 2019), and a team often rewards a well performing player with small increases. When a player's MLS gets to 3 or more at the end of a season, they enter the next stage of the rookie contract called arbitration.

Arbitration This next stage is the first time the player will have some control over what he makes. Here players and teams will negotiate a salary, using comparable players who have signed contracts in recent seasons. Often arbitration leads to higher salaries than rookie contracts, though salary reductions are possible if a player performs poorly the prior year. If a salary is not agreed upon between the player and team, they each give their salary figure to a panel of arbitrators. The arbitrators will decide which salary figure is more appropriate for the player and that will be the player's salary for the upcoming season. Arbitration happens for the last 3 years of the rookie contract and is meant to build off each prior year salary (assuming consistent or improving performance). Arbitration numbers in the final year can get over $20 million for the best players in the league.

Free Agency After the rookie contract is over, so a player has 6 or more years of MLS, then they become a free agent. Unlike the previous 2 stages where only 1 team is involved, now they can negotiate and get a contract from any of the 30 teams in the league. This is often the most lucrative contract of a player's career, and they usually only get one shot in free agency so most players will try to maximize this contract. This is where past performance can become a key indicator in salary negotiations, but also a player's age and how a team projects them to play going forward will be important as well. How a specific market forms for a player can also be a factor. For example if there are a lot of free agent shortstops in a certain year, that could suppress their earning potential. On the other hand if a player is the only quality shortstop on the market, many teams may be involve which can help them in contract negotiations.

Data Sources

The two main categories of data we focused on for this problem were salary data and performance data. We got the salary data from Cot's Baseball Contracts and performance data we got from FanGraphs.

Salary Data Cot's Baseball Contracts is an unofficial record of salaries and payrolls collected from published reports and sources (https://legacy.baseballprospectus.com/compensation/cots/). It has a Google Sheets workbook that contains opening day salaries for players going back to 2000. This data source was used because of the simplicity of the format and ease of use. It also contained the 2 pieces of information we were looking for: major league service time and salary. We used this data from 2012-2020.

Performance Data FanGraphs is an open source website for advanced baseball analysis and stats (https://www.fangraphs.com/). FanGraphs has player leaderboards that have both traditional and advanced stats for each player for every baseball season, though some advanced stats can not be collected for older seasons. We collected data for batters and pitchers from 2012-2019, each data set having a combination of traditional baseball statistics and advanced statistics. Some examples:

Batters:

- Traditional Stats
  - Batting Average (AVG)
  - Home Runs (HR)
  - Runs Batted In (RBI)
  - Plate Appearances (PA)
- Advanced Stats
  - Weighted On-Base Average (wOBA)
  - Weighted Runs Created Plus (wRC+)
  - Wins Above Replacement (WAR)

Pitchers:

- Traditional Stats
  - Wins (W)
  - Earned Run Average (ERA)
  - Innings Pitched (IP)
- Advanced Stats
  - Fielding Independent Pitching (FIP)
  - Skill-Interactive ERA (SIERA)
  - Wins Above Replacement (WAR)

Data Set Creation

We needed to combine these data sets into player records. Each record would contain a key column identifying the player and the current year (ie 2016_Kris_Bryant) along with current year performance and salary data, the previous year performance and salary data, and the 2nd previous year performance and salary data. Current year fields will have a suffix of "_C", previous year a suffix of "_P1", and 2nd previous year a suffix of "_P2". For example, the Wins Above Replacement (WAR) columns for year will be WAR_C, WAR_P1, and WAR_P2. Only 3 fields will not have previous year data: the key identifier column, Age, and major league service (MLS). We did not feel Age or MLS were relevant outside of the current values. We will also add the response variable, which is the player's next year salary. This will be denoted as "Salary_Y". Because of the difference in performance statistics between batters and pitchers, we created separate data sets for batters and pitchers. We will then train models on each, with the result being 2 unique predictive models for batters and pitchers.

A lot of cleanup was needed for this. Some players did not have previous year data because they are either in the first 2 years of their major league career or for whatever reason were out of MLB. The previous years data was filled with zeroes in this case. If a current year player did not have a salary for the next year we would remove that player from the data set. We used last and first name to join these data sets together, and on a couple of occasions players had the same name. This was a rare occurrence and very hard to account for, so any duplicate names were dropped for simplicity. Here are the final data set attributes:

- Batters: 1857 records with 63 predictors
- Pitchers: 1771 records with 72 predictors

For a full view of the columns in each data set, please see the Appendix.

Proposed Methodology

We will use a few regression methods and models to try and best predict next year's salaries. The three methods we will highlight in this report are Multiple Linear Regression with Regularization, Random Forest, and XGBoost.

Multiple Linear Regression with Regularization This method was chosen as a simple, baseline model to use as a starting point. This model will be more interpretable than the others and can aid us in feature selection for further model tuning. We expect (and will show later) multicollinearity to be present in this data set, so we will use regularization to help account for that. While ridge regression is best to handle multicollinearity, we will use cross-validation to choose the penalty parameter for the model.

Random Forest Random Forest is a well known and popular model, it is an ensemble of decision trees using re-sampling to create many trees in parallel. The average of these trees helps to reduce variance and better model performance. We expect this model, as well as the XGBoost model, to have higher performance and make better predictions.

XGBoost XGBoost is another powerful model that uses trees like Random Forest, but it builds trees sequentially to account for previous errors. It uses re-weighting instead of re-sampling which helps improve training error.

We will split both data sets into 75/25 train and test sets. Any hyperparameters will be tuned using 10 fold cross-validation of the train set, and then the best hyperparameters will be trained on the full train set and then used on the test set to make predictions and evaluate performance. After that we will do some feature selection and engineering to try and improve performance for each model.

Model Performance

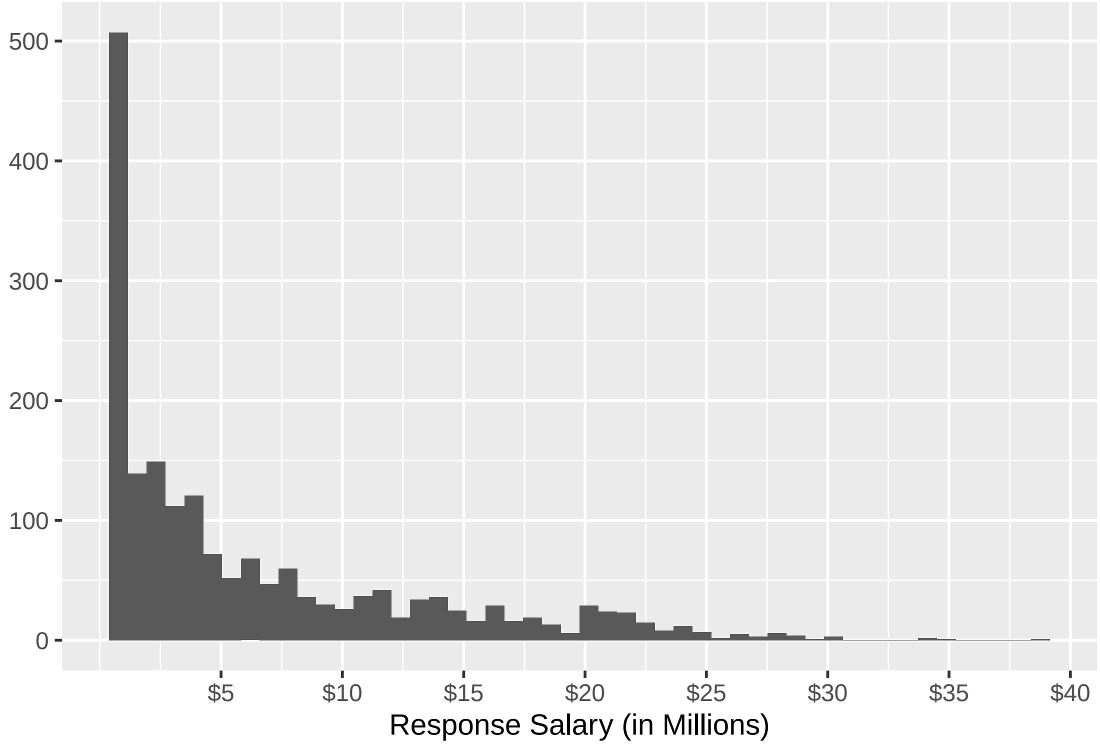
We wanted a way to determine if our models are performing well or not, to have some kind of baseline to try to improve upon. We originally thought the Multiple Linear Regression model would be this, but during our analysis we found a better baseline: Current Salary. As we will show later, a player's current salary was very well correlated with their salary the following year. So we calculated the metrics on our test set if we just used a player's current salary as the prediction, and now we have a baseline to try to improve upon.

For performance metrics, we decided to use two metrics for evaluation: Mean Average Error (MAE) and Mean Average Percent Error (MAPE). We chose these metrics because we found them more interpretable than Root Mean Squared Error (RMSE) or R-squared. MAE would be the average difference in salary our predictions were off, and MAPE would account for the magnitude of the response salary in its value. Being $1 million off on a response salary that's $20 million, in our opinion, is not as bad as being $1 million off on a response salary of $750K. By MAE these are the same error, but MAPE would account for the response salary magnitude and have a greater error on the latter, which we think is more appropriate. When picking the best hyperparameters we use MAPE.

Analysis

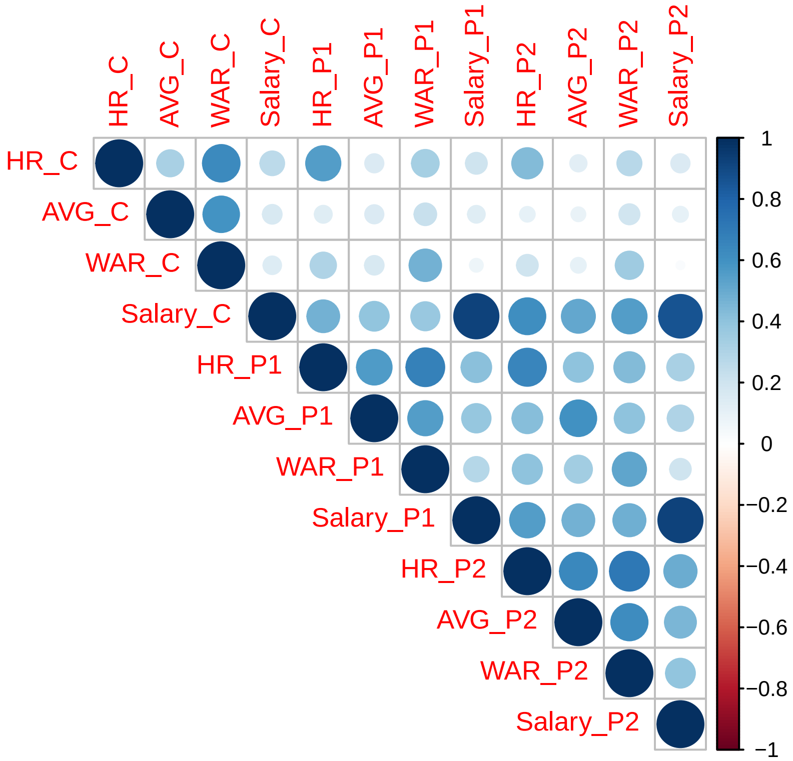
Batter Data Set

The batter data set contained 1857 records with 63 predictors. After splitting into train and test sets we had 1393 records for training and 464 for predictions. First we will look at the distribution of the response salary.



We see this is very right skewed, with most values falling around or under $1 million. A few values scattered above $30 million, which represents the few highest earning batters. Most batters appear to be on rookie deals making close to or at the league minimum.

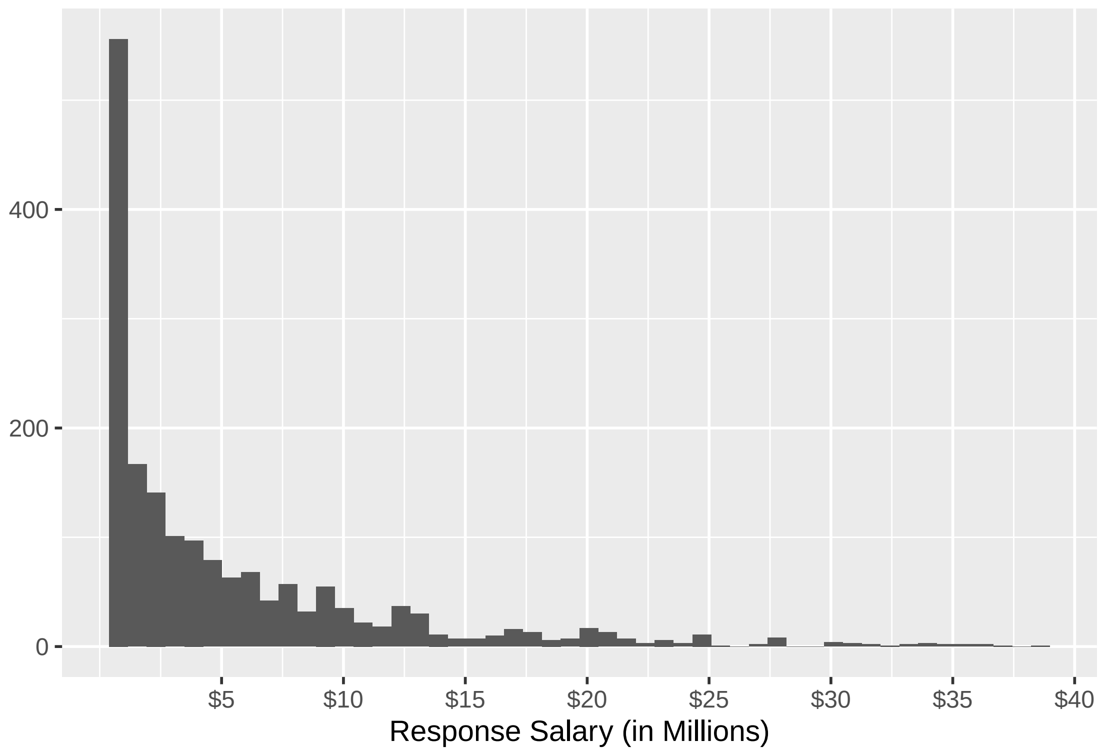
We expect multicollinearity to be present because of how we created the data set. For example a hitter that hits a lot of home runs one year will probably hit a lot of home runs the next year, so we would expect “HR_C”, “HR_P1”, and “HR_P2” to be correlated with each other. Below is an example of 4 columns, HR, AVG, WAR, and Salary, and how they correlate with their previous years.



For predictors that correlate well with the response salary, we found that the salary predictors correlate the strongest with Salary_Y, which makes sense because in a lot of cases these salaries are used as starting points for a player’s next salary. Some key performance statistics that correlate with Salary_Y are PA, HR, RBI, wRC+, and WAR. A full correlation plot can be found in the Appendix.

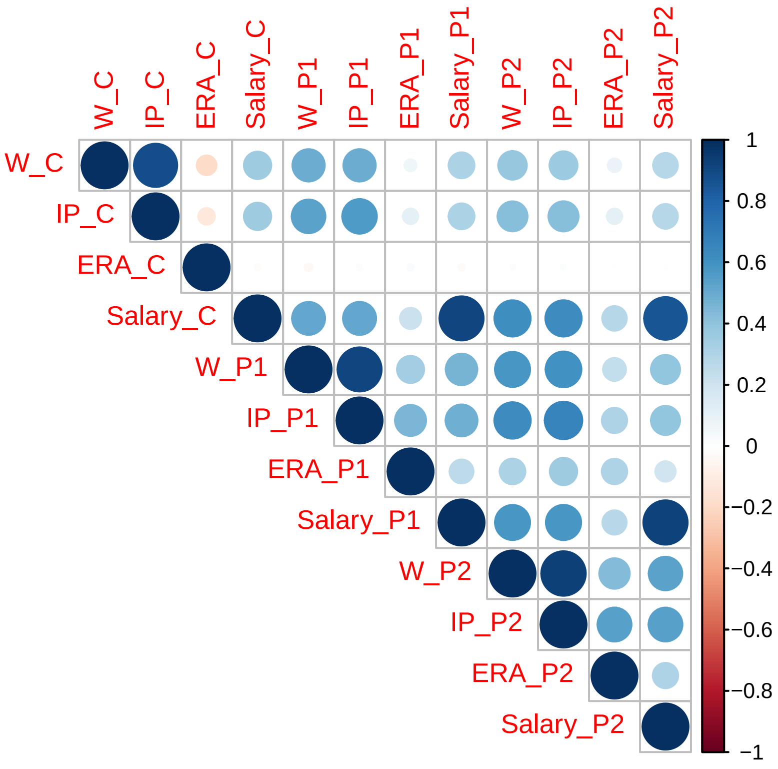
Pitcher Data Set

The pitcher data set contained 1771 records with 72 predictors. After splitting into train and test sets we had 1328 records for training and 443 for predictions. First we will look at the distribution of the response salary.



Similar to batters, the response is heavily right skewed. Pitchers have a few more top end salaries compared to batters, but once again most pitchers are making close to league minimum.

Multicollinearity will also be existent in this dataset, as we can see in the example plot below.



Once again salary predictors correlate strongest with the response salary. The performance predictors that correlated strongest were W, GS, IP, and WAR. Overall there seemed to be fewer predictors correlated with the response for pitchers than batters. The full pitcher correlation plot can be found in the Appendix.

Results

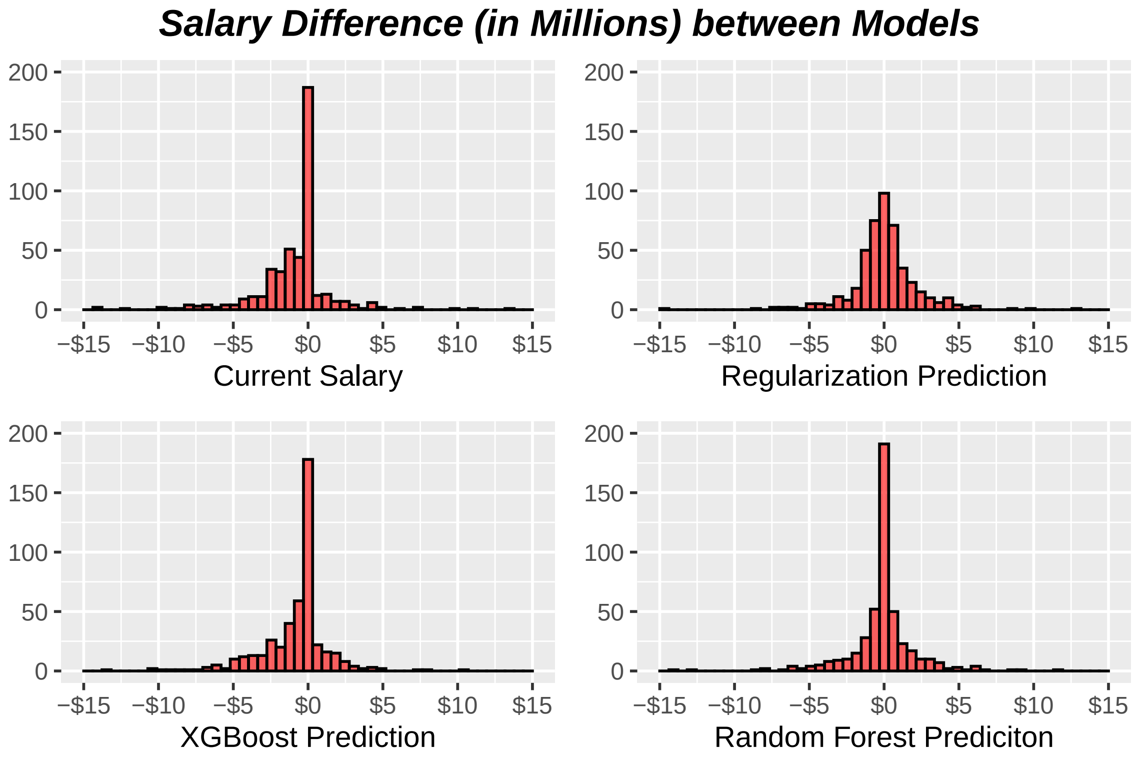
As mention in the Proposed Methodology section, we will tune and train each of the 3 models on 2 data sets. One with all of the predictors present and then another with feature selection and engineering applied to try and optimize performance.

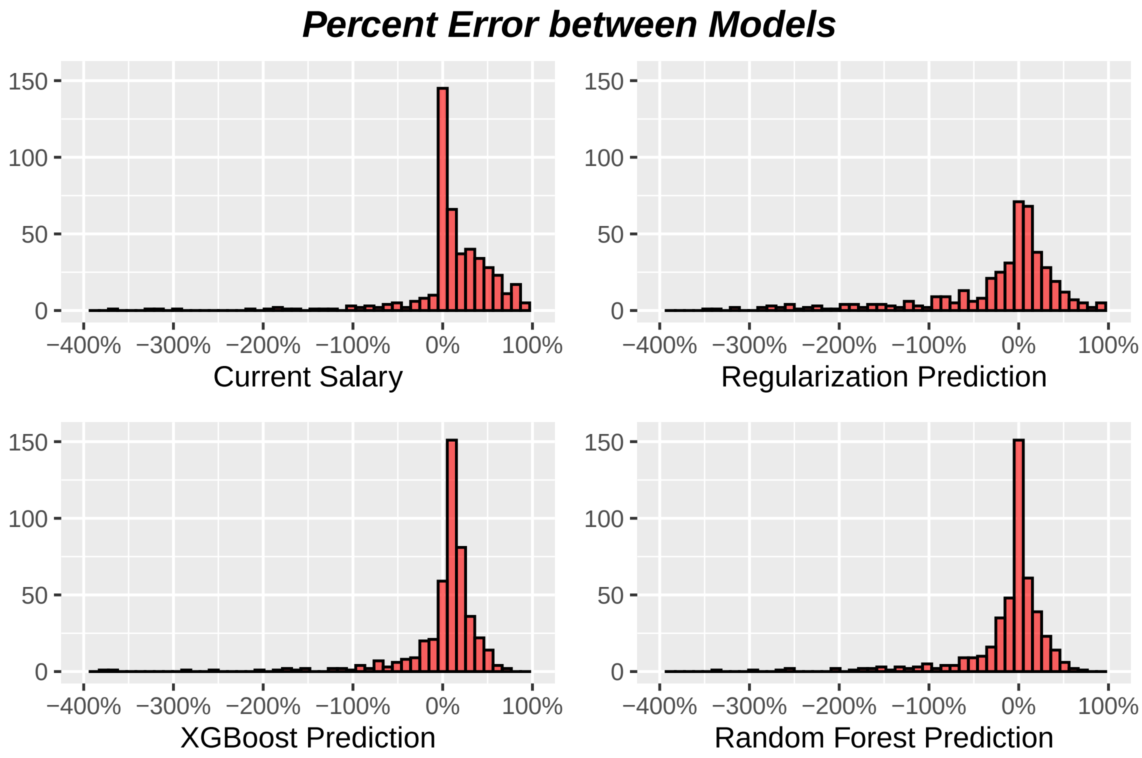
Batters Full Data Set

Model	Mean Average Error	Mean Average Percent Error
Current	$1,542,474	32.1%
Regularization	$1,433,836	58.2%
XGBoost	$1,459,431	26.2%
Random Forest	$1,261,312	27.8%

Above are the results of the three models along with the metrics for the current salary predictions against our batter test set. We can see that the multiple linear regression with regularization model does not fare very well according to MAPE, however it does have a better MAE than the current and XGBoost model. Random Forest performs best taking into account both metrics, while the XGBoost does the best on our preferred metric MAPE.

If you look at the MAE and MAPE distributions below, you'll notice the primary error of the current salary is that is underestimates the response salary. All 3 models help to normalize the error distribution around zero, with Regularization having a wider variance than XGBoost and Random Forest.





Batters Tuned Data Set

The following changes were made to the data set to optimize performance.

- Removed predictors so only 20 remained, using correlations with Salary_Y and coefficients from the regularization model to help pick features
- Created a "Salary_change" predictor which is the difference between last year's and current year's salary
- Added interaction between current salary and the other 19 predictors.

Model	Mean Average Error	Mean Average Percent Error
Current	$1,542,474	32.1%
Regularization	$1,453,113	51.6%
Boosting	$1,397,992	25.7%
Random Forest	$1,278,381	27.1%

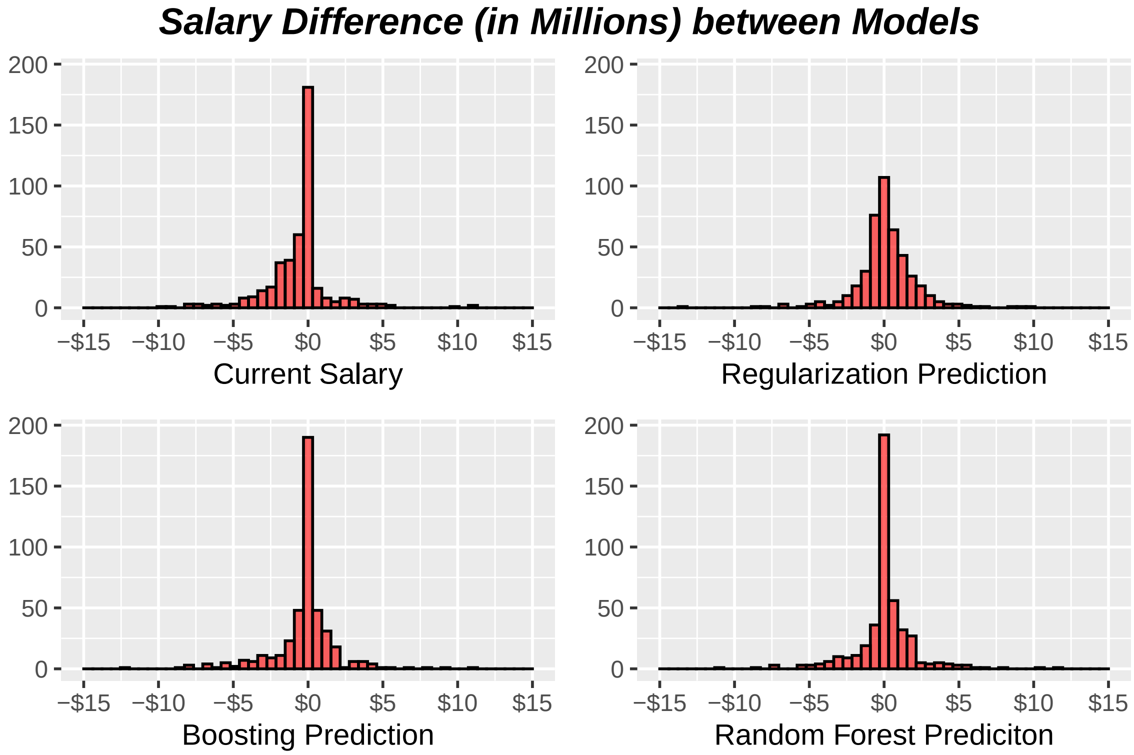
Not a lot of improvement looking at MAE for any of our models. Regularization improved the most by MAPE, dropping roughly 7%. XGBoost and Random Forest saw some improvement but not much compared to the original models.

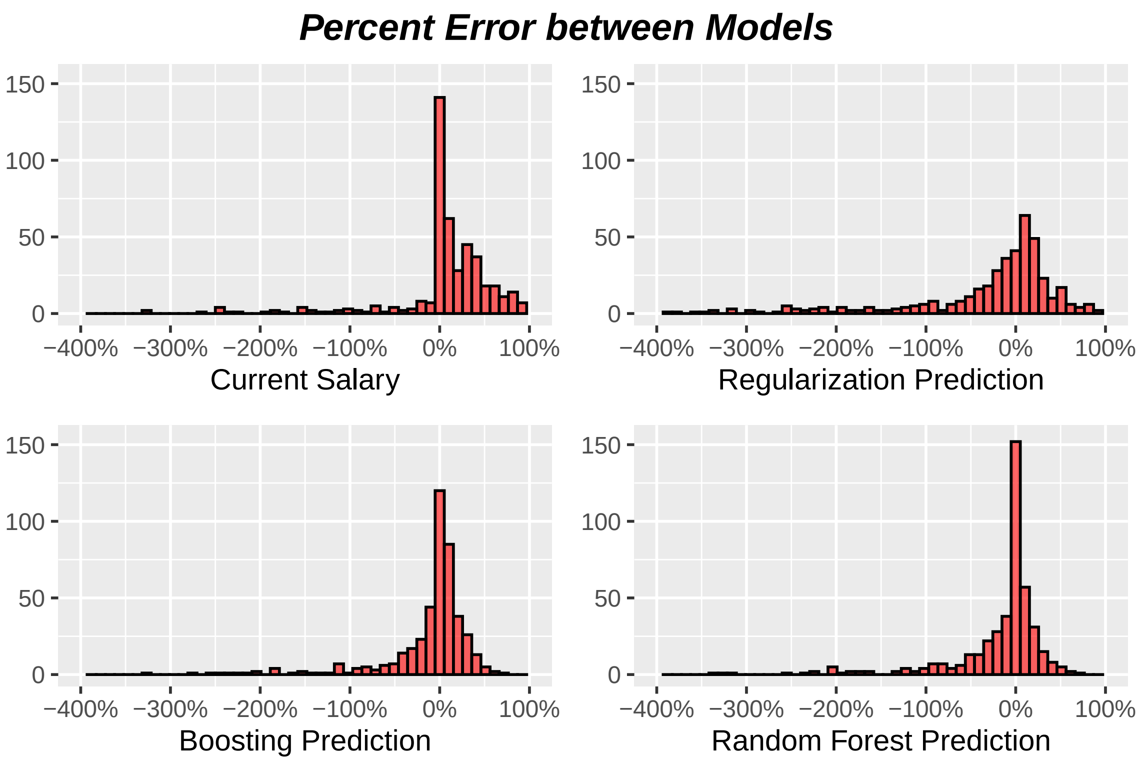
Pitchers Full Data Set

Model	Mean Average Error	Mean Average Percent Error
Current	$1,385,693	38.2%
Regularization	$1,335,969	64.6%
XGBoost	$1,193,866	31.3%
Random Forest	$1,153,853	33.2%

Above are the results of the three models along with the metrics for the current salary predictions against our pitcher test set. Similar results for each model, with Regularization performing poorly by MAPE and XGBoost and Random Forest performing better then the current salary baseline. Despite lower MAE compared to batters, the pitcher models' MAPE is higher across each model and our baseline.

Our error distribution charts for pitchers show a similar story to batters.





Pitchers Tuned Data Set

The following changes were made to the data set to optimize performance:

- Removed predictors so only 23 were left, using correlations with Salary_Y and coefficients from the regularization model to help pick features
- Tried creating a "Salary_change" like for batters, but this did not improve performance and was then taken out
- Added interaction between current salary and the other 22 predictors.
- XGBoost and Random Forest performed worse with the trimmed features, so all predictors were used with the interactions included

Model	Mean Average Error	Mean Average Percent Error
Current	$1,385,693	38.2%
Regularization	$1,394,779	58.9%
Boosting	$1,462,353	31.8%
Random Forest	$1,170,812	33.8%

Regularization showed improvement by MAPE, but the other models barely improve or even performed worse looking at MAE.

To see what hyperparameters were chosen for each model shown, you can find them in the Appendix.

Conclusions

Predicting next year player salaries proved to be a very challenging problem, both from a data collection and prediction aspect. The main challenge was improving upon what is already a very strong baseline in a player's current salary. Multiple Linear Regression with Regularization proved to be a poor model using our metric of choice MAPE. It improved slightly looking at MAE, but it seemed to keep more uniform error for all values of the response compared to the other models. Given that most salaries are below $1 million, having a more uniform MAE of $1.3-1.4 million increases the MAPE. For batters and pitchers both XGBoost and Random Forest were improvements on the current salary, but not as much as we were hoping. Batters had a 5-6% less MAPE and at its best Random Forest had MAE $281K below the current salary. The XGBoost model for pitchers had the biggest difference in MAPE from the current salary, being 7% lower, but only less than $200K difference in MAE.

Feature selection and engineering also had little to no effect on our models other than Regularization. I'd like to think this is because we did a good job of pre-selecting our data features in the beginning of this analysis, but it's possible there was some feature engineering that we didn't explore that would have resulted in better performance.

Overall I believe, after this analysis, that baseball contracts are far too complex to truly predict with the information we had. I think this analysis helped proved that current salary and performance data are key aspects to salary negotiations, but are far from encompassing all the variables that go in to these kinds of decisions. To build off this report, I think more contract information would help in predictions. We highlighted the 3 main stages to baseball contracts earlier, and finding a way to classify players into each stage could add some more information for our models to use in prediction.

One thing we did not account for are players in this data set that are already on long-term contracts. These players already have their salaries known for an extended time, so trying to predict their salaries does not make much sense. We thought about removing players in this situation, but decided it was far to challenging to do by hand. We hoped our models could use the past salaries and major league service time to understand if a player was on a long-term contract, but either removing these players or creating an indicator would probably help in our predictions.

There are also many factors that are unquantifiable that go into salary decisions. These are human-beings negotiating and oftentimes emotions can play a factor in them, no matter how hard they try to keep them out. Team locations, family, team relationships, and many other things could be factors in why a player or team chooses a salary or contract. Also the fluidity of a baseball free agent market will be hard to account for in these types of models. Factors like this will always make this problem hard to perfectly predict.

Lesson Learned

There were many lessons I learned working on this project. Being an avid baseball fan I was aware of how complex the baseball economic system can be, but even I was surprised by the complexity when I dove in. I was hoping to capture some of that complexity, but the time constraint on this project made this almost impossible to do, so I tried to keep it simple. I was hoping major league service time would help, and perhaps it did, but not enough and I think finding a way to add the contracts details into the data set would, in my opinion, be the best way to improve upon this report. I also learned just how few baseball players are actually millionaires. The vast majority of players are hanging around the league minimum (still not a bad wage), even though it's the top earning players we hear the most about.

For this class I learned how to present data analysis and models in a report. Most of the techniques I've heard about before, but being able to use them while telling a story of my analysis is what I'll really take away from this class. I will say I'm not a fan of the presentation being due a week before the report. In order for the presentation to be made you need to have the report done or at least 90% of it in my opinion. It made me rush what I wanted to do with this report so I could get the presentation done. I know it probably helps the TAs with grading having them due a week apart, but due to the fact this whole class is report driven maybe just drop the presentation and make the report worth more.