Final Project: Baseball Contract Predictor

Author: Joseph Aguilar

Discussants:

BaseballR: https://billpetti.github.io/baseballr/

Lahman Baseball Database: https://cran.r-project.org/web/packages/Lahman/Lahman.pdf

Introduction

It is currently the off season for Major League Baseball, and the off season begets signings of players who are free agents. A big part of the job for general managers of teams interested in signing players is deciding a fair contract for the players they are signing. I would like to see if there is any statistically-driven way to help create these contracts and decide on an average annual value (AAV) for the player. Of course, I anticipate that numbers generated by this statistically analysis would only act as a starting point for negotiations, as contracts can also be influenced by non-statistic factors (endorsements, marketability, home town discount, etc.). Shohei Ohtani, for example, signed a 700 million dollar deal with the Los Angeles Dodgers. From his stats, that may be seen as a massive overpay (even for his immense talent and skill), but his jersey sales and his global popularity as a Japanese two-way (hitting AND pitching) player more than make up for it. In short, the problem I am addressing is if there is a way to statistically decide fair contracts for players AND teams. If there is, it would be an immensely useful tool for agents of players and general managers alike.

I will be taking in the stats and salaries of position players (NOT pitchers) across the 2002-2016 seasons from the Sean Lahman Baseball Database, using the "Lahman" R Package. (https://cran.r-project.org/web/packages/Lahman/Lahman.pdf) Given that these are not modern seasons, this will also serve as a study of how baseball contract prices have changed as of recent. Modern baseball contracts are far more expensive than contracts of the past. I am not currently using this data in another class or for another research project.

There have been similar analyses done (https://ajaypatell8.medium.com/modeling-free-agent-contracts-in-mlb-4fada2220add), especially with the rise of Google's Statcast tool to help analyse statistics across full baseball seasons. There is a very popular book (and movie) called *Moneyball* detailing the story of the 2002 Oakland Athletics and how their use of statistics allowed them to compile a very cheap and talented team that would go on a famous 20-game win streak. This "moneyball" strategy took off, and I imagine many major league teams have similar tools for acquiring underrated players on team-friendly deals. This project will mainly be focused on discovering how these teams perform this analysis.

Creating a Data Frame of Relevant and Useful Statistics

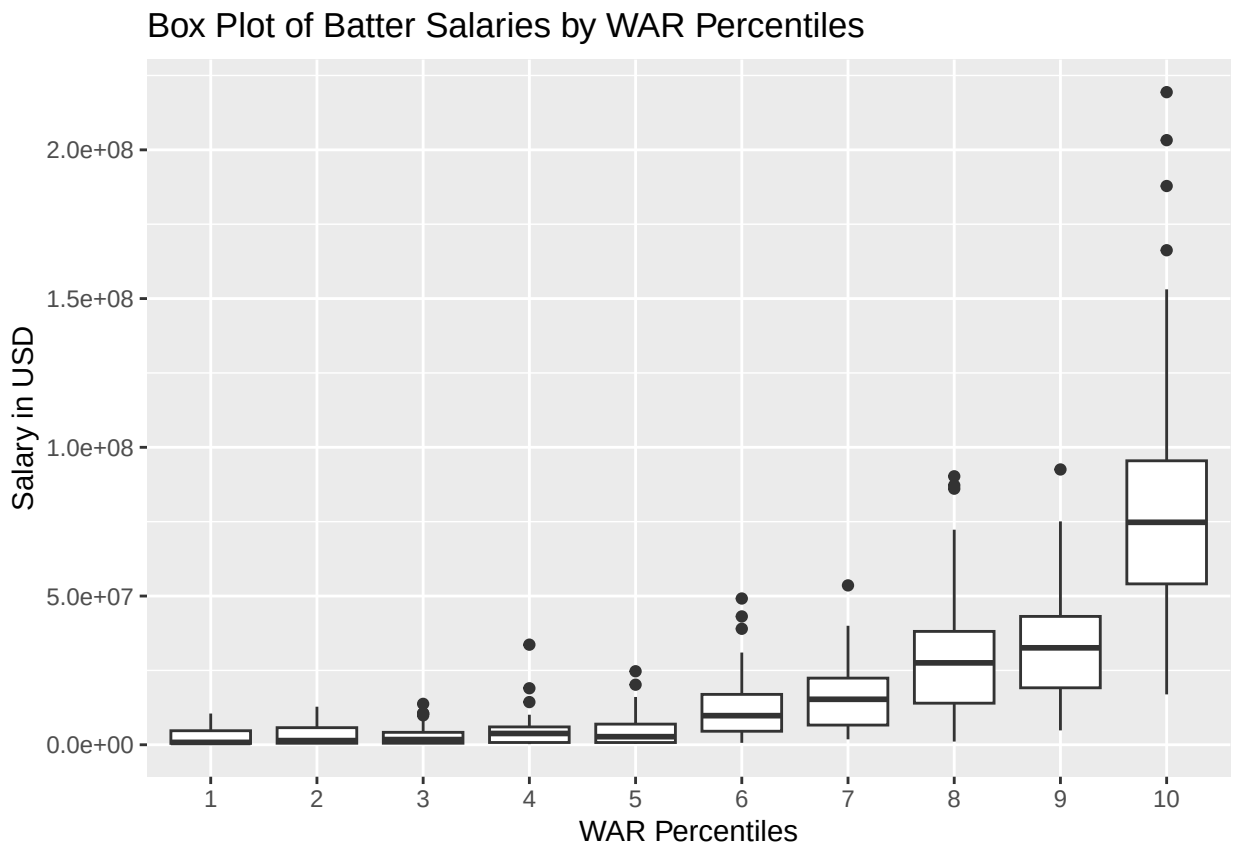
```
## # A tibble: 10 x 11
##    playerID  years   salary   WAR    HR   RBI    BA   SLG   OBP   OPS
##    <chr>     <int>    <int> <dbl> <int> <int> <dbl> <dbl> <dbl> <dbl>
##  1 abreubo01     9 93133333 33      173   835 0.286 0.463 0.389 0.852
##  2 abreujo02     3 27332667 12.3     91   308 0.299 0.515 0.360 0.875
##  3 ackledu01     2  3800000  4.44    26   115 0.234 0.360 0.294 0.653
##  4 adamsma01     1   516000  2.23    15    68 0.288 0.457 0.321 0.779
##  5 alfoned01     2 10500000  0.84    24   158 0.274 0.399 0.342 0.741
##  6 alomaro01     1  7939664  0.6     11    53 0.266 0.376 0.331 0.708
##  7 alonsyo01     1  1400000  1.49     9    62 0.273 0.393 0.348 0.741
##  8 aloumo01      2 19000000  5.06    61   197 0.286 0.511 0.359 0.871
##  9 altuvjo01     5  8426200 20.0     58   310 0.314 0.442 0.357 0.800
## 10 alvarpe01     2  2900000  5.91    66   185 0.238 0.470 0.307 0.777
## # i 1 more variable: WAR_percentile <int>
```

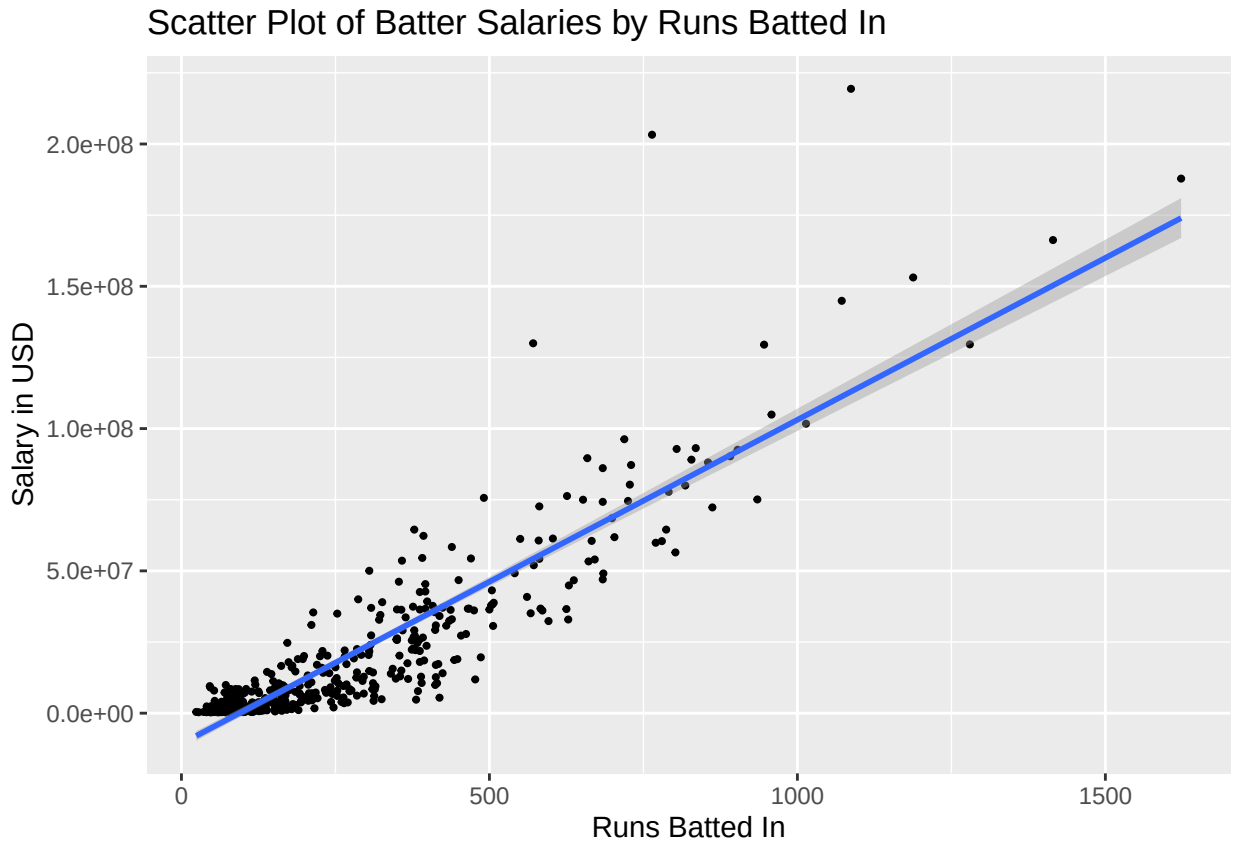
The original batting data created a separate case for each year that a player played for, as well as a separate case for each team that the player played for during that year. This clouded our statistical analysis, since even though a single player should be treated as a single case with their single salary, those players became multiple different cases with the same salary but different stats. If a player was signed to an expensive long-term contract, and then they got traded to a different for a few months of a season and only put up a relatively low number of at-bats, this player's salary would be attached to a pitiful stat line and treated similarly to their all-star numbers that earned them their high salary in the first place. In short, I grouped all of the stats by player ID, and summed up their stats to generate career stats and career money earned from 2002 - 2016 (the "Moneyball" era of baseball).

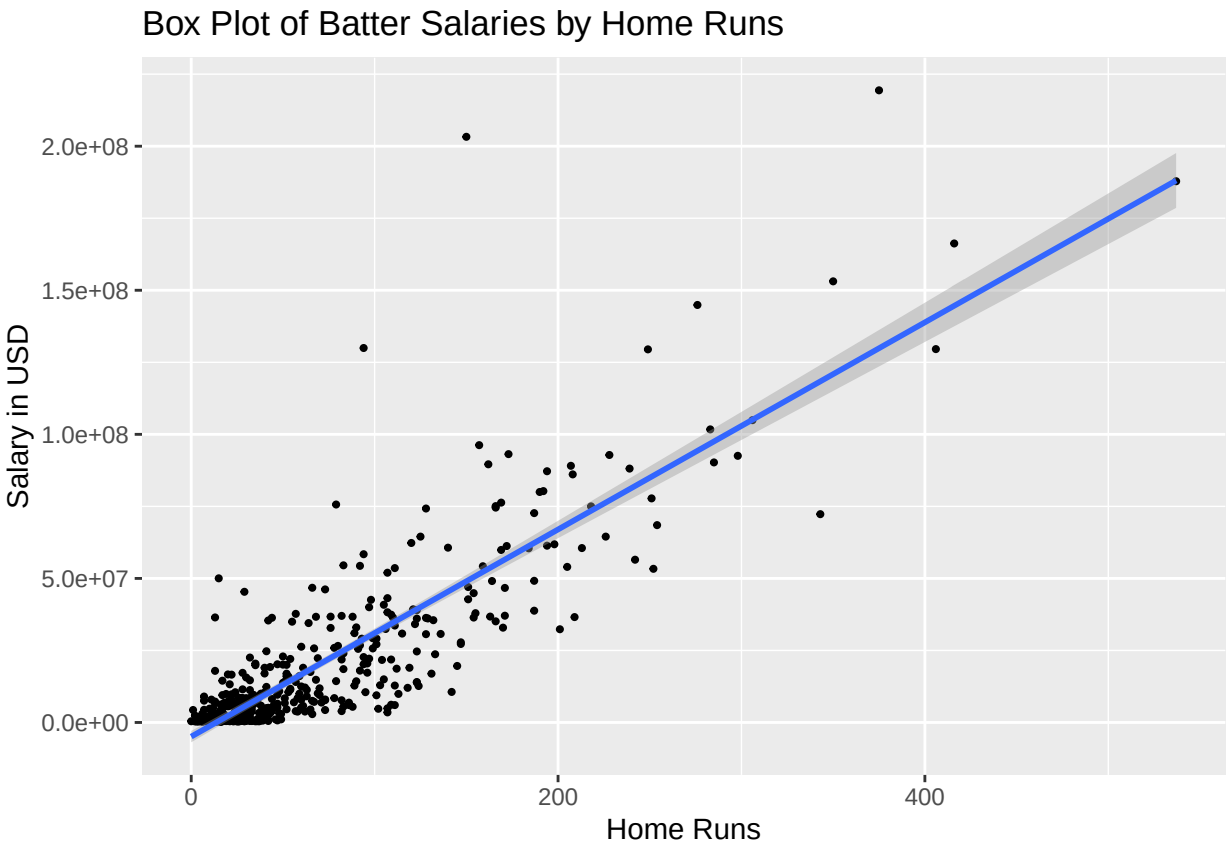
I then attached these stats to a different dataset that contained information on a player's WAR (wins above replacement), a statistic that takes in a multitude of nuanced league-wide factors and tells us how many wins that the player contributed to that a similar player would not have. I attached this combined data set to another data set containing the salaries of players throughout their careers. This helped with a more straightforward statistical analysis.

I also created functions to generate more nuanced offensive metrics from their stats, like on-base percentage (OBP), slugging percentage (SLG), and on-base plus slugging (OPS).

Different Statistics Plotted Against Player Salary







The first box plot of a player's salary against where they fall on a WAR percentile scale gives us the clearest indication that there is a statistical correlation between offensive production and total salary. While this seems obvious, the box plot showing that obvious of a trend is positive indication that we can help predict salaries through statistical analysis. Players that fall in the highest WAR percentile have the highest average salaries, while players that fall into the lower half of league WAR make the least amount of money.

The other plots also show how salaries correlate with runs batted in and home runs. There seems to be a clear positive linear trend with RBIs, and a little more subtle positive linear trend with home runs. RBIs are an important metric, runs will win baseball games and a player that drives in more of them will be more sought after. This also shows us that more straightforward stats like RBIs can also contribute to our investigation.

Analyses: Linear Regression Model Test, and Polynomial Regression Model Evaluation

```
n <- nrow(battingSalaries)
folds <- sample(rep(1:3, length.out = n))

all_degree_results <- numeric(5)
all_r <- numeric(5)
all_adj_r <- numeric(5)
all_AIC <- numeric(5)
all_BIC <- numeric(5)
```

```r
for (i in 1:5) {
    curr_CV_results <- numeric(3)
    curr_model <- lm(salary ~ poly(years * (WAR + RBI + HR) +
        BA + OPS + OBP, i), data = battingSalaries)
    all_r[i] <- summary(curr_model)$r.squared
    all_adj_r[i] <- summary(curr_model)$adj.r.squared
    all_AIC[i] <- AIC(curr_model)
    all_BIC[i] <- BIC(curr_model)

    for (fold in 1:3) {
        test_indices <- which(folds == fold)
        train_indices <- setdiff(1:n, test_indices)

        train_data <- battingSalaries[train_indices, ]
        test_data <- battingSalaries[test_indices, ]

        curr_model <- lm(salary ~ poly(years * (WAR + RBI + HR) +
            BA + OPS + OBP, i), data = train_data)

        test_predictions <- predict(curr_model, newdata = test_data)

        mspe <- mean((test_data$salary - test_predictions)^2)
        curr_CV_results[fold] <- mspe
    }

    all_degree_results[i] <- mean(curr_CV_results)

}

all_verification_metrics <- c(which.min(all_degree_results),
    which.max(all_r), which.max(all_adj_r), which.min(all_AIC),
    which.min(all_BIC))
bestPoly <- median(all_verification_metrics)
bestPoly
```

```
## [1] 2
```

```
##
## Call:
## lm(formula = salary ~ years * (WAR + RBI + HR) + BA + OPS + OBP,
##     data = battingSalaries)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -41737380  -4633791   -282470   3911240  98476525
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -30105648    9458681  -3.183 0.001558 **
## years        -1826646    1035176  -1.765 0.078306 .
## WAR           -452044     240262  -1.881 0.060547 .
## RBI            -48550      26634  -1.823 0.068985 .
## HR             519019      92256   5.626 3.23e-08 ***
```

```
## BA          139309413   45673295   3.050 0.002421 **
## OPS         -82627115   21931043  -3.768 0.000186 ***
## OBP         161685406   40445112   3.998 7.46e-05 ***
## years:WAR      128280      32046   4.003 7.30e-05 ***
## years:RBI       18393       3194   5.759 1.56e-08 ***
## years:HR       -65963      10734  -6.145 1.75e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 11220000 on 455 degrees of freedom
## Multiple R-squared:  0.8636, Adjusted R-squared:  0.8606
## F-statistic: 288.1 on 10 and 455 DF,  p-value: < 2.2e-16
```

```
batting_fit_poly <- lm(salary ~ poly(years * (WAR + RBI + HR) +
    BA + OPS + OBP, bestPoly), data = battingSalaries)

# Juan Soto's Contract
js_career <- 15
js <- predict(batting_fit_poly, data.frame(years = js_career,
    WAR = 7.9 * js_career, RBI = 109 * js_career, BA = 0.288,
    OPS = 0.989, OBP = 0.419, HR = 41 * js_career)) * 2

# Shohei Ohtani's Contract
so_career <- 10
so <- predict(batting_fit_poly, data.frame(years = so_career,
    WAR = 9.2 * so_career, RBI = 130 * so_career, BA = 0.31,
    OPS = 1.036, OBP = 0.39, HR = 54 * so_career)) * 2

# Fernando Tatis Jr.'s Contract
ft_career <- 14
ft <- predict(batting_fit_poly, data.frame(years = ft_career,
    WAR = 6.6 * ft_career, RBI = 97 * ft_career, BA = 0.282,
    OPS = 0.975, OBP = 0.364, HR = 42 * ft_career)) * 2

# Jurickson Profar Deserved Salary
jps_career <- 1
jps <- predict(batting_fit_poly, data.frame(years = jps_career,
    WAR = 3.6 * jps_career, RBI = 85 * jps_career, BA = 0.28,
    OPS = 0.776, OBP = 0.38, HR = 24 * jps_career)) * 2

# Jurickson Profar Projected Contract
jpc_career = 3
jpc <- predict(batting_fit_poly, data.frame(years = jpc_career,
    WAR = 3.6 * jpc_career, RBI = 85 * jpc_career, BA = 0.28,
    OPS = 0.776, OBP = 0.38, HR = 24 * jpc_career)) * 2

# Possible Jackson Merrill Extension
jm_career = 14
jm <- predict(batting_fit_poly, data.frame(years = jm_career,
    WAR = 4.4 * jm_career, RBI = 90 * jm_career, BA = 0.292,
    OPS = 0.826, OBP = 0.326, HR = 24 * jm_career)) * 2

cat("Based on Juan Soto's stats last year, he should make", format(js,
    big.mark = ","), "USD over", js_career, "years.\n\n")
```

```
## Based on Juan Soto's stats last year, he should make 371,837,141 USD over 15 years.
```

```
cat("Based on Shohei Ohtani's stats last year, he should make",
    format(so, big.mark = ","), "USD over", so_career, "years.\n\n")
```

```
## Based on Shohei Ohtani's stats last year, he should make 304,827,643 USD over 10 years.
```

```
cat("Based on Fernando Tatis Jr.'s stats before his extension, he should have made",
    format(ft, big.mark = ","), "USD over", ft_career, "years.\n\n")
```

```
## Based on Fernando Tatis Jr.'s stats before his extension, he should have made 363,811,783 USD over 14
```

```
cat("Based on Jurickson Profar's stats last year, he should have made",
    format(jps, big.mark = ","), "USD.\n\n")
```

```
## Based on Jurickson Profar's stats last year, he should have made 5,022,416 USD.
```

```
cat("Based on Jurickson Profar's stats last year, he should make",
    format(jpc, big.mark = ","), "USD over", jpc_career, "years.\n")
```

```
## Based on Jurickson Profar's stats last year, he should make 24,425,490 USD over 3 years.
```

```
cat("Based on Jackson Merrill's stats last year, the Padres should extend him for",
    format(jm, big.mark = ","), "USD over", jm_career, "years.")
```

```
## Based on Jackson Merrill's stats last year, the Padres should extend him for 336,407,958 USD over 14
```

Before discussing the linear model's results, I will discuss how it was developed. I included seven variables for the model's consideration, which resulted in an R^2 value of 0.86, meaning 86% of contract variability could be explained through these seven variables. The first and most obvious variables I chose were WAR, RBIs, and home runs, as those were the variables that I saw bad the most obvious correlation with salary. In addition, I included batting average (BA), OBP, and OPS, as these are regarded as important metrics in the baseball world for discussing a player's offensive output. The correlation between these stats and player salaries was less obvious from my findings but I still included them to see if they actually are impactful when discussing salary. Finally, I included the length of a contract/signing in years as an interaction factor between WAR, RBIs, and HRs. These variables will often depend on how long a player has been playing, since you will accrue more WAR, RBI, and HR as your career continues. I would like my model to take that into consideration.

I tried to find if a polynomial regression model would fit my data better. I tested this by trying different degree polynomials and used adjusted R^2 values, AIC, BIC, and cross-validation to determine which degree polynomial fit my model and data the best. Seeing which each degree polynomial each test output and taking the median best degree, I found that a 2nd degree polynomial was the best fit for my model.

As for applying my model, let's go to Juan Soto's 15 year 765 million dollar contract. If Soto were to maintain his all-star stat line last year over the course of those 15 years, the model predicts that he would only be worth about 371 million dollars. This is nearly half of what he is actually making. Initial reactions to this deal have deemed it a massive overpay, and my model seems to agree.

Going to the previous most expensive contract in sports history, Shohei Ohtani is currently in a 10 year 700 million dollar contract. If Ohtani were to maintain his all-star stat line last year over the course of those 10

years, the model predicts that the contract would only be worth about 300 million dollars. This is, again, nearly half of what he is actually making. However, there are actually outside factors that make this a lot less of an overpay than it may seem. Ohtani is the only two-way player in the league, meaning he both hits AND pitches. The Dodgers are not only paying for a hitter, but also for a star pitcher. In addition, Ohtani is one of the few Japanese players in the league, and so he generates a lot of endorsements and ad-revenue from Japan. The Dodgers felt that 700 million dollars would be covered by Ohtani's endorsements, as well as be worth the cost of paying for both an MVP hitter and an ace pitcher. I am inclined to agree.

Let's move to a contract that is deemed a very cheap and team-friendly, Fernando Tatis Jr.'s 14 year 340 million dollar deal. Based on the stats from 2021 that Tatis Jr. was signed for, the model predicts that over 14 years, he would make 363 million dollars. This is about 23 million dollars over what he is actually making. Many analysts have called this deal very agreeable and friendly for both the San Diego Padres and Tatis Jr., and my model and I agree.

I've also wondered if players can make less than what they're actually worth. I fed Jurickson Profar's breakout season last year into the model, as Profar's burst of power seemed to appear out of nowhere to most sports analysts. The San Diego Padres signed Profar to a 1 year, 1 million dollar deal to be a backup outfielder. However, Profar had an all-star season and became their starting outfielder and leadoff hitter. Putting his breakout stats into the model, it seemed that he actually deserved 5 million dollars this past year. The Padres saved about 4 million dollars on an allstar outfielder. Going forward, the model predicts that a 3-year contract (which is what analysts believe he is getting) for Profar would go for about 24 million dollars.

This model would also be useful in negotiations for rookies looking to have their contract extended by their teams. I took Jackson Merrill's all-star rookie season with the San Diego Padres last year to see how much his rookie season could earn him. Fernando Tatis Jr. had a similar all-star rookie season, which earned him a 14-year contract extension. I asked the model to predict how much money Jackson Merrill could earn in the future if he were to keep up his rookie stat line for 14 years, and the model predicted that he is worth about 340 million dollars (about the same as Tatis!). The San Diego Padres may consider signing him now to prevent him from becoming even more expensive in the future, or wait to see if he can continue this same level of offensive production.

Conclusion

After using the model to predict current-day contracts, I found that contract prices have indeed skyrocketed in recent years. This was obvious from my observations. The most expensive contract from the "Moneyball" era (2002-2016) that my data is looking at was a 10-year 252 million dollar contract signed by Alex Rodriguez. The most expensive contract as of 2024 is Juan Soto's 765 million dollar contract with the Mets. From my testing, I found that my model, when used to predict current day contracts, would predict almost 1/2 of the actual value of their contract. For example, Fernando Tatis Jr. signed a 14-year 340 million dollar contract in 2021. When I input Tatis Jr.'s stats from 2021 into the model to predict a 14 year contract, it estimated about 180 million dollars, which is nearly half of his actual earnings over 14 years. So, a rule of thumb I found was to multiply my model's output by 2 when trying to predict modern-day contracts. It's interesting to see that modern baseball contracts have doubled in the last 8 years.

From my research, it does seem very possible to generate reasonable and realistic contracts purely from offensive baseball stat lines, which is was the problem I was seeking to address. A possible shortcoming of my project is that which teams were doing the signings was not considered in the contract process. Certain teams, like the Dodgers, Yankees, and Mets, are able to spend a lot more money than teams like the Athletic's or the Guardians. I would be curious to see if the teams interested in a player would have any impact on how much money they are projected to earn. Overall, I am satistified with my findings and will be curious to see how my model will stack up with reality as the off season continues.

Reflection

I think most of the project was spent getting the data into an easy-to-analyze format, as generating career stats and thinking of different ways to make the data more clear was very time consuming. Once the data was all sorted, I went very smoothly from there. I was able to see trends in the data through visualization, and the model was actually able to read trends and output realistic values. I did try to make a model designed around pitching statistics, but those are a lot more nebuluous and less concrete. The real data that goes into determining pitching value a lot more nuanced and a lot harder to utilize effectively. I could not get anything concrete, so I left it out of the writeup. I think spent about 6 hours on this project.

Appendix

```
# Code for downloading a dataset with players and their WAR
# (wins above replacement)
# if(!file.exists('./data')){dir.create('./data')} fileUrl
# <-
# 'https://www.baseball-reference.com/data/war_daily_bat.txt'
# download.file(fileUrl, destfile='war_daily_bat.csv',
# method='curl')
war <- read.csv("war_daily_bat.csv", header = TRUE)

# A function to calculate a metric of offensive production
# (SLG)
slgCalculator <- function(hits, doubles, triples, hrs, abs) {
    singles <- hits - (doubles + triples + hrs)
    SLG <- (singles + (2 * doubles) + (3 * triples) + (4 * hrs))/abs
    return(SLG)
}

# A function to calculate how often a player gets on base
# (OBP)
obpCalculator <- function(hits, walks, hbp, abs, sf) {
    OBP <- (hits + walks + hbp)/(abs + walks + hbp + sf)
    return(OBP)
}

# Takes a data set of statistics related to batting and
# attaches salaries and WAR to it, among other data
# cleaning.
battingSalaries <- Lahman::Batting |>
    # Combine the batting dataset with the salary dataset
left_join(select(Lahman::Salaries, playerID, yearID, salary),
    by = c("playerID", "yearID")) |>
    # Combine our current dataset with the WAR dataset.
left_join(select(war, player_ID, year_ID, WAR, pitcher), by = c(playerID = "player_ID",
    yearID = "year_ID")) |>
    # Take only seasons from 2002 onwards (the 'Moneyball'
    # era). Remove any cases with no noted salary. Remove
    # any unqualified hitters (hitters with less than 502
    # at bats). Remove any pitchers (pitchers are paid on
```

```
    # pitching stats, not hitting stats)
filter(yearID >= 2002, !is.na(salary), AB >= 502, pitcher ==
    "N") |>
    # Turn WAR into a number.
mutate(WAR = as.numeric(WAR)) |>
    # Sum the stats of each player by each year they
    # played, regardless of team. This prevents the data
    # being ruined by any mid-season trades. If a player
    # was traded to a different team during the season,
    # their stats would be double counted (one for each
    # team they played on during the year) and
    # overrepresented in the data set.
group_by(playerID, yearID) |>
    summarise(WAR = sum(WAR), H = sum(H), AB = sum(AB), BB = sum(BB),
        HBP = sum(HBP), X2B = sum(X2B), X3B = sum(X3B), HR = sum(HR),
        RBI = sum(RBI), SF = sum(SF), salary = sum(salary)) |>
    # Sum the career stats of each player. This allows us
    # to get the total amount of a money that a player
    # earned during years they qualified as a hitter. This
    # gives us an idea of how much money they really
    # 'earned' over the course of their career, and the
    # stats that accompany that salary.
group_by(playerID) |>
    summarise(H = sum(H), AB = sum(AB), BB = sum(BB), HBP = sum(HBP),
        X2B = sum(X2B), X3B = sum(X3B), SF = sum(SF), years = n(),
        salary = sum(salary), WAR = sum(WAR), HR = sum(HR), RBI = sum(RBI),
        ) |>
    # Add additional offensive metrics for more data
mutate(BA = H/AB, SLG = slgCalculator(H, X2B, X3B, HR, AB), OBP = obpCalculator(H,
    BB, HBP, AB, SF), OPS = SLG + OBP) |>
    # Create their WAR percentiles.
mutate(WAR_percentile = ntile(WAR, 10))
head(battingSalaries[, c(1, 9:18)], 10)
```

```
## # A tibble: 10 x 11
##    playerID  years   salary   WAR    HR   RBI    BA   SLG   OBP   OPS
##    <chr>     <int>    <int> <dbl> <int> <int> <dbl> <dbl> <dbl> <dbl>
##  1 abreubo01     9 93133333 33      173   835 0.286 0.463 0.389 0.852
##  2 abreujo02     3 27332667 12.3     91   308 0.299 0.515 0.360 0.875
##  3 ackledu01     2  3800000  4.44    26   115 0.234 0.360 0.294 0.653
##  4 adamsma01     1   516000  2.23    15    68 0.288 0.457 0.321 0.779
##  5 alfoned01     2 10500000  0.84    24   158 0.274 0.399 0.342 0.741
##  6 alomaro01     1  7939664  0.6     11    53 0.266 0.376 0.331 0.708
##  7 alonsyo01     1  1400000  1.49     9    62 0.273 0.393 0.348 0.741
##  8 aloumo01      2 19000000  5.06    61   197 0.286 0.511 0.359 0.871
##  9 altuvjo01     5  8426200 20.0     58   310 0.314 0.442 0.357 0.800
## 10 alvarpe01     2  2900000  5.91    66   185 0.238 0.470 0.307 0.777
## # i 1 more variable: WAR_percentile <int>
```