Technical Report

Group 6

Abstract

The competitive balance tax (implemented in 2002) is a rule implementing salary caps for baseball players in a given team. This paper attempts to uncover if there is any underlying difference in the number of lost potential players (called “couldabeens”) in the pre-rule (1968-2002) and post-rule eras (2002-2018).

Introduction

Major League Baseball’s “competitive balance tax,” first implemented in 2002, has become a favorite subject of outrage among players and fans alike in recent years. On paper, the policy was meant to make the sport more competitive and boost salaries for players on lower-revenue teams. The owners and the players’ union would negotiate the largest reasonable amount a team could spend on its roster, and any team that wanted to spend beyond that cap would pay a “tax” (a share of their excess payroll) to be redistributed to the poorer teams. In practice the policy has effectively become a salary cap, particularly after the 2016 renegotiations. In 2018 only two MLB teams out of thirty went over the salary threshold, and only one of those by any significant margin (the team that did so, the Boston Red Sox, unsurprisingly went on to win the World Series). With leaguewide revenue growth far outpacing growth in the revenue cap and most younger players effectively locked out of salary negotiations by free agency rules, players have understandably chafed at the limitation on their salaries, with mutterings of a player strike unless the rule is altered.

But while the depressive effect on players’ salaries is not in dispute, the question of whether the rule *hurts the game* (a far graver sin among baseball fans than merely conspiring to lower salaries) is more open. At least in the conventional wisdom, the tax encourages teams to cultivate a massive pool of recruits and then pay the best of them the minimum permitted salary for up to seven years before they age into free agency. Once these players enter free agency, the narrative goes, the team dumps them for younger players whom they can pay the minimum salary. The remainder of their salary cap goes towards retaining a few older superstars with the name recognition to bring in fans, with many good-but-not-Mike-Trout players pushed into early retirement. The implication, then, is that the salary policy “hurts the game” because the MLB is less likely to be putting the best 30 shortstops in the world on the field at any given time. I would like to test this anecdotal observation empirically.

So, the working hypothesis states that since the advent of the competitive balance tax, the number of better players forced into retirement annually will have increased. A “better player forced into retirement” will be defined as a player under the age of 32 who a) left baseball due to their contract not being renewed, rather than injury or personal circumstances, and b) was outperforming the median rookie in their position at the time of their retirement. In other words, we might expect to find the number of couldabeens to increase after the implementation of the rule.

The Couldabeen Classification Problem

First Look: A Logistic Model

Computing Retiree Proportions

Here is what the proportion-appended couldabeen dataframe looks like:

Year as Predictor: Linear Modeling

```r
# Partition dataset into years before and after rule
couldabeens_pre <- prerule(couldabeens)
couldabeens_post <- postrule(couldabeens)
```

```r
# Obtain linear model for pre-rule years
model_pre <- lm(formula = prop ~ I(Year), data = couldabeens_pre)
coefs_pre <- model_pre$coefficients
# Obtain linear model for post-rule years
model_post <- lm(formula = prop ~ I(Year), data = couldabeens_post)
coefs_post <- model_post$coefficients
```

```r
# Obtain linear model for all years
model_comp <- lm(formula = prop ~ I(Year), data = couldabeens)
coefs_comp <- model_comp$coefficients
```

Couldabeens: A Comprehensive Look

Simpson's Paradox

Couldabeens: Pre-rule Era (1969-2002)

Couldabeens: Post-rule Era (2003-2018)

Methods

Discussion

Code Appendix

Importing the Datasets

```
# Load rookies datasets
df_pit_rkes <- read_csv("data/rookie-pitcher.csv")
df_pos_rkes <- read_csv("data/rookie-position.csv")
# Load retirees datasets
df_pit_ret <- read_csv("data/retirees-pitcher.csv")
df_pos_ret <- read_csv("data/retirees-position.csv")
```

Wrangling the Datasets

```
# select appopriate columns from player datasets
wrangle_init <- function(dataset){
  colnames(dataset)[3] <- "WAR"
  dataset %>% select(WAR, Year)
}
```

```
# Obtain wrangled datasets
pit_rkes <- wrangle_init(df_pit_rkes)
pit_ret <- wrangle_init(df_pit_ret)
pos_rkes <- wrangle_init(df_pos_rkes)
pos_ret <- wrangle_init(df_pos_ret)
```

Finding the Couldabeen Thresholds

```
# obtain summary of WAR: median and variance
find_thresholds <- function(dataset, sds = 0){
  dataset %>%
    group_by(Year) %>%
    summarize(mean_WAR = median(WAR), sd_WAR = sqrt(var(WAR))) %>%
    mutate(threshold = mean_WAR + sds*sd_WAR) %>%
    select(Year, threshold)
}
```

```
# Get thresholds in each year
pit_thresholds <- find_thresholds(pit_rkes)
pos_thresholds <- find_thresholds(pos_rkes)
```

```
head(pit_thresholds)
```

```
## # A tibble: 6 x 2
##    Year threshold
##   <dbl>     <dbl>
## 1  1969     1
## 2  1970     0.8
## 3  1971     0.650
## 4  1972     1.1
## 5  1973     0.8
## 6  1974     1
```

Classifying Retiree Couldabeens

```
# Appends above_threshold column to retirees dataset (checks if a retiree exceeds a threshold)
compare_thresholds <- function(dataset, summary_dataset){
  above_threshold <- rep(NA, nrow(dataset))
  for(i in 1:nrow(dataset)){
    year <- as.numeric(dataset[i,2])
    above_threshold[i] <- (dataset[i,1] > summary_dataset[year - 1968, 2])
  }
  dataset <- cbind(dataset,above_threshold)
  colnames(dataset)[3] <- "above_threshold"
  dataset
}
```

```
# See and record which players cross that year's adjusted threshold from rookie players
pit_ret <- compare_thresholds(pit_ret, pit_thresholds)
pos_ret <- compare_thresholds(pos_ret, pos_thresholds)
# Get all retired couldabeens by combining the rows
retirees <- rbind(pit_ret,pos_ret)
```

```
head(retirees)
```

```
##    WAR Year above_threshold
## 1  1.8 1972            TRUE
## 2  0.1 1974           FALSE
## 3  0.3 1976           FALSE
## 4 -0.5 1977           FALSE
## 5  0.4 1977           FALSE
## 6 -1.8 1974           FALSE
```

Counting Couldabeens by Year

```
# Counts couldabeens in a given year
count_cbns <- function(dataset){
  dataset %>%
    group_by(Year) %>%
    summarize(cbns = sum(above_threshold))
}
```

```
# Count the retiree couldabeens by year
couldabeens <- count_cbns(retirees)
```

```
head(couldabeens)
```

```
## # A tibble: 6 x 2
##    Year  cbns
##   <dbl> <int>
## 1  1969     7
## 2  1970     3
## 3  1971     6
## 4  1972    10
## 5  1973     9
## 6  1974     6
```

Counting Retirees in a Given Year

```
# Yields the sum of retirees in two datasets (pitchers, position commonly)
total_retirees_by_yr <- function(pitchers, position){
  year_count_pitchers <- retirees_by_yr(pitchers)$retirees
  year_count_position <- retirees_by_yr(position)$retirees
  data.frame(Year = 1969:2018, retirees = year_count_position + year_count_pitchers)
}
# Counts the retirees in a single dataset
retirees_by_yr <- function(dataset){
  year_count <- dataset %>%
    group_by(Year) %>%
    summarize(retirees = n())
}
```

```
# Find number of retirees by year
num_retirees <- total_retirees_by_yr(df_pit_ret, df_pos_ret)
num_retirees <- data.frame(retirees = num_retirees)
```

```
head(num_retirees)
```

```
##   retirees.Year retirees.retirees
## 1          1969                32
## 2          1970                35
## 3          1971                44
## 4          1972                49
## 5          1973                41
## 6          1974                45
```

Retiree Proportion of Couldabeens

```
# Append number of retirees that year
couldabeens <- cbind(couldabeens, num_retirees)
# Find and append proportion of couldabeens : retirees
couldabeens <- couldabeens %>% mutate(prop = cbns/retirees)
```

```
head(couldabeens)
```

```
##   Year cbns retirees       prop
## 1 1969    7       32 0.21875000
## 2 1970    3       35 0.08571429
## 3 1971    6       44 0.13636364
## 4 1972   10       49 0.20408163
## 5 1973    9       41 0.21951220
## 6 1974    6       45 0.13333333
```

Modeling Helper Functions

```r
# Filters year to be in the post-rule era
postrule <- function(dataset, rule_year = 2002){
  dataset %>% filter(Year > rule_year)
}
# Filters year to be in the pre-rule era
prerule <- function(dataset, rule_year = 2002){
  dataset %>% filter(Year <= rule_year)
}
# Prep the boolean data for a logistic model
prep_booleans <- function(dataset){
  bools <- as.numeric(dataset$above_threshold)
  dataset[,3] <- bools
  dataset
}
```

Model Functions

```r
# Logisitic regression model on exceeding the threshold
logistic_model <- function(dataset){
  logistic_model <- glm(formula = above_threshold ~ Year, data = dataset, family = "binomial")
  logistic_model
}
# Linear model predicting proportion of couldabeens based on the year
linear_model <- function(dataset){
  linear_model <- lm(formula = prop ~ I(Year), data = dataset)
  linear_model
}
```

Logistic Models

```r
# Partition dataset into years before and after rule
retirees_pre <- prerule(retirees)
retirees_post <- postrule(retirees)
# On the retirees pre-rule dataset
model_log_pre <- logistic_model(retirees_pre)
model_log_post <- logistic_model(retirees_post)
```

Linear Models

```r
# Partition dataset into years before and after rule
couldabeens_pre <- prerule(couldabeens)
couldabeens_post <- postrule(couldabeens)
# Obtain linear model for pre-rule years
model_pre <- lm(formula = prop ~ I(Year), data = couldabeens_pre)
coefs_pre <- model_pre$coefficients
# Obtain linear model for post-rule years
model_post <- lm(formula = prop ~ I(Year), data = couldabeens_post)
coefs_post <- model_post$coefficients
```

References

(1) https://stathead.com/baseball/