

NESCAC Position Player WAR and wRC

Tyler Marshall

12/23/2019

Packages Used

```
library(tidyverse)

## -- Attaching packages ----- tidyverse 1.2.1 --

## v ggplot2 3.2.1      v purrr  0.3.3
## v tibble  2.1.3      v dplyr  0.8.3
## v tidyr   1.0.0      v stringr 1.4.0
## v readr   1.3.1      v forcats 0.4.0

## -- Conflicts ----- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()     masks stats::lag()

library(robotstxt)
library(knitr)
library(rvest)

## Loading required package: xml2

##
## Attaching package: 'rvest'

## The following object is masked from 'package:purrr':
##
##   pluck

## The following object is masked from 'package:readr':
##
##   guess_encoding
```

Web Scraping

First, I want to make sure that the NESCAC website allows us to scrape data from it. In this case it does, so we can continue by scraping what we need. I have to scrape the data for each team individually and will bind it together in one dataset.

```
paths_allowed("https://www.nescac.com/sports/bsb/2018-19/players?sort=ab&view=&pos=h&r=0")

##
## www.nescac.com          No encoding supplied: defaulting to UTF-8.

## [1] TRUE
```

Amherst

```
amherst_page <- read_html("http://athletics.amherst.edu/sports/bsb/2018-19/teams/amherst?view=lineup&r=")
#URL for Amherst stats

amherst_tables <- html_nodes(amherst_page, "table") #Read in data
amherst <- html_table(amherst_tables[[4]]) #Hitting Data
amherst2 <- html_table(amherst_tables[[5]]) #Extended Hitting Data

amhersthit <- inner_join(amherst, amherst2, by="Name") #Join two hitting datasets
names(amhersthit)[9]<-"Doubles" #Naming column Doubles
names(amhersthit)[10]<-"Triples" #Naming column Triples

amhersthit2 <- amhersthit %>%
  mutate(School= "Amherst") #Add school to be used later
```

Bates

```
bates_page <- read_html("https://www.gobatesbobcats.com/sports/bsb/2018-19/teams/bates")
#URL for Bates data

bates_tables <- html_nodes(bates_page, "table") #Read in data
bates <- html_table(bates_tables[[4]]) #Hitting Data
bates2 <- html_table(bates_tables[[5]]) #Extended Hitting data

bateshit <- inner_join(bates, bates2, by="Name") #Join two hitting datasets
names(bateshit)[9]<-"Doubles" #Naming column Doubles
names(bateshit)[10]<-"Triples" #Naming column Triples

bateshit2 <- bateshit %>%
  mutate(School= "Bates") #Add school to be used later
```

Bowdoin

```
bowdoin_page <- read_html("https://athletics.bowdoin.edu/sports/bsb/2018-19/teams/bowdoin")
#URL for Bowdoin data

bowdoin_tables <- html_nodes(bowdoin_page, "table") #Read in data
bowdoin <- html_table(bowdoin_tables[[4]]) #Hitting data
bowdoin2 <- html_table(bowdoin_tables[[5]]) #Extended hitting data

bowdoinhit <- inner_join(bowdoin, bowdoin2, by="Name") #Join two hitting datasets
names(bowdoinhit)[9]<-"Doubles" #Naming column Doubles
names(bowdoinhit)[10]<-"Triples" #Naming column Triples

bowdoinhit2 <- bowdoinhit %>%
  mutate(School= "Bowdoin") #Add school to be used later
```

Colby

```
colby_page <- read_html("https://gocolbymules.com/sports/bsb/2018-19/teams/colby")  
#URL for data  
  
colby_tables <- html_nodes(colby_page, "table") #Read in data  
colby <- html_table(colby_tables[[4]]) #Hitting data  
colby2 <- html_table(colby_tables[[5]]) #Extended hitting data  
  
colbyhit <- inner_join(colby, colby2, by="Name") #Join two hitting datasets  
names(colbyhit)[9]<-"Doubles" #Naming column Doubles  
names(colbyhit)[10]<-"Triples" #Naming column Triples  
  
colbyhit2 <- colbyhit %>%  
  mutate(School= "Colby") #Add school to be used later
```

Hamilton

```
hamilton_page <- read_html("https://athletics.hamilton.edu/sports/bsb/2018-19/teams/hamilton")  
#URL for Hamilton data  
  
hamilton_tables <- html_nodes(hamilton_page, "table") #Read in data  
hamilton <- html_table(hamilton_tables[[4]]) #Hitting data  
hamilton2 <- html_table(hamilton_tables[[5]]) #Extended hitting data  
  
hamiltonhit <- inner_join(hamilton, hamilton2, by="Name") #Join two hitting datasets  
names(hamiltonhit)[9]<-"Doubles" #Naming column Doubles  
names(hamiltonhit)[10]<-"Triples" #Naming column Triples  
  
hamiltonhit2 <- hamiltonhit %>%  
  mutate(School="Hamilton") #Add school to be used later
```

Middlebury

```
mid_page <- read_html("https://athletics.middlebury.edu/sports/bsb/2018-19/teams/middlebury")  
#URL for Middlebury data  
  
mid_tables <- html_nodes(mid_page, "table") #Read in data  
middlebury <- html_table(mid_tables[[5]]) #Hitting data  
middlebury2 <- html_table(mid_tables[[7]]) #Extended hitting data  
  
middleburyhit <- inner_join(middlebury, middlebury2, by="Name") #Join two hitting datasets  
names(middleburyhit)[9]<-"Doubles" #Naming column Doubles  
names(middleburyhit)[10]<-"Triples" #Naming column Triples  
  
middleburyhit2 <- middleburyhit %>%  
  mutate(School= "Middlebury") #Add school to be used later
```

Trinity

```
trinity_page <- read_html("https://www.bantamsports.com/sports/bsb/2018-19/teams/trinityconn")
#URL for Trinity data

trinity_tables <- html_nodes(trinity_page, "table") #Read in data
trinity <- html_table(trinity_tables[[4]]) #Hitting data
trinity2 <- html_table(trinity_tables[[5]]) #Extended hitting data

trinityhit <- inner_join(trinity, trinity2, by="Name") #Join two hitting datasets
names(trinityhit)[9]<-"Doubles" #Naming column Doubles
names(trinityhit)[10]<-"Triples" #Naming column Triples

trinityhit2 <- trinityhit %>%
  mutate(School= "Trinity") #Add school to be used later
```

Tufts

```
tufts_page <- read_html("https://www.gotuftsjumbos.com/sports/bsb/2018-19/teams/tufts")
#URL for Tufts data

tufts_tables <- html_nodes(tufts_page, "table") #Read in data
tufts <- html_table(tufts_tables[[4]]) #Hitting data
tufts2 <- html_table(tufts_tables[[5]]) #Extended hitting data

tuftshit <- inner_join(tufts, tufts2, by="Name") #Join two hitting datasets
names(tuftshit)[9]<-"Doubles" #Naming column Doubles
names(tuftshit)[10]<-"Triples" #Naming column Triples

tuftshit2 <- tuftshit %>%
  mutate(School= "Tufts") #Add school to be used later
```

Wesleyan

```
wesleyan_page <- read_html("https://athletics.wesleyan.edu/sports/bsb/2018-19/teams/wesleyanconn")
#URL for Wesleyan data

wesleyan_tables <- html_nodes(wesleyan_page, "table") #Read in data
wesleyan <- html_table(wesleyan_tables[[4]]) #Hitting data
wesleyan2 <- html_table(wesleyan_tables[[5]]) #Extended hitting data

wesleyanhit <- inner_join(wesleyan, wesleyan2, by="Name") #Join two hitting datasets
names(wesleyanhit)[9]<-"Doubles" #Naming column Doubles
names(wesleyanhit)[10]<-"Triples" #Naming column Triples

wesleyanhit2 <- wesleyanhit %>%
  mutate(School= "Wesleyan") #Add school to be used later
```

Williams

```
williams_page <- read_html("https://ephsports.williams.edu/sports/bsb/2018-19/teams/williams")
#URL for Williams data

williams_tables <- html_nodes(williams_page, "table") #Read in data
williams <- html_table(williams_tables[[4]]) #Hitting data
williams2 <- html_table(williams_tables[[5]]) #Extended hitting data

williamshit <- inner_join(williams, williams2, by="Name") #Join two hitting datasets
names(williamshit)[9]<-"Doubles" #Naming column Doubles
names(williamshit)[10]<-"Triples" #Naming column Triples

williamshit2 <- williamshit %>%
  mutate(School= "Williams") #Add school to be used later
```

Binding Datasets

```
nescachit <- rbind(amhersthit2, bateshit2, bowdoinhit2, colbyhit2, hamiltonhit2,
                  middleburyhit2, trinityhit2, tuftshit2, wesleyanhit2, williamshit2)
#Bind each NESCAC school hitting data into one dataset
```

Cleaning Up the Data

This step cleans up the data, so it is in the proper format to allow the analysis to be done.

```
nescachit <- nescachit %>%
  filter(Name != "Totals") %>% #Remove Totals
  filter(Name != "Opponent") %>% #Remove Opponents
  mutate(Doubles= as.numeric(Doubles)) %>% #Make numeric variables numeric
  mutate(Triples= as.numeric(Triples)) %>%
  mutate(hr= as.numeric(hr)) %>%
  mutate(hbp= as.numeric(hbp)) %>%
  mutate(sf= as.numeric(sf)) %>%
  mutate(sh= as.numeric(sh)) %>%
  mutate(Singles= h-Doubles-Triples-hr) %>% #Create a variable for Singles
  mutate(bb= as.numeric(bb)) %>%
  mutate(obp= as.numeric(obp)) %>%
  mutate(sb= as.numeric(sb)) %>%
  mutate(cs= as.numeric(cs)) %>%
  mutate(slg= as.numeric(slg))
```

```
## Warning: NAs introduced by coercion
```

```
## Warning: NAs introduced by coercion
```

```
## Warning: NAs introduced by coercion
```

```
## Warning: NAs introduced by coercion
```

```
## Warning: NAs introduced by coercion
## Warning: NAs introduced by coercion
## Warning: NAs introduced by coercion
## Warning: NAs introduced by coercion
## Warning: NAs introduced by coercion
## Warning: NAs introduced by coercion
## Warning: NAs introduced by coercion
```

```
nescachit[is.na(nescachit)] <- 0 #Replace all NAs with 0s
```

Calculations

In the following section, all of the calculations for the necessary statistics are made. The preliminary calculations are done to determine the coefficient for wOBA and wOBA scale for the NESCAC in 2019. Also, lgwSB is calculated which is necessary for the wSB calculation. Runs Per Win is calculated to determine how many runs was worth one win in the NESCAC in 2019 and runs per plate appearance was calculated to be used in the wRC calculation.

Once the necessary preliminary calculations were made OPS, wOBA, wRAA, wSB, fOWar and wRC were all calculated for each individual player. A Shiny App was then created to display the results of this project.

```
nescactotal <- nescachit %>%
  mutate(totalbb= sum(bb)) %>% #Finding totals to be used to determine wOBA coefficient
  mutate(totalhbp = sum(hbp)) %>%
  mutate(totalsingles= sum(Singles)) %>%
  mutate(totaldoubles= sum(Doubles)) %>%
  mutate(totaltriples= sum(Triples)) %>%
  mutate(totalhr= sum(hr)) %>%
  mutate(totalpa= sum(ab + bb + sf + hbp)) %>%
  mutate(totalsb= sum(sb)) %>%
  mutate(totalcs= sum(cs)) %>%
  mutate(totalruns= sum(r)) %>%
  mutate(averageobp= sum(obp)/195) %>% #Finding average OBP since average wOBA equals average OBP
  mutate(coef = (.55*totalbb + .57 * totalhbp + .7 * totalsingles +
    totaldoubles + 1.27*totaltriples +
    1.65* totalhr)/(totalpa)) %>% #Coefficient for wOBA
  mutate(wOBAscale= averageobp/coef) %>% #Found wOBAscale for NESCAC
  mutate(lgwSB= (totalsb *.2 + totalcs * -.4)/(totalsingles+totalbb+totalhbp)) %>%
  #Found lgwSB for NESCAC using .2 for runSB and -.4 for runCS (based on MLB recent years)
  mutate(RPW = 9 *(totalruns/2871.6)*1.5 +3) %>%
  #2871.6 is total innings, calculating Runs per Win
  mutate(RperPA = totalruns/totalpa) #Calculate runs per plate appearance needed for wRC

nescacwar <- nescachit %>%
  mutate(OPS= obp+slg) %>%
  mutate(wOBA = ((1.29*.55)*bb + (1.29*.57) *hbp + (1.29*.7)*Singles
```

```

      + (1.29) * Doubles + (1.29*1.2) * Triples
      + (1.29*1.65)* hr) / (ab+bb+hbp+sf)) %>%
mutate(wRAA= ((wOBA- .356)/1.29)*(ab+bb+hbp+sf+sh)) %>%
mutate(wSB= (sb*.2) + (cs*-.4) - (.0176 * (Singles+bb+hbp))) %>%
mutate(fOWar= (wRAA + wSB + 20/600 * pa) / 13.36) %>%
#Using Fangraphs War (only offense), divided by runs per win which is 13.36 for the NESAC
mutate(wRC= (((wOBA- .356/1.29)) + .16)*pa) #wRC Calculation

#Add in Class Year to be used in Shiny App
nescacwar2 <- nescacwar %>%
  mutate(Year = Yr.x) %>%
  mutate(Year= case_when(grepl("Sr", Year) ~ "Senior",
    grepl("Jr", Year) ~ "Junior",
    grepl("So", Year) ~ "Sophomore",
    grepl("Fr", Year) ~ "First-Year",
    grepl("Fy", Year) ~ "First-Year",
    grepl("19", Year) ~ "Senior",
    grepl("20", Year) ~ "Junior",
    grepl("21", Year) ~ "Sophomore",
    grepl("22", Year) ~ "First-Year",
    grepl("Christian Beal", Name) ~ "Sophomore",
    grepl("Giovanni Torres", Name) ~ "Junior",
    grepl("Jack Arend", Name) ~ "Junior",
    grepl("Kyle Carter", Name) ~ "Junior",
    grepl("Justin White", Name) ~ "Senior",
    grepl("Bryan Gotti", Name) ~ "First-Year",
    grepl("Jon Lindgren", Name) ~ "Junior",
    grepl("Antonio Jareno", Name) ~ "First-Year",
    grepl("Dan Trulli", Name) ~ "Senior",
    grepl("Andrew Chi", Name) ~ "First-Year",
    grepl("Will Sylvia", Name) ~ "Junior",
    grepl("Pat Beaton", Name) ~ "Junior",
    grepl("Noah Loughlin", Name) ~ "Sophomore",
    grepl("Kevin Lucey", Name) ~ "First-Year",
    grepl("Colin Coyne", Name) ~ "Sophomore",
    grepl("Leo Watson", Name) ~ "Junior",
    grepl("Dan Brady", Name) ~ "Sophomore",
    grepl("Chris Capo", Name) ~ "First-Year",
    grepl("Rob Matson", Name) ~ "Sophomore",
    TRUE ~ as.character(Year)))

#Add in Position to be used in Shiny App
nescacwar2 <- nescacwar2 %>%
  mutate(Position= Pos.x) %>%
  mutate(Position= case_when(grepl("Sr", Position) ~ "Senior",
    grepl("INF", Position) ~ "Infield",
    grepl("IF", Position) ~ "Infield",
    grepl("1B", Position) ~ "Infield",
    grepl("2B", Position) ~ "Infield",
    grepl("SS", Position) ~ "Infield",
    grepl("3B", Position) ~ "Infield",
    grepl("CF", Position) ~ "Outfield",
    grepl("C", Position) ~ "Catcher",

```

```

grepl("OF", Position) ~ "Outfield",
grepl("UTIL", Position) ~ "Utility",
grepl("UTL", Position) ~ "Utility",
grepl("UT", Position) ~ "Utility",
grepl("DH", Position) ~ "Designated Hitter",
grepl("P", Position) ~ "Designated Hitter",
grepl("Matt Santos", Name) ~ "Infield",
grepl("Ryan Nakajima", Name) ~ "Outfield",
grepl("Ryan Noone", Name) ~ "Catcher",
grepl("Nate Bozzella", Name) ~ "Infield",
grepl("Adam Geibel", Name) ~ "Infield",
grepl("Jamie Mills", Name) ~ "Outfield",
grepl("Kyle Sylvester", Name) ~ "Infield",
grepl("Spencer Nastala", Name) ~ "Catcher",
grepl("Ryan Young", Name) ~ "Utility",
grepl("Matt Koperniak", Name) ~ "Outfield",
grepl("Geoff Stillman", Name) ~ "Outfield",
grepl("Johnny Stamatidis", Name) ~ "Infield",
grepl("Ben Reinisch", Name) ~ "Infield",
grepl("Alex Rodriguez", Name) ~ "Catcher",
grepl("Mike Guanci", Name) ~ "Infield",
grepl("Robbie Cronin", Name) ~ "Infield",
grepl("Mack Lauder", Name) ~ "Cathcer",
grepl("Vincent Capone", Name) ~ "Infield",
grepl("Eric Thronson", Name) ~ "Catcher",
grepl("Tyler Buckley", Name) ~ "Utility",
grepl("Brett Stevenson", Name) ~ "Outfield",
grepl("Ian Steckel", Name) ~ "Outfield",
grepl("Gianni Valentini", Name) ~ "Infield",
grepl("Jimmy Edwards", Name) ~ "Infield",
grepl("Joseph Celio", Name) ~ "Outfield",
grepl("Kevin Bomba", Name) ~ "Outfield",
grepl("Ian Maldonado", Name) ~ "Infield",
grepl("Alex Steele", Name) ~ "Infield",
grepl("Cameron Crowley", Name) ~ "Infield",
TRUE ~ as.character(Position))

```

```
write.csv(nescacwar2, file= "nescacwar2.csv")
```

```
save(nescacwar2, file="nescacwar2.Rda")
```

References

The following sources were used to find the necessary formuals to calculate fOWar, wOBA, wRAA, wSB, and wRC.

fOWar: <https://library.fangraphs.com/war/war-position-players/D>

wOBA: <https://library.fangraphs.com/offense/woba/>

Deriving wOBA: <https://library.fangraphs.com/the-beginners-guide-to-deriving-woba/>

wSB: <https://library.fangraphs.com/offense/wsb/>

wRAA: <https://library.fangraphs.com/offense/wraa/>

wRC: <https://library.fangraphs.com/offense/wrc/>

Runs Per Win: <https://library.fangraphs.com/misc/war/converting-runs-to-wins/>

MLB wOBA and other constants: <https://www.fangraphs.com/guts.aspx?type=cn>