Does Baseball's Quality At Bat % Have Statistical Significance in Run Scoring?

Daniel Thompson

2022-12-07

Nowadays, in the baseball analytics revolution, more and more stats are becoming the main focus of the game. Not just for coaches and players, but for fans. We're still in the hay day of the launch angle revolution, and now broadcasts before games will actually display the starting lineup's .OPS rather than batting average. We've moved in this direction because we like finding added value in some statistics that one really hasn't thought much of before.

However, there are some statistics out there in use for lower levels of baseball but are not in the main discussion for baseball analytics enthusiasts. One of them, Quality At Bat %, is used commonly at the high school varsity level to shift a player's goal setting from result based to effort based. But does this stat actually improve the results of run scoring?

According to qualityatbats.com, a QAB is any plate appearance that results in a:

- Hard it ball
- Walk
- 8 Pitch At-Bat
- Sac Bunt/Fly
- Move Runners Over w/ Less than 2 Outs
- Base Hit

When it comes to getting a base hit or a walk, these are fairly obvious as they are the main ingredients in measuring OPS. However, for example, let's say a batter, Player A, hits a 110 mph rocket lining out right to the shortstop. Let's say in his next at bat he makes the pitcher really work for an out before grounding out and taking the fatigued pitcher out of the game. In the third at bat he moves a runner from second to third on hard ground ball to third base, and in his fourth he narrowly misses hitting a home run on a sacrifice fly 100mph off the bat. His stat sheet shows 0-3 (sacrifice flyouts don't count against batting average) and is punished statistically with a lower .OPS for not reaching any bases. Now let's take another batter on the team, Player B, who bats 1-4 with a home run and three strikeouts in key RISP positions and a high whiff % trying to hit another home run. Who played better? Who showed more discipline and skill? Hint: It's not the one who ended with the higher .OPS for the game.

Mainstream stats are great and highly useful. That's why they're mainstream. However it's always beneficial to add more context to show how fluid these analytics really are. This paper gathers data from the 2017-2021 MLB seasons (2020 excluded), measures each team's QAB%, regresses the stat onto run scoring in a variety of ways, and find ways to improve the model of what a QAB could be better defined as, to test its efficacy on team's offensive performances.

```
##
## Attaching package: 'dplyr'
## The following objects are masked from 'package:plyr':
##
##     arrange, count, desc, failwith, id, mutate, rename, summarise,
##     summarize

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union

## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --
## v ggplot2 3.3.6     v purrr   0.3.5
## v tibble  3.1.8     v stringr 1.4.1
## v tidyr   1.2.1     v forcats 0.5.2
## v readr   2.1.3
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::arrange()   masks plyr::arrange()
## x purrr::compact()   masks plyr::compact()
## x dplyr::count()     masks plyr::count()
## x dplyr::failwith()  masks plyr::failwith()
## x dplyr::filter()    masks stats::filter()
## x dplyr::id()        masks plyr::id()
## x dplyr::lag()       masks stats::lag()
## x dplyr::mutate()    masks plyr::mutate()
## x dplyr::rename()    masks plyr::rename()
## x dplyr::summarise() masks plyr::summarise()
## x dplyr::summarize() masks plyr::summarize()
## Rows: 719573 Columns: 92
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (16): pitch_type, player_name, events, description, des, game_type, sta...
## dbl  (67): release_speed, release_pos_x, release_pos_z, batter, pitcher, zon...
## lgl   (8): spin_dir, spin_rate_deprecated, break_angle_deprecated, break_len...
## date  (1): game_date
##
## i Use 'spec()' to retrieve the full column specification for this data.
## i Specify the column types or set 'show_col_types = FALSE' to quiet this message.
## Rows: 707551 Columns: 92
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (16): pitch_type, player_name, events, description, des, game_type, sta...
## dbl  (67): release_speed, release_pos_x, release_pos_z, batter, pitcher, zon...
## lgl   (8): spin_dir, spin_rate_deprecated, break_angle_deprecated, break_len...
## date  (1): game_date
##
## i Use 'spec()' to retrieve the full column specification for this data.
## i Specify the column types or set 'show_col_types = FALSE' to quiet this message.
## Rows: 680232 Columns: 92
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (16): pitch_type, player_name, events, description, des, game_type, sta...
## dbl  (67): release_speed, release_pos_x, release_pos_z, batter, pitcher, zon...
## lgl   (8): spin_dir, spin_rate_deprecated, break_angle_deprecated, break_len...
## date  (1): game_date
##
## i Use ‘spec()‘ to retrieve the full column specification for this data.
## i Specify the column types or set ‘show_col_types = FALSE‘ to quiet this message.
## Rows: 763042 Columns: 92
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (16): pitch_type, player_name, events, description, des, game_type, sta...
## dbl  (67): release_speed, release_pos_x, release_pos_z, batter, pitcher, zon...
## lgl   (8): spin_dir, spin_rate_deprecated, break_angle_deprecated, break_len...
## date  (1): game_date
##
## i Use ‘spec()‘ to retrieve the full column specification for this data.
## i Specify the column types or set ‘show_col_types = FALSE‘ to quiet this message.
```

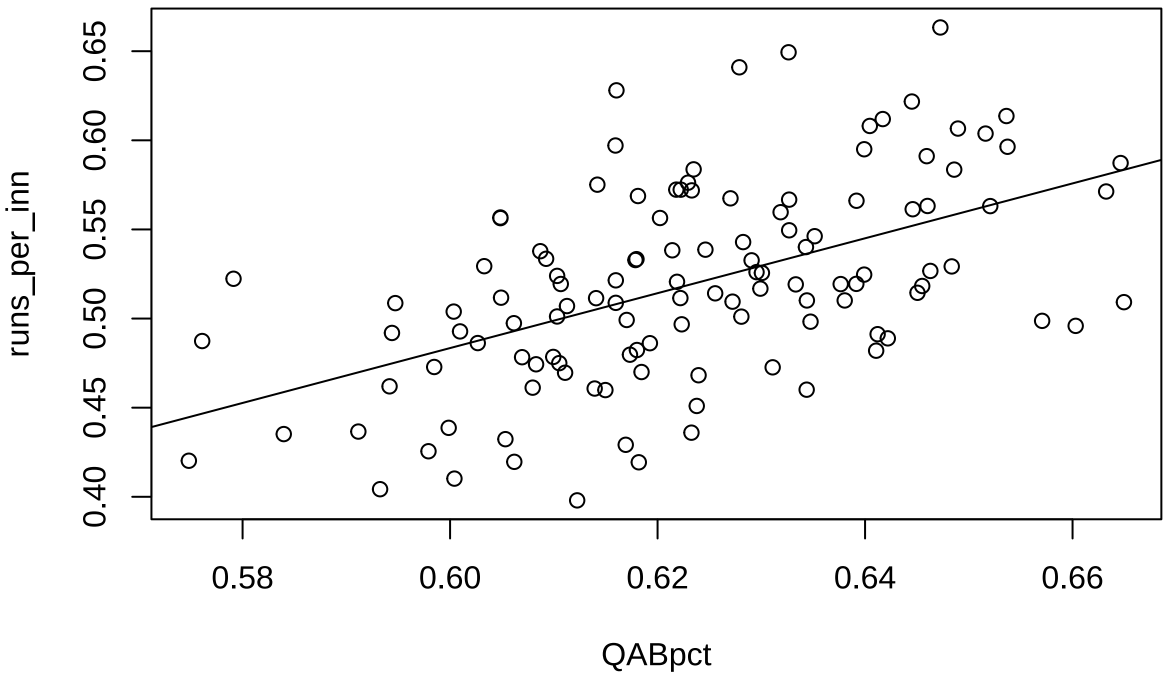
```
##  [1] "strikeout"                    NA
##  [3] "field_out"                    "force_out"
##  [5] "grounded_into_double_play"    "single"
##  [7] "double"                       "field_error"
##  [9] "home_run"                     "walk"
## [11] "hit_by_pitch"                 "sac_fly"
## [13] "fielders_choice"              "sac_fly_double_play"
## [15] "fielders_choice_out"          "caught_stealing_2b"
## [17] "strikeout_double_play"        "triple"
## [19] "double_play"                  "sac_bunt"
## [21] "wild_pitch"                   "pickoff_caught_stealing_home"
## [23] "game_advisory"                "pickoff_2b"
## [25] "other_out"                    "caught_stealing_3b"
## [27] "catcher_interf"               "triple_play"
## [29] "pickoff_1b"                   "caught_stealing_home"
## [31] "pickoff_caught_stealing_3b"   "pickoff_caught_stealing_2b"
## [33] "stolen_base_2b"               "pickoff_3b"
## [35] "sac_bunt_double_play"         "passed_ball"
## [37] "runner_double_play"           "stolen_base_home"
```

The overall QAB% for all 30 teams in this timespan was 62.35%. Here are the QAB% for each team.

The information for total innings batted (as apposed to innings pitched) can be found here: https://www.teamrankings.com/mlb/stat/opponent-outs-pitched-per-game?date=2021-11-03 It is needed to calculate runs scored per inning. Regressing QAB% onto a per inning stat is more than fair given each team bats a different amount of innings every season.

```
## Rows: 120 Columns: 5
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (1): team
## dbl (4): yearID, N, QABpct, Outs_G
##
## i Use ‘spec()‘ to retrieve the full column specification for this data.
## i Specify the column types or set ‘show_col_types = FALSE‘ to quiet this message.
```

```
##
## Call:
## lm(formula = runs_per_inn ~ QABpct, data = teamsDF)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.104302 -0.030534  0.000127  0.033060  0.119913
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -0.4404     0.1429  -3.082  0.00256 **
## QABpct        1.5397     0.2292   6.717 6.88e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.04771 on 118 degrees of freedom
## Multiple R-squared:  0.2766, Adjusted R-squared:  0.2705
## F-statistic: 45.12 on 1 and 118 DF,  p-value: 6.883e-10
```



This model is barely an adequate fit for the data, for it only explains 27.66% of the variation of runs scored despite a highly significant predictor of runs scored. This is not an invitation to only look at a player's QAB% to determine their offensive value. Not all quality at bats are created equal. A home run is a quality at bat, while a strikeout on a 10 pitch at bat is also a quality at bat. One is guaranteed to score at least a run. The other is only good for forcing the other team to bring in another pitcher, so that *potentially a run can be scored.

With this, QAB will be broken down into its appropriate components and will all be regressd onto runs scored to see which values statistically matter and which do not. .OPS will be used to measure the on base portions of QAB, and the rest are listed here:

- .OPS
- Productive Out % (outs that move a base runner/opportunities for an out to move the base runner (does not include batter reaching base))
- Hard Hit % (95+ mph)
- Long AB % (8+ pitches)

```
## The following objects are masked from teamsDF:
##
##     AB, attendance, BB, BBA, BPF, CG, CS, divID, DivWin, DP, E, ER,
##     ERA, FP, G, Ghome, H, HA, HBP, HR, HRA, innings_batted, IPouts, L,
##     lgID, LgWin, name, Outs_G, park, PPF, QABpct, R, RA, Rank,
##     runs_per_inn, SB, SF, SHO, SO, SOA, SV, team, team_yearID, teamID,
##     teamIDBR, teamIDlahman45, teamIDretro, W, WCWin, WSWin, X2B, X3B,
##     yearID
```

```
##
## Call:
## lm(formula = runs_per_inn ~ OPS + productive + hardhitpct + longabpct)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.036813 -0.012731 -0.001391  0.009658  0.057159
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.63549    0.04108 -15.468  < 2e-16 ***
## OPS          1.35141    0.04445  30.406  < 2e-16 ***
## productive   0.14919    0.05763   2.589   0.0109 *
## hardhitpct   0.27800    0.05970   4.657 8.69e-06 ***
## longabpct    0.34877    0.36907   0.945   0.3466
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.01698 on 115 degrees of freedom
## Multiple R-squared:  0.9107, Adjusted R-squared:  0.9076
## F-statistic: 293.2 on 4 and 115 DF,  p-value: < 2.2e-16
```

This model fits much better in the original model of boxed-up QAB% that explains 27.66% of the variance. This model explains 91.07% of the variance of run scoring. The only aspect of a QAB that does not contribute significantly to scoring runs is long at bat % so it should be dropped from the model and it's best to continue with the best fitted model with the available variables.

```
## The following objects are masked from context (pos = 3):
##
##     AB, attendance, BB, BBA, BPF, CG, CS, divID, DivWin, DP, E, ER,
##     ERA, FP, G, Ghome, H, HA, hardhitpct, HBP, HR, HRA, innings_batted,
##     IPouts, L, lgID, LgWin, longabpct, N, N.x, N.y, name, OPS, Outs_G,
##     park, PPF, productive, QABpct, R, RA, Rank, runs_per_inn, SB, SF,
##     SHO, SO, SOA, SV, team, team_yearID, teamID, teamIDBR,
##     teamIDlahman45, teamIDretro, W, WCWin, WSWin, X2B, X3B, yearID
## The following objects are masked from teamsDF:
##
##     AB, attendance, BB, BBA, BPF, CG, CS, divID, DivWin, DP, E, ER,
##     ERA, FP, G, Ghome, H, HA, HBP, HR, HRA, innings_batted, IPouts, L,
##     lgID, LgWin, name, Outs_G, park, PPF, QABpct, R, RA, Rank,
##     runs_per_inn, SB, SF, SHO, SO, SOA, SV, team, team_yearID, teamID,
##     teamIDBR, teamIDlahman45, teamIDretro, W, WCWin, WSWin, X2B, X3B,
##     yearID
```

```
##
## Call:
## lm(formula = runs_per_inn ~ OPS + productive + hardhitpct)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.036743 -0.013220 -0.000665  0.010201  0.055147
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.62089    0.03805 -16.317  < 2e-16 ***
## OPS          1.35257    0.04441  30.458  < 2e-16 ***
## productive   0.14643    0.05753   2.545   0.0122 *
## hardhitpct   0.27841    0.05967   4.666 8.31e-06 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 0.01697 on 116 degrees of freedom
## Multiple R-squared:   0.91,  Adjusted R-squared:  0.9077
## F-statistic:   391 on 3 and 116 DF,  p-value: < 2.2e-16
```

In conclusion, the ability to score runs is not just dependent on the .OPS of your players but also significantly correlates to their ability to get an out productively when given the chance and consistently get velocity off the bat. The ability to work counts is the only aspect of QAB% proven ineffective for run scoring. However, as I worked through this project and learned more about the statcast data, it's become more clear to me that QAB% is not a necessary statistic in and of itself. It's a great mental tool for players to use at any level to keep them task focused and cool headed. If a player is in a slump, maybe he might want to turn his outs into quality outs to get himself back into a groove (perhaps basis for a future project).

What statcast provides, especially with the delta run expectancy (difference of run expectancy before and after a pitch) tells a more honest tale of what teams need to do to score runs. I've enjoyed this project, and it has set inspiration for ideas to test in the future. Thank you for reading.