



Research Article

Ryan S. Brill* and Abraham J. Wyner

Introducing Grid WAR: rethinking WAR for starting pitchers

<https://doi.org/10.1515/jqas-2023-0095>

Received October 21, 2023; accepted May 8, 2024;
published online June 4, 2024

Abstract: The baseball statistic “Wins Above Replacement” (WAR) has emerged as one of the most popular evaluation metrics. But it is not readily observed and tabulated; WAR is an estimate of a parameter in a vaguely defined model with all its attendant assumptions. Industry-standard models of WAR for starting pitchers from FanGraphs and Baseball Reference all assume that season-long averages are sufficient statistics for a pitcher’s performance. This provides an invalid mathematical foundation for many reasons, especially because WAR should not be linear with respect to any counting statistic. To repair this defect, as well as many others, we devise a new measure, Grid WAR, which accurately estimates a starting pitcher’s WAR on a per-game basis. The convexity of Grid WAR diminishes the impact of “blow-up” games and up-weights exceptional games, raising the valuation of pitchers like Sandy Koufax, Whitey Ford, and Catfish Hunter who exhibit fundamental game-by-game variance. Although Grid WAR is designed to accurately measure historical performance, it has predictive value insofar as a pitcher’s Grid WAR is better than Fangraphs’ FIP WAR at predicting future performance. Finally, at <https://gridwar.xyz> we host a Shiny app which displays the Grid WAR results of each MLB game since 1952, including career, season, and game level results, which updates automatically every morning.

Keywords: mathematical modeling; baseball; pitching; wins above replacement

1 Introduction

1.1 Why calculate WAR?

Player valuation is one of the fundamental goals of sports analytics. In team sports, the goal in particular is to quantify the contributions of individual players towards the collective performance of their team. In baseball player roles are clearly defined and events are discrete (see Appendix A for a review of the rules of baseball). This has led to the development of separate player valuation measures for each aspect of the game: hitting, pitching, fielding, and baserunning (Baumer et al. 2015). Historically fundamental measures for evaluating hitters include batting average (BA), on-base percentage (OBP), and slugging percentage (SLG). Classical measures of pitching include earned run average (ERA) and walks and hits per inning pitched (WHIP). For a more thorough review of other player performance measures in baseball, we refer to the reader to Thorn and Palmer (1984), Lewis (2003), Albert and Bennett (2003), Schwarz and Gammons (2005), Tango et al. (2007), Baumer and Zimbalist (2014), and Baumer et al. (2015).

The benefit of separately measuring different aspects of baseball is that it isolates different aspects of player ability. The drawback is that it doesn’t provide a comprehensive measure of overall player performance. This makes it difficult to compare the value of players across positions. For instance, is a starting pitcher with a 2.50 ERA more valuable than a shortstop with a 2.80 batting average? As the fundamental result of a baseball game is a win or loss, the ideal unified measure of a player’s performance is his contribution to the number of games his team wins. On this view, an analyst’s task is to apportion this total number of wins to each player.

To this end, *wins above replacement (WAR)* estimates an individual baseball player’s contribution on the scale of team wins relative to a replacement level player. Win contribution is not estimated relative to a league average player because average players themselves are valuable. A better baseline is a *replacement level* player who is freely available to add to your team in the absence of the player being evaluated (e.g., a minor league free agent). With this

*Corresponding author: Ryan S. Brill, Graduate Group in Applied Mathematics and Computational Science, University of Pennsylvania, Philadelphia, PA, USA, E-mail: ryguy123@sas.upenn.edu. <https://orcid.org/0000-0001-6387-7713>

Abraham J. Wyner, Department of Statistics and Data Science, University of Pennsylvania, Philadelphia, USA, E-mail: ajw@wharton.upenn.edu

baseline, Sandy Koufax having 11.5 wins above replacement in 1966 means that losing Koufax to injury is associated, on average, with his team dropping about 11.5 games in the standings over an entire season.

1.2 Standard WAR calculations

Traditional measures of player performance like ERA and others listed previously are counting statistics. In other words, they are statistics which tabulate what factually happened during a season. For instance, in 1966 Sandy Koufax had 62 earned runs in 323 innings, resulting in a 1.73 ERA. WAR, on the other hand, is not a readily observed counting statistic. In other words, how many of the 1966 Dodgers' 95 wins were due to Koufax's contributions is not a known or observable quantity. Rather, WAR must be *estimated* from historical data.

Estimating a player's wins above replacement is an inherently difficult task because the ultimate win/loss result of a game is the culmination of a complex interaction of event outcomes involving players of varying roles. Accordingly, there have been many different attempts to estimate a baseball player's WAR. The fundamental idea behind each of these estimation procedures is to capture the contribution of a player's observed performance isolated from the factors outside of his control. The way in which we measure observed performance is a crucial component of estimating his contribution to winning games.

To estimate WAR, a baseball analyst first chooses a base metric of performance and then maps this base metric to wins. Different choices of base metric yield substantially different estimates for WAR. For instance, FanGraphs builds separate WAR values for pitchers from two counting stats: fielding-independent pitching (FIP) and average runs allowed per nine innings (RA/9). FIP is a weighted average of a pitcher's isolated pitching metrics¹ (e.g., home runs, walks, and strikeouts) (Slowinski 2012). Baseball Reference builds WAR for pitchers from expected runs allowed (xRA), which assigns to each potential at-bat outcome² an expected run value (Baseball Reference 2011). For instance, the expected runs allowed off a single reflects the average number of runs that follow a single in a half-inning.

¹ FanGraphs builds WAR from ifFIP (FIP with infield flies),

$$\text{if FIP} := \frac{13 \cdot \text{HR} + 3 \cdot (\text{BB} + \text{HBP}) - 2 \cdot (\text{K} + \text{IFFB})}{\text{IP}}$$

+ if FIP constant,

which is built from home runs (HR), walks (BB), hit by pitches (HBP), strikeouts (K), infield fly balls (IFFB), and innings pitched (IP).

² The seven potential outcomes of an at-bat are out, walk, hit-by-pitch, single, double, triple, and home run.

The difference between Runs Allowed, FIP, and xRA, and hence the associated WAR values for players, can be substantial. FIP, unlike runs allowed and xRA, ignores at-bat outcomes involving fielders in order to fully disentangle pitching performance from fielding. Specifically, FIP excludes balls-in-play (singles, doubles, triples, ground outs, fly outs, etc.) because fielders' actions play a role in the outcome of these plays. Furthermore, FIP and xRA, unlike runs allowed, do not depend on the sequencing of events in an inning. For example, consider an inning where a pitcher strikes out three, while allowing a home run, two walks, and a single. Depending on the sequence of the events, the pitcher could be charged with one to four runs. The pitcher's FIP and xRA for this inning, however, are the same regardless of the sequence. For a more thorough review of the differences in their methodologies, we refer the reader to the supplementary materials of Baumer et al. (2015).

After choosing a base measure of player performance, a baseball analyst then decides how to aggregate player performance over the course of a season. Current implementations of WAR from FanGraphs and Baseball Reference average pitcher performance over the entire season (e.g., RA per nine innings, FIP per inning, and xRA per out).

1.3 Problems with standard WAR calculations for starting pitchers

For starting pitchers in particular, there are two problems with standard WAR calculations. First, a base measure of pitcher performance should account for sequencing context; FIP and xRA do not. Second, pitcher performance should not be a simple average across innings and games.

Problem 1: excluding sequencing context. Standard WAR calculations are functions of pitcher performance, traditionally one of FIP, xRA, or runs allowed. If the pitcher cannot affect the sequence of events in an inning, then it makes sense to measure his performance using FIP or xRA. If sequencing variation were due to chance alone, it would not be controlled by the pitcher and should reasonably be excluded. For starting pitchers, however, sequencing variability has other causes. Blake Snell in 2023 is a prime example: he has the highest walk rate since 2000³ yet has the best ERA in baseball (2.33) and an exceptional 1.2 ERA through the last 23 games at the time of this writing. High-strikeout pitchers like Snell can give up walks strategically without giving up runs; he can increase his effort to strike out a batter when necessary. All this indicates

³ At this time of this writing, Snell has about five walks per nine innings and a walk rate of 13.4 %, the 14th highest walk rate since 1912.

that he exerts some control over the sequencing of events.⁴ Thus, we believe starting pitchers bear responsibility for their sequence of outcomes and that their performance in situations of varying leverage should be taken into account. For batters, there is enough evidence of this to generate a substantial debate – see for example, Bill James’ comparison of Altuve and Judge⁵ – but for pitchers the argument against it is nonsense. Thus, we should not use FIP or xRA, which remove sequencing context, as a base measure of pitcher performance to estimate a starting pitcher’s WAR. Rather, in this paper we use runs allowed.⁶

Problem 2: averaging across games. After selecting a base measure of pitcher performance, standard WAR calculations from FanGraphs and Baseball Reference then average pitcher performance over the entire season (e.g., RA per nine innings, FIP per inning, and xRA per out). If the pitcher doesn’t affect his variability across games, then it makes sense to average his performance across games. If variation across games were due to chance alone, it would not be controlled by the pitcher and should reasonably be excluded. For starting pitchers, however, we believe that game-by-game variability has causes other than chance. Sandy Koufax in 1966 is a prime example: he had an astounding 17 complete games in which he allowed at most one run⁷ yet had three terrible “blow-up” games. The variability of pitchers like Koufax appears in noticeable patterns, indicating that his non-stationarity across games is a fundamental trait. Our method will account for the possibility that the version of Sandy Koufax that starts a game and gets tagged for six runs in two innings is fundamentally different from the Koufax who strikes out the first six batters (colloquially known as the “the left arm of God”).

Since a starting pitcher’s game-by-game variability is not entirely due to chance, averaging pitcher performance across games is *wrong*, specifically because *not all runs have the same value* within each game. To understand this, think of a starting pitcher’s WAR in a single game as a function $R \mapsto \text{WAR}(R)$ where R is the number of runs allowed in that game.⁸ We expect WAR to be a decreasing function of R

⁴ Great pitchers are much better in high leverage situations, where the game is on the line. See for example <https://www.baseball-reference.com/players/split.fcgi?id=koufasa01&year=1966&t=p>.

⁵ https://www.billjamesonline.com/judge_and_altuve/.

⁶ Critics of runs allowed as a base measure of pitcher performance argue it is confounded with the fielding, but we argue in Section 4 that the impact of fielding on WAR is small (smaller than ballpark, which itself is small).

⁷ In 1996 Koufax had eight complete game shutouts and nine one-run complete games.

⁸ As we will discuss in Section 2, think of game WAR as measuring a context-neutral version of win probability added derived only from

because allowing more runs in a game should correspond to fewer wins above replacement. Additionally, we expect WAR to be a *convex* function in R (i.e. its second derivative is positive.) As R increases, we expect the relative impact of allowing an extra run, given by $\text{WAR}(R+1) - \text{WAR}(R)$, to decrease. For instance, allowing two runs instead of one should have a much steeper drop off in WAR than allowing eight runs instead of seven.⁹ Therefore, by Jensen’s inequality,

$$\text{WAR}(\mathbb{E}[R]) \leq \mathbb{E}[\text{WAR}(R)]. \quad (1)$$

Traditional methods for computing WAR are reminiscent of the left side of Equation (1): first average a pitcher’s performance, then compute his WAR from the resulting average scaled by the number of innings pitched. Averaging weighs each run allowed equally, causing a pitcher’s “blow-up” games to unfairly dilute the value of high quality performances. Because winning a baseball game is defined by the runs allowed during a game, WAR should look like the right side of Equation (1) – compute the WAR of each of a pitcher’s individual games, and then aggregate. Crucially, these quantities are not the same.

For concreteness, consider Max Scherzer’s six game stretch from June 12, 2014 through the 2014 All Star game, shown in Table 1 (ESPN 2014). We re-arrange the order of these games to aid our explanation. He was excellent in games one through five and had a 1.2 ERA (five runs in 37 innings). This corresponds to about 2 WAR according to standard metrics (the Detroit Tigers did in fact win all 5). In game six, Scherzer was rocked for ten runs in four innings, exiting in the fifth inning with runners on second and third and no outs. His ERA over the six game stretch ballooned to 3.3 (15 runs allowed in 41 innings), reducing his total WAR to about 1/2 according to standard metrics. This is a complete absurdity: because a game can’t be lost more than once, accumulated “real” WAR cannot drop from 2 to

Table 1: Max Scherzer’s performance over six games prior to the 2014 all star break, consisting of five excellent games and one bad “blow-up” game.

Game	1	2	3	4	5	6	total
Earned runs	0	1	2	1	1	10	15
Innings pitched	9	6	7	8	7	4	41

a pitcher’s performance, invariant to factors outside of his control (e.g., his team’s batting). This is very different from the usual win-probability-added calculation, which is completely dependent on the starting pitcher’s team’s offense.

⁹ Because you can only lose a game once.

1/2 with the addition of one game! The correct assessment would charge Scherzer with the maximum possible damage, about -0.40 wins. So Scherzer's "real" WAR over the six games should be about 1.5, which is three times higher than the standard calculation. By evaluating Scherzer's performances using only the average, standard WAR significantly devalues his contributions during this six game stretch. The correct approach would be to calculate WAR per game and sum them up.

Here is another revealing albeit hypothetical example. Suppose a pitcher tosses two nearly flawless eight inning starts, allowing one run in each start, followed by a terrible two-inning blow-up where he gives up eight runs. His averaged performance over the three games is a thoroughly mediocre five runs per nine innings, which translates to a WAR of about 0.0 when calculated using standard metrics. In contrast, it is clear that over the three starts his team will win, with near certainty, two out of three, which translates to a "real" WAR of about 0.8 in total. Our hypothetical pitcher, who is great in two out of three starts and terrible in every third, would accumulate more than eight WAR over a full season, constituting an all-time great season worthy of a Cy Young award.¹⁰ In contrast, standard WAR metrics would suggest he be designated for assignment. What drives the difference? A poor performance can greatly affect the average, effectively allowing a single game to be "lost" more than once. Specifically, standard metrics allow the one blow-up game to count for two losses, resulting in 0 WAR. The example is somewhat extreme, but not that rare.¹¹

1.4 Paper organization

These examples illustrate that calculating WAR as a function of pitcher performance averaged across games is wrong. Hence in Section 2 we devise Grid WAR, which estimates a starting pitcher's WAR in each of his games. Grid WAR estimates the completely context-neutral win probability added above replacement at the point when a pitcher exits the game. Then in Section 3 we discuss our results. We find that averaging pitcher performance across games tends to, in general, undervalue mediocre and highly variable pitchers. This is because the convexity of GWAR diminishes

the impact of games in which a pitcher allows many runs, and these pitchers have more of those games. We also find that Grid WAR has predictive value: past Grid WAR is more predictive than past FanGraphs WAR of future Grid WAR. This suggests that some pitchers' game-by-game variance in performance is not just "bad luck" but a measurable characteristic. Finally, in Section 4 we conclude with a discussion. Notably, we compare the careers of starting pitchers via Grid WAR across baseball history. Although Grid WAR values many pitchers' careers similarly as other metrics, it substantially changes our view of some pitchers who exhibit fundamental game-by-game variance. Sandy Koufax is a prime example: although his 1966 season is just the 20th best season of all time according to FanGraphs WAR, it is the best season of all time according to Grid WAR. Other methods incorrectly overweight his three outlying blow-up games.

2 Defining Grid WAR for starting pitchers

Our task is to estimate a starting pitcher's WAR for an individual game, which we call Grid WAR (GWAR). The idea is to estimate a context-neutral version of win probability added derived only from a pitcher's performance, invariant to factors outside of his control such as his team's batting.¹² This is very different from the usual win-probability-added calculation, which is completely dependent on the starting pitcher's team's offense. In Section 2.1 we detail our mathematical formulation of Grid WAR. We give a brief overview of our data in Section 2.2, and then we discuss how we estimate the grid functions f and g (Sections 2.3 and 2.4), the constant w_{rep} (Section 2.5), and the park effects α (Section 2.6) which allow us to compute a starting pitcher's Grid WAR for a baseball game.

2.1 Grid WAR formulation

In baseball, each team's starting pitcher is the first pitcher in the game for that team. A starter usually exits midway through a game according to the discretion of the field manager (the equivalent of a head coach in baseball).

¹⁰ Just two pitchers since 2016 have eclipsed eight Grid WAR.

¹¹ Yankees' starting pitcher Domingo Germán is a perfect real life example. In two June 2023, he threw a perfect game one game after being rocked for ten runs allowed in three innings. Across an entire season, a pitcher like Germán who alternates having great and terrible games would accumulate more than 3 real WAR and -4.5 standard WAR.

¹² Note that computing a player's context neutral value added in each individual game was first introduced as "Support Neutral Win Loss" (Wolverton 1993, 1999, 2004). In our study, we use different methods, explore the statistical aspects of Grid WAR in much greater depth, and conduct a much more extensive comparison of Grid WAR to other WAR metrics which average pitcher performance over the entire season (specifically, FanGraphs WAR).

Generally a starter pitches for a significant portion of the game and sometimes pitches the entire game (known as a complete game). Often times, a starter is removed between innings.¹³ We first define a starter's Grid WAR for a game in which he exits at the end of an inning. To do so, we define the function $f = f(I, R)$ which, assuming both teams have league-average offenses, computes the probability a team wins a game after giving up R runs through I innings (the values of $f(I, R)$ for integer values of I and R can be displayed in a simple grid). In short, f is a context-neutral version of win probability, as it depends only on the starter's performance.

Note that f also depends on the league (AL vs. NL), season, and ballpark. For example, games in which the home team is in the National League (NL) prior to 2022 did not feature designated hitters, whereas American League (AL) games did, leading to different run environments. Additionally, baseballs have different compositions across seasons, leading to different proportions of home runs and base hits, and hence different run environments. Finally, it is easier to score runs at some ballparks than at others. For instance, Coors Field at high altitude in Denver features many more home runs than other parks. Consequently, $f = f(I, R)$ is implicitly a function of league, season, and ballpark.

To compute a wins *above replacement* metric, we need to compare this context-neutral win-contribution to that of a potential replacement-level pitcher. A replacement-level player is freely available to add to your team in the absence of the player being evaluated (e.g., a minor league free agent). We use a constant w_{rep} which denotes the probability a team wins a game with a replacement-level starting pitcher, assuming both teams have league-average offenses. We expect $w_{\text{rep}} < 0.5$ since replacement-level pitchers are worse than league-average pitchers. Then, we define a starter's Grid WAR during a game in which he gives up R runs through I complete innings as

$$f(I, R) - w_{\text{rep}}. \quad (2)$$

We call our metric Grid WAR because the function $f = f(I, R)$ is defined on the 2D grid $\{1, \dots, 9\} \times \{0, \dots, R_{\max} = 10\}$.

Next, we define a starting pitcher's Grid WAR for a game in which he exits midway through an inning. In this case, a starter exits an inning having thrown a certain number

¹³ Recall that in a baseball game, the two teams switch back and forth between batting and fielding; the batting team's turn to bat is over once the fielding team records three outs. One turn batting for each team constitutes an inning. A game is usually composed of nine innings, and the team with the greater number of runs at the end of the game wins (Wikipedia 2023).

of outs $O \in \{0, 1, 2\}$ and having potentially left baserunners on first, second, or third base, encoded by the base-state

$$S \in \{000, 100, 010, 001, 110, 101, 011, 111\}.$$

We define an auxiliary function $g = g(r|S, O)$ which, assuming both teams have league-average offenses, computes the probability that, starting midway through an inning with O outs and base-state S , a team scores exactly r runs through the end of the inning. Given g , we define a starter's Grid WAR during a game in which he gives up R runs and leaves midway through inning I with O outs and base-state S as the expected Grid WAR at the end of the inning,

$$\sum_{r \geq 0} g(r|S, O) f(I, r + R) - w_{\text{rep}}. \quad (3)$$

Finally, we define a starting pitcher's Grid WAR for an entire season as the sum of the Grid WAR of his individual games.

2.2 Our data

In the remainder of Section 2, we discuss how we estimate the grid functions f and g , the constant w_{rep} , and the park effects α which allow us to compute a starting pitcher's Grid WAR for a baseball game (Equations (2) and (3)). In our analysis we use play-by-play data from Retrosheet. We scraped every plate appearance from 1990 to 2020 from the Retrosheet database. For each plate appearance, we record the pitcher, batter, home team, away team, league, park, inning, runs allowed, base state, and outs count. Our final dataset is publicly available online.¹⁴ In our study, we restrict our analysis to every plate appearance from 2010 to 2019 featuring a starting pitcher. Additionally, we scraped FanGraphs RA/9 WAR (abbreviated henceforth as FWAR(RA/9)) and FanGraphs FIP WAR (abbreviated henceforth as FWAR(FIP)) using the baseballr package (Petti and Gilani 2021). All computations in our analysis are performed in R, and our code is publicly available online.¹⁵

2.3 Estimating the grid function f

Now, we estimate the grid function $f = f(I, R)$ which, assuming both teams have average offenses,¹⁶ computes the probability a team wins a game after giving up R runs

¹⁴ <https://upenn.app.box.com/v/retrosheet-pa-1990-2000>.

¹⁵ https://github.com/snoopyryan123/grid_war.

¹⁶ Technically, we assume both teams have offenses that are *randomly drawn* from our dataset, rather than league-average offenses, since we don't explicitly adjust for offensive quality.

through I complete innings. We call f a grid because the values of $f(I, R)$ for integer values of I and R can be displayed in a simple 2D grid. To account for different run environments across different seasons, leagues (NL vs. AL), and ballparks, it is imperative to estimate a different grid for each league-season-ballpark. Thus we estimate f using a parametric mathematical model rather than a statistical or machine learning model fit from historical data. Here, we give an overview as to why; see Appendix B for a more detailed discussion.

The naive solution to estimating f is the empirical grid: across all combinations of I and R , simply take the observed proportion of times a starter's team won the game. Due to a lack of data, the empirical grid fit on all half-innings from any given league-season massively overfits. In particular, it fails to be monotonic in I or R even though it should be.¹⁷ Thus any ballpark-adjusted empirical grid would also massively overfit. To force the grid to be monotonic, we try XGBoost with monotonic constraints. While the fitted f is monotonic and also convex in R as expected, it still overfits, especially towards the tails (e.g., for R large). Refitting the grid using a statistical model for each league-season is clearly not optimal.

As there is not enough data to use machine learning to fit a separate grid for each league-season without overfitting, we turn to a parametric mathematical model. Specifically, we use an Empirical Bayes Poisson model from which we explicitly compute context-neutral win probability. We find that our Poisson model is a powerful approximation technique, yielding reasonable grids for our application which vary across each league, season, and ballpark without overfitting. Indeed, by distilling the information of our dataset into just a few parameters, our Poisson model creates a strong model from limited data.

Because the runs allowed in a half-inning is a natural number, we begin our parametric modeling process by supposing that the runs allowed in a half-inning is a Poisson random variable. In particular, denoting the runs scored by the pitcher's team's batters in inning i by X_i and the runs scored by the opposing team in inning i by Y_i (for innings after the pitcher exits the game), we model

$$X_i \stackrel{\text{i.i.d.}}{\sim} \text{Poisson}(\lambda_X) \quad \text{and} \quad Y_i \stackrel{\text{i.i.d.}}{\sim} \text{Poisson}(\lambda_Y). \quad (4)$$

The two teams have their own runs per inning parameters λ_X and λ_Y because a baseball season involves teams of varying strength playing against each other. The assumption

that runs scored in an inning is independent across innings given a team's strength, while technically false due to non-stationarity across innings,¹⁸ is justified by *working*. In particular, it yields a grid which looks like a smoothed, non-overfit version of the empirical and XGBoost grids, and so we deem it a reasonable enough assumption. In other words, we view the independence and Poisson assumptions as tools for creating flexible, non-overfit grids which vary across league, season, and ballpark.

Given the team strength parameters, the probability that a pitcher wins the game after allowing R runs through I innings, assuming win probability in overtime is $1/2$, is

$$\begin{aligned} f(I, R | \lambda_X, \lambda_Y) := & \mathbb{P}\left(\sum_{i=1}^9 X_i > R + \sum_{i=I+1}^9 Y_i\right) \\ & + \frac{1}{2} \cdot \mathbb{P}\left(\sum_{i=1}^9 X_i = R + \sum_{i=I+1}^9 Y_i\right). \end{aligned} \quad (5)$$

If $I = 9$, this is equal to

$$\mathbb{P}(\text{Poisson}(9\lambda_X) > R) + \frac{1}{2} \cdot \mathbb{P}(\text{Poisson}(9\lambda_X) = R). \quad (6)$$

If $I < 9$, it is equal to

$$\begin{aligned} & \mathbb{P}(\text{Skellam}(9\lambda_X, (9 - I - 1)\lambda_Y) > R) \\ & + \frac{1}{2} \cdot \mathbb{P}(\text{Skellam}(9\lambda_X, (9 - I - 1)\lambda_Y) = R), \end{aligned} \quad (7)$$

noting that the Skellam distribution arises as a difference of two independent Poisson random variables. To capture variability in team strength across each of the 30 MLB teams, we impose a positive normal prior,

$$\lambda_X, \lambda_Y \sim \mathcal{N}_+(\lambda, k \cdot \sigma_\lambda^2). \quad (8)$$

We estimate the prior hyperparameters λ and σ_λ^2 separately for each league-season by computing each team's mean and variance of the runs allowed in each half inning, respectively, and then averaging over all teams. The initial estimated values of σ_λ^2 are too large (e.g., the prior is overdispersed), so we include a tuning parameter k designed to tune the dispersion across team strengths to match observed data. In particular, we use $k = 0.28$, which minimizes the log-loss between the observed win/loss column and predictions from the induced grid. In particular,

¹⁸ For instance, f should be monotonic decreasing in R because as a pitcher allows more runs through a fixed number of innings, his team is less likely to win the game.

the induced grid is given by the posterior mean grid, which we estimate using Monte Carlo integration with $B = 100$ samples,

$$f(I, R | \lambda, \sigma_\lambda^2, k) := \frac{1}{B} \sum_{b=1}^B f(I, R | \lambda_X^{(b)}, \lambda_Y^{(b)}), \quad (9)$$

where $\lambda_X^{(b)}$ and $\lambda_Y^{(b)}$ are i.i.d. samples from the prior distribution (8).

Additionally, recall that $f = f(I, R)$ is implicitly a function of ballpark. To adjust for ballpark, we first define the *park effect* α of a ballpark as the expected runs allowed in one half-inning at that park above that of an average park, if an average offense faces an average defense. With our parametric model, the ballpark adjustment is easy: we simply incorporate the park effect into our Poisson parameters. With a statistical model, the ballpark adjustment is more difficult and prone to overfitting, providing yet another justification of our mathematical model. As λ represents the mean runs allowed in a half-inning for a given league-season, $\lambda + \alpha$ represents the mean runs allowed in a half-inning at a given ballpark during that league-season. So, to adjust for ballpark, we use $\lambda + \alpha$ in place of λ in our Poisson model (9) and positive Normal prior (8). In Section 2.6 we estimate the park effects α .

In Figure 1 we visualize the estimated grid f according to our Poisson model (9), with prior (8), using the 2019 NL λ and σ_λ^2 , without a park adjustment. Note that the f grid for other league-seasons are similar, but differ slightly according to the differing run environments λ and σ_λ^2 . We see that f is monotonic decreasing in R because as a pitcher allows more runs through a fixed number of innings, his team is

less likely to win the game. Also, f is monotonic increasing in I because giving up R runs through I innings is worse than giving up R runs through $I + i$ innings for $i > 0$, since giving up R runs through $I + i$ innings implies a pitcher gave up no more than R runs through I innings. Further, f is convex in R for large values of R because the marginal impact of allowing an additional run diminishes to zero as R increases because, after giving up a certain number of runs, the game has essentially already been lost. Succinctly, “you can only lose a game once”. Finally, the grid f is smooth.

2.4 Estimating the grid function g

Now, we estimate the function $g = g(r|S, O)$ which, assuming both teams have league-average offenses, computes the probability that, starting midway through an inning with $O \in \{0, 1, 2\}$ outs and base-state

$$S \in \{000, 100, 010, 001, 110, 101, 011, 111\},$$

a team scores exactly r runs through the end of the inning. We estimate $g(r|S, O)$ using the empirical distribution. Specifically, we bin and average over the variables (r, S, O) , using data from every game from 2010 to 2019. Because g isn't significantly different across innings, we use data from each of the first eight innings. In Figure 2 we visualize $g(r|S, O = 0)$, with $O = 0$ outs, for each base-state S .

2.5 Estimating the constant w_{rep}

To estimate wins *above replacement*, we need to compare a starting pitcher's context-neutral win contribution to that of a potential replacement-level pitcher. Thus we

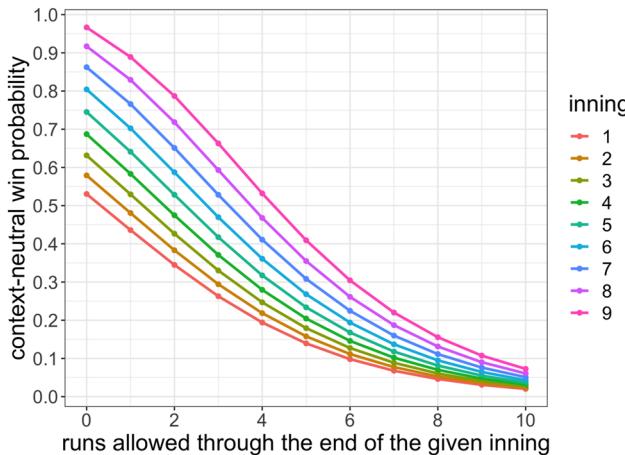


Figure 1: Context-neutral win probability (y-axis) if a starter allowed R runs (x-axis) through I complete innings (color) according to the 2019 National League grid function f , fit from our Poisson model (9) with positive normal prior (8).

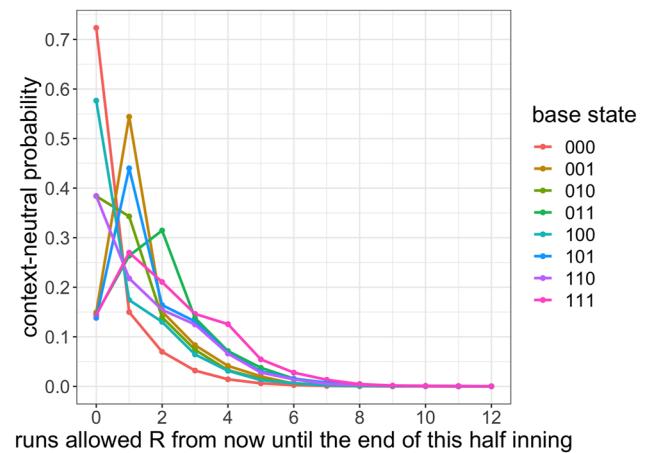


Figure 2: From base-state S (color) and $O = 0$ outs through the end of an inning, the context-neutral probability (y-axis) that the pitcher allows R runs (x-axis) according to the grid function g .

estimate a constant w_{rep} which represents the context-neutral probability a team wins a game with a replacement-level starting pitcher, assuming both teams have a league-average offense and league-average fielding. Fangraphs (2010) defines *replacement-level* as the “level of production you could get from a player that would cost you nothing but the league minimum salary to acquire.” We could estimate w_{rep} in a given season by averaging the context neutral win probability at the point when the starter exits the game across all games featuring a replacement-level starting pitcher. To facilitate fair comparison of Grid WAR to FanGraphs WAR, we instead estimate w_{rep} so as to match FanGraphs’ definition of replacement-level. In particular, we choose $w_{\text{rep}} = 0.428$ so that the sum of GWAR across all starting pitchers from 2010 to 2019 equals the sum of FWAR(RA/9).

2.6 Estimating the park effects α

Finally, we estimate the park effect α of each ballpark, which measures the expected runs scored in one half-inning at that park above that of an average park, if an average offense faces an average defense. To compute the park effects for 2019, we take all half-innings from 2017 to 2019 and fit a ridge regression, using cross validation to tune the ridge hyperparameter, where the outcome is runs scored during the half-inning and the covariates are fixed effects for each park, team-offensive-season, and team-defensive-season. We compute similar three-year park effects for

other seasons. We visualize the 2019 park effects in Figure 3. We use ridge regression, as opposed to ordinary least squares or existing park effects from ESPN, FanGraphs, or Baseball Reference, because, as detailed in Appendix D, it performs the best in two simulation studies and has the best out-of-sample predictive performance on observed data.

3 Results

After estimating the grid functions f and g , the constant w_{rep} , and the park effects α , we compute Grid WAR for each starting pitcher in each game from 2010 to 2019. As a quick exposition, we show the full 2019 Grid WAR rankings (Figure 4) and a full game-by-game breakdown of Justin Verlander’s 2019 season (Figure 5). The remainder of this section is organized as follows. In Section 3.1 we compare GWAR to FanGraphs WAR (FWAR) in order to understand the effect of averaging pitcher performance over the entire season on valuing pitchers. We find that averaging over the season allows a pitcher’s terrible performances to dilute the contributions of his great ones. This is because the convexity of GWAR diminishes the impact of games in which a pitcher allows many runs, whereas averaging across games doesn’t. Thus traditional measures of WAR have generally undervalued mediocre pitchers and higher variance pitchers who tend to have more of these terrible games. Then in Section 3.2 we quantify the value

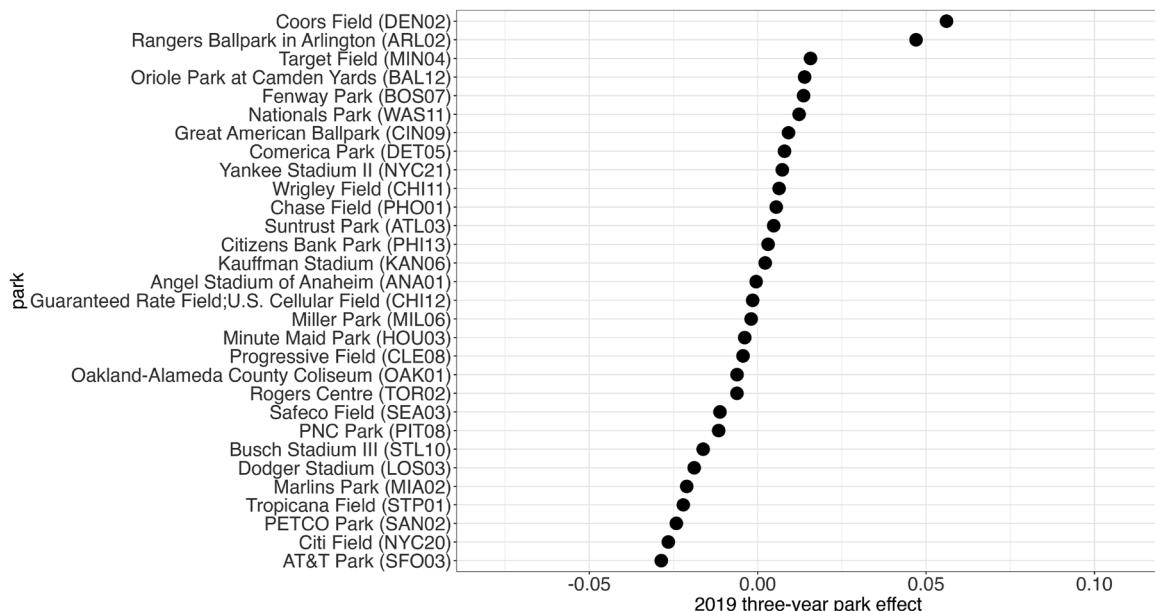


Figure 3: Our 2019 three-year park effects (x-axis), fit from half-inning data from 2017 to 2019, for each ballpark (y-axis). The abbreviations are retrosheet ballpark codes.

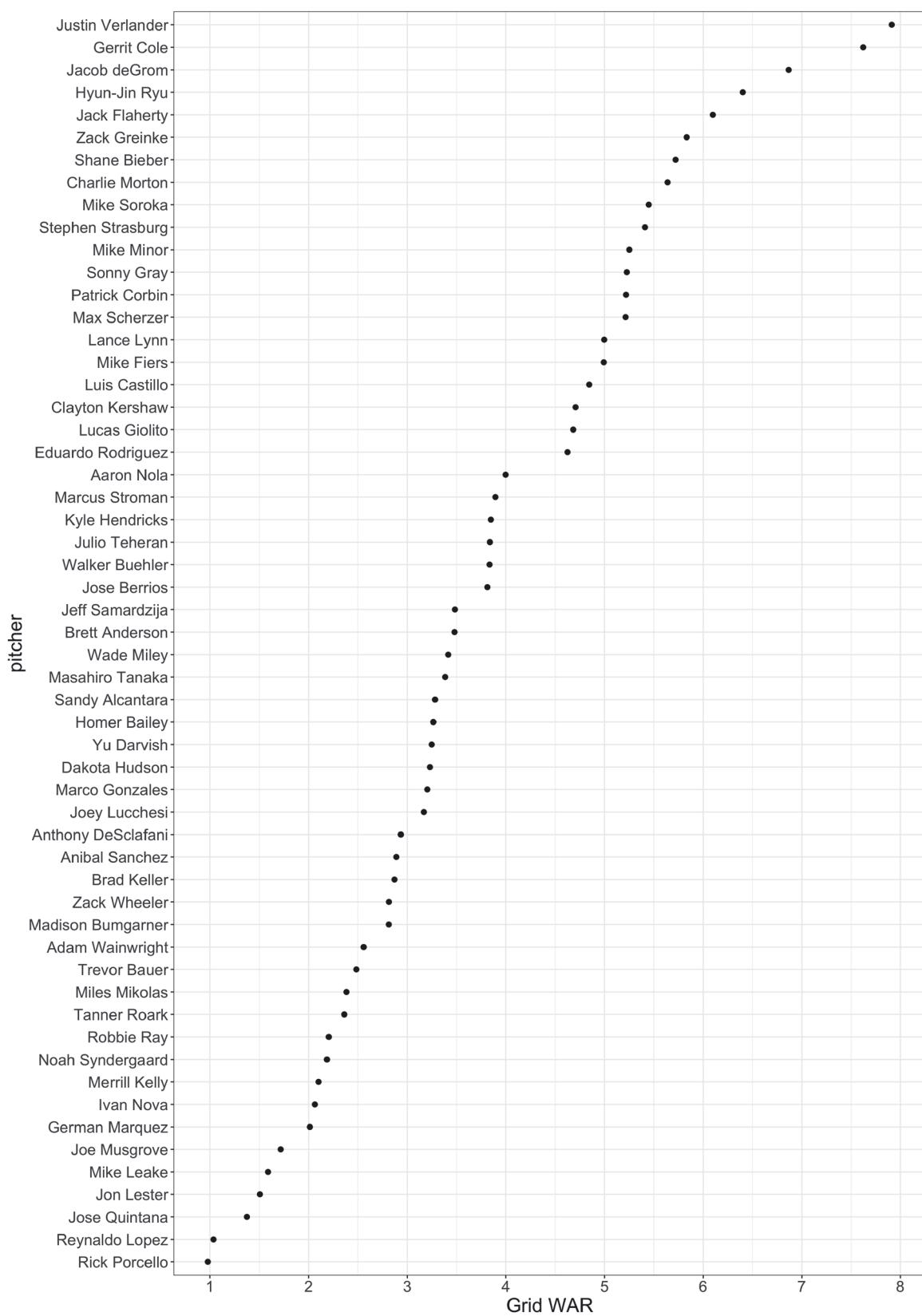


Figure 4: Ranking starting pitchers in 2019 by Grid WAR.

DATE	HOME TEAM	AWAY TEAM	PARK	RUNS ALLOWED	EXIT INNING	EXIT BASE STATE	EXIT OUTS	GWAR
2019-03-28	TBA	HOU	STP01	1	7			0.358
2019-04-02	TEX	HOU	ARL02	4	5	101	0	-0.207
2019-04-08	HOU	NYA	HOU03	3	6			0.074
2019-04-13	SEA	HOU	SEA03	1	6			0.302
2019-04-19	TEX	HOU	ARL02	1	7			0.370
2019-04-24	HOU	MIN	HOU03	1	8			0.430
2019-04-29	MIN	HOU	MIN04	1	6			0.317
2019-05-05	ANA	HOU	MNT01	4	7	000	1	-0.007
2019-05-10	HOU	TEX	HOU03	0	7			0.461
2019-05-15	DET	HOU	DET05	1	7			0.376
2019-05-21	HOU	CHA	HOU03	1	8			0.430
2019-05-26	HOU	BOS	HOU03	3	7			0.136
2019-06-01	OAK	HOU	OAK01	1	8			0.428
2019-06-06	SEA	HOU	SEA03	1	7	110	1	0.259
2019-06-12	HOU	MIL	HOU03	3	7			0.136
2019-06-18	CIN	HOU	CIN09	4	7			0.000
2019-06-23	NYA	HOU	NYC21	3	7			0.146
2019-06-29	HOU	SEA	HOU03	4	5			-0.089
2019-07-05	HOU	ANA	HOU03	4	7			0.016
2019-07-14	TEX	HOU	ARL02	2	6			0.203
2019-07-19	HOU	TEX	HOU03	2	6			0.192
2019-07-24	HOU	OAK	HOU03	1	6			0.306
2019-07-30	CLE	HOU	CLE08	0	7			0.461
2019-08-04	HOU	SEA	HOU03	1	7	000	0	0.312
2019-08-11	BAL	HOU	BAL12	4	5			-0.079
2019-08-16	OAK	HOU	OAK01	2	7			0.256
2019-08-21	HOU	DET	HOU03	2	9			0.389
2019-08-27	HOU	TBA	HOU03	0	6	010	1	0.333
2019-09-01	TOR	HOU	TOR02	0	9			0.550
2019-09-07	HOU	SEA	HOU03	1	7			0.369
2019-09-12	HOU	OAK	HOU03	3	6			0.074
2019-09-17	HOU	TEX	HOU03	0	6			0.407
2019-09-22	HOU	ANA	HOU03	2	5			0.129
2019-09-28	ANA	HOU	ANA01	3	6			0.076

Figure 5: A game-by-game breakdown of Justin Verlander's 2019 season.

lost by using FWAR to estimate pitcher quality rather than using GWAR. We find that pitcher rankings built from past GWAR are better than pitcher rankings built from past FWAR at predicting future pitcher rankings according to GWAR. This provides evidence that game-by-game variability in pitcher performance is a fundamental and measurable trait.

3.1 Averaging pitcher performance across games dilutes the contributions of his great games

In this section we compare GWAR to FanGraphs WAR (FWAR) in order to understand the effect of averaging pitcher performance over the entire season on valuing pitchers. We find that averaging over the season allows a pitcher's terrible performances to dilute the contributions of his great ones. Note that to compare the *relative* value of starting pitchers according to GWAR relative to FWAR(RA/9), in this section we rescale GWAR in each year so that the sum of GWAR across all starters in each season equals the corresponding sum in FWAR(RA/9).

We begin with Figure 6, which visualizes GWAR versus FWAR(RA/9) for starting pitchers in 2019. Pitchers who lie above the line $y = x$ are undervalued according to GWAR relative to FWAR and pitchers who lie below the line are overvalued. To understand why some players are undervalued and others are overvalued, we compare players who have similar FWAR but different GWAR values in 2019. In Figure 7 we compare Homer Bailey's 2019 season to Tanner Roark's. They have the same FWAR(RA/9), 2.7, but Bailey has a much higher GWAR (Bailey 3.43, Roark 2.48). Similarly, in Figure 8 we compare Mike Fiers's 2019 season to Aaron

Nola's. They have the same FWAR(RA/9), 4.1, but Fiers has a higher GWAR (Fiers 5.25, Nola 4.2).

In each of these comparisons, we see a similar trend explaining the differences in GWAR. The pitcher with higher GWAR allows fewer runs in more games and allows more runs in fewer games. This is depicted graphically in the “Difference” histograms, which show the difference between the histogram on the left and the histogram on the right. The green bars denote positive differences (i.e. the pitcher on the left has more games with a given number of runs allowed than the pitcher on the right) and the red bars denote negative differences (i.e. the pitcher on the left has fewer games with a given number of runs allowed than the pitcher on the right). In each of these examples, the green bars are shifted towards the left (pitchers with higher GWAR allow few runs in more games) and the red bars are shifted towards the right (pitchers with lower GWAR allow many runs in more games). In Figure 7, Bailey pitches four more games than Roark in which he allows two runs or fewer and Roark pitches four more games than Bailey in which he allows three runs or more. Similarly, in Figure 8, Fiers pitches four more games than Nola in which he allows zero runs and Nola pitches five more games than Fiers in which he allows one run or more. The convexity of GWAR diminishes the impact of games in which a pitcher allows many runs, explaining why Bailey has more GWAR than Roark and why Fiers has more GWAR than Nola.

Next, in Figure 9 we visualize GWAR versus FWAR(RA/9) for each starting pitcher-season across all years from 2010 to 2019. Pitchers who lie above the line $y = x$ (the solid black line) are undervalued according to GWAR relative to FWAR and pitchers who lie below the line are overvalued. Worse pitchers generally lie above the line

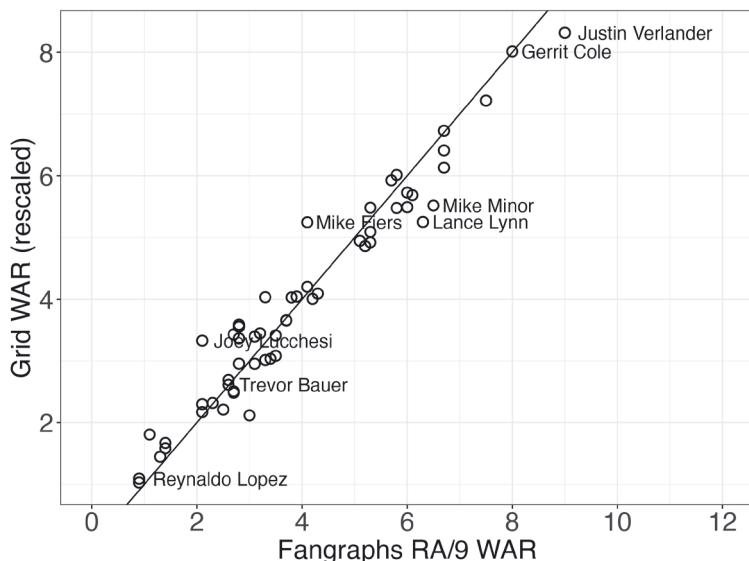


Figure 6: Grid WAR (y-axis) versus FanGraphs WAR (RA/9) (x-axis) for each pitcher-season in 2019. The pitcher name refers to the dot on its immediate left.

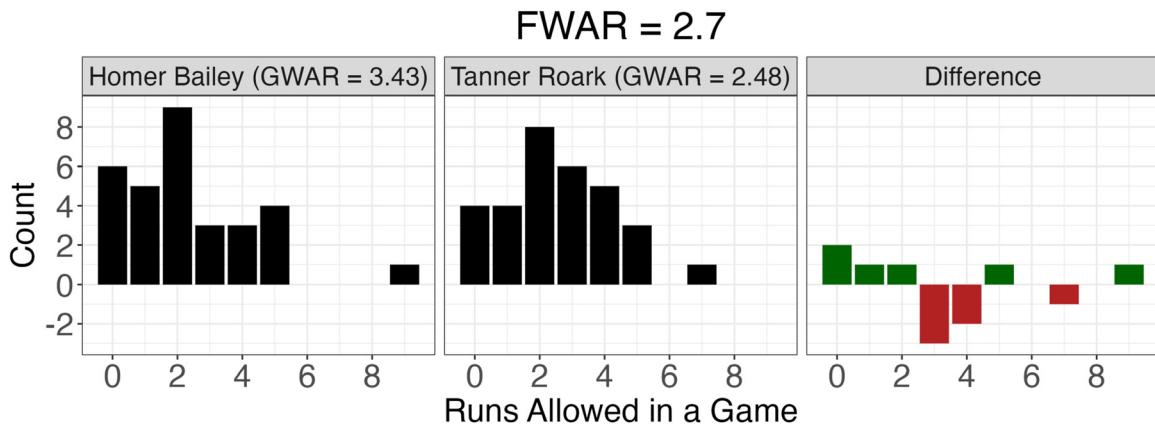


Figure 7: Histogram of runs allowed in a game in 2019 for Homer Bailey (left), Tanner Roark (middle), and the difference between these two histograms (right). Even though they have the same FWAR, Bailey has a higher GWAR than Roark because he has more games in which he allows fewer runs.

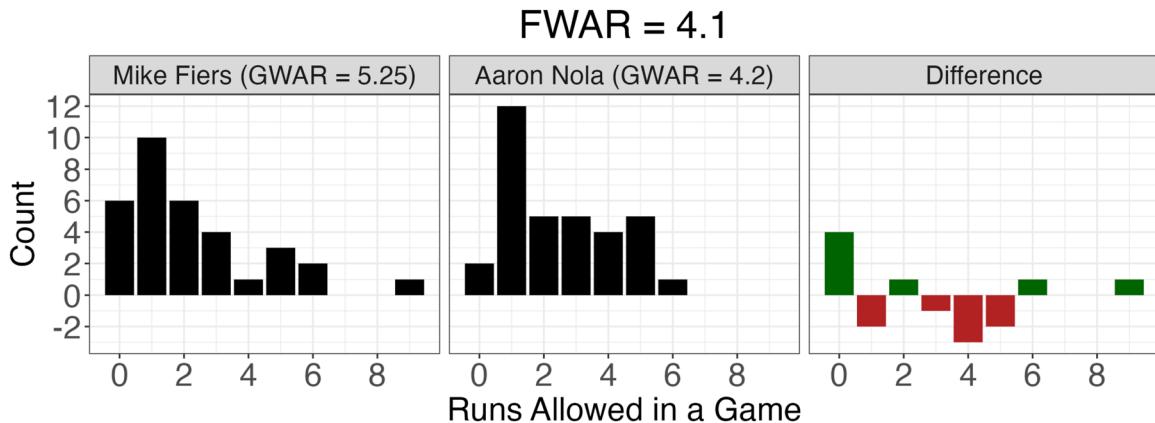


Figure 8: Histogram of runs allowed in a game in 2019 for Mike Fiers (left), Aaron Nola (middle), and the difference between these two histograms (right). Even though they have the same FWAR, Fiers has a higher GWAR than Nola because he has more games in which he allows fewer runs.

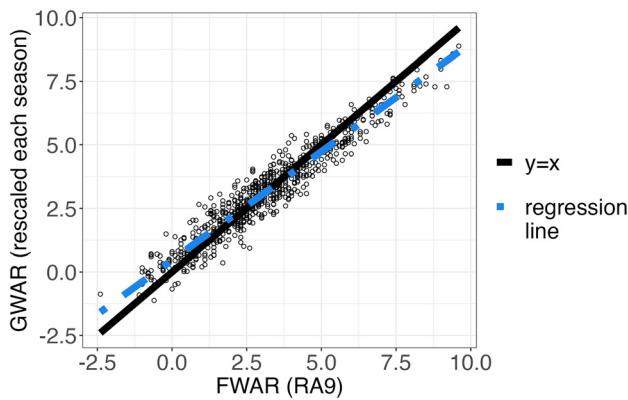


Figure 9: Grid WAR (y-axis) versus FanGraphs WAR (RA/9) (x-axis) for each pitcher-season from 2010 to 2019. The solid black line is $y = x$ and the dashed blue line is the regression line $y = 0.47 + 0.85x$. The slope of the regression line is less than 1, indicating that worse pitchers are generally undervalued and better pitchers are generally overvalued by FWAR(RA/9) relative to GWAR.

$y = x$ (the solid black line), and so are undervalued, whereas better pitchers generally lie below the line. The regression line $y = 0.47 + 0.85x$ (the blue dashed line), which has slope less than one, summarizes this phenomenon. This occurs because FWAR averages pitcher performance across an entire season, which dilutes the contributions of his good games. Since Grid WAR is (mostly) convex in runs allowed, GWAR downweights the contribution of the R th run of a game for large R , whereas FWAR weighs all runs allowed in a season equally. Since worse pitchers have many more occurrences than better pitchers of allowing a large number of runs allowed in a game, FWAR(RA/9) undervalues worse pitchers in general. As we have constrained GWAR and FWAR to have the same sum, FWAR(RA/9) must therefore overvalue better pitchers in general.

In Figure 10 we show the six most undervalued and overvalued pitchers according to GWAR relative to

Pitcher	Career GWAR (Rescaled) Minus FWAR (RA9)
Yovani Gallardo	4.32
Madison Bumgarner	3.46
Tim Lincecum	3.35
Edinson Volquez	3.01
Ryan Vogelsong	2.81
Matt Cain	2.76

(a)

Pitcher	Career GWAR (Rescaled) Minus FWAR (RA9)
Justin Verlander	-5.55
R.A. Dickey	-5.39
Corey Kluber	-4.41
Max Scherzer	-4.17
Mark Buehrle	-3.74
Clayton Kershaw	-3.68

(b)

Figure 10: The six most undervalued (a) and overvalued (b) pitchers according to GWAR relative to FWAR(RA/9), aggregated across all seasons from 2010 to 2019. The undervalued pitchers are generally considered worse pitchers and the overvalued pitchers are generally considered better pitchers.

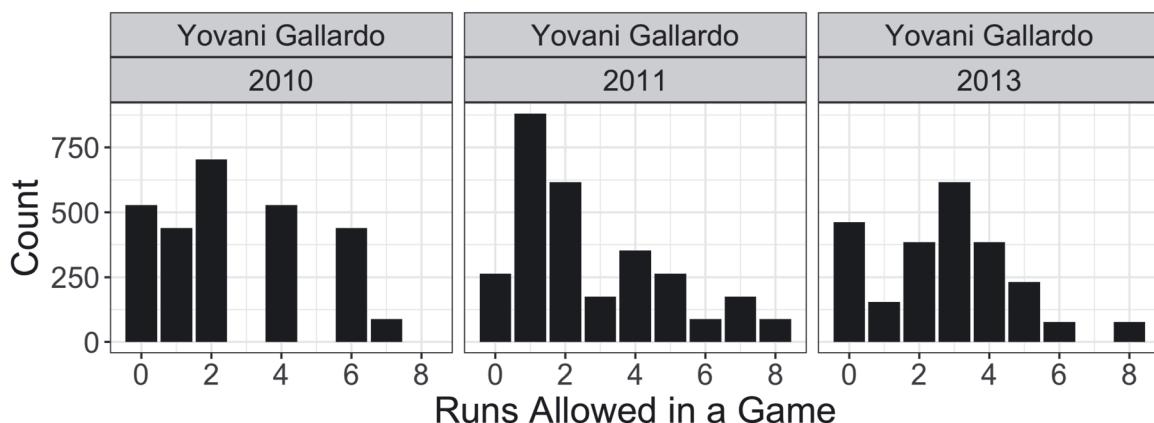


Figure 11: Histogram of Yovani Gallardo’s runs allowed in each game across three seasons. Gallardo has many great and terrible games. Traditional WAR metrics allow his terrible performances to dilute his great ones.

FWAR(RA/9), aggregated across all seasons from 2010 to 2019. As expected, the undervalued pitchers are generally considered worse pitchers and the overvalued pitchers are generally considered better pitchers. In Figure 11 we visualize the runs allowed distribution of the most undervalued pitcher, Yovani Gallardo, in his three most undervalued seasons. Gallardo has many great games (say, zero or one run allowed) and many terrible games (say, six or more runs allowed). GWAR diminishes the impact of these terrible games and magnifies the impact of these great games, increasing his estimated contribution to winning.

As averaging pitcher performance across games allows a pitcher’s worse performances to dilute his better performances, we also expect FanGraphs WAR to undervalue higher variance pitchers. We see this in Figure 12 in which we visualize the relationship between a pitcher’s variability in performance across games and the extent to which he is undervalued by FanGraphs WAR. For each starting pitcher-season from 2010 to 2019 consisting of at least 30 games, we plot the difference between his seasonal GWAR and FWAR(RA/9) against the game-by-game standard

deviation of his GWAR. Higher variance pitchers indeed generally have higher GWAR than FWAR.

3.2 Grid WAR has predictive value

Recall that it is wrong to average pitcher performance across games in estimating a starting pitcher’s WAR. This is why we devised Grid WAR, which estimates WAR on a per-game basis and is the right way to estimate historical WAR for starting pitchers. Nonetheless, it is possible *a priori* that averaging provides a stabler estimate of pitcher quality which is more predictive of future WAR. In particular, if game-by-game variability in pitcher performance is due mostly to chance, accounting for these variations introduces noise into estimates of pitcher quality.

Because a pitcher’s historical WAR at the end of a season defines how valuable he was during the season, we want a predictive pitcher quality measure to predict his next season’s historical WAR as best as possible. In other words, pitcher quality is simply predicted future historical WAR. Hence our goal is to predict a starting pitcher’s future Grid WAR. *A priori*, it is not immediately obvious whether

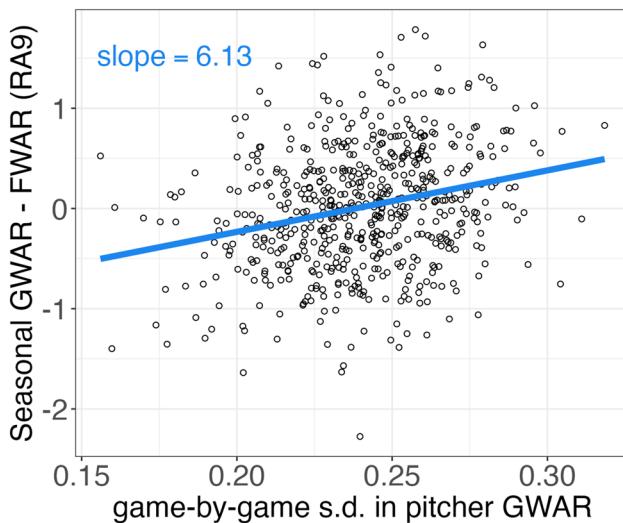


Figure 12: For each starting pitcher-season from 2010 to 2019 consisting of at least 30 games, the difference between his seasonal GWAR and FWAR(RA/9) (y-axis) versus the game-by-game standard deviation of his GWAR (x-axis). Higher variance pitchers are undervalued by FWAR(RA/9) because they allow a pitcher's worse performances to dilute his better performances.

a pitcher's past Grid WAR is predictive of his future Grid WAR. In particular, if a pitcher's game-by-game variance in runs allowed is due mostly to randomness rather than a fundamental identifiable trait, a WAR which averages pitcher performance over the season may be more predictive than Grid WAR of future Grid WAR. Thus, in this section, we compare the predictive capabilities of Grid WAR and FanGraphs WAR. We find that, in predicting future pitcher rankings according to Grid WAR, our predicted pitcher ranking built from Grid WAR is more predictive than that built from FanGraphs WAR. This suggests that some pitchers' game-by-game variance in performance is a fundamental trait.

To value a starting pitcher using his previous seasons' WAR and number of games played, we could simply use his mean game WAR. The fewer games a pitcher has played, however, the less reliable his mean game WAR is in predicting his latent pitcher quality. Therefore, we use shrinkage estimation to construct a pitcher quality metric. In calculating pitcher p 's quality estimate $\hat{\mu}_p$, the fewer games he has played, the more his mean game WAR is shrunk towards the overall mean pitcher quality. Specifically, we construct three shrinkage estimators of pitcher p 's quality, denoted $\hat{\mu}_p^{\text{GWAR}}$, $\hat{\mu}_p^{\text{FWAR (FIP)}}$, and $\hat{\mu}_p^{\text{FWAR (RA/9)}}$, built from the three respective WAR metrics. We use a parametric Empirical Bayes approach in the spirit of Brown (2008) to formulate these shrinkage estimators, detailed in Appendix C.

Recall that our goal is to predict each starting pitcher's next season's cumulative Grid WAR, which at the end of next

season will represent his historical value added. So, using the 2019 season as a hold-out validation set, our goal is to predict each starting pitcher's 2019 Grid WAR. We use our remaining data from 2010 to 2018 to estimate pitcher quality, built separately from GWAR and FWAR, in order to predict 2019 Grid WAR. Thus we restrict our analysis to the set of starting pitchers who have a FanGraphs' WAR in at least one season from 2010 to 2018 (so, they must have at least 25 starts in that season). Our pitcher quality estimators, however, are on different scales since each WAR metric is on its own scale. To ensure fair comparison of Grid WAR and FanGraphs WAR, we map each estimator to a starting pitcher ranking, ranking each pitcher from one (best) to the number of pitchers (worst). We denote the three ranks of pitcher p according to $\hat{\mu}_p^{\text{GWAR}}$, $\hat{\mu}_p^{\text{FWAR (FIP)}}$, and $\hat{\mu}_p^{\text{FWAR (RA/9)}}$ by \hat{R}_p^{GWAR} , $\hat{R}_p^{\text{FWAR (FIP)}}$, and $\hat{R}_p^{\text{FWAR (RA/9)}}$, respectively. In Figure 22 of Appendix C we visualize these starting pitcher rankings prior to the 2019 season according to these estimators $\hat{\mu}_p$ (left) and their associated ranks \hat{R}_p (right). Additionally, we rank pitchers in 2019 by their observed cumulative 2019 Grid WAR, denoted R_p^{GWAR} . Finally, we use root mean squared error (rmse) to measure how well the predicted pitcher rankings \hat{R} predict the observed rankings R , shown in Table 2. We see that pitcher rankings built from Grid WAR are more predictive than those built from FanGraphs WAR. Formally, $A < B$ and $A < C$, where A , B , and C are defined in Table 2. In other words, baseball analysts lose value by not using Grid WAR to value pitchers.

In Tables 3 and 4 we conduct a similar analysis, but restricting the test set to just the five most *undervalued* starting pitchers in 2019 according to \hat{R}^{GWAR} relative to $\hat{R}^{\text{FWAR (RA/9)}}$ (in Table 3) and relative to $\hat{R}^{\text{FWAR (FIP)}}$ (in Table 4). Conversely, in Tables 5 and 6 we conduct a similar analysis, but restricting the test set to just the five most *overvalued* starting pitchers in 2019 according to \hat{R}^{GWAR} relative to $\hat{R}^{\text{FWAR (RA/9)}}$ (in Table 5) and relative to $\hat{R}^{\text{FWAR (FIP)}}$ (in Table 6). We again find that baseball analysts lose value by not using Grid WAR to estimate pitcher quality. In particular, for these “extreme” pitchers who are highly undervalued or highly overvalued, analysts do worse predicting their quality when they use

Table 2: The rmse of the observed pitcher ranking R^{GWAR} in 2019 and three pitcher ranking estimates \hat{R} computed from three different WAR metrics. GWAR is more predictive than FWAR of future GWAR.

Symbol	Observed ranking R	Predicted ranking \hat{R}	rmse(R, \hat{R})
A	R^{GWAR}	\hat{R}^{GWAR}	10.2
B	R^{GWAR}	$\hat{R}^{\text{FWAR (RA/9)}}$	12.4
C	R^{GWAR}	$\hat{R}^{\text{FWAR (FIP)}}$	13.1

Table 3: The rmse of the observed pitcher ranking R^{GWAR} in 2019 and pitcher ranking estimates \hat{R} computed from three different WAR metrics, using just the five most *undervalued* starting pitchers in 2019 according to \hat{R}^{GWAR} relative to each $\hat{R}^{\text{FWAR(RA/9)}}$. GWAR is more predictive than FWAR(RA/9) of future GWAR, particularly for undervalued pitchers.

Observed ranking R	Predicted ranking \hat{R}	rmse(R, \hat{R})
R^{GWAR}	\hat{R}^{GWAR}	7.2
R^{GWAR}	$\hat{R}^{\text{FWAR(RA/9)}}$	15.7

Table 4: The rmse of the observed pitcher ranking R^{GWAR} in 2019 and pitcher ranking estimates \hat{R} computed from three different WAR metrics, using just the five most *undervalued* starting pitchers in 2019 according to \hat{R}^{GWAR} relative to each $\hat{R}^{\text{FWAR(FIP)}}$. GWAR is more predictive than FWAR(FIP) of future GWAR, particularly for undervalued pitchers.

Observed ranking R	Predicted ranking \hat{R}	rmse(R, \hat{R})
R^{GWAR}	\hat{R}^{GWAR}	5.1
R^{GWAR}	$\hat{R}^{\text{FWAR(FIP)}}$	15.0

FWAR rather than GWAR. In Figure 13 we visualize how our GWAR- and FWAR-based pitcher ranking predictions fare against the observed 2019 GWAR pitcher rankings. Specifically, the blue triangles (our 2019 GWAR-based predictions) are closer to the black squares (the observed 2019 pitcher rankings according to GWAR) than the red dots (our 2019 FWAR(FIP)-based predictions).

Further, in Figure 14 we visualize the variability of Grid WAR and FanGraphs WAR from season to season. For each pitcher-season from 2011 to 2019, we plot a pitcher's seasonal WAR against his previous season's WAR. We see that FWAR(FIP) is more stable from season to season than GWAR

Table 5: The rmse of the observed pitcher ranking R^{GWAR} in 2019 and pitcher ranking estimates \hat{R} computed from three different WAR metrics, using just the five most *overvalued* starting pitchers in 2019 according to \hat{R}^{GWAR} relative to each $\hat{R}^{\text{FWAR(RA/9)}}$. GWAR is more predictive than FWAR(RA/9) of future GWAR, particularly for overvalued pitchers.

Observed ranking R	Predicted ranking \hat{R}	rmse(R, \hat{R})
R^{GWAR}	\hat{R}^{GWAR}	10.7
R^{GWAR}	$\hat{R}^{\text{FWAR(RA/9)}}$	14.0

Table 6: The rmse of the observed pitcher ranking R^{GWAR} in 2019 and pitcher ranking estimates \hat{R} computed from three different WAR metrics, using just the five most *overvalued* starting pitchers in 2019 according to \hat{R}^{GWAR} relative to each $\hat{R}^{\text{FWAR(FIP)}}$. GWAR is more predictive than FWAR(FIP) of future GWAR, particularly for overvalued pitchers.

Observed ranking R	Predicted ranking \hat{R}	rmse(R, \hat{R})
R^{GWAR}	\hat{R}^{GWAR}	10.2
R^{GWAR}	$\hat{R}^{\text{FWAR(FIP)}}$	18.0

and FWAR(RA/9), which have similar levels of season-to-season variability. This makes sense: runs allowed, from which GWAR and FWAR(RA/9) are constructed, is inherently more noisy than FIP, which doesn't account for balls in play. Having greater stability across seasons, however, does not mean that FWAR(FIP) is a better pitcher valuation metric than GWAR. As shown previously in this section, an estimate of latent pitcher quality built from GWAR is more predictive of future GWAR than estimates built from FWAR(FIP) and FWAR(RA/9). Predictiveness of future Grid WAR is a more important quality than year-to-year stability, as a general manager should want to acquire a starting pitcher who will have more Grid WAR in future seasons.

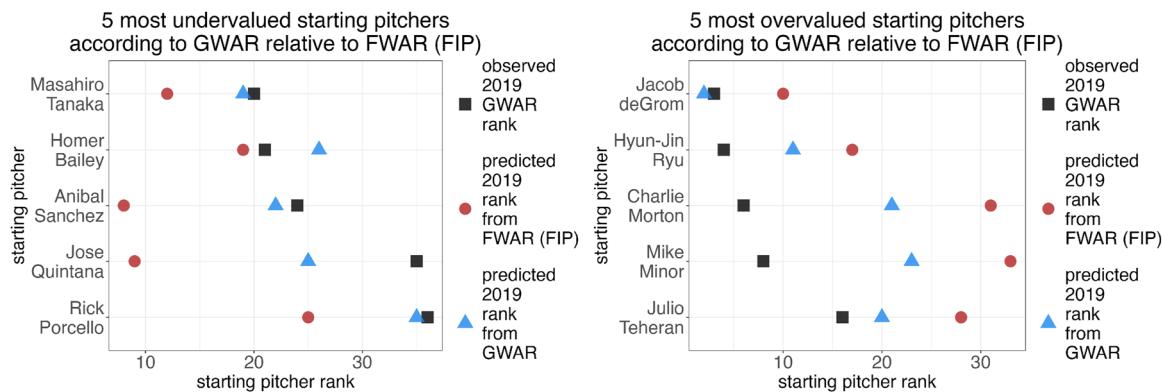


Figure 13: Visualizing the observed pitcher rankings R^{GWAR} in 2019 and pitcher ranking estimates $\hat{R}^{\text{FWAR(FIP)}}$ of the five most undervalued (left) and overvalued (right) starting pitchers in 2019 (according to \hat{R}^{GWAR} relative to $\hat{R}^{\text{FWAR(FIP)}}$). As the blue triangles (our 2019 GWAR-based predictions) are closer to the black squares (the observed 2019 pitcher rankings according to GWAR) than the red dots (our 2019 FWAR(FIP)-based predictions), GWAR is more predictive than FWAR(FIP).

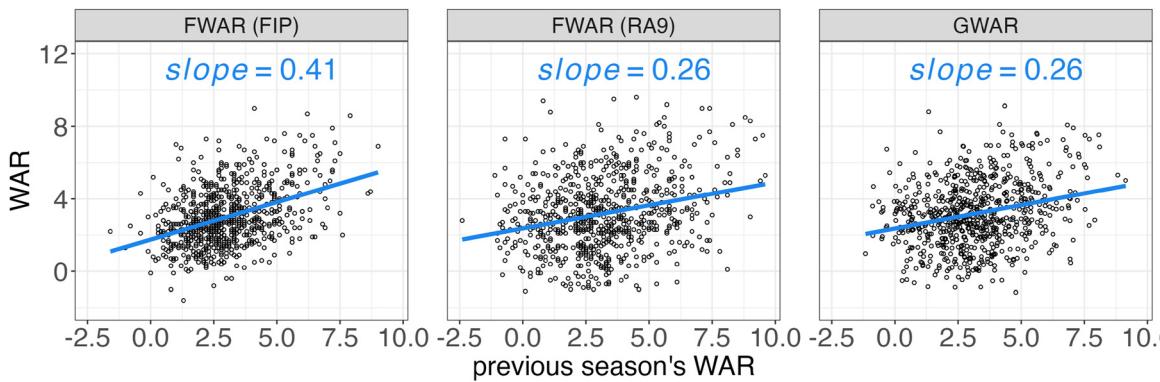


Figure 14: For each pitcher-season from 2011 to 2019, a pitcher’s seasonal WAR (y-axis) versus his previous season’s WAR (x-axis). FWAR(FIP) is more stable from year to year than GWAR or FWAR(RA/9), which have similar levels of season-to-season variability.

So, a game-by-game measure of WAR like Grid WAR is not only the right way to measure historical WAR for starting pitchers, but is also predictive of future Grid WAR. In particular, an estimator of latent pitcher talent should be built using Grid WAR or some other game-by-game metric. Now that we have such an estimator of pitcher talent at our disposal, we explore the relationship between a pitcher’s talent according to $\hat{\mu}_p^{\text{GWAR}}$ and his game-by-game performance. We find that all pitchers have great games, but great pitchers have few terrible games. Therefore, averaging pitcher performance over the entire season dilutes the value of mediocre pitchers’ good games, causing existing WAR metrics to undervalue mediocrity. This agrees with our assessment from the previous section.

In Figure 15 we provide a sense of the distribution of pitcher talent $\hat{\mu}_p^{\text{GWAR}}$ (left) and of the distribution of game-by-game Grid WAR (right). Then, in Figure 16a, we visualize the distribution of game-by-game Grid WAR conditional on

being a bad pitcher (red), a typical pitcher (green), and a great pitcher (blue), according to $\hat{\mu}_p^{\text{GWAR}}$. Bad pitchers, typical pitchers, and great pitchers all have great games. Great pitchers, on the other hand, pitch many fewer bad games than bad and mediocre pitchers do. In Figure 16b we view this phenomenon through another lens. We visualize the distribution of pitcher quality $\hat{\mu}_p^{\text{GWAR}}$ conditional on having a bad game (red), a typical game (green), and a great game (blue). Bad pitchers, typical pitchers, and great pitchers all have great games, but bad games feature a higher proportion of bad pitchers.

Averaging pitcher performance over the season allows a pitcher’s bad performances to dilute the value of his good ones. Consequently, such WAR metrics like FanGraphs WAR devalue the contributions of mediocre and bad pitchers, who have many more bad games than great pitchers. In short, the baseball community has been undervaluing the contributions of the mediocre.

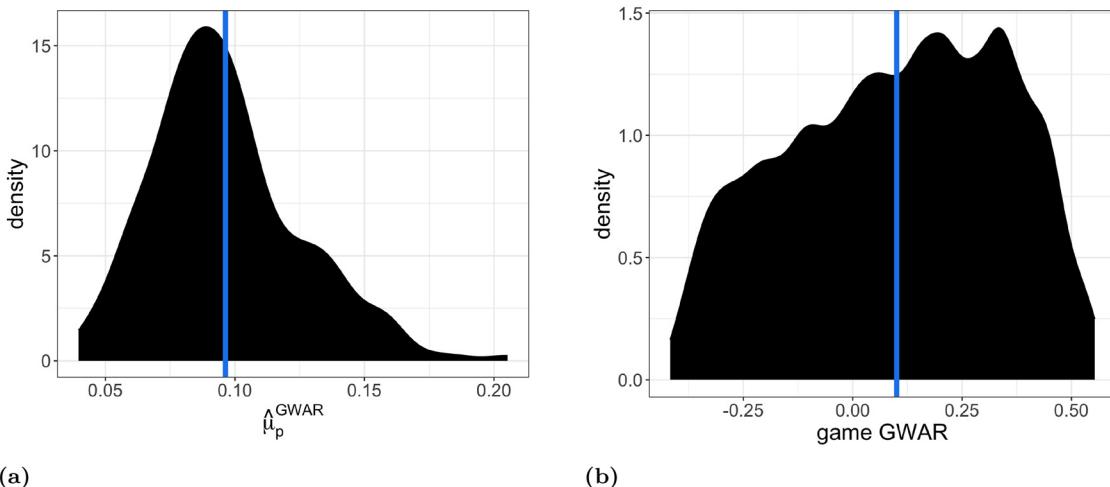


Figure 15: Distribution of estimated pitcher talent $\hat{\mu}_p^{\text{GWAR}}$ (a) and game-by-game Grid WAR (b). The blue lines denote the respective means.

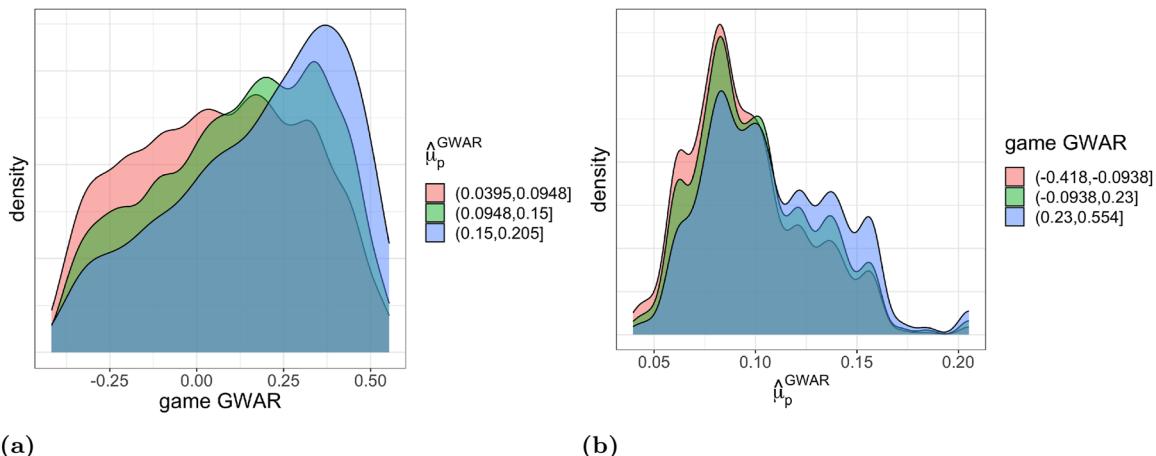


Figure 16: All pitchers have terrible games, but great pitchers have fewer of them. (a) Shows the distribution of game-by-game Grid WAR conditional on being a bad pitcher (red), a typical pitcher (green), and a great pitcher (blue), according to $\hat{\mu}_p^{\text{GWAR}}$. (b) Shows the distribution of estimated pitcher talent $\hat{\mu}_p^{\text{GWAR}}$ conditional on having a bad game (red), a typical game (green), and a great game (blue).

4 Discussion

Traditional implementations of WAR for starting pitchers estimate WAR as a function of pitcher performance averaged over the entire season. Averaging pitcher performance, however, allows a pitcher's bad games to dilute the performances of his good games. One bad “blow-up” game after averaging can reduce a pitcher's WAR by more than minimum possible WAR in a game. Therefore, a starters' seasonal WAR should be the sum of the WAR of each of his individual games. Hence we devise Grid WAR, which estimates a starting pitcher's WAR in each of his games. Grid WAR estimates the context-neutral win probability added above replacement at the point when a pitcher exits the game. Grid WAR is convex in runs allowed, capturing the fundamental baseball principle that you can only lose a game once.

Comparing starting pitchers' Grid WAR to his FanGraphs WAR from 2010 to 2019, we find that standard WAR calculations undervalue mediocrity and variability relative to Grid WAR. Because all starters pitch great games, but great starters don't pitch many terrible games, averaging pitcher performance over a season discounts the contributions of great games by mediocre and bad pitchers. We also show that past performance according to Grid WAR is predictive of future Grid WAR, providing evidence that a pitcher's runs allowed profile is not entirely random, but is the result of an identifiable game-by-game variation.

To compare starting pitchers across baseball history through the lens of Grid WAR, we created an interactive

Shiny app,¹⁹ hosted at <https://gridwar.xyz>, which displays the Grid WAR results of every starting pitcher game, season, and career since 1952. For many starters, Grid WAR is similar to FanGraphs WAR.²⁰ Grid WAR, however, looks much more favorably upon the careers of some starters with intrinsic game-by-game variance (that is, the occasional tendency to pitch an awful game).²¹ There are many starters that have substantial differences including Whitey Ford and Catfish Hunter. Whitey Ford (resp., Catfish Hunter) has a whopping 25 (resp., 15) more career GWAR than FWAR! Ford is the 49th best starter since 1952 according to FanGraphs (53 FWAR) but is the 19th best according to Grid WAR (78 GWAR). Similarly, Hunter is the 107th best starter since 1952 according to FanGraphs (37 FWAR) but is the 32nd best according to Grid WAR (52 GWAR). What drives this difference? Ford and Hunter are extreme boom-bust pitchers. Standard WAR, from either FanGraphs and Baseball Reference, average pitcher performance across games, allowing these pitchers' many blow-up games to dilute their great performances which accumulate into huge discrepancies across careers, significantly devaluing them relative

¹⁹ The website is built using pre-2008 play-by-play data from Retrosheet (2021) and play-by-play data since 2008 from Statcast (2023). We use the `baseballr` package in R to scrape from each of these data sources (Petti and Gilani 2021). We automatically scrape Statcast data each morning, so the website is up-to-date.

²⁰ <https://www.fangraphs.com/leaders/major-league?pos=all&lg=all&qual=y&type=8&month=0&ind=0&team=0&rost=0&players=0&startdate=&enddate=&season1=1952&season=2023&stats=sta&sortcol=20&sortdir=default&pagenum=1&pageitems=2000000000>.

²¹ There is an asymmetry due to selection bias: pitchers that are usually awful and occasionally brilliant don't pitch for long.

to other starters. To understand the fundamental game-by-game variance of these pitchers, consider Ford's 1961 season and Hunter's 1967 season. In 1961 Ford (7.2 GWAR) started 39 games with six complete game shutouts accompanied by seven blow-up games (lower than -0.1 GWAR). Similarly, in 1967 Hunter (4.5 GWAR) started 35 games with five complete game shutouts and four one-run complete games accompanied by eight blow-up games (lower than -0.1 GWAR). Grid WAR, which correctly values pitcher performance in each individual game, sees the value of pitchers having such strong variability across games. The public agrees with us: Catfish Hunter made the Hall-of-Fame (despite having just the 107th best career according to FanGraphs) and Whitey Ford won a Cy Young award in 1961. We continue our discussion of Grid WAR across baseball history in Appendix E.

While we focus in this paper on developing WAR for starting pitchers, some of the principles we discussed should be used to estimate WAR for relief pitchers, opening pitchers, and batters and some should not. The primary difference between valuing starting and relief pitchers is that opposing batter context matters for valuing the latter but not the former. A relief pitcher who pitches the ninth inning up or down, say, three runs only marginally impacts the game, as its outcome has essentially already been decided. Conversely, a relief pitcher who pitches the ninth inning up or down, say, one run hugely impacts the game. Grid WAR for starting pitchers, as developed in this paper, doesn't account for opposing batter context and so is not appropriate to value relief pitchers. Nonetheless, due to the massive variability in leverage for relief pitchers across games, relief pitcher WAR should be estimated separately for each game. Perhaps we should construct relief pitcher game WAR using a context-neutral version of win probability added that adjusts for score differential at the time the relief pitcher enters the game.

Opening pitchers are starting pitchers, so we can use Grid WAR to estimate their value. Grid WAR provides a strong justification for the use of an opener. For concreteness, suppose you are the manager of the 2023 Padres, featuring closer Josh Hader who had a 1.28 ERA through 56 innings in 61 games. For simplicity, suppose Hader pitched for exactly one inning, the ninth inning, in each of 56 games, resulting in 1.7 FanGraphs WAR.²² Now suppose he had instead pitched just the first inning in 56 games with that ERA, allowing an average of about 0.14 runs per inning, or 0 runs in 86 % of innings and 1 run in 14 % of innings. His

average context neutral win probability per game would be about

$$(0.86) \cdot f(I = 1, R = 0) + (0.14) \cdot f(I = 1, R = 1) \\ \approx (0.86) \cdot (0.53) + (0.14) \cdot (0.44) = 0.52,$$

taking these values of f from Figure 1, resulting in

$$(0.52 - w_{\text{rep}}) \cdot (56 \text{ games}) \approx 5.15$$

seasonal Grid WAR. Our preliminary analysis suggests that Hader would be much more valuable as an opener than as a closer. What drives the difference? Closers aren't that valuable because many of the games they pitch in are low leverage situations in which the outcome of the game has essentially already been decided. Conversely, the performance of an opening pitcher *always matters*. Allowing few runs through the first inning is very valuable since those innings always impact the outcome of the game. Of course, our initial analysis is overly simplistic and is intended to be just an impetus for further research. A closing pitcher's dominant ERA doesn't necessarily extrapolate to the first inning because, for instance, in each first inning but not necessarily in each ninth inning, a pitcher faces the opposing team's best batters. A more thorough analysis would also need to account for the following telescoping phenomenon: if we moved the closing pitcher to the opening pitcher position, the setup man would become the closer and the middle relief pitcher would become the setup man; but, we expect the resulting potential loss of WAR to be much smaller than the positive gain in WAR from switching the closer to the opener position. Finally, if many teams used an opening pitcher replacement-level w_{rep} for openers may change.

Game-by-game WAR, on the other hand, doesn't make sense for batters. This largely stems from the fact that each individual batter has just three to five plate appearances in a typical nine-inning game. A batter doesn't influence any individual game enough for the non-convexity of game-level WAR to meaningfully change his valuation over the course of a season. Also, these aren't enough plate appearances to allow strong evidence of fundamental variance in batter performance across games to appear in the data. Finally, in each game the starting pitcher is responsible for the vast majority of his team's (defensive) context (until he is pulled), which is not true for any individual batter. Having said that, in estimating seasonal batter WAR, batter performance should still be adjusted for opposing pitcher quality and park.

Although Grid WAR improves substantially upon existing estimates of WAR for starting pitchers, our analysis is not without limitations. In particular, the version of Grid WAR described in this paper, as well WAR estimates from

²² <https://www.fangraphs.com/players/josh-hader/14212/stats?position=P>.

FanGraphs and Baseball Reference, doesn't adjust for opposing batter quality. Thus, for a pitcher who faces good offensive teams more often than other pitchers do, Grid WAR underestimates his WAR. In our updated version of Grid WAR at gridwar.xyz, we adjust for opposing batter quality via GWAR+ (details are included on the website). Additionally, the current version of Grid WAR doesn't adjust for the pitcher's team's fielding. Thus, for a pitcher who plays with great fielders who reduce his runs allowed, Grid WAR overestimates his WAR. Nonetheless, we expect these adjustments to have a very small impact. In particular, we expect the effect of fielding to have a smaller total impact than ballpark, which itself has a small impact, except at extreme parks like Coors Field. This can be seen in Figure 27: Grid WAR computed with our ridge-adjusted park effects is extremely similar to Grid WAR without park effects. We leave the addition of batting and fielding adjustments to future work.

Moreover, the distribution of runs scored in a half-inning is not Poisson; more likely it is a zero-inflated Poisson, a more general Conway-Maxwell-Poisson, or a similar distribution on the non-negative integers. Computationally, it is straightforward to modify the f grid formula (Equation (5)) to accommodate different distributions. One interesting modification would adjust to allow different parameters for each inning depending on when a starting pitcher is pulled. In particular, middle relievers tend to be worse than starting pitchers, suggesting a higher value of λ for those innings, and closers are often very good pitchers, suggesting a lower value of λ . But there are several substantial benefits to sticking with a simpler Poisson model. First, it produces a closed-form formula which is quick to evaluate. Second, a simple parametric model makes it easier to adjust for ballpark (and other confounders like batting quality and fielding quality). Finally, the resulting f grid is reasonable and quite accurate for our purposes. For example, while the Poisson model systematically underestimates the probability of big-deficit late inning comeback, these differences have an insignificant impact on Grid WAR. We leave any adjustment to the half-inning runs distribution as future work.

Finally, there is a flaw in our Empirical Bayes shrinkage estimator of latent pitcher quality μ_p . Formula (20) assumes that μ_p remains constant over the entire decade from 2010 to 2019, but player quality is non-stationary over time, and a more elaborate estimator should account for this. In future work we suggest using a similar Empirical Bayes approach to estimate pitcher quality, with weights that decay further back in time (e.g., using exponential decay weighting as

in Medvedovsky and Patton 2022) in the posterior mean Formulas (20) and (32).

Acknowledgments: The authors thank Eric Babitz and Sam Bauman, who first calculated Grid WAR for starting pitchers and created the name Grid WAR, and Justin Lipitz, Emma Segerman, and Ezra Troy, who contributed to an early version of this paper.

Research ethics: Not applicable.

Author contributions: The authors have accepted responsibility for the entire content of this manuscript and approved its submission.

Competing interests: The authors state no conflict of interest.

Research funding: None declared.

Data availability: There is a link to the data in the paper and the raw data can be obtained on request from the corresponding author.

Appendix A: A review of the rules of baseball

In a baseball game, two teams of nine players each take turns on offense (batting and baserunning) and defense (fielding and pitching). The game occurs as a sequence of plays, with each play beginning when the pitcher throws a ball that a batter tries to hit with a bat. The objective of the offensive team (batting team) is to hit the ball into the field of play, away from the other team's players, in order to get on base. The goal of a baserunner is to eventually advance counter-clockwise around all four bases to score a "run", which occurs when he touches home plate (the position where the batter initially began batting). The defensive team tries to prevent batters from reaching base and scoring runs by getting batters or baserunners "out", which forces them out of the field of play. The pitcher can get the batter out by throwing three pitches which result in "strikes". Fielders can get the batter out by catching a batted ball before it touches the ground, and can get a runner out by tagging them with the ball while the runner is not touching a base. The batting team's turn to bat is over once the defensive team records three outs. The two teams switch back and forth between batting and fielding; one turn batting for each team constitutes an inning. A game is usually composed of nine innings, and the team with the greater number of runs at the end of the game wins. Most games end after the ninth inning, but if scores are tied at that point, extra innings are played (Wikipedia 2023).

Appendix B: Estimating f using a mathematical, not a statistical, model

In this section, we detail our modeling process for estimating the grid function $f = f(I, R)$ which, assuming both teams have randomly drawn offenses, computes the probability a team wins a game after giving up R runs through I complete innings. In particular, we compare statistical models fit from observational data to mathematical probability models, which are superior.

To account for different run environments across different seasons and leagues (NL vs. AL), we estimate a different grid for each league-season. We begin by estimating f from our observational dataset of half-innings from 2010 to 2019. The response variable is a binary indicator denoting whether the pitcher's team won the game, and the features are the inning number I , the runs allowed through that half-inning R , the league, and the season. Note that if a home team leads after the top of the 9th inning, then the bottom of the 9th is not played. Therefore, to avoid selection bias, we exclude all 9th inning instances in which a pitcher pitches at home.

With enough data, the empirical grid (e.g., binning and averaging over all combinations of I and R within a league-season) is a great estimator of f . In Figure 17a we visualize the empirical grid fit on a dataset of all half-innings from 2019 in which the home team is in the National League. The function f should be monotonic decreasing in R . In particular, as a pitcher allows more runs through a fixed number of innings, his team is less likely to win the game. It should also be monotonic increasing in I because giving up R runs through I innings is worse than giving up R runs through $I + i$ innings for $i > 0$, since giving up R runs

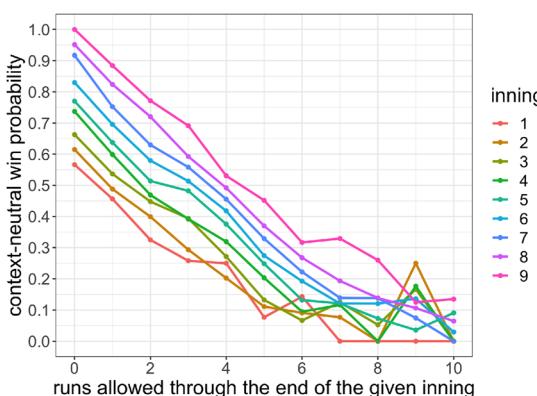
through $I + i$ innings implies a pitcher gave up no more than R runs through I innings. The empirical grid, however, is not monotonic in either R or I because each league-season dataset is not large enough. Moreover, even when we use our entire dataset of all half-innings from 2010 to 2019, the empirical grid is still not monotonic in R or I .

To force our fitted f to be monotonic, we use XGBoost with monotonic constraints, tuned using cross validation (Chen and Guestrin 2016). We visualize our 2019 NL XGBoost fit in Figure 17b. We indeed see that the fitted f is decreasing in R and increasing in I . Additionally, $R \mapsto f(I, R)$ is mostly convex: if a pitcher has already allowed a high number of runs, there is a lesser relative impact of throwing an additional run on winning the game. Nonetheless, XGBoost overfits, especially towards the tails (e.g., for R large). For instance, the 2019 NL XGBoost model indicates that the probability of winning a game after allowing 10 runs through 9 innings is about 0.11, which is too large.

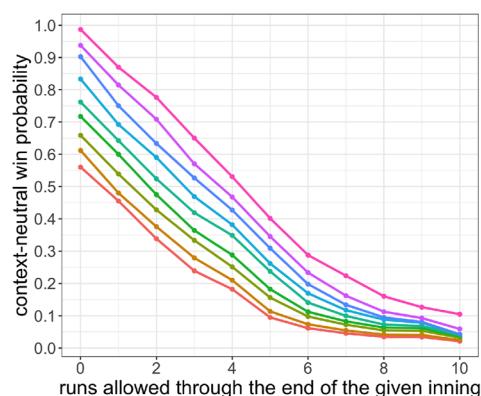
As there is not enough data to use machine learning to fit a separate grid for each league-season without overfitting, we turn to a parametric mathematical model. Indeed, the power of parameterization is that it distills the information of a dataset into a concise form (e.g., into a few parameters), allowing us create a strong model from limited data. Because the runs allowed in a half-inning is a natural number, we begin our parametric quest by supposing that the runs allowed in a half-inning is a Poisson(λ) random variable. In particular, denoting the runs allowed by the pitcher's team's batters in inning i by X_i and the runs allowed by the opposing team in inning i (for innings i after the pitcher exits the game), we assume

$$X_i, Y_i \stackrel{\text{i.i.d.}}{\sim} \text{Poisson}(\lambda). \quad (10)$$

Then the probability that a pitcher wins the game after allowing R runs through I innings, assuming win probability



(a)



(b)

Figure 17: Estimates of the 2019 National League function $R \mapsto f(I, R)$ using the empirical grid (a) and XGBoost with monotonic constraints (b).

in overtime is $1/2$, is

$$\begin{aligned} f(I, R|\lambda) := & \mathbb{P}\left(\sum_{i=1}^9 X_i > R + \sum_{i=I+1}^9 Y_i\right) \\ & + \frac{1}{2} \cdot \mathbb{P}\left(\sum_{i=1}^9 X_i = R + \sum_{i=I+1}^9 Y_i\right). \end{aligned} \quad (11)$$

If $I = 9$, this is equal to

$$\mathbb{P}(\text{Poisson}(9\lambda) > R) + \frac{1}{2} \cdot \mathbb{P}(\text{Poisson}(\lambda) = R). \quad (12)$$

If $I < 9$, it is equal to

$$\begin{aligned} & \mathbb{P}(\text{Skellam}(9\lambda, (9-I-1)\lambda) > R) \\ & + \frac{1}{2} \cdot \mathbb{P}(\text{Skellam}(9\lambda, (9-I-1)\lambda) = R), \end{aligned} \quad (13)$$

noting that the Skellam distribution arises as a difference of two independent Poisson distributed random variables. Then, we estimate λ separately for each league-season by computing each team's mean runs allowed in each half inning, and then averaging over all teams.

In Figure 18a we visualize the estimated f according to our Poisson model (11) using the 2019 NL λ . We see that f is decreasing in R , increasing in I , convex in the tails of R , and is smooth. Nonetheless, some of the win probability values from this model are unrealistic. For instance, it implies the probability of winning the game after shutting out the opposing team through 9 innings is about 99 %, which is too high, and the probability of winning the game after allowing 10 runs through 9 innings is about 1 %, which is too low.

The win probability values at both tails of R are too extreme in our original Poisson model (11) because we assume both teams have the same mean runs per inning

λ . This is an unrealistic assumption: in real life, a baseball season involves teams of varying strength playing against each other. When teams of differing batting strength play each other, win probabilities differ. For instance, a great hitting team down seven runs has a larger probability of coming back to win the game than a worse hitting team would. Thus, accounting for random differences in team strength across games should flatten the $f(I, R)$ grid.

On this view, it is more realistic to assume the pitcher's team and the opposing team have their own runs scored per inning parameters,

$$X_i \stackrel{\text{i.i.d.}}{\sim} \text{Poisson}(\lambda_X) \quad \text{and} \quad Y_i \stackrel{\text{i.i.d.}}{\sim} \text{Poisson}(\lambda_Y), \quad (14)$$

and

$$\begin{aligned} f(I, R|\lambda_X, \lambda_Y) := & \mathbb{P}\left(\sum_{i=1}^9 X_i > R + \sum_{i=I+1}^9 Y_i\right) \\ & + \frac{1}{2} \cdot \mathbb{P}\left(\sum_{i=1}^9 X_i = R + \sum_{i=I+1}^9 Y_i\right). \end{aligned} \quad (15)$$

Moreover, to capture the variability in team strength across each of the 30 MLB teams, we impose a positive normal prior,

$$\lambda_X, \lambda_Y \sim \mathcal{N}_+(\lambda, \sigma_\lambda^2). \quad (16)$$

We estimate the prior hyperparameters λ and σ_λ separately for each league-season by computing each team's mean and s.d. of the runs allowed in each half inning, respectively, and then averaging over all teams.

Given λ_X and λ_Y , we compute Formula (15) similarly as before using the Poisson and Skellam distributions. We use Monte Carlo integration with $B = 100$ samples to estimate the posterior mean grid,

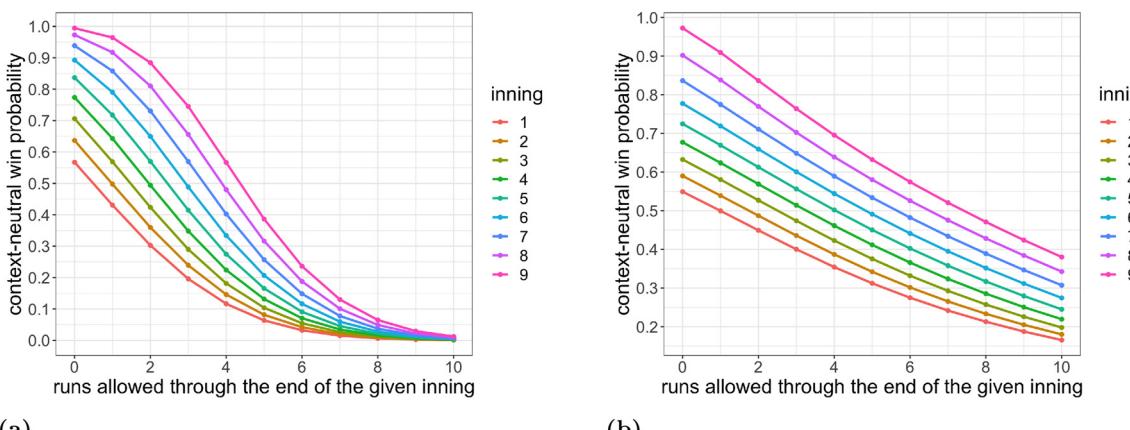


Figure 18: Estimates of the 2019 National League function $R \mapsto f(I, R)$ using our Poisson model (11) with constant λ (a) and our Poisson model (17) with a truncated normal prior (16) on two team strength parameters λ_X and λ_Y (b).

$$f(I, R | \lambda, \sigma_\lambda^2) \approx \frac{1}{B} \sum_{b=1}^B f(I, R | \lambda_X^{(b)}, \lambda_Y^{(b)}), \quad (17)$$

where $\lambda_X^{(b)}$ and $\lambda_Y^{(b)}$ are i.i.d. samples from the prior distribution (16).

In Figure 18b we visualize the estimated f according to this Poisson model (17), with prior (16), using the 2019 NL λ and σ_λ^2 . We see that f is mostly linear in R , rather than convex, and the values of f when R is large are highly unrealistic. For instance, this model indicates that the probability of winning the game after allowing 10 runs through 9 innings is about 38 %, which is way too high. This is because our model is overdispersed, i.e. the estimated prior variance σ_λ^2 is too large. For example, too large of a σ_λ^2 allows λ_X and λ_Y to be very far apart, so if a pitcher allows 10 runs through 9 innings and λ_X is much larger than λ_Y , then his team will have a significant chance of coming back to win.

To resolve the overdispersion issue, we introduce a tuning parameter k designed to tune the dispersion across team strengths to match observed data,

$$\lambda_X, \lambda_Y \sim \mathcal{N}_+(\lambda, k \cdot \sigma_\lambda^2). \quad (18)$$

In particular, we use $k = 0.28$, which minimizes the log-loss between the observed win/loss column and predictions from the induced grid $f(I, R | \lambda, \sigma_\lambda^2, k)$. In Figure 19 we visualize the estimated f according to our Poisson model (17), with tuned dispersion prior (18), using the 2019 NL λ and σ_λ^2 . We see that f is decreasing in R , increasing in I , and convex when R is large. In particular, it looks like a smoothed version of the XGBoost grid from Figure 17b. Additionally, the values of the grid at both tails of R seem reasonable. For instance, the model indicates that allowing 0 runs through

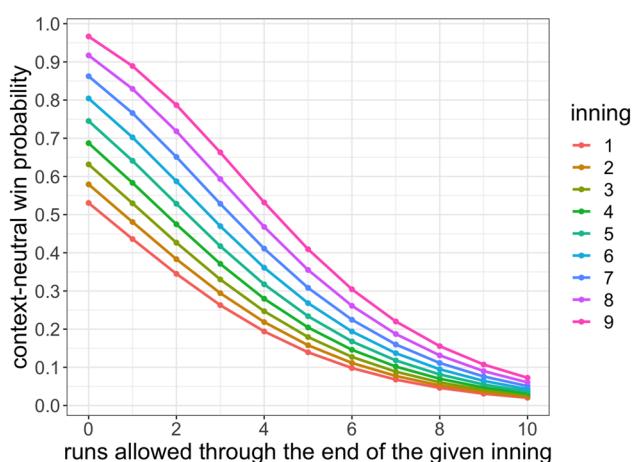


Figure 19: Estimates of the 2019 National League function $R \mapsto f(I, R)$ using our Poisson model (17) with tuned dispersion prior (18).

9 innings has about a 97 % win probability, which is more reasonable than before. For all of these reasons, we use this model for the grid f to compute Grid WAR for starting pitchers.

Appendix C: Estimating pitcher quality using Empirical Bayes

In this section, we describe how we estimate pitcher quality. Given enough data, a pitcher's mean game WAR would suffice to capture his quality. In the MLB, however, a pitcher starts just a finite number of games per season, so for many pitchers there is not enough data to use just his mean game WAR to represent his quality. Therefore, in this section we use a parametric Empirical Bayes approach in the spirit of Brown (2008) to devise shrinkage estimators $\hat{\mu}_p^{\text{GWAR}}$ and $\hat{\mu}_p^{\text{FWAR}}$, built from Grid WAR and FanGraphs WAR respectively, to represent pitcher p 's quality. In particular, $\hat{\mu}_p$ shrinks his mean game WAR to the overall mean in proportion to his number of games played.

C.1 Empirical Bayes estimator of pitcher quality built from Grid WAR

To begin, index each starting pitcher by $p \in \{1, \dots, \mathcal{P}\}$ and index pitcher p 's games by $g \in \{1, \dots, N_p\}$. Let X_{pg} denote pitcher p 's observed Grid WAR in game g . After observing his N_p games, we model

$$X_{pg} \sim \mathcal{N}(\mu_p, \sigma_p^2), \quad (19)$$

$$\mu_p \sim \mathcal{N}(\mu, \tau^2).$$

In this model, μ_p represents pitcher p 's unobservable "true" pitcher quality, or his latent underlying mean game Grid WAR. Similarly, σ_p^2 represents pitcher p 's latent game-by-game variance in pitcher quality, or his game-by-game variance in mean game Grid WAR. The prior parameters μ and τ^2 represent the mean and variance, respectively, of pitcher quality across all pitchers. In Figure 20 we visualize the game-level Grid WAR of four starting pitchers. While Grid WAR is not actually normally distributed, it isn't too unreasonable an approximation (particularly for typical pitchers). In particular, we use a Gaussian model because it produces a good and interpretable estimator of pitcher p 's latent pitcher quality, not because of accuracy.

We estimate pitcher p 's pitcher quality μ_p using the posterior mean, which as a result of our normal-normal conjugate model (19) is

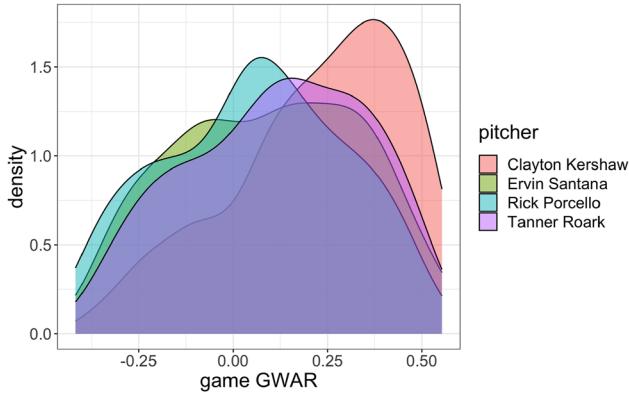


Figure 20: For four starting pitchers p , the distribution of his game-level GWAR $\{X_{pg}\}$ from 2010 to 2018.

$$\hat{\mu}_p := \mathbb{E}[\mu_p | X_{p1}, \dots, X_{pN_p}] = \frac{\frac{1}{\sigma_p^2} \sum_{g=1}^{N_p} X_{pg} + \frac{\mu}{\tau^2}}{\frac{N_p}{\sigma_p^2} + \frac{1}{\tau^2}}. \quad (20)$$

The posterior mean is a weighted sum between the observed total Grid WAR and the overall mean pitcher quality, weighted by the variances σ_p^2 and τ^2 and the number of games played N_p . In particular, the more games a pitcher has played, the closer his estimated quality is to his observed mean game Grid WAR. Conversely, the fewer games he has played, the closer his estimated quality is to the overall mean quality.

Estimator (20), however, is defined in terms of unknown parameters μ , τ^2 , and σ_p^2 . Thus, to effectively use this estimator, we employ an Empirical Bayes approach in the spirit of Brown (2008). Specifically, in place of these estimators in Equation (20), we plug in their maximum likelihood estimates (MLE) $\hat{\mu}$, $\hat{\tau}^2$, and $\hat{\sigma}_p^2$, estimated from the data $\{X_{pg}\}$.

We begin by finding the MLE by noting the marginal distribution of X_{pg} according to model (19),

$$X_{pg} \sim \mathcal{N}(\mu, \tau^2 + \sigma_p^2). \quad (21)$$

Thus the likelihood of pitcher p 's data $\{X_{pg}: 1 \leq g \leq N_p\}$ is

$$\begin{aligned} & \mathbb{P}(\{X_{pg}: 1 \leq g \leq N_p\} | \mu, \tau^2, \sigma_p^2) \\ &= \left(\frac{1}{\sqrt{2\pi(\tau^2 + \sigma_p^2)}} \right)^{N_p} \\ & \cdot \exp \left(\frac{-1}{2(\tau^2 + \sigma_p^2)} \sum_{g=1}^{N_p} (X_{pg} - \mu)^2 \right). \end{aligned} \quad (22)$$

Therefore the log-likelihood of the full dataset $\{X_{pg}: 1 \leq g \leq N_p, 1 \leq p \leq \mathcal{P}\}$ is

$$\begin{aligned} & \ell(\{X_{pg}\} | \mu, \tau^2, \sigma_p^2) \\ &= \sum_{p=1}^{\mathcal{P}} \log \mathbb{P}(\{X_{pg}: 1 \leq g \leq N_p\} | \mu, \tau^2, \sigma_p^2) \\ &\propto -\sum_{p=1}^{\mathcal{P}} \frac{N_p}{2} \cdot \log(\tau^2 + \sigma_p^2) \\ &\quad - \sum_{p=1}^{\mathcal{P}} \frac{1}{2(\tau^2 + \sigma_p^2)} \sum_{g=1}^{N_p} (X_{pg} - \mu)^2. \end{aligned} \quad (23)$$

To find the MLE of μ , we set the derivative of the log-likelihood with respect to μ equal to 0 and solve for μ ,

$$\frac{\partial \ell}{\partial \mu} = -\sum_{p=1}^{\mathcal{P}} \frac{1}{2(\tau^2 + \sigma_p^2)} \sum_{g=1}^{N_p} 2(X_{pg} - \mu) = 0, \quad (24)$$

which yields

$$\mu = \frac{\sum_{p,g} X_{pg} / (\tau^2 + \sigma_p^2)}{\sum_{p,g} 1 / (\tau^2 + \sigma_p^2)}. \quad (25)$$

We use a similar approach to find the MLE of τ^2 and σ_p^2 . In particular,

$$\frac{\partial \ell}{\partial \tau^2} = -\frac{1}{2} \sum_{p,g} \frac{1}{\tau^2 + \sigma_p^2} + \frac{1}{2} \sum_{p,g} \frac{(X_{pg} - \mu)^2}{(\tau^2 + \sigma_p^2)^2}, \quad (26)$$

or equivalently

$$\sum_{p,g} \frac{1}{\tau^2 + \sigma_p^2} = \sum_{p,g} \frac{(X_{pg} - \mu)^2}{(\tau^2 + \sigma_p^2)^2}. \quad (27)$$

Additionally, for each pitcher p ,

$$\frac{\partial \ell}{\partial \sigma_p^2} = -\frac{1}{2} \sum_{g=1}^{N_p} \frac{1}{\tau^2 + \sigma_p^2} + \frac{1}{2} \sum_{g=1}^{N_p} \frac{(X_{pg} - \mu)^2}{(\tau^2 + \sigma_p^2)^2}, \quad (28)$$

or equivalently

$$\sum_{g=1}^{N_p} \frac{1}{\tau^2 + \sigma_p^2} = \sum_{g=1}^{N_p} \frac{(X_{pg} - \mu)^2}{(\tau^2 + \sigma_p^2)^2}. \quad (29)$$

This process yields $\mathcal{P} + 2$ Equations (25), (27), and (29) in $\mathcal{P} + 2$ unknown variables μ , τ^2 , and $\{\sigma_p^2\}$. As suggested in Brown (2008), we solve these equations by iterating until convergence, detailed in iterative Algorithm 1.

Algorithm 1. Compute the MLE of μ , τ^2 , and $\{\sigma_p^2\}$ from model (21)

```

1: Input: Grid WAR  $\{X_{pg}: 1 \leq p \leq P, 1 \leq g \leq N_p\}$ ,  $\epsilon$ 
2: Initialization:
3:  $\mu(t=0) = \frac{1}{P} \sum_p \frac{1}{N_p} \sum_g X_{pg}$ 
4:  $\sigma_p^2(t=0) = \text{Var}(\{X_{pg}: 1 \leq g \leq N_p\})$ 
5:  $\tau^2(t=0) = \text{Var}(\{X_{pg}: 1 \leq p \leq P, 1 \leq g \leq N_p\}) - \frac{1}{P} \sum_p \sigma_p^2(t=0)$ 
6:  $t = 1$ 
7: while TRUE do
8:   Step 1. Solve for  $\mu$  and save the result as  $\mu(t)$ :  $\mu(t) = \frac{\sum_{p,g} X_{pg}/(\tau^2(t-1)+\sigma_p^2(t-1))}{\sum_{p,g} 1/(\tau^2(t-1)+\sigma_p^2(t-1))}$ .
9:   Step 2. Solve for  $\tau^2$  (e.g., using a root finder) and save the result as  $\tau^2(t)$ :  $\sum_{p,g} \frac{1}{\tau^2(t)+\sigma_p^2(t-1)} = \sum_{p,g} \frac{(X_{pg}-\mu(t))^2}{(\tau^2(t)+\sigma_p^2(t-1))^2}$ .
10:  for  $p = 1$  to  $P$  do
11:    Step 3. Solve for  $\sigma_p^2$  (e.g., using a root finder) and save the result as  $\sigma_p^2(t)$ :  $\sum_{g=1}^{N_p} \frac{1}{\tau^2(t)+\sigma_p^2(t)} = \sum_{g=1}^{N_p} \frac{(X_{pg}-\mu(t))^2}{(\tau^2(t)+\sigma_p^2(t))^2}$ .
12:  end for
13:  if  $|\mu(t) - \mu(t-1)| < \epsilon$ ,  $|\tau^2(t) - \tau^2(t-1)| < \epsilon$ , and  $|\sigma_p^2(t) - \sigma_p^2(t-1)| < \epsilon \forall p$  then
14:    break the while loop
15:  else
16:     $t = t + 1$ 
17:  end if
18: end while
19: Output:  $\hat{\mu} = \mu(t)$ ,  $\hat{\tau}^2 = \tau^2(t)$ ,  $\hat{\sigma}_p^2 = \sigma_p^2(t)$ 

```

Using our dataset of all starting pitchers from 2010 to 2018,²³ we run Algorithm 1, yielding maximum likelihood estimators of μ , τ^2 , and $\{\sigma_p^2\}$. With $\epsilon = 10^{-5}$, the algorithm converges after just four iterations. Then, we plug these estimators into Formula (20), yielding parametric Empirical Bayes estimators of $\{\mu_p\}$. In Figure 21 we compare these estimates $\{\hat{\mu}_p^{\text{GWAR}}\}$ to each pitcher's observed mean game Grid WAR. For players with fewer games played (small gray dots), $\hat{\mu}_p$ is shrunk towards the overall mean $\hat{\mu}$. For players with enough games played (large blue dots), $\hat{\mu}_p$ is essentially pitcher p 's mean game GWAR, lying on the line $y = x$.

In Figure 22 we visualize starting pitcher rankings prior to the 2019 season according to $\hat{\mu}_p$ (left) and the associated ranks \hat{R}_p (right). Clayton Kershaw has the highest $\hat{\mu}_p^{\text{GWAR}}$ and Ivan Nova has the lowest.

C.2 Empirical Bayes estimator of pitcher quality built from FanGraphs WAR

Our estimator $\hat{\mu}_p^{\text{FWAR}}$ of latent pitcher quality built from pitcher p 's previous seasons' observed FanGraphs WAR differs methodologically from our estimator built from Grid WAR in that FWAR is computed on the seasonal level and

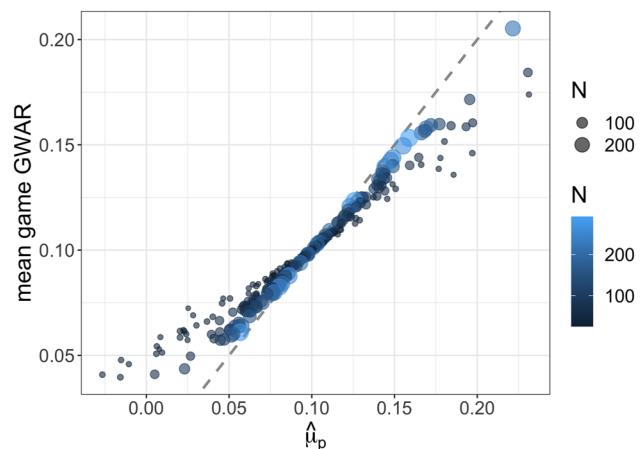


Figure 21: For starting pitchers p from 2010 to 2018, his mean game GWAR versus his Empirical Bayes estimator $\hat{\mu}_p^{\text{GWAR}}$. The dashed gray line is the line $y = x$.

²³ Here we just use starting pitcher-seasons whose FanGraphs WAR is available online, because the purpose of crafting our pitcher quality estimates $\hat{\mu}_p$ is to compare the predictive capability of Grid WAR to FanGraphs WAR. Specifically, these are the starting pitcher-seasons with at least 25 starts in a season.

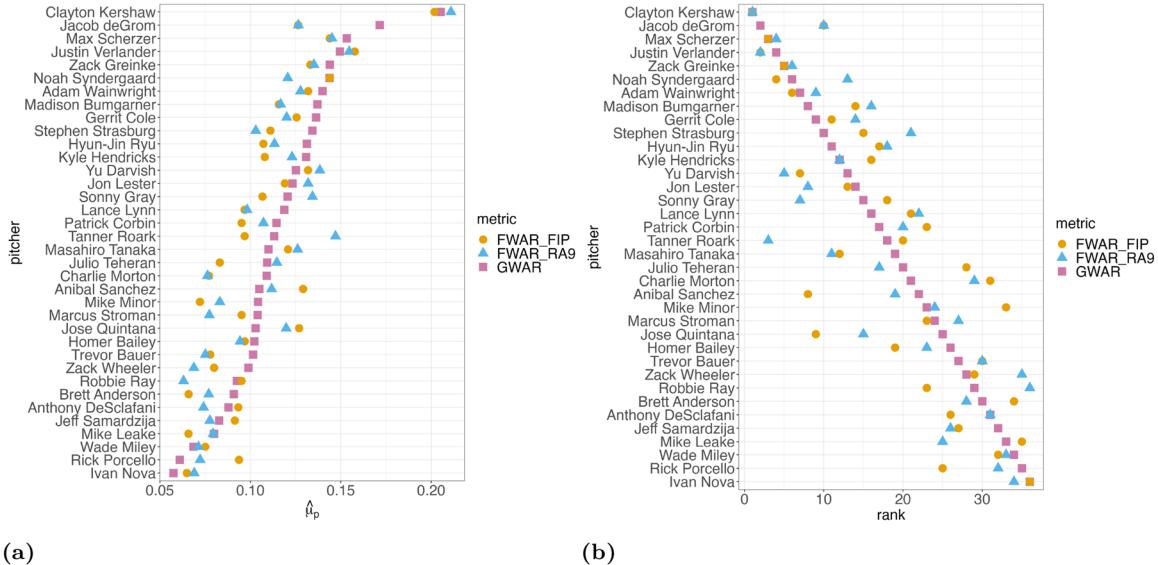


Figure 22: Pitcher quality estimates $\hat{\mu}_p$ (a) and their associated ranks \hat{r}_p (b) for a set of starting pitchers prior to the 2019 season.

GWAR is computed on the game level. Accordingly, we slightly modify the procedure from the previous section.

To begin, again index each starting pitcher by $p \in \{1, \dots, P\}$ and index pitcher p 's games by $g \in \{1, \dots, N_p\}$. Let X_{pg} denote pitcher p 's unobserved FanGraphs WAR in game g . Note that we observe his total FWAR,

$$X_p = \sum_{g=1}^{N_p} X_{pg}. \quad (30)$$

As before, we use model (19), which implies

$$\begin{aligned} X_p &\sim \mathcal{N}\left(N_p \mu_p, N_p \sigma_p^2\right), \\ \mu_p &\sim \mathcal{N}(\mu, \tau^2). \end{aligned} \quad (31)$$

Therefore the posterior mean of pitcher p 's latent pitcher quality μ_p is

$$\hat{\mu}_p := \mathbb{E}[\mu_p | X_p] = \frac{\frac{X_p}{\sigma_p^2} + \frac{\mu}{\tau^2}}{\frac{N_p}{\sigma_p^2} + \frac{1}{\tau^2}}. \quad (32)$$

This estimator is analogous to that from Equation (20), using FWAR instead of GWAR.

As before, we use a parametric Empirical Bayes approach to estimate each starting pitcher's latent quality from his FanGraphs WAR. In particular, we compute maximum likelihood estimates for μ , τ^2 , and $\{\sigma_p^2\}$ using the FanGraphs data $\{X_p\}$, which we plug in to Formula (32). We again begin finding the MLE by noting the marginal distribution of X_p according to model (31),

$$X_p \sim \mathcal{N}(N_p \cdot \mu, N_p \cdot (\tau^2 + \sigma_p^2)). \quad (33)$$

Thus the log-likelihood of the full FanGraphs dataset $\{X_p\}$ is proportional to

$$\ell(X_p | \mu, \tau^2, \sigma_p^2) \propto -\frac{1}{2} \sum_{p=1}^P \log(\tau^2 + \sigma_p^2) - \frac{1}{2} \sum_{p=1}^P \frac{\left(\frac{X_p}{N_p} - \mu\right)^2}{\tau^2 + \sigma_p^2}. \quad (34)$$

Setting the derivative of the log-likelihood with respect to μ (resp., τ^2) equal to 0 and solving for μ (resp., τ^2) yields the following equations,

$$\mu = \frac{\sum (X_p / N_p) / (\tau^2 + \sigma_p^2)}{\sum 1 / (\tau^2 + \sigma_p^2)}. \quad (35)$$

and

$$\sum_p \frac{1}{\tau^2 + \sigma_p^2} = \sum_p \frac{(X_p / N_p - \mu)^2}{(\tau^2 + \sigma_p^2)^2}. \quad (36)$$

So, in designing an iterative algorithm analogous to Algorithm 1 but for FanGraphs WAR, we replace Equations (25) (Step 1) and (27) (Step 2) with Equations (35) and (36).

Setting the derivative of the log-likelihood with respect to σ_p^2 equal to 0 and solving for σ_p^2 , however, yields a trivial equation which doesn't include σ_p^2 . Intuitively this is because we can't glean information about σ_p^2 since we don't observe the game-level FanGraphs WAR X_{pg} . Therefore, in designing an iterative algorithm analogous to Algorithm 1

but for FanGraphs WAR, we eliminate Equation (29) (Step 3) and replace σ_p^2 in Steps 1 and 2 with a constant hyperparameter σ^2 . We then choose the value of σ^2 which minimizes the

rmse between resulting estimated pitcher quality $\hat{\mu}_p$ and observed mean game FWAR in a hold-out set. We detail the full procedure in Algorithm 2.

Algorithm 2. Compute the MLE of μ and τ^2 from model (31)

```

1: procedure 1
2:   Input: FanGraphs WAR  $\{X_p : 1 \leq p \leq P\}, \epsilon, \sigma^2$ 
3:   Initialization:
4:    $\mu(t=0) = \frac{1}{P} \sum_p X_p$ 
5:    $\tau^2(t=0) = \text{Var}(\{X_p/N_p : 1 \leq p \leq P\}) - \frac{1}{P} \sigma^2$ 
6:    $t = 1$ 
7:   while TRUE do
8:     Step 1. Solve for  $\mu$  and save the result as  $\mu(t)$ : 
$$\mu = \frac{\sum_p (X_p/N_p)/(\tau^2+\sigma^2)}{\sum_p 1/(\tau^2+\sigma^2)}.$$

9:     Step 2. Solve for  $\tau^2$  (e.g., using a root finder) and save the result as  $\tau^2(t)$ : 
$$\sum_p \frac{1}{\tau^2+\sigma^2} = \sum_p \frac{(X_p/N_p-\mu)^2}{(\tau^2+\sigma^2)^2}.$$

10:    if  $|\mu(t) - \mu(t-1)| < \epsilon$  and  $|\tau^2(t) - \tau^2(t-1)| < \epsilon$  then
11:      break the while loop
12:    else
13:       $t = t + 1$ 
14:    end if
15:   end while
16:   Output:  $\hat{\mu} = \mu(t), \hat{\tau}^2 = \tau^2(t)$ 
17: end procedure
18:
19: procedure 2
20:   Input: FanGraphs WAR  $\{X_p : 1 \leq p \leq P\}, \epsilon$ 
21:   Initialization:
22:   Split  $\{X_p\}$  into a training set  $\{X_p^{(\text{train})}\}$  and a validation set  $\{X_p^{(\text{test})}\}$ 
23:   In particular,  $\{X_p^{(\text{train})}\}$  is all FWAR from 2010 to 2016,  $\{X_p^{(\text{test})}\}$  is all FWAR from 2017 to 2018
24:   Sigmas = vector of smartly chosen positive values
25:   Losses = empty vector
26:   for  $\sigma^2$  in Sigmas do
27:     Step 1. Run Procedure 1 with inputs  $\{X_p^{(\text{train})}\}, \epsilon$ , and  $\sigma^2$ 
28:     Step 2. Use  $\hat{\mu}$  and  $\hat{\tau}^2$  from Step 1 to estimate pitcher quality  $\hat{\mu}_p$  (Formula (32))
29:     Step 3. Append rmse( $\{\hat{\mu}_p, X_p^{(\text{test})}\}$ ) to Losses
30:   end for
31:   Output:  $\hat{\sigma}_p^2 \equiv \sigma^2$ , where  $\sigma^2$  corresponds to the minimum value in Losses
32: end procedure

```

Using the same dataset of starting pitchers from 2010 to 2018 as before, we run Algorithm 2, yielding maximum likelihood estimators of μ and τ^2 and an estimate of $\sigma_p^2 \equiv \sigma^2$. With $\epsilon = 10^{-5}$, the algorithm converges again after just four iterations. Then, we plug these estimators into Formula (32), yielding parametric Empirical Bayes estimators of $\{\mu_p\}$. In Figure 23 we compare these estimates $\{\hat{\mu}_p^{\text{FWAR}}\}$ to each pitcher's mean game FanGraphs WAR from 2010 to 2018. As before, for players with fewer games played (small gray dots), $\hat{\mu}_p$ is shrunk towards the overall mean μ . Also, for players with enough games played (large blue dots), $\hat{\mu}_p$ is

essentially pitcher p 's mean game FWAR, lying on the line $y = x$.

A weakness of our Empirical Bayes approach is that it assumes latent pitcher quality μ_p is constant over the decade from 2010 to 2019. Player quality is, however, non-stationary over time. Therefore, in future work we suggest using a similar Empirical Bayes approach to estimate pitcher quality, except downweighting data further back in time (e.g., using exponential decay weighting as in Medvedovsky and Patton 2022) in the posterior mean Formulas (20) and (32).

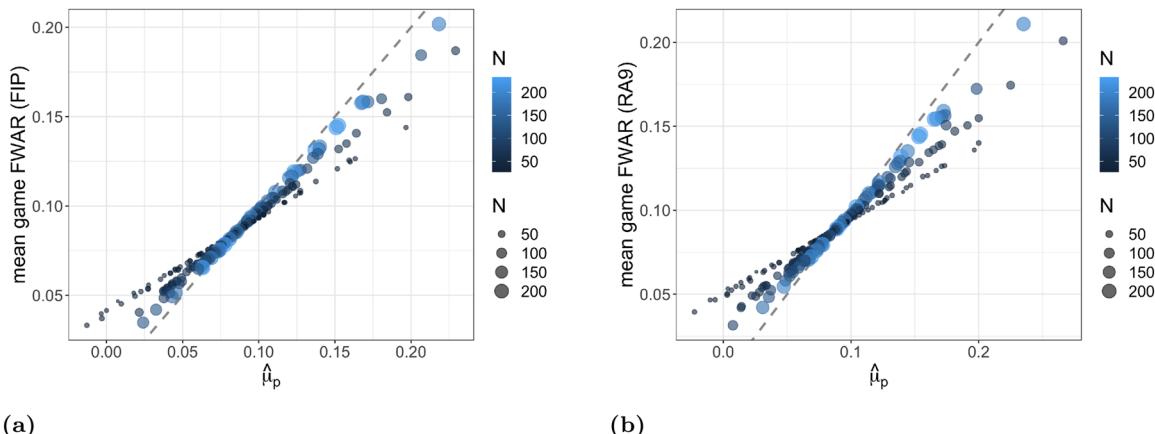


Figure 23: For starting pitchers p from 2010 to 2018, his mean game FWAR versus his Empirical Bayes estimator $\hat{\mu}_p^{\text{FWAR}}$, built from FWAR(FIP) in (a) and FWAR(RA/9) in (b). The dashed gray line is the line $y = x$.

In Figure 22 we visualize starting pitcher rankings prior to the 2019 season according to $\hat{\mu}_p$ (left) and the associated ranks \hat{R}_p (right). Clayton Kershaw has the highest $\hat{\mu}_p^{\text{FWAR}(\text{FIP})}$ and $\hat{\mu}_p^{\text{FWAR}(\text{RA/9})}$, and Ivan Nova has the lowest such values. We see that there is a nontrivial difference between pitcher quality estimates and rankings built from Grid WAR and FanGraphs WAR.

Appendix D: Estimating the park effects α

In this section, we detail why we use ridge regression to estimate park effects. First, in Section D.1, we discuss existing park factors from ESPN, FanGraphs, and Baseball Reference. Then, in Section D.2, we discuss problems with these existing park effects. In Section D.3, we introduce our park effects model, designed to yield park effects which represent the

expected runs scored in a half-inning at a ballpark above that of an average park, if an average offense faces an average defense. Then, in Sections D.4 and D.5, we conduct two simulation studies which show that ridge repression works better than other methods at estimating park effects. Then, in Section D.6, we show that our ridge park effects have better out-of-sample predictive performance than existing park effects from ESPN and FanGraphs. Finally, in Sections D.7 and 2.6, we discuss our final ridge park effects, fit on data from all half-innings from 2017 to 2019.

D.1 Existing park effects

FanGraphs, Baseball Reference, and ESPN each have runs-based park factors, which are all variations of a common formula: the ratio of runs per game at home to runs per game on the road. In particular, ESPN's park factors are an unadjusted version of this formula (ESPN 2022),

$$\alpha_{\text{ESPN}} = \frac{(\text{home runs scored} + \text{home runs allowed})/(\text{home games})}{(\text{road runs scored} + \text{road runs allowed})/(\text{road games})}. \quad (37)$$

FanGraphs modifies this formula by imposing a form of regression onto the park factors (Appelman 2016). In particular, the FanGraphs park factors $\alpha_{\text{FanGraphs}}$ are computed via

$$\begin{cases} \xi = ((\text{home runs per game}) - (\text{road runs per game})) / (\text{number of MLB teams}), \\ PF_{\text{raw}} = (\text{home runs per game}) / ((\text{road runs per game}) + \xi), \\ iPF = (PF_{\text{raw}} + 1) / 2, \\ \alpha_{\text{FanGraphs}} = 1 - (1 - iPF) \cdot w, \end{cases} \quad (38)$$

where w is a regression weight determined by the number of years in the dataset (e.g., for a three year park factor, $w = 0.8$).

Finally, Baseball Reference's park factors are a long series of adjustments on top of ESPN's park factors, computed separately for batters and pitchers (Baseball Reference 2022). In particular, Baseball Reference begins with

$$\frac{(\text{home runs scored})/(\text{home games})}{(\text{road runs scored})/(\text{road games})} \quad (39)$$

for batters and

$$\frac{(\text{home runs allowed})/(\text{home games})}{(\text{road runs allowed})/(\text{road games})} \quad (40)$$

for pitchers as base park factors. Then, they apply several adjustments on top of these base values. For instance, they adjust for the quality of the home team and the fact that the batter doesn't face its own pitchers. These adjustments, however, are a long series of convoluted calculations, so we do not repeat them here.

D.2 Problems with existing park effects

There are several problems with these existing runs-based park effects. First, ESPN and FanGraphs do not adjust for offensive and defensive quality at all, and Baseball Reference adjusts for only a fraction of team quality. It is important to adjust for team quality in order to de-bias the park factors. For example, the Colorado Rockies play in the NL West, a division with good offensive teams such as the Dodgers, Giants, and Padres. So, by ignoring offensive quality in creating park factors, the Rockies' park factor may be an overestimate, since many of the runs scored at their park may be due to the offensive power of the NL West rather than the park itself. By ignoring team quality, the ESPN and FanGraphs park factors are biased. Baseball Reference's park factors adjust for the fact that a team doesn't face its own pitchers, albeit through a convoluted series of ad-hoc calculations. Although adjusting for not facing one's own pitchers slightly de-biases the park factors, it does not suffice as a full adjustment of the offensive and defensive quality of a team's schedule.

Second, these existing runs-based park effects do not come from a statistical model. This makes it more difficult to quantitatively measure which park factors are the "best", for instance via some loss function. In other words, it is more difficult to quantitatively know that Baseball Reference's park factors are actually more accurate than FanGraphs', in some mathematical sense, besides that it claims to adjust for some biases in its derivation, although we discuss a way to

do so in Section D.6. Another benefit of a statistical model is that it will allow us to adjust for the offensive and defensive quality of a team and its opponents simultaneously. Finally, a statistical model will give us a firm physical interpretation of the park factors.

Hence, in this paper, we create our own park factors, which are the fitted coefficients of a statistical model that adjusts for team offensive and defensive quality.

D.3 Our park effects model

In this section, we introduce our park effects model, designed to yield park effects which represent the expected runs scored in a half-inning at a ballpark above that of an average park, if an average offense faces an average defense.

We index each half-inning in our dataset by i , each park by j , and each team-season by k . We define the park matrix \mathbf{P} so that \mathbf{P}_{ij} is 1 if the i th half-inning is played in park j , and 0 otherwise. Similarly, we define the offense matrix \mathbf{O} so that \mathbf{O}_{ik} is 1 if the k th team-season is on offense during the i th half-inning, and 0 otherwise, and define the defense matrix \mathbf{D} so that \mathbf{D}_{ik} is 1 if the k th team-season is on defense during the i th half-inning, and 0 otherwise. We denote the runs scored during the i th half-inning by y_i . Then, we model y_i using a linear model,

$$y_i = \beta_0 + \sum_{j \neq \text{ANA}} \mathbf{P}_{ij} \beta_j^{(\text{park})} + \sum_{k \neq \text{ANA2017}} \mathbf{O}_{ik} \beta_k^{(\text{off})} \\ + \sum_{k \neq \text{ANA2017}} \mathbf{D}_{ik} \beta_k^{(\text{def})} + \epsilon_i, \quad (41)$$

where ϵ_i is mean-zero noise,

$$\mathbb{E}[\epsilon_i] = 0. \quad (42)$$

Succinctly, we model

$$y_i = \mathbf{X}_{i*} \boldsymbol{\beta} + \epsilon_i, \quad (43)$$

where

$$\mathbf{X} = [1, \mathbf{P}, \mathbf{O}, \mathbf{D}] \quad (44)$$

and

$$\boldsymbol{\beta}^\top = (\beta_0, \beta^{(\text{park})\top}, \beta^{(\text{off})\top}, \beta^{(\text{def})\top}). \quad (45)$$

The coefficients are fitted relative to the first park ANA (the Anaheim Angels) and relative to the first team-season ANA2017 (the Angels in 2017). By including distinct coefficients for each offensive-team-season and each defensive-team-season, we adjust for offensive and defensive quality simultaneously in fitting our park factors. Finally, in order to make our park effects represent the expected runs scored

in a half-inning at a ballpark above that of an average park, we subtract the mean park effect from each park effect,

$$\beta^{(park)} \leftarrow \beta^{(park)} - \frac{1}{n} \sum_{j=1}^n \beta_j^{(park)}. \quad (46)$$

D.4 First simulation study

We have a park effects model, Formula (41), but it is not immediately obvious which algorithm we should use to fit the model. In particular, due to multicollinearity in the observed data matrix \mathbf{X} , ordinary least squares is sub-optimal. Hence we run a simulation study in order to test various methods of fitting model (41), using the method which best recovers the “true” simulated park effects as the park factor algorithm to be used in computing Grid WAR: ridge regression.

Simulation setup. In our first simulation study, we assume that the park, team offensive quality, and team defensive quality coefficients are independent. Specifically, we simulate 25 “true” parameter vectors $\{\beta^{[m]}\}_{m=1}^{25}$ according to

$$\begin{cases} \beta_0 = 0.4, \\ \beta_j^{(park)\text{iid}} \sim \mathcal{N}(0.04, 0.065), \\ \beta_k^{(\text{off})\text{iid}} \sim \mathcal{N}(0.02, 0.045), \\ \beta_k^{(\text{def})\text{iid}} \sim \mathcal{N}(0.03, 0.07). \end{cases} \quad (47)$$

Then, we assemble our data matrix \mathbf{X} to consist of every half-inning from 2017 to 2019. Then, we simulate 25 “true” outcome vectors $\{y^{[m]}\}_{m=1}^{25}$ according to

$$y_i = \text{Round}(\mathcal{N}_+(\mathbf{X}_{i*}\beta, 1)). \quad (48)$$

We use a truncated normal distribution, denoted by \mathcal{N}_+ , in order to make y_i positive. We round y_i so that it is a positive integer, since y_i represents the runs scored in the i th inning. Although we don’t directly simulate ϵ_i , our simulated y_i still adheres to model (43), as it has mean $\mathbf{X}_{i*}\beta$. We choose the values in Formula (47) so that the simulated outcome vectors $\{y^{[m]}\}_{m=1}^{25}$ seem reasonable in representing the runs scored in a half-inning.

Our goal is to recover the park effects $\beta^{(park)}$, so our evaluation metric of an estimator $\hat{\beta}$ is the average simulation error of the fitted park effects,

$$\frac{1}{25} \sum_{m=1}^{25} \|\hat{\beta}^{(park)[m]} - \beta^{(park)[m]}\|_2. \quad (49)$$

Note that it doesn’t make sense to compare the existing ESPN and FanGraphs park factors to park effects methods based on model (41) as part of this simulation study because

they are not based on model (41). In fact, ESPN and FanGraphs park effects are not based on any statistical model. Rather, in Section D.6, we separately compare these existing park factors to our model-based park factors.

Method 1: OLS without adjusting for team quality. The naive method of estimating the park factors $\beta^{(park)}$ is ordinary least squares regression while ignoring team offensive quality and team defensive quality, as done in Baumer et al. (2015, Formula 11). In other words, fit the park coefficients using OLS on the following model,

$$y_i = \beta_0 + \sum_{j \neq \text{ANA}} \mathbf{P}_{ij} \beta_j^{(park)} + \epsilon_i. \quad (50)$$

In failing to adjust for offensive and defensive quality, we expect this algorithm to perform poorly.

Method 2: OLS. Next, we adjust for offensive quality, defensive quality, and park simultaneously using ordinary least squares (OLS) on model (41). This method is similar to that from Acharya et al. (2008), although they compute game-level park factors and we compute half-inning-level park factors. This yields an unbiased estimate of the park effects, and so we expect this method to perform better than the previous one.

Method 3: Three-Part OLS. Although OLS using model (41) is unbiased, the fitted coefficients have high variance due to the multicollinearity of the data matrix \mathbf{X} . In particular, the park matrix \mathbf{P} is correlated with the offensive team matrix \mathbf{O} and the defensive team matrix \mathbf{D} because in each half-inning, either the team on offense or defense is the home team. We may visualize the collinearity in \mathbf{X} by denoting all of the half-innings (rows) in which the road team is batting by road, denoting the half-innings in which the home team is batting by home, and writing \mathbf{X} (e.g., for one season of data) as

$$\mathbf{X} = \begin{bmatrix} 1 & \mathbf{P}_{\text{road},*} & \mathbf{O}_{\text{road},*} & \mathbf{P}_{\text{road},*} \\ 1 & \mathbf{P}_{\text{home},*} & \mathbf{P}_{\text{home},*} & \mathbf{D}_{\text{home},*} \end{bmatrix}. \quad (51)$$

To address this collinearity issue, we propose a three-part OLS algorithm. First, we estimate the offensive quality coefficients during half-innings in which the road team is batting. We do so via OLS on the following model,

$$y_{\text{road}} = \beta_0 1 + \mathbf{P}_{\text{road},*} \beta^{(\text{park}-\text{def})} + \mathbf{O}_{\text{road},*} \beta^{(\text{off})} + \epsilon. \quad (52)$$

This yields a decent estimate $\hat{\beta}^{(\text{off})}$ of $\beta^{(\text{off})}$, in particular because for one season of innings, $\mathbf{P}_{\text{road},*} = \mathbf{D}_{\text{road},*}$, and for multiple seasons of innings, $\mathbf{P}_{\text{road},*} \simeq \mathbf{D}_{\text{road},*}$.

Second, we estimate the defensive quality coefficients during half-innings in which the home team is batting. We do so via OLS using the following model,

$$y_{\text{home}} = \beta_0 1 + \mathbf{P}_{\text{home},*} \beta^{(\text{park}-\text{off})} + \mathbf{D}_{\text{home},*} \beta^{(\text{def})} + \epsilon. \quad (53)$$

This yields a decent estimate $\hat{\beta}^{(\text{def})}$ of $\beta^{(\text{def})}$, in particular because $\mathbf{P}_{\text{home},*} \simeq \mathbf{O}_{\text{home},*}$.

Third, we use the fitted team quality coefficients $\hat{\beta}^{(\text{off})}$ and $\hat{\beta}^{(\text{def})}$ on all half-innings to obtain the park effects. Specifically, we run OLS on the following model,

$$y_i = \beta_0 + \beta_1 \cdot \sum_{k \neq \text{ANA}2017} \mathbf{O}_{ik} \hat{\beta}_k^{(\text{off})} + \beta_2 \cdot \sum_{k \neq \text{ANA}2017} \mathbf{D}_{ik} \hat{\beta}_k^{(\text{def})} + \sum_{j \neq \text{ANA}} \mathbf{P}_{ij} \beta_j^{(\text{park})} + \epsilon_i, \quad (54)$$

yielding fitted park coefficients $\hat{\beta}^{(\text{park})}$.

Method 4: Ridge. Finally, we use ridge regression to fit model (41). In the presence of multicollinearity, ridge regression coefficient estimates may improve upon OLS estimates by introducing a small amount of bias in order to reduce the variance of the estimates (Hoerl and Kennard 1970). We tune the ridge parameter λ using cross validation.

Simulation Results. Recall from the start of this section that we simulate 25 sets of “true” parameters $\{\beta^{[ml]}\}_{m=1}^{25}$ and 25 “true” outcome vectors $\{y^{[ml]}\}_{m=1}^{25}$. Then, using the observed data matrix \mathbf{X} , which consists of all half-innings from 2017 to 2019, we run each of our five methods, yielding parameter estimates, and evaluate them using the average simulation error from Formula (49). We report the results in Table 7.

The OLS estimator without adjusting for team quality performs worst, as ignoring team quality leads to a biased estimate of the park effects. OLS and three-part OLS, which include proper adjustments for team quality, perform similarly and are second best. Three-part OLS turns out not to be an improvement over OLS because despite the multicollinearity, there is enough linear independence between the batting-road and batting-home half-innings to obtain reasonably accurate team quality estimates. Also, if \mathbf{X} contains multiple years worth of data, three-part OLS leads to slightly biased estimates of $\beta^{(\text{off})}$ and $\beta^{(\text{def})}$ since steps one and two only adjust for park. Additionally, three-part OLS uses half as much data to estimate team quality, which is significant because the outcome variable inning runs is

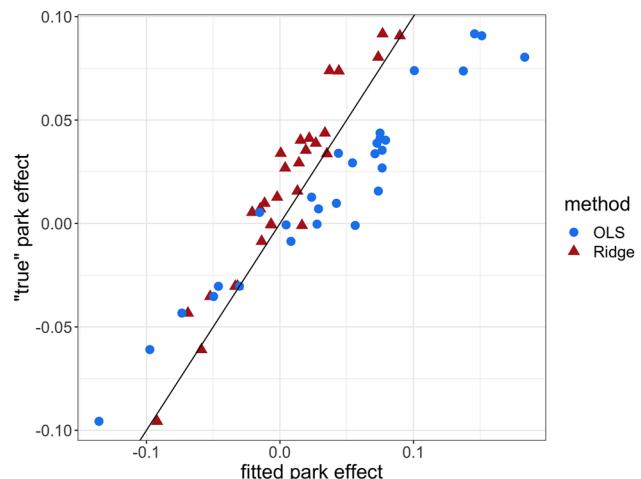


Figure 24: For one of 25 simulations from our first simulation study from Section D.4, plot the “true” park effects against the ridge estimates and the OLS estimates. The line $y = x$, shown in black, represents a perfect fit between the “true” and fitted park effects. The OLS estimates are biased, whereas the ridge estimates lie more evenly around the line $y = x$.

so noisy. Lastly, ridge regression performs the best, and is significantly better than OLS. In Figure 24, we visualize one of the 25 simulations by plotting the “true” park effects against the ridge estimates and the OLS estimates. We see that OLS is biased, whereas ridge lies more evenly around the line $y = x$.

D.5 Second simulation study

A primary criticism of the first simulation study from Section D.4 is that in actual baseball, the offensive and defensive quality coefficients are not independent. Rather, often times offensive and defensive qualities are correlated within MLB divisions. For instance, in 2021, the Rays, Red Sox, Yankees, and Blue Jays of the AL East each had at least 91 wins, and so were all good offensive teams. Correlated offensive and defensive qualities within divisions introduces additional collinearity into the data matrix \mathbf{X} , since teams play other teams within their division at a disproportionately high rate.

Another criticism of the first simulation study is that it treats Colorado’s park effect as a draw from the same distribution as the other park effects, whereas in real life we know Colorado’s park effect is an outlier as a result of the high altitude.

So, in this section, we conduct a second simulation study which incorporates intra-divisional collinearity and forces Colorado’s park effect to be an outlier. Specifically, we simulate 25 “true” parameter vectors $\{\beta^{[ml]}\}_{m=1}^{25}$ according to

Table 7: Results of our first simulation study.

Method	Simulation error
Ridge	0.0244
OLS	0.0326
Three-part OLS	0.0330
OLS without adjusting for team quality	0.0610

The bold value indicates the best method that has the lowest error.

$$\left\{ \begin{array}{l} \beta_0 = 0.15, \\ \beta_j^{(\text{park} \neq \text{DEN})} \stackrel{\text{iid}}{\sim} \mathcal{N}(0.04, 0.065), \\ \beta_j^{(\text{park} = \text{DEN})} = 0.32, \\ \beta_k^{(\text{div}, \text{off})} \stackrel{\text{iid}}{\sim} \mathcal{N}(0.02, 0.05), \\ \beta_k^{(\text{off})} \stackrel{\text{iid}}{\sim} \mathcal{N}(\beta^{(\text{div}, \text{off})}, 0.02), \\ \beta_k^{(\text{div}, \text{def})} \stackrel{\text{iid}}{\sim} \mathcal{N}(0.03, 0.0), \\ \beta_k^{(\text{def})} \stackrel{\text{iid}}{\sim} \mathcal{N}(\beta^{(\text{div}, \text{def})}, 0.033), \end{array} \right. \quad (55)$$

where DEN refers to Coors Field in Denver, Colorado. In other words, each division has its own offensive and defensive quality means, coercing the offensive and defensive qualities within each division to be correlated. We choose the values in Formula (55) so that the simulated outcome vectors $\{y^{[m]}\}_{m=1}^{25}$ seem reasonable in representing the runs scored in a half-inning.

Because Colorado is an outlier in this simulation study, we judge our methods based on two loss functions: the Colorado park effect average simulation error,

$$\frac{1}{25} \sum_{m=1}^{25} \left| \hat{\beta}_{j=\text{DEN}}^{(\text{park})[m]} - \beta_{j=\text{DEN}}^{(\text{park})[m]} \right|, \quad (56)$$

and the non-Colorado park effect average simulation error,

$$\frac{1}{25} \sum_{m=1}^{25} \left\| \hat{\beta}_{j \neq \text{DEN}}^{(\text{park})[m]} - \beta_{j \neq \text{DEN}}^{(\text{park})[m]} \right\|_2. \quad (57)$$

The remainder of the second simulation study proceeds identically to the first simulation study from the previous section.

We report the results of our second simulation study in Table 8. Again, ridge regression performs the best. In particular, ridge performs significantly better than the other methods on the outlier Colorado, and better than the other methods on the other parks. In Figure 25, we visualize one of the 25 simulations by plotting the “true” park effects against the ridge estimates and the OLS estimates. We see that

Table 8: Results of our second simulation study.

Method	Non-Colorado simulation error	Colorado error
Ridge	0.0442	0.0502
OLS	0.0483	0.1791
Three-part OLS	0.0490	0.1807
OLS without adjusting for team quality	0.0751	0.1893

The bold value indicates the best method that has the lowest error.

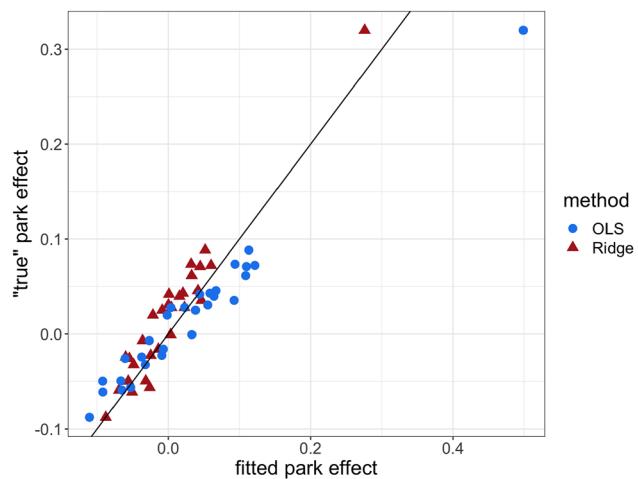


Figure 25: For one of the 25 simulations from our second simulation study from Section D.5, plot the “true” park effects against the ridge estimates and the OLS estimates. The line $y = x$, shown in black, represents a perfect fit between the “true” and fitted park effects. The OLS estimates are biased, whereas the ridge estimates lie more evenly around the line $y = x$. In particular, ridge regression much better captures the park effect of the outlier, Denver.

ridge regression successfully fits the Colorado park effect, whereas OLS significantly overestimates Colorado. Colorado as an outlier in OLS exerts high leverage over the rest of the park effects, swaying their estimates upwards. One might suggest removing the outlier Colorado from the dataset and estimating it separately, but doing weakens the estimates of the other teams in its division as a result of removing too many games from the set schedule which determines the data matrix X . So, in both the first simulation study from Section D.4 and the second simulation study from this section, ridge regression most successfully estimates the “true” simulated park effects.

D.6 Comparing existing park effects and ridge park effects

Now, in this section, we compare our ridge park factors, which perform best in our simulation studies from Sections D.4 and D.5, to existing park effects from ESPN and FanGraphs.

Transforming ESPN and FanGraphs park factors to an “additive” scale. Our ridge and OLS park effects of a ballpark, based on model (41), are “additive” in the sense that they represent the expected runs scored in a half-inning at that park above that of an average park, if an average offense faces an average defense. On the other hand, ESPN and FanGraphs park factors, defined in Formulas (37) and (38), are “multiplicative” in the sense that they represent the ratio of runs created at home to runs created on the road.

Therefore, in order to compare these existing park factors to our park factors, we need to put them on the same scale. In particular, we transform the ESPN and FanGraphs park effects into “additive” park effects. To do so, we take the mean runs scored in a half-inning,

$$\bar{y} = 0.5227, \quad (58)$$

and multiply it by a “multiplicative” park factor subtracted by 1. For example, if the ESPN Colorado park factor α in 2019 is 1.34, representing that teams score 34 % more runs in Colorado than in other parks, then the transformed “additive” ESPN park factor is

$$(\alpha - 1) \cdot \bar{y} = (0.34) \cdot (0.5227) = 0.178. \quad (59)$$

After this transformation, the ESPN and FanGraphs park factors also represent the expected runs scored in a half-inning at that park above that of an average park.

Visualizing these park effect methods. In Figure 26 we visualize the ridge, OLS, ESPN, and FanGraphs park effects (where the latter two are transformed to an “additive” scale), fit on all half-innings from 2017 to 2019. We use the park abbreviations from Retrosheet, our data source, as discussed in Section 2.2. As expected, we see that ridge park factors are a shrunk version of OLS park factors, and FanGraphs park factors are a shrunk version of ESPN park factors. The FanGraphs and ridge park factors are remarkably similar. Also, as expected, Coors Field (DEN02) has the largest park effect for all four methods. The Texas Ranger’s

ballpark in Arlington (ARL02) has the second highest park effect for all four methods. These two parks have signifas the lowest park effect for all four methods.

Additionally, we visualize how these various park effects impact the Grid WAR of various starting pitchers. In Figure 27, we show the 2019 seasonal Grid WAR for a set of starters without park effects, with ridge park effects, and with (transformed) ESPN park effects. For most pitchers, the impact of including a park adjustment is small. For some pitchers, the impact of an ESPN park adjustment is massive. For instance, the GWAR of Mike Minor and Lance Lynn, who pitched for the Rangers in 2019, each increases by a staggering one whole WAR. Ridge park factors have a much more muted impact than ESPN park factors. This makes sense, as ridge regression shrinks the park coefficients closer to 0. For a few pitchers, however, even the ridge park effects make a nontrivial impact on their GWAR. This also makes sense, as some park effects, such as those of the Mets, Rockies, and Rangers, are far enough from zero. For instance, the GWAR of Noah Syndergaard and Jacob deGrom, who pitched for the Mets, each decreases by about one-quarter of a WAR.

Comparing these park effects quantitatively. In deciding which park effects to use in our final Grid WAR calculations, we quantitatively compare the ridge, OLS, ESPN, and FanGraphs park factors. In particular, we compare the out-of-sample predictive performance of these park effects.

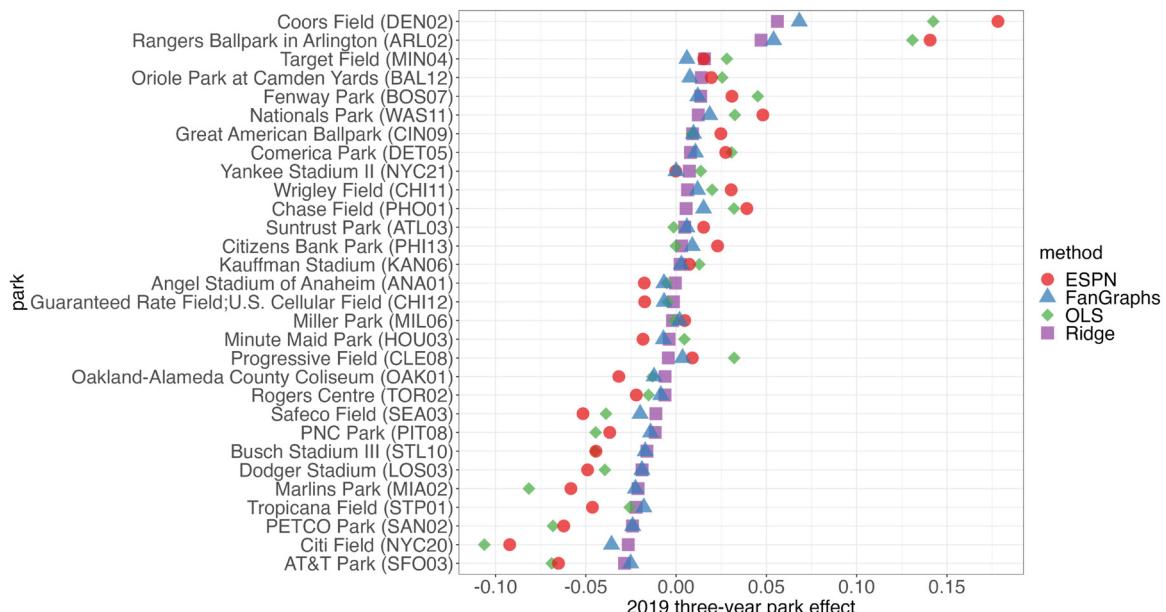


Figure 26: Ridge, OLS, ESPN, and FanGraphs 2019 three-year park effects (where the latter two are transformed to an “additive” scale). The abbreviations are the retrosheet ballpark codes.

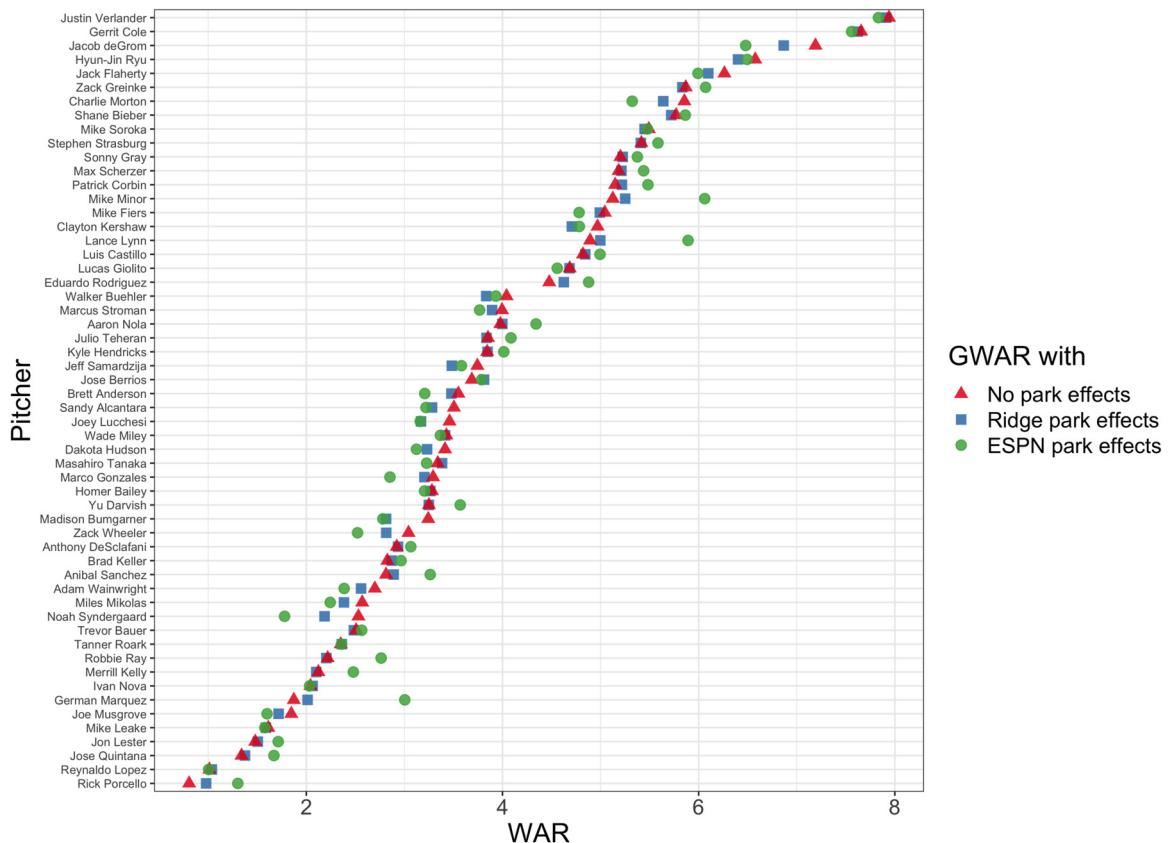


Figure 27: Grid WAR on a set of pitchers from 2019 without park effects (red triangles), with ridge park effects (blue squares), and with ESPN park effects (green circles).

We begin by fitting each park factor method using data from all half-innings from 2014 to 2016. Note that the OLS and ridge park factors adjust for team offensive and defensive quality, whereas the ESPN and FanGraphs park factors don't. So, in order to fairly quantitatively compare which of these park factor methods is “best”, we adjust for team quality on all of these methods. Specifically, using the fitted park factors $\hat{\alpha}$ from 2014 to 2016 for a given method, we regress out team offense and team defense indicators via OLS on the following model, where \mathbf{P} , \mathbf{O} , and \mathbf{D} are the data matrices consisting of all half-innings from 2017 to 2019,

$$\begin{aligned} y_i = \beta_0 + \beta_1 \sum_{j \neq \text{ANA}} \mathbf{P}_{ij} \hat{\alpha}_j + \cdot \sum_{k \neq \text{ANA2017}} \mathbf{O}_{ik} \beta_k^{(\text{off})} \\ + \cdot \sum_{k \neq \text{ANA2017}} \mathbf{D}_{ik} \beta_k^{(\text{def})} + \epsilon_i. \end{aligned} \quad (60)$$

Then, using these adjusted models based on Formula (60), we predict the expected runs scored \hat{y}_i in each half-inning i from 2017 to 2019, which are out-of-sample predictions relative to the park effects $\hat{\alpha}$ which were estimated on data from 2014 to 2016. Finally, we compute the out-of-sample RMSE.

Each of the four methods has the same out-of-sample RMSE, 1.504. For reference, the RMSE of the overall mean is 1.508, so park factors do improve prediction, albeit slightly. But, because the runs scored in a half-inning is so noisy, and because the differences across parks are so slight, out-of-sample RMSE isn't sensitive enough to quantitatively show which park factors have the best predictive performance.

To more clearly understand the differences in methods, we calculate the ecological RMSE to quantitatively compare the various park factor methods. This is done by first fixing a ballpark p . Then for each park factor method, we take the mean of the vector of predicted runs scored in each half-inning at the given park p , yielding $\hat{\bar{y}}_p$. Then we find the mean of the vector of observed runs scored in each half-inning at that park p , yielding \bar{y}_p . Finally, we compute the RMSE of the regression of (\bar{y}_p) on $(\hat{\bar{y}}_p)$ (each vectors of length 30), yielding the out-of-sample ecological RMSE. In Table 9 we show the out-of-sample ecological RMSE for several park factor methods. The ridge park effects perform best, outperforming the FanGraphs and ESPN park factors, mainly since Ridge adjusts for offensive and defensive quality. The ridge park effects outperform the OLS park effects

Table 9: Out-of-sample ecological RMSE of predicting the runs scored in a half-inning using various park effect methods.

Park effect	Ecological RMSE
Ridge	0.01516
FanGraphs	0.01578
ESPN	0.01579
OLS	0.01672
Overall mean \bar{y}	0.04658

The bold value indicates the best method that has the lowest error.

for the same reasons discussed in our simulation studies from Sections D.4 and D.5.

Hence the ridge regression park effects lead to better out-of-sample predictions of the runs scored in a half-inning than existing park factor methods from ESPN and FanGraphs.

D.7 2019 three-year park effects

As discussed in Sections D.4 and D.5, the ridge park effects outperform other park effect methods in two simulation studies. Further, as discussed in Section D.6, ridge park effects outperform existing park effect methods from ESPN and FanGraphs. Therefore, we use ridge park effects in computing Grid WAR. In particular, we use ridge regression on our observed dataset $\{y, X\}$ consisting of all half-innings from 2017 to 2019, tuning the ridge parameter λ using cross

validation, to fit our park effects, shown in Figure 3 in Section 2.6.

Appendix E: Grid WAR across modern baseball history

We visualize the top 15 starting pitcher-seasons of all time by total Grid WAR in Figure 28a. The pitcher with the highest total Grid WAR of all time in a single season is Sandy Koufax and it is not even close. In 1966, he accumulated 11.54 GWAR over 41 games in his final season which is half a game more GWAR than the second best season (Bob Gibson had 11.05 Grid WAR over 34 games in 1968) and the third best (Dwight Gooden had 11.04 Grid WAR over 35 games in 1985). Koufax's 1966 season is an illuminating example of the value of Grid WAR compared to standard formulations. While his 1966 is the standout season of all time in terms of Grid WAR, it is just the sixth highest seasonal FWAR(RA/9) and the 20th highest seasonal FWAR(FIP). The other methods incorrectly overweight his three outlying blow-up games (i.e. less than -0.1 GWAR). This is an excellent example of why it is a philosophical mistake to ignore variance and convexity. In 1966, there were two versions of Koufax: the “left arm of God” or worse than replacement. The “left arm of God” threw eight complete game shutouts and 9 one-run complete games. Grid WAR properly accounts for this variation while the standard metrics do not. Among the 15 all time best

#	Year	Pitcher	Team	GWAR	GWAR per start	Num Games
1	1966	Sandy Koufax	LAN	11.543	0.282	41
2	1968	Bob Gibson	SLN	11.045	0.325	34
3	1985	Dwight Gooden	NYN	11.039	0.315	35
4	1997	Roger Clemens	TOR	10.970	0.323	34
5	1972	Steve Carlton	PHI	10.712	0.261	41
6	1953	Robin Roberts	PHI	10.429	0.254	41
7	1963	Sandy Koufax	LAN	10.405	0.260	40
8	1978	Ron Guidry	NYA	10.332	0.295	35
9	2000	Pedro Martinez	BOS	10.294	0.355	29
10	1972	Gaylord Perry	CLE	9.997	0.250	40
11	1964	Dean Chance	LAA	9.782	0.279	35
12	1971	Wilbur Wood	CHA	9.733	0.232	42
13	1971	Tom Seaver	NYN	9.670	0.276	35
14	1971	Vida Blue	OAK	9.670	0.248	39
15	1965	Sandy Koufax	LAN	9.595	0.234	41

(a)

#	Year	Pitcher	Team	GWAR	GWAR per start	Num Games
1	2000	Pedro Martinez	BOS	10.294	0.355	29
2	1994	Greg Maddux	ATL	8.244	0.330	25
3	1995	Greg Maddux	ATL	9.196	0.328	28
4	1968	Bob Gibson	SLN	11.045	0.325	34
5	1997	Roger Clemens	TOR	10.970	0.323	34
6	1985	Dwight Gooden	NYN	11.039	0.315	35
7	1999	Pedro Martinez	BOS	9.098	0.314	29
8	1953	Warren Spahn	MLN	8.940	0.298	30
9	1978	Ron Guidry	NYA	10.332	0.295	35
10	2014	Clayton Kershaw	LAD	7.690	0.285	27
11	2015	Zack Greinke	LAD	9.046	0.283	32
12	1966	Sandy Koufax	LAN	11.543	0.282	41
13	1990	Roger Clemens	BOS	8.750	0.282	31
14	1997	Randy Johnson	SEA	8.127	0.280	29
15	1964	Dean Chance	LAA	9.782	0.279	35

(b)

Figure 28: The top 15 starting pitcher-seasons post 1951 by total Grid WAR (a) and Grid WAR per game (b).

seasons, no other pitcher appears more than once, while Koufax appears on the list three times (1963, 1965, 1966) with his 1964 season only falling short because Koufax lost a quarter of the season due to injury. Koufax's "duality" is not just chance variation, it is a systematic attribute and a significant contributor to his early retirement after the 1966 season.

Grid WAR certainly favors pitchers that start more often and pitch deeper into the game. Consequently, all of the top pitcher-seasons of all time, according to cumulative Grid WAR, occurred prior to 2001. Moreover, no pitcher-season since 2001 has eclipsed 9.1 GWAR (Greinke in 2009 has the highest at 9.068). Just five pitcher seasons since 2009 have eclipsed 8.5 GWAR (Weaver in 2011, deGrom in 2018, Arrieta in 2015, and Greinke in 2009 and 2015), and only two pitcher-seasons since 2016 have eclipsed eight GWAR (deGrom in 2018 and Verlander in 2019). Starting pitchers are fundamentally less valuable today because top starters used to pitch many more games per season and more innings per game. Koufax started every fourth game in 1966, whereas high-end pitchers today start in every fifth game. Furthermore, the complete game is a thing of the past. Today, starters are typically removed prior to the seventh inning as a reaction to the time through the order penalty (Brill et al. 2023).

We visualize the top 15 starting pitcher-seasons of all time by Grid WAR per game in Figure 28b. Today's pitchers fare substantially better on a per-start basis. The most efficient individual pitcher-season of all time, subject to starting at least 25 games, is Pedro Martinez who in 2000

finished with 0.355 Grid WAR over 29 games. Greg Maddux's 1994 season is second all-time, with 0.330 Grid WAR over 25 games. Greg Maddux has third place too, with 0.328 Grid WAR over 28 games in 1995. These seasons were remarkable in terms of consistency. Pedro had zero blow-up games in 2000. All but two of his games resulted in positive GWAR, and his worst game featured just -0.068 GWAR. Similarly, across 1994 and 1995, Maddux had just five negative GWAR games in 53 starts, with at worst -0.13 GWAR in a game.

We visualize the top 15 starting pitchers of all time by career Grid WAR per game in Figure 29b. The most efficient starter of all time over his career (minimum of 100 games) is Clayton Kershaw who currently averages 0.208 Grid WAR per start over 410 games. Jacob deGrom is a close second with 0.206 over his 215 games. Pedro Martinez is third with an average of 0.201 over 410 games. Interestingly, only two of the top 10 pitchers of all time by career efficiency played their entire careers prior to 2000 (the exceptions are Whitey Ford and Sandy Koufax).

We visualize the top 15 starting pitchers of all time by total career Grid WAR in Figure 29a. The pitcher with the most career Grid WAR is Roger Clemens, who accumulated 139.8 GWAR across 24 seasons from 1984 to 2007. The starter with the second highest career Grid WAR, Greg Maddux, doesn't come close to Clemens, as he accumulated 115.6 GWAR over 23 seasons from 1986 to 2008. Three of the top ten starters according to career GWAR (Clemens, Maddux, and Randy Johnson) played in the 1980s, 90s, and 00s, and the remaining seven (Tom Seaver, Nolan Ryan, Don Sutton, Gaylord Perry, Bert Blyleven, Steve Carlton, and Jim Palmer)

#	Pitcher	GWAR	GWAR per start	Num Games	Seasons
1	Roger Clemens	139.779	0.198	707	1984-2007
2	Greg Maddux	115.617	0.156	740	1986-2008
3	Tom Seaver	108.610	0.168	647	1967-1986
4	Nolan Ryan	105.442	0.136	773	1966-1993
5	Randy Johnson	104.795	0.174	603	1988-2009
6	Don Sutton	100.065	0.132	756	1966-1988
7	Gaylord Perry	99.069	0.144	690	1962-1983
8	Bert Blyleven	97.706	0.143	685	1970-1992
9	Steve Carlton	93.345	0.132	709	1965-1988
10	Jim Palmer	91.162	0.175	521	1965-1984
11	Tom Glavine	90.700	0.133	682	1987-2008
12	Tommy John	87.532	0.125	700	1963-1989
13	Phil Niekro	86.886	0.122	714	1965-1987
14	Mike Mussina	86.475	0.161	536	1991-2008
15	Clayton Kershaw	85.736	0.208	412	2008-2023

(a)

#	Pitcher	GWAR	GWAR per start	Num Games	Seasons
1	Clayton Kershaw	85.736	0.208	412	2008-2023
2	Jacob deGrom	44.219	0.206	215	2014-2023
3	Pedro Martinez	82.305	0.201	410	1992-2009
4	Roger Clemens	139.779	0.198	707	1984-2007
5	Whitey Ford	78.460	0.185	425	1953-1967
6	Johan Santana	52.452	0.185	284	2000-2012
7	Sandy Koufax	57.567	0.183	314	1955-1966
8	Chris Sale	45.460	0.180	253	2012-2023
9	Roy Halladay	69.822	0.179	390	1998-2013
10	Walker Buehler	18.561	0.177	105	2018-2022
11	Shane Bieber	22.061	0.176	125	2018-2023
12	Jim Palmer	91.162	0.175	521	1965-1984
13	Justin Verlander	85.723	0.175	489	2005-2023
14	Randy Johnson	104.795	0.174	603	1988-2009
15	Max Scherzer	73.536	0.171	430	2008-2023

(b)

Figure 29: The top 15 starting pitchers post 1951 by career Grid WAR (a) and career Grid WAR per game (b).

played in the 1960s, 70s, and 80s. Just five of the top 25 starters by career Grid WAR, on the other hand, began their career after 2000. The best such starters are Justin Verlander (85.7 GWAR over 19 seasons from 2005 to 2003) and Clayton Kershaw (85.1 GWAR over 16 seasons from 2008 to 2023). These are the 15th and 16th ranked pitchers, and they don't come anywhere particularly close to Clemens or even Maddux, although their careers aren't over. As discussed before,

the pitchers with the highest career Grid WAR come from the previous millennium because top starters back then pitched more games per season and pitched more innings per game.

Finally, we measure a starters peak performance by the maximum total Grid WAR across all his contiguous four year stretches. We combine a starter's peak ranking with his ranking by total career Grid WAR using the geometric

#	Pitcher	Peak WAR	Total WAR	HOF	Rank Career	Rank Peak	Geo Mean Rank
1	Greg Maddux	33.23	115.60	BBWAA	2	4	2.828
2	Randy Johnson	35.47	104.80	BBWAA	5	2	3.162
3	Roger Clemens	30.72	139.80		1	11	3.317
4	Tom Seaver	31.88	108.60	BBWAA	3	6	4.243
5	Jim Palmer	33.04	91.16	BBWAA	10	5	7.071
6	Pedro Martinez	34.80	82.30	BBWAA	18	3	7.348
7	Sandy Koufax	38.69	57.57	BBWAA	57	1	7.550
8	Gaylord Perry	30.19	99.07	BBWAA	7	13	9.539
9	Clayton Kershaw	31.16	85.56		15	7	10.250
10	Kevin Brown	31.10	80.71		19	8	12.330
11	Bert Blyleven	28.09	97.71	BBWAA	8	21	12.960
12	Don Sutton	26.74	100.10	BBWAA	6	28	12.960
13	Bob Gibson	30.68	82.32	BBWAA	17	12	14.280
14	Nolan Ryan	24.00	105.40	BBWAA	4	51	14.280
15	Justin Verlander	28.84	85.45		16	17	16.490
16	Juan Marichal	30.95	70.09	BBWAA	29	10	17.030
17	Jim Bunning	31.05	67.63	Veterans	34	9	17.490
18	Steve Carlton	25.86	93.35	BBWAA	9	35	17.750
19	Whitey Ford	28.69	78.46	BBWAA	20	18	18.970
20	Tom Glavine	25.76	90.70	BBWAA	11	36	19.900
21	Roy Halladay	29.65	69.82	BBWAA	30	14	20.490
22	Fergie Jenkins	27.32	78.28	BBWAA	21	25	22.910
23	Warren Spahn	27.77	75.66	BBWAA	23	23	23.000
24	Don Drysdale	28.28	69.19	BBWAA	31	20	24.900
25	Max Scherzer	27.33	73.54		26	24	24.980
26	CC Sabathia	26.80	74.90		24	27	25.460
27	Mike Mussina	24.17	86.48	BBWAA	14	50	26.460
28	Robin Roberts	29.51	59.52	BBWAA	52	15	27.930
29	Dave Stieb	29.28	60.16		50	16	28.280
30	Curt Schilling	25.93	72.28		27	34	30.300

Figure 30: The hall-of-fame chart, or the top 30 starting pitchers post 1951 by the geometric mean of career Grid WAR and peak four-year Grid WAR.

mean, $\text{GeomMean}(a, b) = \sqrt{a \cdot b}$. We use the geometric mean as opposed to the arithmetic mean so as to value pitchers much more the closer they are to being the best (rank one). In Figure 30 we show the peak GWAR rank and geometric mean rank of the top 30 starting pitchers post 1951. Greg Maddux has the best geometric mean rank, as he has the second highest career GWAR and the fourth highest peak GWAR. We also include whether the pitcher was elected to the Hall of Fame by the Baseball Writers' Association of America (BBWAA) or the veteran's committee (Veterans), and hence call Figure 30 the hall-of-fame chart. Each of these pitchers who aren't still playing (Kershaw, Verlander, Scherzer), recently retired (Sabathia), or mired by controversy (Clemens, Schilling) made the Hall of Fame, except notably Kevin Brown and Dave Steib (#10 and #29 ranked geometric mean, respectively). These latter two have without a doubt been snubbed and should be in Cooperstown. There are many articles online corroborating this view.²⁴

References

- Acharya, R.A., Ahmed, A.J., D'Amour, A.N., Lu, H., Morris, C.N., Oglevee, B.D., Peterson, A.W., and Swift, R.N. (2008). Improving major league baseball park factor estimates. *J. Quant. Anal. Sports* 4: 6.
- Albert, J. and Bennett, J. (2003). *Curve ball: baseball, statistics, and the role of chance in the game*. Copernicus Books, New York.
- Appelman, D. (2016). Park factors – 5 year regressed, Available at: <https://library.fangraphs.com/park-factors-5-year-regressed/>.
- Baseball Reference (2011). Pitcher WAR calculations and details, Available at: https://www.baseball-reference.com/about/war_explained_pitch.shtml.
- Baseball Reference (2022). Park adjustments, Available at: <https://www.baseball-reference.com/about/parkadjust.shtml>.
- Baumer, B. and Zimbalist, A. (2014). *The sabermetric revolution: assessing the growth of analytics in baseball*. University of Pennsylvania Press, Philadelphia, Pennsylvania.
- Baumer, B.S., Jensen, S.T., and Matthews, G.J. (2015). OpenWAR: an open source system for evaluating overall player performance in major league baseball. *J. Quant. Anal. Sports* 11: 69–84.
- Brill, R.S., Deshpande, S.K., and Wyner, A.J. (2023). A Bayesian analysis of the time through the order penalty in baseball. *J. Quant. Anal. Sports* 1, <https://doi.org/10.1515/jqas-2022-0116>.
- Brown, L.D. (2008). In-season prediction of batting averages: a field test of empirical Bayes and Bayes methodologies. *Ann. Appl. Stat.* 2: 113–152.
- Chen, T. and Guestrin, C. (2016). XGBoost: a scalable tree boosting system. In: *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining, KDD '16*. ACM, New York, NY, USA, pp. 785–794.
- ESPN (2014). Max Scherzer 2014 game log, Available at: https://www.espn.com/mlb/player/gamelog/_id/28976/year/2014/category/pitching.
- ESPN (2022). MLB park factors – 2019, Available at: https://www.espn.com/mlb/stats/parkfactor/_year/2019.
- Fangraphs (2010). Replacement level, Available at: <https://library.fangraphs.com/misc/war/replacement-level/>.
- Hoerl, A.E. and Kennard, R.W. (1970). Ridge regression: biased estimation for nonorthogonal problems. *Technometrics* 12: 55–67.
- Lewis, M. (2003). *Moneyball: the art of winning an unfair game*. WW Norton & Company, New York.
- Medvedovsky, K. and Patton, A. (2022). *Daily adjusted and regressed Kalman optimized projections | DARKO*, Available at: <https://apanalytics.shinyapps.io/DARKO/>.
- Petti, B. and Gilani, S. (2021). Baseballlr: acquiring and analyzing baseball data. R package version 1.6.0, Available at: <https://billpetti.github.io/baseballr/>.
- Retrosheet (2021). Retrosheet play-by-play data files (event files), Available at: <https://www.retrosheet.org/game.htm>.
- Schwarz, A. and Gammons, P. (2005). *The numbers game: baseball's lifelong fascination with statistics*. Thomas Dunne Books, New York.
- Slowinski, P. (2012). WAR for pitchers, Available at: <https://library.fangraphs.com/war/calculating-war-pitchers/>.
- Statcast (2023). Statcast search, Available at: https://baseballsavant.mlb.com/statcast_search.
- Tango, T., Lichtman, M., and Dolphin, A. (2007). *The book: playing the percentages in baseball*. Potomac Books, Dulles, Virginia.
- Thorn, J. and Palmer, P. (1984). *The hidden game of baseball: a revolutionary approach to baseball and its statistics*. Doubleday, Garden City, NY.
- Wikipedia (2023). Baseball, Available at: <https://en.wikipedia.org/wiki/Baseball>.
- Wolverton, M. (1993). “Support-neutral” statistics – a method of evaluating the true quality of a pitcher’s start. By *The Numbers* 5: 4–14.
- Wolverton, M. (1999). The top pitchers of the 1990s: a support-neutral approach, Available at: <https://www.baseballprospectus.com/news/article/416/the-top-pitchers-of-the-1990s-a-support-neutral-approach/>.
- Wolverton, M. (2004). Baseball prospectus basics: the support-neutral stats, Available at: <https://www.baseballprospectus.com/news/article/2590/baseball-prospectus-basics-the-support-neutral-stats/>.

²⁴ <https://blogs.fangraphs.com/should-kevin-brown-be-in-the-hall-of-fame/>, <https://www.fishstripes.com/22914710/kevin-brown-hall-of-fame-case>.