Challenge Run Value (*cRV*): quantifying strategic value in MLB's ABS challenge era

Jacob Goldstein

Millburn High School, Millburn, NJ 07041, United States

jacobgold392@gmail.com • ORCID: 0009-0004-2659-1256

August 13, 2026

Abstract

We introduce Challenge Run Value (*cRV*), a pitch-level metric for valuing Automated Ball-Strike (ABS) challenges in Major League Baseball. Unlike framing metrics, which measure physical receiving skill under human umpiring, *cRV* treats the challenge as a scarce strategic resource whose value depends on when it is spent, whether it succeeds, and which future options it forgoes. The model combines run-expectancy deltas from count- and plate-appearance-ending state changes with a time-decaying, scarcity-weighted penalty for failed challenges, calibrated against an empirically derived RE288 run-expectancy surface built from public Statcast data. A central methodological contribution is the valuation of terminal outcomes: challenges that convert a called strike three into ball four (or the reverse) end the plate appearance and must be priced from the resulting base/out state and runs scored rather than zeroed. We implement a reproducible five-module pipeline (ingestion, event isolation, alternate-state generation, RE288 mapping, and scoring) with game-clustered bootstrap uncertainty, leverage stratification, and sensitivity analysis. Applying the model to the 1,971 plate-appearance-ending challenges recoverable from the 2026 regular season through August 12, we estimate a mean *cRV* of 0.159 runs per challenge (95% CI $[0.147, 0.172]$); catchers generate roughly two-and-a-half times the per-challenge value of batters (0.231 versus 0.088). Zeroing terminal states would instead drive the aggregate negative (-0.016), inverting the sign of the result. The metric gives teams a principled basis for deciding when to spend a scarce challenge and for separating high-leverage conversion skill from reckless early usage.

Keywords: run expectancy; opportunity cost; catcher framing; sabermetrics; decision quality; resource allocation.

1 Introduction

MLB's adoption of the Automated Ball-Strike (ABS) challenge system reframes pitch evaluation. Under the framing-era paradigm, catcher value at the margin of the strike zone was largely a function of receiving technique and umpire interaction (Fast, 2011; Marchi, 2013; MLB Advanced Media, 2026). A catcher earned runs by making borderline pitches *look* like strikes, and the skill was continuous, repeatable, and accumulated over thousands of receptions per season.

The ABS challenge system replaces that continuous skill with a discrete, scarce resource. Each team carries a limited number of challenges; a successful challenge is retained, a failed challenge is consumed. The strategic problem is therefore no longer "how do I make this pitch look like a strike" but "is this pitch worth spending a challenge on, given how much game remains and how likely I am to be right." That is a

resource-allocation problem over a nine-inning (or longer) horizon, and it is not measured by any existing public metric.

Existing public ABS coverage has been largely descriptive: challenge rates, success percentages, and umpire-agreement rates (Major League Baseball, 2026). These are useful accuracy diagnostics but they are denominated in *challenges*, not runs, and they are indifferent to context. A catcher who wins 60% of his challenges on 0–0 counts in the first inning and a catcher who wins 60% of his challenges on full counts with the bases loaded in the ninth have identical success rates and vastly different value.

What has been missing is a unified, run-denominated valuation that (i) credits the immediate state change produced by an overturn and (ii) charges the opportunity cost of a failed challenge against the remaining game horizon. This paper formalizes that valuation as Challenge Run Value (*cRV*), implements it as an open and reproducible pipeline, validates the implementation against a designed corpus of edge cases, and applies it to the 2026 regular season through August 12: 1,971 plate-appearance-ending challenge events scored against an empirically estimated RE288 surface.

1.1 Contributions

1. A formal run-denominated metric, *cRV*, combining an RE288 pitch-state delta with a time-decaying opportunity-cost penalty (Section 2.1).

2. Correct treatment of plate-appearance-ending challenges: walk force chains with runs scored, and strikeouts with out increments and inning-end zeroing (Section 2.1.5). We show this is not a detail: it is where nearly all of the metric's signal lives.

3. A reproducible five-module pipeline with provenance tracking, bootstrap uncertainty, leverage stratification, an empirical EV_c estimate, and a win-denominated overlay (Section 2.2), backed by a designed validation corpus and unit-test suite that exercises each state-transition path (Section 2.3).

4. An empirical application to the 2026 regular season through August 12 (1,971 plate-appearance-ending challenges scored against an empirically estimated RE288 matrix) that quantifies the run value of the challenge as used in practice and shows catchers extract roughly two-and-a-half times the per-challenge value of batters (Section 3).

1.2 Related work

Count-dependent run expectancy is foundational in sabermetric modeling (T. M. Tango, Lichtman, & Dolphin, 2007). The classical RE24 matrix conditions expected runs on the 24 base-out states. The RE288 extension (8 base states $\times$ 3 outs $\times$ 12 counts) conditions additionally on the ball-strike count, which is precisely the dimension an ABS challenge manipulates (T. Tango, 2018). Without the count dimension, a challenge that moves a count from 0–1 to 1–0 is invisible; RE288 makes it measurable.

Catcher framing research established that called-strike environments are measurable and strategically relevant (Fast, 2011; Marchi, 2013; MLB Advanced Media, 2026). That literature is the natural predecessor to this work, and also its natural contrast: framing measured *how often* borderline pitches were converted to strikes through receiving skill, whereas *cRV* measures *when* a challenge should be spent and *whether* spending it paid off. Framing runs and *cRV* are complementary rather than competing: in a full-ABS world the former decays toward zero while the latter becomes the live skill.

The strategic problem of *when* to spend a scarce, retained-on-success review resource has an established economics and operations-research literature. Abramitzky, Einav, Kolkowitz, and Mill (2012) characterize optimal Hawk-Eye line-call challenge behavior in professional tennis; Clarke and Norman (2012) solve the challenge decision as a dynamic program; and Shivakumar (2018) documents strategic review use in cricket's

Decision Review System, whose "retained on success" rule mirrors ABS. Broader work on umpiring-context effects and decision incentives in sport (Moskowitz & Wertheim, 2011) frames the same problem at the behavioral level. Our opportunity-cost penalty is a deliberately simple approximation to the option value these dynamic-program treatments compute exactly; Section 4.4 returns to the gap between the two.

2 Materials and Methods

2.1 Methodology

2.1.1 Notation

For challenge event i, let the *pre-pitch state* be the triple (c_i, b_i, o_i) where c_i is the ball-strike count, $b_i \in \{0,1\}^3$ is the base occupancy string (1B, 2B, 3B), and $o_i \in \{0,1,2\}$ is the out count. Let $RE(\cdot)$ denote the run value of a post-pitch state, defined to include both any runs scored immediately on the play and the run expectancy of the resulting state.

Every challenge concerns a called pitch that the umpire (or ABS system) ruled either a ball or a strike. Two realities follow:

- the **original** reality, in which the original call stands, and
- the **overturned** reality, in which the call is reversed.

If the challenge fails, the overturned reality never materializes and the two coincide.

2.1.2 Pitch-state delta

$$\Delta RV_{pitch,i} = \begin{cases} RE(\text{overturned}_i) - RE(\text{original}_i) & \text{batter challenge succeeds,} \\ RE(\text{original}_i) - RE(\text{overturned}_i) & \text{catcher challenge succeeds,} \\ 0 & \text{challenge fails.} \end{cases} \tag{1}$$

The sign convention encodes whose interest the challenge serves. Run expectancy is always stated from the batting team's perspective, so a batter benefits when the overturn *raises* RE (a strike becomes a ball) and a catcher benefits when the overturn *lowers* it (a ball becomes a strike). Taking the catcher's delta as $RE_{\text{original}} - RE_{\text{overturned}}$ makes a successful defensive challenge score positive, so that *cRV* is directly comparable across roles.

Failed challenges leave the on-field state unchanged. The pitch delta is exactly zero (not merely small) because the umpire's call stands and play proceeds from the original reality.

2.1.3 Opportunity-cost penalty

Let EV_c be the baseline expected value of *retaining* a challenge. In v1 we fix $EV_c = 0.15$ runs as a transparent constant; Section 3.4 reports sensitivity over $[0.034, 0.30]$. Conceptually,

$$EV_c = P(\text{future need}) \times P(\text{win}) \times \overline{\Delta RV}_{\text{late}}, \tag{2}$$

the probability a challenge will be wanted again, times the conversion rate, times the average value of a late-game conversion. Estimating the factors from the 2026 corpus gives $P(\text{future need}) = 0.206$ (the share of challenges that are not the challenging team's last challenge of the game), $P(\text{win}) = 0.465$, and $\overline{\Delta RV}_{\text{late}} = 0.353$ runs, for an empirical $EV_c = 0.034$ runs (cluster-bootstrap 95% CI $[0.030, 0.038]$; Table 1).

This is a calibration of the penalty constant rather than a direct measurement of option value: $P(\text{future need})$ is a within-game frequency proxy, not a model of future game states. Because the empirical value sits below the 0.15 baseline, the headline results that follow are computed under a deliberately conservative EV_c.

Table 1: Empirical estimate of the retained-challenge value EV_c.

Metric	Value	CI lower	CI upper
P(future need)	0.206	0.191	0.222
P(win)	0.465	0.444	0.486
Mean late-game delta RV	0.353	0.326	0.384
Empirical EV_c	0.034	0.030	0.038

Failed challenges incur

$$Penalty_i = -\left(\frac{\max(0,\, 9 - \text{inning}_i)}{9}\right)\frac{EV_c}{C_i}, \tag{3}$$

where C_i is the number of challenges the team held immediately before the failure. The penalty prices the opportunity cost of the challenge the failure destroys: each team enters the game with a two-challenge budget (Major League Baseball, 2026) (a failure consumes one, a success retains it), so losing the *last* challenge forfeits the entire retained option value, while losing one of two forfeits only a $1/C$ share. In the 2026 corpus, 973 of the 1,055 failed challenges were the team's first failure of the game ($C_i = 2$, a half charge) and 82 were its last ($C_i = 1$, a full charge).

Four properties of Equation 3 deserve comment. First, successful challenges are retained under ABS rules (Major League Baseball, 2026), so $Penalty_i = 0$ on wins, since the resource was not consumed. Second, the penalty conditions on the actual scarce resource: a failure with a reserve challenge still in hand ($C_i = 2$) is charged half of one with none in reserve ($C_i = 1$), because what is destroyed is the marginal challenge, not the whole stock. Third, the penalty is *linearly decaying* in the inning: a failed first-inning challenge forfeits eight innings of option value and is charged $-(8/9) \times EV_c/C_i$, whereas a failed ninth-inning challenge forfeits nothing and is charged zero. Fourth, extra innings floor at zero via the $\max(0, \cdot)$ term, consistent with inning-level challenge resets.

Both the linear decay and the $1/C$ share are modeling choices, not derived results: they are the simplest forms with the correct boundary behavior, and they avoid free parameters the current data cannot identify. The full solution (which the tennis and cricket challenge literature obtains by dynamic programming (Abramitzky et al., 2012; Clarke & Norman, 2012; Shivakumar, 2018)) would price the marginal value of the k-th challenge and the option value of challenges held. A convex decay, reflecting that late innings are more leveraged per inning than early ones, is a plausible refinement.

2.1.4 Unified metric

$$cRV_i = \Delta RV_{pitch,i} + Penalty_i. \tag{4}$$

Because exactly one of the two terms is nonzero for any given event (successes have no penalty, failures have no delta), *cRV* decomposes cleanly: positive values come entirely from conversions, negative values entirely from wasted challenges.

2.1.5 Terminal outcomes

The central modeling question is how to value challenges that *end the plate appearance*. A called strike on an 0–0 count merely moves the count; a called strike on a 3–2 count ends the at-bat. Since challenges are

disproportionately spent on full counts and two-strike counts (exactly where the stakes are highest), getting terminal outcomes right dominates the metric's behavior.

We handle three cases:

Walk (Ball 4). Runners advance under standard force rules: the batter takes first; a runner on first is forced to second; a runner on second advances only if first was occupied; a runner on third scores only if the bases were loaded. Any forced run is added immediately to the state's value. Remaining run expectancy is then looked up at the post-walk base state, unchanged outs, and a reset 0–0 count (a new batter is up).

Strikeout (Strike 3). Outs increment by one. If outs reach three, the half-inning ends and the remaining run expectancy is zero. Otherwise RE is looked up at the same bases, outs+1, and a reset 0–0 count. A pure strikeout scores no runs, so the immediate-runs term is zero.

Non-terminal count change. Bases and outs are unchanged; only the count state updates in the RE288 lookup. These are the low-magnitude events: an 0–1 to 1–0 swing is worth a few hundredths of a run.

Formal transition rules are given in Appendix A.

2.1.6 Why the terminal treatment matters

Earlier drafts of this pipeline assigned $RE = 0$ to *all* terminal states, on the reasoning that a completed plate appearance has no count-conditioned run expectancy. That is a category error: the plate appearance ends, but the *inning* does not. A bases-loaded walk leaves the bases loaded with a run already in; a strikeout with one out leaves runners on with two outs. Both states carry substantial run expectancy.

The practical consequence is severe. Under the old treatment, a bases-loaded walk conversion scores $\Delta RV = 0$ instead of the $+1.91$ runs it is actually worth, and an inning-ending strikeout conversion scores 0 instead of $+1.72$. On the 2026 terminal-challenge corpus (Section 3), zeroing terminal states drives mean *cRV* to -0.016: the metric reports that challenging is, on net, *harmful*, because only the penalties survive. With terminal states valued correctly, mean *cRV* is $+0.159$. The sign of the headline result flips on this one modeling decision, and it flips on real data, not just a constructed example.

2.2 Data and pipeline

The repository implements five core modules plus a validation layer (Figure 1):

1. **Ingestion** (`ingestion.py`): Statcast pitch-by-pitch pull via `pybaseball`, defaulting to the 2026 ABS-era window. Returns an empty frame on network failure rather than raising, so the pipeline degrades gracefully.

2. **Event isolation** (`events.py`): filters the `des` field for challenge language, infers challenger role (batter/catcher), called pitch type, and success/failure, and flags parse-quality issues.

3. **Alternate realities** (`realities.py`): from the pre-pitch count, bases, and outs, constructs the umpire-call state and the overturned state, including walk force chains and strikeout out increments.

4. **RE288 mapping** (`re288.py`): loads an empirical matrix estimated from play-by-play when available, else a structured synthetic surface; maps both realities to run values.

5. **Scoring** (`calculate.py`): applies ΔRV and the opportunity-cost penalty.

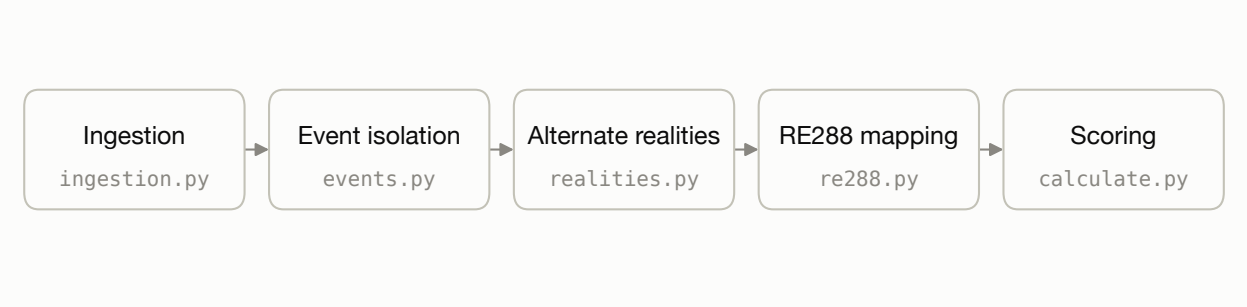

Figure 1: Five-module *cRV* pipeline: ingestion, event isolation, alternate-state generation, RE288 mapping, and scoring. Each stage writes an inspectable intermediate file, so any challenge event can be traced from its raw description text through to its final score.

2.2.1 Event isolation and parse quality

Challenge events are identified from free-text descriptions requiring both challenge language and an outcome token ("overturned" / "upheld" / "stands"). Role and called-pitch-type are inferred from the same text. When inference fails (for example, a description reading only "Challenge overturned to a ball" with no role named), the event is retained but flagged via a `parse_quality` column rather than silently dropped or silently mis-assigned.

This matters for honest accounting: in the 2026 corpus, only 2 of 1,971 events carry a `missing_challenger` flag. Such events contribute zero to ΔRV (no role means no sign convention) and appear as an `unknown` row in the role summary. Production ABS feeds with structured challenge flags would eliminate this ambiguity entirely.

A structural feature of the extraction deserves note. Statcast records the descriptive `des` text on the final pitch of a plate appearance, so the challenges recoverable from it are precisely those that end the plate appearance: 1,413 overturned or upheld strikeout calls and 558 walk calls, concentrated on two-strike and three-ball counts. Non-terminal count-change challenges (for example an 0–1 call overturned to 1–0 mid at-bat) are not separately identifiable from `des` and are absent from this corpus. This is a selection property of the data source, not of the metric; Section 3 reports on the terminal challenges that dominate actual challenge usage.

2.2.2 RE288 estimation

The empirical estimator groups play-by-play pitches by (base state, outs, count), computes runs scored from that pitch to the end of the half-inning, and averages. It also reports a per-cell `sample_size`, since rare states (3–0 counts with the bases loaded and two outs) are noisy and a multi-season baseline is required for stability. Across the 288 cells the median cell rests on 585 pitches and the mean on 1,858, but the distribution is right-skewed: 28 cells (10%) have fewer than 100 pitches and 188 cells (65%) fewer than 1,000 (Table 2). The sparsest cells are precisely the rare base-out-count states that most need a multi-season baseline.

When no play-by-play is available, a structured synthetic surface is used: $RE = 0.56 + 0.18r - 0.32o + 0.04b - 0.03s$, where r is runners on base, o is outs, and b, s are balls and strikes. This is monotone in the right directions and adequate for exercising the pipeline, but it is not a baseball estimate and results derived from it are labeled accordingly.

2.2.3 Provenance and reproducibility

All artifacts regenerate with:

Table 2: Per-cell sample sizes of the empirical RE288 surface.

Metric	Value
RE288 cells	288
Min pitches per cell	11
Median pitches per cell	585
Mean pitches per cell	1858.160
Max pitches per cell	33388
Cells under 100 pitches	28
Cells under 1000 pitches	188

```
python pipeline.py -output-dir outputs
python pipeline.py -force-synthetic # offline demo corpus
```

A `run_metadata.csv` records the data source (`statcast`/`synthetic`) and RE288 source (`empirical`/`synthetic`/`csv`) for every run, so no figure can be silently misattributed. The results in this paper were produced with `data_source = statcast` and `re288_source = empirical`: real 2026 challenge events scored against an RE288 matrix estimated from the same season's play-by-play.

2.3 Validation design

The state-transition logic is the part of the pipeline most prone to silent error, so its correctness is established independently of any particular data pull, against a *designed* corpus that exercises every transition path:

- non-terminal count changes (0–0 strike overturned to a ball),
- a bases-loaded walk conversion (run forced in, bases stay loaded),
- an inning-ending strikeout conversion (RE drops to zero),
- a two-out strikeout that does *not* end the inning,
- an extra-inning failure (penalty must be exactly zero),
- early and late failures (penalty must scale with innings remaining),
- a parse-ambiguous description (role inference must fail loudly).

A unit-test suite (`tests/`) asserts the expected behavior of each path independently of the pipeline: walk force chains for all eight base states, terminal RE zeroing, extra-inning penalty nullification, and parser role extraction. All tests pass on the current implementation. This harness makes the empirical results of Section 3 regression-safe: the transition rules that value a bases-loaded walk are the same rules the tests pin down.

3 Results

We apply the pipeline to every plate-appearance-ending ABS challenge recoverable from the 2026 regular season: 1,971 events spanning 2026-03-26 through 2026-08-12, issued by 446 distinct challengers, scored against an RE288 matrix estimated from the same season's play-by-play ($EV_c = 0.15$). All artifacts live in `outputs/`; the provenance metadata confirms the `statcast` and `empirical` sources.

Table 3: Validation summary metrics for the current run.

Metric	Value	CI lower	CI upper	Notes
Total events	1971	1971	1971	Count of challenge events
Success rate	0.465	0	1	Share of successful challenges
Mean cRV	0.159	0.147	0.172	Game-clustered bootstrap CI for mean cRV

Mean *cRV* is 0.159 runs per challenge, with a game-clustered bootstrap 95% CI of $[0.147, 0.172]$, comfortably above zero, so the challenge as actually used in 2026 added run value on net within this terminal-challenge census. A positive mean is the expected-value answer for the challenge system, not a scoring artifact: a successful overturn realizes a positive run swing, while a failure is charged only the opportunity cost of the challenge it destroyed (Section 4.4). The overall success rate is 46.5%. At $n = 1{,}971$ the interval is tight, a marked contrast to the wide bands a small validation corpus would produce, and it is narrow enough to support role- and leverage-level comparisons rather than only a directional claim.

Table 4: Top player-level cumulative cRV values.

Player	Role	Challenges	Success rate	Cumulative *cRV*	Mean *cRV*
Shea Langeliers	catcher	33	0.697	8.791	0.266
Edgar Quero	catcher	18	0.667	7.031	0.391
Hunter Goodman	catcher	26	0.692	6.481	0.249
Carson Kelly	catcher	17	0.882	5.483	0.323
Victor Caratini	catcher	24	0.792	5.433	0.226
Freddy Fermin	catcher	27	0.667	5.422	0.201
Ryan Jeffers	catcher	25	0.640	5.168	0.207
Jonah Heim	catcher	21	0.762	5.118	0.244
Francisco Alvarez	catcher	18	0.667	5.095	0.283
Carlos Narváez	catcher	18	0.889	5.069	0.282

Table 5: Aggregate cRV by challenger role.

Role	Challenges	Cumulative *cRV*	Mean *cRV*
catcher	984	227.642	0.231
batter	985	86.700	0.088
unknown	2	−0.017	−0.008

Figure 2 shows the player-level distribution and Figure 3 the same totals aggregated by role. The shape is the one the framework predicts: value is concentrated in terminal-state conversions, while failures contribute a thin band of small negative values whose magnitude is set entirely by inning. The role split is the headline empirical finding. Batters and catchers issue challenges in almost equal numbers (985 versus 984), but catchers accumulate 227.6 runs of *cRV* to the batters' 86.7, a per-challenge mean of 0.231 against 0.088. This comparison is descriptive, not causal: catchers and batters issue challenges in structurally different spots (converting two-strike takes into outs versus avoiding called third strikes), so it does not by itself identify superior judgment. Catchers convert borderline two-strike takes into outs, the single most valuable thing a challenge can do; batters more often convert a called third strike into a mere continuation of the at-bat.

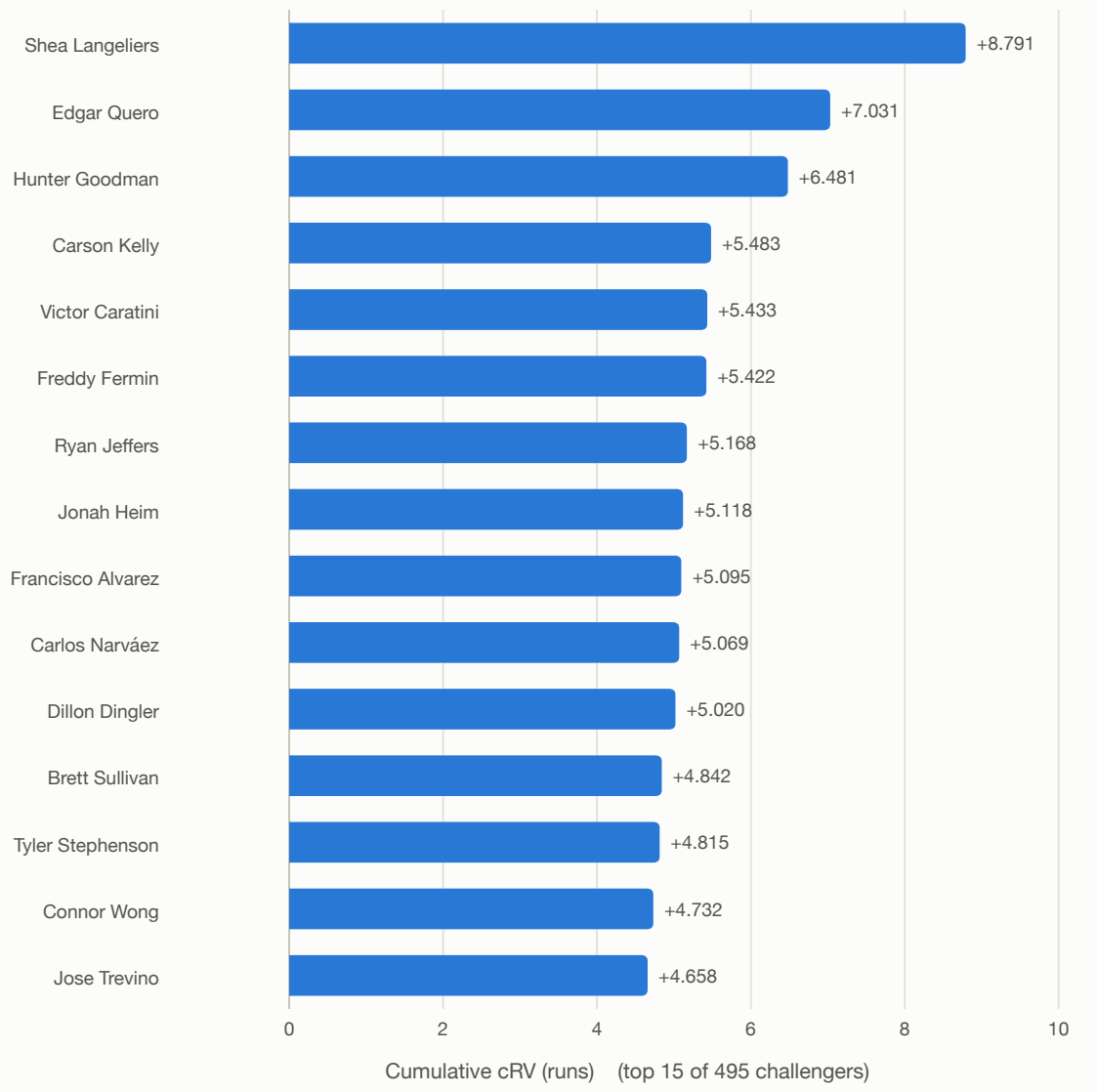

Figure 2: Player-level cumulative *cRV* for the top challengers of the 2026 regular season. Bars extending right of the zero rule (blue) are positive and arise entirely from successful conversions; bars extending left (red) are negative and arise entirely from opportunity-cost penalties on failed challenges. Bar direction relative to the zero rule encodes the sign independently of color, so the comparison survives grayscale reproduction. The leaderboard is dominated by catchers, the primary users of the challenge.

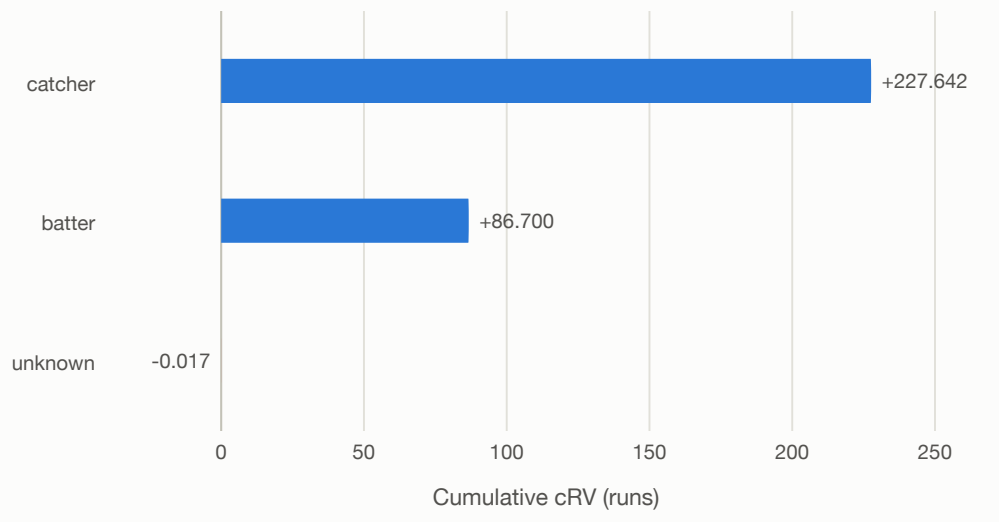

Figure 3: Cumulative *cRV* aggregated by challenger role (batter, catcher, and a small number of events whose role could not be parsed). Batters and catchers issue challenges in nearly equal numbers, but catchers account for the large majority of aggregate value, converting borderline two-strike calls into inning-ending outs.

3.1 Information and reliability

Two checks bear on whether *cRV* measures anything beyond raw accuracy, and whether it is stable enough to be a skill rather than noise (Table 6). First, across the 47 players with at least 10 challenges, mean *cRV* correlates 0.72 (Pearson) and 0.68 (Spearman) with challenge success rate. Accuracy explains roughly half the variance, but the remaining half is leverage and timing, the information the run-denominated metric adds over a success-rate ranking. Second, split-half reliability within 2026 is weaker: for the 45 players with at least four challenges in each half of the season, mean *cRV* in the first half correlates 0.34 (Pearson) with the second half. Player-level *cRV* is therefore only modestly stable within a half-season, consistent with a young, sparse sample; the run-denominated ordering is robust in aggregate but not yet a per-player skill signal.

Table 6: Information and reliability of player-level cRV.

Metric	Value
Qualifying players	47
Min challenges per player	10
Pearson r (cRV vs success rate)	0.723
Spearman rho (cRV vs success rate)	0.678
Qualifying players	45
Min challenges per half	4
Pearson r (split-half cRV)	0.344
Spearman rho (split-half cRV)	0.333

3.2 Case studies

Three events illustrate the model's economic content. Each is traced from pre-pitch state to final score.

Bases-loaded walk conversion (Eli White, batter, +1.91). Fifth inning, bases loaded, 1 out, 3–2 count. The umpire calls strike three; White challenges and wins, converting the pitch to ball four.

- *Original reality:* strikeout. Outs $1 \to 2$, bases loaded unchanged. $RE(\text{original}) = 0.721$.
- *Overturned reality:* walk. The runner on third is forced home (+1 run, Ozzie Albies scores), bases remain loaded, outs stay at 1. $RE(\text{overturned}) = 2.633$.

Batter delta: $2.633 - 0.721 = 1.912$ runs, with no penalty (the challenge is retained). A forced run plus the gap between one and two outs with the bases loaded is an enormous swing, and it is the single most valuable challenge outcome available.

Inning-ending strikeout conversion (Luis Campusano, catcher, $+1.72$). Fourth inning, bases loaded, 2 outs, 3–2 count. The umpire calls ball four; Campusano challenges and wins, converting the walk into strike three.

- *Original reality:* walk. A run is forced in (+1), bases stay loaded, still 2 outs. $RE(\text{original}) = 1.721$.
- *Overturned reality:* strikeout, third out. Inning over. $RE(\text{overturned}) = 0$.

Catcher delta: $1.721 - 0 = 1.721$ runs. This is exactly the case the old zero-everything treatment got most wrong: it scored the inning-ending conversion (erasing a run already in and ending the threat) at zero.

Early failed challenge (Carter Jensen, catcher, -0.07). A first-inning failed catcher challenge (the team's first failure, so two challenges were still held) yields $\Delta RV = 0$ and $Penalty = -(8/9) \times 0.15 \times (1/2) \approx -0.067$. Had it been the team's last challenge the charge would double to -0.133, and any ninth-inning failure yields exactly 0.00 because no innings of option value remain. Same outcome and accuracy, but the price depends on both how much game remains and how much of the challenge budget survives, the resource-management incentive the framework exists to encode.

3.3 Leverage stratification

Table 7: cRV by inning leverage bucket.

Leverage bucket	Events	Mean *cRV*	Total *cRV*
early(1-3)	578	0.168	97.005
mid(4-6)	631	0.171	107.993
late(7-9)	735	0.144	105.576
extra(10+)	27	0.139	3.752

Total value tracks event count: the late bucket (innings 7–9) carries the most challenges (735, 105.6 runs), followed by mid (631 events, 108.0) and early (578 events, 97.0), with a small extra-inning tail (27 events, 3.8). The *mean* is mildly higher early and mid-game than late: 0.168, 0.171, 0.144, and 0.139 runs respectively. Two forces pull against each other: the inning decay makes an early failure costlier, but the scarcity weight makes a team's *first* failure (disproportionately early) only half as costly as its last. Terminal conversions dominate the numerator in every inning, so per-challenge value stays within a narrow band while total value accumulates wherever challenges are most frequent.

3.4 Sensitivity to EV_c

Total corpus *cRV* ranges from 338.9 runs at the empirical $EV_c = 0.034$ to 282.6 runs at $EV_c = 0.30$ (Figure 4), a spread of 56.2 runs across the grid, with the mean falling from 0.172 to 0.143. The relationship is

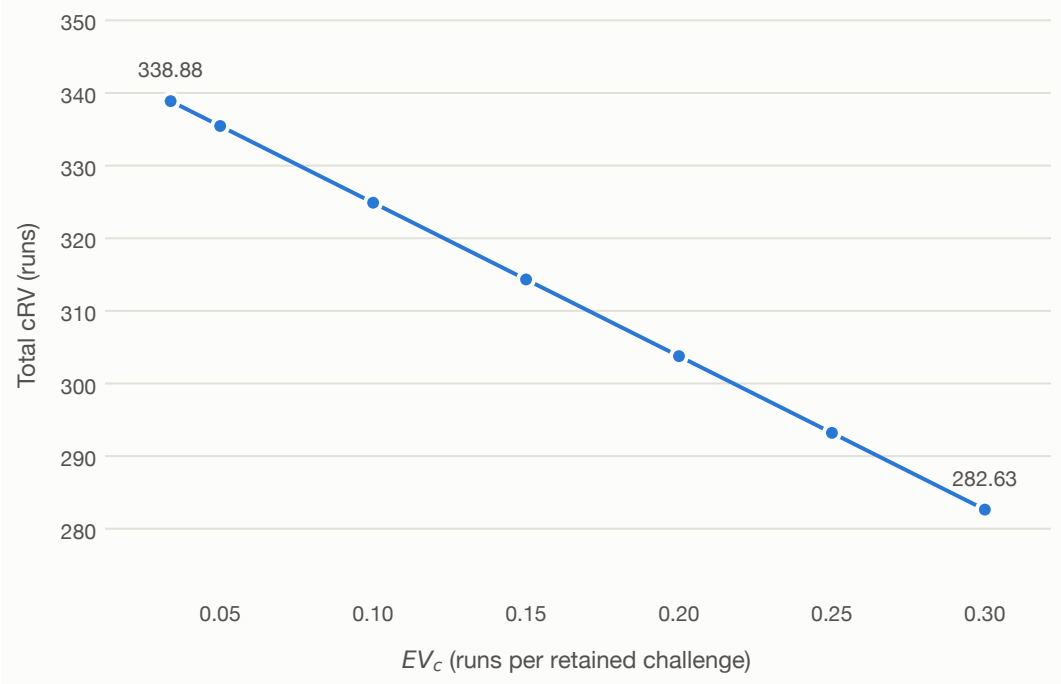

Figure 4: Total corpus *cRV* as a function of the baseline retained-challenge value EV_c, over the range $[0.034, 0.30]$ runs spanning the empirical estimate and the fixed baseline. The relationship is exactly linear because the opportunity-cost penalty applies only to failed challenges and scales linearly in EV_c; successful challenges carry no penalty term. Endpoint values are labeled.

Table 8: Sensitivity of aggregate cRV to the baseline challenge value EV_c.

EV_c	Mean *cRV*	Total *cRV*
0.034	0.172	338.876
0.050	0.170	335.453
0.100	0.165	324.889
0.150	0.159	314.325
0.200	0.154	303.761
0.250	0.149	293.198
0.300	0.143	282.634

exactly linear: from Equation 3, $\sum_i cRV_i = \sum_i \Delta RV_i - EV_c \sum_{i \in \text{fail}} (9 - \text{inning}_i)/(9C_i)$, and only the second term depends on EV_c.

Two conclusions follow. First, because *cRV* is monotone nonincreasing in EV_c (only the penalty depends on it), the positive sign of mean *cRV* holds across the entire grid (at the data-driven value 0.034 as well as the fixed baseline), so the conclusion that challenging adds value is not an artifact of the EV_c choice. Second, the *ranking* of successful high-leverage events is completely invariant to EV_c, since successes carry no penalty term at all. EV_c affects only how harshly failures are judged relative to one another, and even there it is a monotone rescaling.

3.5 Win-probability overlay

As a win-denominated complement to the run-denominated results, we define a descriptive win-denominated analog of *cRV*: a successful overturn is priced by its win-probability swing, and a failed challenge carries a win-denominated opportunity-cost penalty. Statcast records `delta_home_win_exp`, the change in the home team's win probability on the resolved pitch; on a successful challenge that pitch is the overturned call, so signing it to the challenging side measures the win probability the overturn actually moved. The penalty mirrors Equation 3, with the retained-challenge value now measured in win probability: 0.0025 win probability per retained challenge, estimated from the same three-factor decomposition as the run version (Table 9). The resulting cWPA preserves the run-denominated ordering: catchers accumulate +14.9 win probability (0.0151 per challenge) to batters' +5.3 (0.0054 per challenge; Table 10), the same catcher dominance found in runs, for a corpus total of +20.2 win probability (Figure 5).

Table 9: Empirical estimate of the win-denominated retained-challenge value.

Metric	Value	CI lower	CI upper
P(future need)	0.206	0.191	0.222
P(win)	0.465	0.444	0.486
Mean late-game win swing	0.026	0.023	0.029
Win-denominated EV_c	0.002	0.002	0.003

Table 10: Realized win-probability swing of successful overturns by role.

Role	Events	Total cWPA	Mean cWPA
catcher	984	14.895	0.015
batter	985	5.324	0.005
unknown	2	0.000	0.000

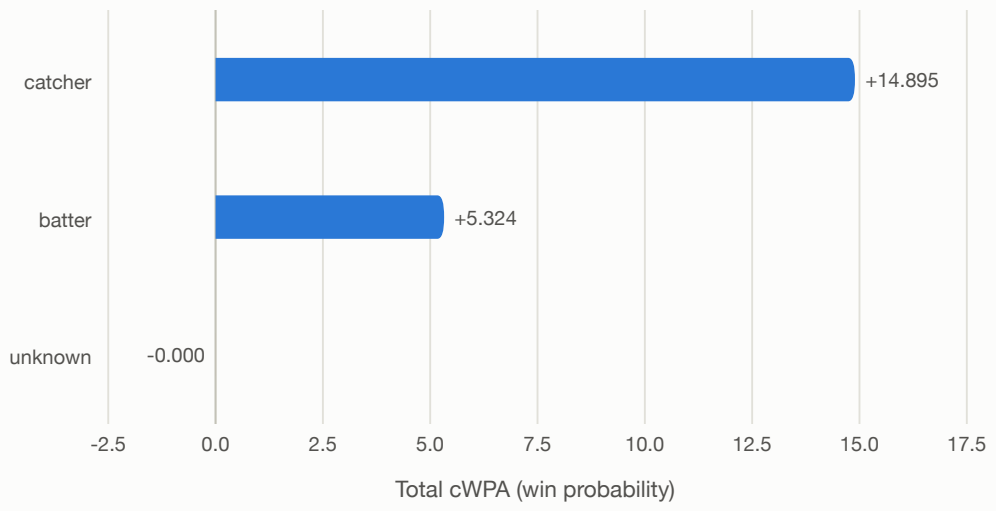

Figure 5: Total challenge win probability (cWPA) by challenger role, the win-denominated counterpart of Figure 3. Successful overturns are priced by their win-probability swing and failed challenges by a win-denominated opportunity-cost penalty. As in runs, catchers accumulate the large majority of win-probability value.

4 Discussion

4.1 Managerial implications

Under this framework, challenging a borderline strike on an 0–0 count is rarely justified. The count move (0–1 $\rightarrow$ 1–0) is worth roughly 0.07 runs in our surface, while a first-inning miss costs $-(8/9) \times 0.15/C$, about -0.07 runs if it is the team's first failure ($C = 2$) and -0.13 if it is the last ($C = 1$) under the baseline EV_c. Break-even therefore requires a win probability of roughly 49% to 65%, depending on how much of the challenge budget remains, which is higher than typical challenge success rates. Under the empirical $EV_c = 0.034$ those first-inning penalties shrink to -0.015 and -0.03 runs, and break-even falls to about 18%–30%, so the strength of this recommendation turns on which EV_c one trusts.

By contrast, full-count terminal challenges (especially with runners on and two outs) can swing between half a run and a run and a half of expectancy. At those magnitudes, break-even win probability falls into the low single digits of percent. Players should be instructed to spend challenges almost freely on plate-appearance-ending calls with runners in scoring position, and almost never on early-count calls with the bases empty.

More generally, cumulative *cRV* is a decision-quality signal distinct from raw challenge accuracy. A catcher who goes 5-for-10 on low-leverage challenges may post a worse *cRV* than one who goes 3-for-5 exclusively in high-leverage spots. Accuracy measures the eye; *cRV* measures the judgment.

4.2 Game-theoretic spillovers

Once a team has exhausted its challenges, the strategic environment changes for both sides. Pitchers may expand the zone and catchers may set up further off the plate, knowing borderline takes cannot be reviewed. Conversely, a team holding challenges late exerts a deterrent effect that never appears in any box score.

cRV as currently specified prices neither effect. It values challenges actually thrown, not the option value of challenges held. A complete game-theoretic treatment would model the equilibrium zone expansion as a function of remaining challenges on both sides, a substantially harder problem requiring pitch-location data

and a behavioral model of umpire and pitcher response.

4.3 Relationship to framing

In a full-ABS-challenge world, framing runs should compress toward zero for challenged pitches while remaining live for unchallenged ones. The two metrics therefore partition the borderline-pitch value space: framing governs the pitches nobody reviews, *cRV* governs the ones somebody does. A complete catcher valuation in the ABS era plausibly sums the two, though care is needed to avoid double-counting pitches that were framed *and* challenged.

4.4 Limitations

- **Data source and coverage.** The corpus is the full set of 2026 regular-season challenge events recoverable from Statcast play-by-play descriptions ($n = 1{,}971$). Because Statcast records an outcome in the free-text description only on the pitch that ends the plate appearance, every recovered challenge is plate-appearance-ending (1,413 strikeout calls, 558 walk calls). Count-changing challenges that leave the plate appearance live are not represented. The results are therefore a census of *terminal* challenges rather than a sample of all challenges, which captures the highest-leverage events but omits the low-value count-changing tail, so the reported mean is upward-biased relative to the mean over all challenges.

- **RE288 surface.** The matrix is estimated empirically from 2026 regular-season play-by-play (`estimate_empirical_re288_matrix`); cells visited rarely inherit correspondingly wide sampling error, and multi-season pooling would stabilize them.

- **Sample size and selection.** $n = 1{,}971$ is a season-to-date census of terminal challenges, not a random sample. The bootstrap interval $[0.147, 0.172]$ reflects resampling variability within that census, not sampling from a larger population, and the reported mean is an average over the highest-leverage subset of challenges.

- **Opportunity-cost specification.** The retained-challenge penalty (Equation 3) conditions on innings remaining and on the number of challenges held before a failure (a $1/C$ share), but it is a first-order approximation, not the full dynamic program solved in the tennis and cricket challenge literature (Abramitzky et al., 2012; Clarke & Norman, 2012; Shivakumar, 2018). A positive mean *cRV* is nonetheless the expected-value answer (a successful overturn realizes a positive run swing while a failure is charged the opportunity cost of the challenge it destroyed), rather than an artifact of an asymmetric scoring rule.

- **Fixed EV_c.** The baseline challenge value is fixed at 0.15 runs for the headline results. The empirical estimate (0.034 runs) is about a quarter of that, so the headline results are conservative; Section 3.4 bounds the model risk across both values and beyond.

- **Decay and share assumptions.** The penalty decays linearly in innings remaining and divides by the number of challenges held. Both have correct boundary behavior but no empirical justification; a convex decay, or a data-driven marginal value for the k-th challenge, may better reflect late-inning leverage and option value.

- **Parse ambiguity.** Role inference from free-text descriptions can fail (2 of 1,971 events). Production feeds with structured challenge flags would eliminate this.

- **Win-probability layer is descriptive.** *cRV* is run-denominated, not win-denominated, so a one-run swing in a blowout and in a tie game score identically. The win-denominated overlay prices overturns and their penalty in win probability, but it leans on Statcast's pitch-level win-expectancy swings rather than a full leverage-index model of challenge value.
- **Challenges held are partially priced.** The penalty charges the $1/C$ share of the challenge a failure destroys, but it does not price the option value of challenges retained for future use, nor the deterrent effect a holding team exerts on an opponent.

4.5 Future work

Immediate priorities, roughly in dependency order:

1. Recover count-changing (non-terminal) challenges that Statcast's free-text descriptions omit, using structured ABS challenge logs, so the corpus becomes a census of *all* challenges rather than terminal ones only.
2. Build a multi-season empirical RE288 matrix and report per-cell sample sizes, flagging low-n states as unstable.
3. Validate description parsers against official ABS game logs and quantify false-positive and false-negative rates.
4. Replace the static three-factor EV_c estimate and the $1/C$ share with the full dynamic program of the tennis and cricket challenge literature (Abramitzky et al., 2012; Clarke & Norman, 2012; Shivakumar, 2018), which prices the marginal value of the k-th challenge and the option value of challenges held.
5. Extend the within-season split-half reliability check (currently $r \approx 0.34$) across multiple seasons and larger per-player samples, to establish whether player *cRV* stabilizes into a skill signal.
6. Explore dynamic $P(\text{win})$ conditioned on pitch location and historical umpire/ABS agreement rates, enabling a prospective "should you challenge this?" model rather than a retrospective score.
7. Layer a full leverage-index model over *cRV*, conditioning the win value of an overturn on the game state rather than Statcast's average pitch-level win-expectancy swing.

5 Conclusion

cRV provides a practical, run-based framework for measuring challenge strategy in the ABS era. By pairing RE288 state deltas (including correct walk and strikeout transitions) with a transparent opportunity-cost penalty, the metric distinguishes high-leverage conversion skill from reckless early usage, a distinction invisible to raw success rates.

The methodological finding we would most emphasize is the terminal-outcome treatment. Plate-appearance-ending challenges are simultaneously the most common high-leverage challenges and the easiest to model incorrectly; on the 2026 terminal-challenge corpus, zeroing them turns a mean *cRV* of $+0.159$ into -0.016, inverting the sign of the headline result. Any future implementation of a challenge-valuation metric should treat walk force chains and strikeout out increments as first-class cases, not edge cases.

The accompanying open pipeline makes every table and figure in this paper regenerable from a single command, with provenance tracking that records whether each run drew on live Statcast data or the synthetic validation corpus. Applied to the 1,971 plate-appearance-ending challenges recoverable from the 2026 season, it establishes catchers as the primary beneficiaries of the ABS challenge and terminal conversions as the events that carry the metric.

A State-transition rules

Let the pre-pitch state be count (b,s), base string $\beta = \beta_1\beta_2\beta_3$, and outs o.

A.1 Called strike

$$(b,s) \to \begin{cases} (b,s+1),\ \beta,\ o & \text{if } s+1<3 \quad \text{(non-terminal)},\\ (0,0),\ \beta,\ o+1 & \text{if } s+1=3,\ o+1<3 \quad \text{(strikeout)},\\ \text{inning over},\ RE=0 & \text{if } s+1=3,\ o+1=3. \end{cases} \tag{5}$$

Runs scored on the play: 0 in all cases.

A.2 Called ball

$$(b,s) \to \begin{cases} (b+1,s),\ \beta,\ o & \text{if } b+1<4 \quad \text{(non-terminal)},\\ (0,0),\ \beta',\ o & \text{if } b+1=4 \quad \text{(walk)}, \end{cases} \tag{6}$$

where the post-walk base string β' and runs scored ρ follow the force chain:

$$\beta_1' = 1 \quad \text{(batter to first, always)}, \tag{7}$$

$$\beta_2' = \begin{cases} 1 & \text{if } \beta_1 = 1 \text{ (forced)},\\ \beta_2 & \text{otherwise}, \end{cases} \tag{8}$$

$$\beta_3' = \begin{cases} 1 & \text{if } \beta_1 = \beta_2 = 1 \text{ (forced from second)},\\ \beta_3 & \text{otherwise}, \end{cases} \tag{9}$$

$$\rho = \begin{cases} 1 & \text{if } \beta = 111 \text{ (bases loaded)},\\ 0 & \text{otherwise}. \end{cases} \tag{10}$$

Enumerating all eight base states:

Table 11: Walk force chain for all base states.

Pre-walk β	Description	Post-walk β'	Runs ρ
000	empty	100	0
100	1B	110	0
010	2B	110	0
001	3B	101	0
110	1B, 2B	111	0
101	1B, 3B	111	0
011	2B, 3B	111	0
111	loaded	111	1

Note the unforced cases: with a runner on second only (010), the batter takes first and the runner *holds*, giving 110. With runners on second and third (011), both hold and the batter takes first, giving 111 with no run scored; the runner on third is not forced.

A.3 Overturn mapping

An overturned called strike is evaluated as a called ball, and an overturned called ball as a called strike. If the challenge fails, the overturned reality is set equal to the original reality, guaranteeing $\Delta RV = 0$ by construction rather than by a separate branch.

A.4 State value

$$RE(\text{state}) = \begin{cases} \rho & \text{if inning over or } o \geq 3, \\ \rho + RE288(\beta, o, \text{count}) & \text{otherwise.} \end{cases} \tag{11}$$

B Reproducing the results

```
python3 -m venv .venv && source .venv/bin/activate
pip install -r requirements.txt
python pipeline.py --output-dir outputs        # full 2026 Statcast pull
python -m pytest tests/ -q
cd paper && make paper
```

The first `pipeline.py` invocation fetches and caches the 2026 regular-season Statcast feed, isolates challenge events, estimates the empirical RE288 surface, and writes every table and figure; add `--force-synthetic` to regenerate the offline validation corpus instead. Every table in Section 3 is generated by the pipeline and included via `\input`; rerunning the pipeline updates the manuscript automatically. A `run_metadata.csv` artifact records the data provenance for the run that produced the current tables. The analysis code and the synthetic validation corpus are deposited at Zenodo (Goldstein, 2026).

Data availability

The pitch-level inputs are publicly available Statcast data, retrieved through the `pybaseball` interface (LeDoux, 2024). The results reported in this manuscript are computed from the 2026 regular-season challenge corpus through August 12 ($n = 1{,}971$) extracted from those feeds. The deterministic synthetic validation corpus described in Section 2.3 is retained as an offline correctness harness, and the analysis code and corpus are deposited at Zenodo (Goldstein, 2026).

Acknowledgments

The author is grateful to the developers of `pybaseball` and to Major League Baseball for making Statcast data publicly available.

Funding

No external funding supported this work.

Conflict of interest

The author declares no conflict of interest.

References

Abramitzky, R., Einav, L., Kolkowitz, S., & Mill, R. (2012). On the optimality of line call challenges in professional tennis. *International Economic Review*, *53*, 939–964. Retrieved from https://doi.org/10.1111/j.1468-2354.2012.00706.x

Clarke, S. R., & Norman, J. M. (2012). Optimal challenges in tennis. *Journal of the Operational Research Society*, *63*, 1765–1772. Retrieved from https://doi.org/10.1057/jors.2011.147

Fast, M. (2011). Spinning yarn: Removing the mask encore presentation. *Baseball Prospectus*. Retrieved 2026-08-13, from https://www.baseballprospectus.com/news/article/15093/spinning-yarn-removing-the-mask-encore-presentation/

Goldstein, J. (2026). *Challenge run value (crv): analysis code and synthetic validation corpus.* Zenodo. Retrieved 2026-08-13, from https://doi.org/10.5281/zenodo.21924168

LeDoux, J. (2024). *pybaseball: Pull current and historical baseball statistics using python.* Software package. Retrieved 2026-08-13, from https://github.com/jldbc/pybaseball

Major League Baseball. (2026). *Everything you need to know about the new abs challenge system.* MLB.com. Retrieved 2026-08-13, from https://www.mlb.com/news/abs-challenge-system-mlb-2026

Marchi, M. (2013). The stats go marching in: Catcher framing before pitchf/x. *Baseball Prospectus*. Retrieved 2026-08-13, from https://www.baseballprospectus.com/news/article/20596/the-stats-go-marching-in-catcher-framing-before-pitchfx/

MLB Advanced Media. (2026). *Statcast catcher framing leaderboard.* Baseball Savant. Retrieved 2026-08-13, from https://baseballsavant.mlb.com/leaderboard/catcher-framing

Moskowitz, T. J., & Wertheim, L. J. (2011). *Scorecasting: The hidden influences behind how sports are played and games are won*. New York: Crown Archetype.

Shivakumar, R. (2018). What technology says about decision-making: Evidence from cricket's decision review system. *Journal of Sports Economics*, *19*. Retrieved from https://doi.org/10.1177/1527002516657218

Tango, T. (2018). *Re288: Run expectancy by the 24 base-out states x 12 plate-count states, recursively.* Tangotiger.net. Retrieved 2026-08-13, from http://tangotiger.com/index.php/site/article/re288-run-expectancy-by-the-24-base-out-states-x-12-plate-count-states-recursive/

Tango, T. M., Lichtman, M. G., & Dolphin, A. E. (2007). *The book: Playing the percentages in baseball*. Washington, DC: Potomac Books.