ChatGPT

FORGE Pillar Weight Tuning Deep Research

What’s actually broken and why it shows up as “ranking anomalies”

Your brief makes one thing painfully clear: FORGE’s *pillar math* is doing what you told it to do, but the *pillar semantics* + *weights* + *calibration* combination is rewarding the wrong player archetypes at RB/TE and underweighting the best signal at QB.

There are two distinct failure modes happening at once:

The first failure mode is **misaligned pillar meaning**. Your “stability” pillar (as currently defined) is behaving like “low weekly scoring variance,” which is *not* the same thing as “reliable fantasy value” for RBs and TEs. A player can be “stable” because he’s consistently meh. [1] The same coefficient-of-variation logic is explicitly used in fantasy analysis to measure consistency (standard deviation divided by mean), and it can absolutely crown low-ceiling producers as “consistent” unless you anchor it to *quality of output*. [2]

The second failure mode is **weight + calibration amplification**. Even when the weighted base score difference is tiny, your calibration step can stretch that tiny gap into a big Alpha gap (and, in your Irving vs CMC table, it appears to *flip ordering*). Stretching is plausible if you’re doing any kind of tier-based min–max scaling or percentile remap; flipping is a red-alert sign that either: - “Weighted Base” isn’t the actual input to calibration, - calibration uses other signals beyond base, or - calibration is not strictly monotone in the input score.

In any ranking system, **non-monotone calibration is basically a betrayal of your own scoring**. If the input score says A>B and the published score says B>A, you’ve created a silent second model that overrides the first.

Why correlation tables are useful but not sufficient for picking weights

Your correlation table is a great diagnostic, but it is not a complete recipe for weights.

Correlations tell you whether each pillar *moves with* PPG in isolation. They do **not** tell you how to combine pillars optimally when pillars are correlated with each other (multicollinearity). That matters because “volume,” “team context,” and even “efficiency” often share information (good offenses create more plays, more red-zone trips, better TD rates, etc.). When predictors are correlated, **weights picked by eyeballing correlations routinely over- or under-shoot**. One standard fix is regularized regression (e.g., ridge), which is explicitly used to handle multicollinearity and stabilize coefficients. [3]

If you want weights that behave like “best linear combination of pillars to track fantasy points,” you should treat it as a constrained optimization problem: - Fit standardized pillar scores → PPG per position - Apply regularization (ridge) for stability [3] - Constrain coefficients to match your product requirement (non-negative, sum to 1)

That last part is not hypothetical; it's literally a known constrained regression setup where coefficients lie on a simplex. [4]

So the punchline: **use your correlations to spot "obviously wrong" signs/semantics (like negative stability), but use constrained regression or grid search to land final weights.**

Position-by-position tuning recommendations that match your stated goals

This section assumes your immediate objective is what your validation approach states: higher alignment between Alpha and PPG (and removal of the most embarrassing inversions), with sane tier counts.

Running back

Your own 2025 table says RB PPG↔Volume is extremely high while PPG↔Stability is strongly negative. Even without trusting the exact coefficients, the direction is obvious: **RB "stability" (as currently defined) is anti-signal**.

This is also conceptually consistent with mainstream fantasy analytics: touches (volume) are the engine of RB fantasy scoring, and efficiency is often far more descriptive than predictive. [5]

Strong opinion: for redraft RBs, you should treat "stability" as guilty until proven innocent. Either near-zero it, or redefine it.

A practical redraft RB weight set that should immediately reduce your listed anomalies:

- Volume: **0.60–0.65**
- Efficiency: **0.20–0.25**
- Team Context: **0.10**
- Stability: **0.00–0.05** (unless redefined)

Why I'm comfortable being aggressive here: - High-touch RB ceilings are inherently spiky; punishing that spikiness is punishing the thing fantasy managers pay for. - Multiple fantasy analyses have shown touches are massively correlated with RB fantasy output (one analysis cites ~0.88 corr for touches vs RB points per game across seasons). [6]

Using **your provided pillar scores** for the Irving vs CMC case, simply reducing the stability weight and shifting mass to volume makes the base gap meaningfully favor CMC (instead of ~0.6 points). Under your own "Option A" style RB weights (0.65/0.15/0.10/0.10), the base gap becomes roughly **+3.0** for CMC; with stability near zero it becomes even larger. (This doesn't require faith—this is just arithmetic on your table.)

Tight end

Your table shows TE's stability correlation is even more negative than RB's. That's also consistent with a broader fantasy scoring reality: TE is widely discussed as the most volatile position on a week-to-week basis, and variance measures (like coefficient of variation) show TE volatility is high compared to the other positions. [7]

Recommendation: for redraft TE, do not reward low variance in weekly fantasy points unless you prove it improves roster outcomes in your product.

- Volume: **0.60–0.70**
- Efficiency: **0.15–0.25**
- Team Context: **0.05–0.10**
- Stability: **0.00–0.05** (unless redefined)

If you keep a non-trivial TE stability weight while stability remains "output consistency," you are incentivizing low-ceiling 6–9 point grinders over spike-week league winners. That's not "ranking accuracy;" that's "ranking vibes."

Wide receiver

Your table says WR stability is strongly positive. That's plausible because WR consistency is commonly tied to consistent targets/route participation and role clarity.

But here's the catch: if your WR "stability" is also "output consistency" (not "role stability"), it can still be noisy. A better approach is: **WR stability should primarily mean role stability** (targets per route, route share, target share volatility), not just points volatility—because points volatility contains TD randomness.

Still, given your reported positive signal, increasing WR stability weight is reasonable:

- Volume: **0.45–0.52**
- Efficiency: **0.12–0.18**
- Team Context: **0.12–0.18**
- Stability: **0.18–0.28**

Quarterback

Your table says QB team context is the strongest single-pillar correlation with QB fantasy production, but the current redraft weight favors efficiency.

That's directionally weird for fantasy QBs because: - Team scoring environment, play volume, and offensive strength strongly drive TD opportunities and yardage ceiling. - Efficiency metrics can be descriptive and noisy, especially if they're not stabilized by attempts. (Again: rate stats can be fragile unless you do serious sample-size handling.) [8]

Recommendation: shift QB weights toward context, but don't scrap efficiency—just stop letting it dominate.

A sane QB redraft set:

- Volume: **0.25–0.30**
- Efficiency: **0.28–0.35**
- Team Context: **0.25–0.32**
- Stability: **0.10–0.15** (only if stability is role/attempt stability, not fantasy-point smoothness)

Redefining “stability” so it stops rewarding boring committee backs

If you only tweak weights, you’ll reduce the worst inversions, but the underlying semantic problem remains: **“stability” is not one concept across positions**. Your own brief already points to the right fix: redefine it per position.

Fantasy analysis commonly distinguishes between: - consistency of production (variance of weekly points) and - consistency of role/opportunity (variance of weekly usage). [1]

That distinction is the whole ballgame here.

Recommended stability definitions by position

For RB and TE, redefine stability as **opportunity stability**, and optionally add a separate “upside” concept.

RB stability (role/opportunity stability): - week-to-week touch share volatility (coefficient of variation on touch share) - snap share volatility - high-value touch stability (red-zone touches, targets)

TE stability (role/opportunity stability): - route participation volatility - target share volatility - end-zone / red-zone target stability

WR stability: - primarily target share + route share stability - optionally keep a small production-variance component, but don’t let TD variance dominate

QB stability: - dropback volume stability (attempts + designed rush attempts) - offensive coordinator / scheme continuity (if you have it) - sack/pressure stability is arguably “risk,” not stability; only include if you can validate it

If you insist on “stability inversion” for RB/TE, do it with guardrails

Your Option C (“invert stability to reward variance”) can work, but it’s dangerous if you just do `new = 100 - old` with no gating. That will over-reward chaos merchants who spike once and disappear.

If you want an “upside variance” pillar, it should be **conditional** on baseline usage. A safe framework is:

- UpsideScore = f(90th_percentile_weekly_points, boom_rate)
- Gate: only apply if average volume score (or touch share) exceeds a threshold

This kind of approach matches how variance is treated in best ball research: variance matters, but it’s meaningful when tied to real opportunity rather than random TD flukes. [7]

Calibration and dampening interactions that can secretly blow up rankings

The monotonicity requirement for calibration

If calibration is a monotone transformation of a raw score, the ordering is preserved. That's not just a "nice to have," it's mathematically fundamental for ranking systems.

In the calibration literature, monotone transforms (including isotonic regression) preserve orderings; this is explicitly noted in discussions of post-hoc calibration where monotonic mapping does not degrade rank-based discrimination. [9]

Your Irving vs CMC example *as written* implies the opposite (base says CMC slightly higher, Alpha says Irving higher). That means you should immediately audit calibration for:

- Non-monotone mapping (piecewise logic mistakes, inverted comparator, percentile buckets applied wrong way)
- Calibration using additional signals besides base (e.g., games played, injury flags, tier, or pillar-specific remaps)
- Tier-based rescaling that changes slopes drastically by region of the score distribution

Why calibration might "over-amplify" small differences even when monotone

Even if calibration is monotone, it can still magnify tiny base gaps. The most common reasons:

Tier band stretching: - If you map "T1" players onto a wide Alpha band (say 78–100) but the raw base scores among T1 are tightly clustered, then a 0.5 raw gap can become a 5-point Alpha gap after min–max scaling.

Percentile mapping: - If Alpha is tied to percentile rank of base, the mapping is non-linear in score space, especially in dense regions of the distribution.

If amplification is unwanted, you have three clean options:

Global linear calibration: - Alpha = a·base + b (per position), then clamp to [0, 100]. Simple, stable, predictable.

Global monotone nonparametric calibration: - Fit isotonic regression from base → expected PPG (or base → Alpha target). Isotonic regression is explicitly used when you want a monotone fit without assuming a functional form. [10]
- If you're worried about overfitting, smooth isotonic variants exist and are discussed as reducing overfit/oscillation. [11]

Hybrid: - Linear fit globally, small isotonic correction only in tails.

Dampening as shrinkage: keep it, but consider pillar-specific confidence

Your dampening approach is basically a manual shrinkage-to-prior scheme. Shrinkage/partial pooling is a standard way to reduce small-sample noise by pulling noisy estimates toward a global mean (baseline). [12]

But your current confidence is based only on games played. That's not always the right reliability unit: - Efficiency reliability should scale with attempts/touches/targets, not just games. - Stability reliability should scale with number of games *and* number of meaningful opportunities per game (e.g., a TE with 2 targets per game over 6 games gives you garbage stability estimates).

So even if this isn't the primary root cause, a pillar-specific confidence system is a strong "quality of life" upgrade that prevents fragile pillars from dominating.

A validation and optimization approach that will converge fast and avoid "whack-a-mole" fixes

Your current validation (post-recompute correlations + tier counts + spot checks) is good as a smoke test, but you can tighten it into a real tuning loop.

Optimize for rank correctness, not just Pearson correlation

Pearson corr (what SQL `corr()` gives you) is sensitive to linear scaling. It can look great even when you have nasty local inversions at the top. For a ranking product, you want at least: - Spearman correlation (rank corr) - Pairwise inversion rate among the top N at each position (how often a higher-PPG player is ranked below a much lower-PPG player) - Top-tier fidelity: how many of the top-X PPG players land in T1/T2

Replace hand-picked weight sets with constrained regression + small grid search

A robust pipeline:

Constrained ridge fit per position: - Standardize the four pillar scores - Fit ridge regression to PPG to stabilize against multicollinearity [3] - Convert coefficients to nonnegative weights that sum to 1 using quadratic programming / simplex constraints (this "simplex coefficients" concept is well-established in constrained regression contexts). [4]

Then local grid search around that solution: - Explore ±0.05 around each weight (maintaining simplex constraint) - Score candidates with a composite objective: rank corr + penalty for tier count out of range + penalty for inversion rate on your anomaly list

This gives you: - a principled starting point (instead of vibes) - a controllable "product constraint" (weights that behave like weights, not regression coefficients) - a quick path to a stable configuration

Build regression tests from your anomaly list

Treat each anomaly as a unit test assertion, not just a human spot-check:

- Irving should not outrank CMC under “redraft RB” if your product definition is “best RBs by fantasy value.”
- Gibbs should not be stuck below low-PPG backs due to a “stability” penalty if the stability pillar is about output smoothness rather than role security.
- Your QB1 should not be a low-PPG QB primarily because he has a huge stability score—unless you explicitly decide that “safe boring QB” is the product goal (which would be… a choice).

Deployment-safe changes that minimize risk while fixing the core issue

If you want the fastest path to “ranking anomalies stop happening” without a full rewrite:

Update redraft RB and TE weights immediately: - Reduce stability weight (near zero) - Move mass to volume (and modestly to efficiency)

Redefine RB/TE stability next: - Switch to “role stability” (opportunity consistency), not “points smoothness.” [13]

Audit calibration monotonicity: - Add a unit test: if baseA > baseB then alphaA ≥ alphaB for same position and season. - If calibration is tier-stretching, cap the stretch factor so small base gaps don’t explode.

Only then revisit dynasty: - Dynasty RB stability at 0.40 is only defensible if stability means “multi-year role security / durability,” not “weekly points uniformity.” Otherwise it’s going to be even more wrong than redraft. The fix is semantic first, weight second.

The headline: **zeroing out RB/TE stability (as currently measured) is the quickest fix; redefining stability is the correct fix; enforcing monotone calibration is the non-negotiable fix.**

---

[1] [2] [13] Metrics that Matter: Consistency in fantasy scoring, role
https://www.pff.com/news/fantasy-football-metrics-that-matter-consistency-in-fantasy-scoring-role

[3] What Is Ridge Regression? | IBM
https://www.ibm.com/think/topics/ridge-regression

[4] Regressions where the coefficients are a simplex.
https://freerangestats.info/blog/2024/11/06/simplex-regression-coefficients

[5] [8] Running Back Stats That Matter for Fantasy Football
https://www.sharpfootballanalysis.com/fantasy/running-back-stats-that-matter-fantasy-football-2025/

[6] The Biggest Gaps Between Real-Life and Fantasy Football Value
https://ftnfantasy.com/nfl/the-biggest-gaps-between-real-life-and-fantasy-football-value-2

[7] Weekly Variance By Position - A Key To Best Ball | Underdog Network
https://underdognetwork.com/football/best-ball-research/weekly-variance-by-position-a-key-to-best-ball

[9] [11] Smooth Isotonic Regression: A New Method to Calibrate Predictive Models - PMC
https://pmc.ncbi.nlm.nih.gov/articles/PMC3248752/

[10] Isotonic regression - Wikipedia
https://en.wikipedia.org/wiki/Isotonic_regression

[12] Hierarchical Partial Pooling — PyMC example gallery
https://www.pymc.io/projects/examples/en/latest/case_studies/hierarchical_partial_pooling.html