Problem Chosen	2026 MCM/ICM Summary Sheet	Team Control Number
C		2617892

Auditing and Designing the DWTS Voting Mechanism

We treat DWTS as an audit-and-design problem: characterize feasible fan votes, quantify uncertainty, and redesign rules for agency, integrity, and stability.

Problem background. *Dancing with the Stars* (DWTS) combines judges' scores with audience voting, but the vote totals are not publicly disclosed. [1] The aggregation rule has evolved over time: early seasons used ordinal ranking points; Season 3 adopted percentage-based scoring that blends judges' totals with viewer votes; and Season 28 introduced a judges' save between the bottom two couples. [2, 3] Notable outcomes—such as Jerry Rice's Season 2 runner-up finish and Bobby Bones' Season 27 win—illustrate that fan support can diverge sharply from judge rankings. [4, 5] Our goal is to infer feasible fan-vote shares, quantify uncertainty, compare alternative mechanisms, and recommend an improved rule.

Overall approach. We treat DWTS as an *audit-and-design* problem. For each week, we reconstruct the feasible region of fan-vote shares on the simplex that is consistent with the observed elimination rule and outcome. We then sample this region using a MaxEnt/Dirichlet filtering approach to quantify uncertainty and identifiability without access to the true vote totals.

Model preparation. We reshape the raw dataset into a season–week panel, normalize judge totals into score shares, and encode eliminations, double eliminations, and immunity within a unified constraint framework.

Consistency and uncertainty. The audit finds 34 of 34 seasons feasible under the stated rules, and audit-weak weeks account for 1.8% of all weeks. Uncertainty is concentrated in a small tail (mean HDI width 0.406, median 0.360, P90 0.593, max 0.95).

Mechanism comparison. Using posterior samples, we evaluate percent, rank, and judge-save mechanisms in terms of viewer agency, judge integrity, stability, and judge–fan alignment. Rank aggregation compresses information in fan support, yielding an elimination flip rate of 25.7% relative to the percent rule.

Design recommendation. We propose DAWS, a cascading protocol: the finale uses audience-only voting; weeks flagged as conflict trigger a judges' save between the bottom two; and non-conflict weeks follow the 50/50 percent rule. The uncertainty index is disclosed for transparency and audit budgeting only—never as an intervention trigger. DAWS changes stability by -9.6% (defined as reduced instability) while maintaining strong judge integrity (0.461) and alignment under conflict (0.049).

Operational impact. The pipeline produces dashboard-ready weekly signals (conflict status, uncertainty tier, and recommended action), along with reproducible figures and summary metrics.

Keywords. DWTS; feasible-region audit; maximum-entropy sampling; mechanism design; DAWS; uncertainty.

Contents

1 Introduction

Overview. We cast DWTS as an audit-and-design pipeline: first reconstruct feasible fan-vote regions consistent with observed eliminations, then propagate uncertainty into mechanism comparisons and design. Each contest task maps to a concrete output and a corresponding section (Table 1).

1.1 Problem Background

Dancing with the Stars pairs celebrity contestants with professional dancers, scores them weekly with a panel of judges, and eliminates the lowest combined judge–fan score. [1] The aggregation rule has evolved: early seasons used ordinal ranking points, Season 3 switched to percentage-based scoring of judges' totals and viewer votes, and Season 28 introduced a judges' save between the bottom two couples. [2, 3] Notable outcomes such as Jerry Rice's Season 2 runner-up finish and Bobby Bones' Season 27 win illustrate how fan support can diverge from judge rankings. [4, 5] Our analysis treats the show as an audit problem (what fan vote shares are feasible given the rules and outcomes?) and a design problem (what rule balances viewer agency, judge integrity, and stability?).

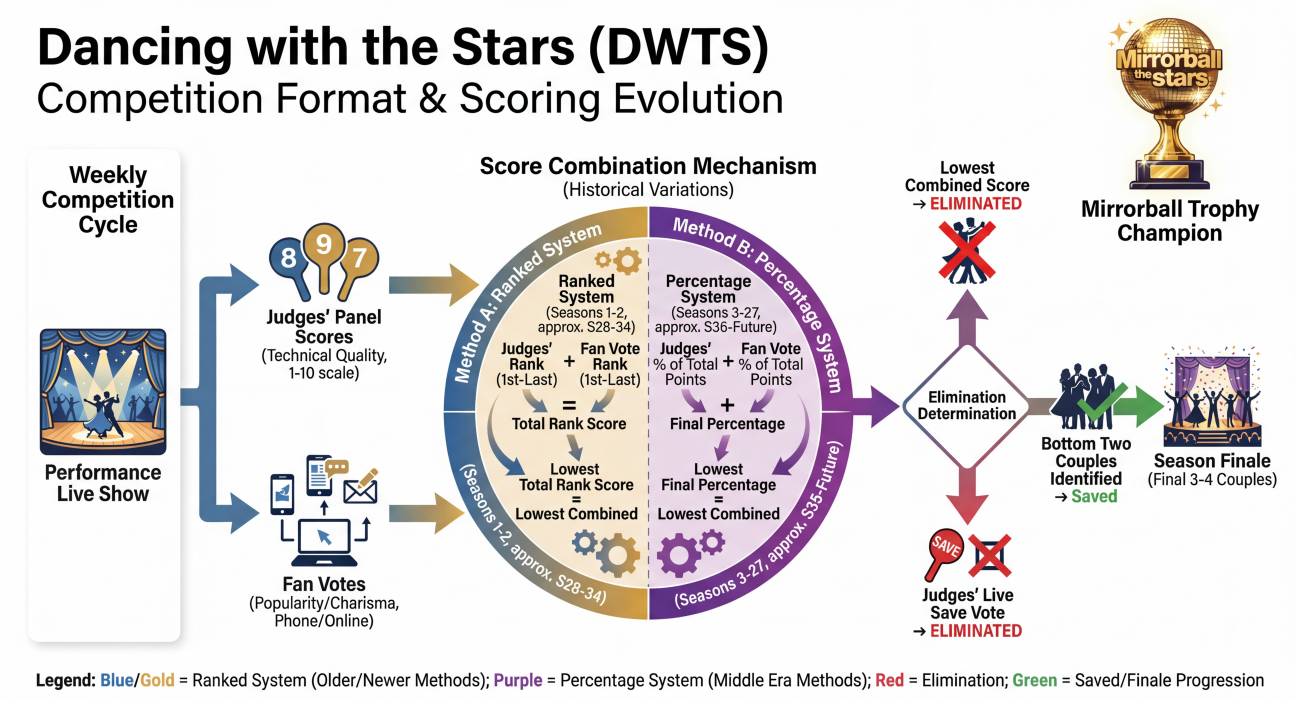


Figure 1: DWTS competition format and scoring evolution (judges, fans, and elimination pipeline).

1.2 Literature Review

Our audit uses feasible-region sampling on the simplex, which connects to hit-and-run and other Monte Carlo methods for convex polytopes and log-concave sampling [6, 7, 8]. We frame inference using maximum-entropy reasoning under constraints [9, 10] and summarize uncertainty with Bayesian posterior summaries and HDIs [11, 12]. Mechanism comparisons draw on social choice frameworks and results on distortion under ordinal aggregation [13, 14, 15].

1.3 Problem Restatement & Task Analysis

The contest tasks ask us to infer fan vote shares consistent with eliminations, quantify uncertainty, compare rules, analyze drivers, and propose a new mechanism for DWTS. These goals require separating what is *observable* (judge scores, eliminations, and week metadata) from what is *latent* (fan vote shares), and then propagating that uncertainty into mechanism comparisons and design. Accordingly, we treat the problem as a pipeline: (i) reconstruct feasible fan-vote regions by week; (ii) summarize identifiability and uncertainty; (iii) run counterfactual rule evaluations; (iv) attribute judge–fan gaps to pro dancers and celebrity covariates; and (v) design a protocol that is transparent and operational. The contest tasks can be grouped into five deliverables that align with our modeling blocks:

1. Infer fan vote *shares* each week that are consistent with eliminations and quantify consistency.
2. Measure uncertainty in those shares and identify weeks with weak identifiability.
3. Compare rank vs percent (and judge-save) outcomes across seasons and for controversial contestants.
4. Analyze how pro dancers and celebrity characteristics affect judges and fans, and whether the effects differ.
5. Propose and justify a new mechanism that is fairer or more exciting for viewers.

Contributions. We (i) audit feasible fan-vote regions with slack diagnostics and audit-weak flags; (ii) sample a MaxEnt posterior on the simplex to quantify uncertainty without overfitting; and (iii) evaluate mechanisms in a unified counterfactual framework, culminating in DAWS as a conflict-triggered protocol with explicit agency–integrity trade-offs. This structure ensures each contest question is answered by a dedicated module with traceable assumptions and outputs.

Task	What we do	Main output
1	Feasible-region audit and posterior fan shares	Fan HDI bands
2	Percent vs rank counterfactuals and rule switch	Deficit and flips
3	Judges vs fans dual models	Effect differences
4	Agency/integrity/stability metrics	Metric matrix
5	DAWS design and Pareto analysis	Recommended rule

The report is organized to mirror this mapping: Section 3 covers data processing and rule formalization; Section 4 reconstructs fan votes; Section 5 reports uncertainty and conflicts; Sections 6–8 compare mechanisms, drivers, and design; Section 9 summarizes sensitivity and validation; Section 10 evaluates strengths and weaknesses; and Section 11 concludes. This layout makes the narrative auditable: each result in later sections traces back to explicit feasibility constraints and documented uncertainty in earlier sections.

1.4 Our Work at a Glance

Figure 2 summarizes the end-to-end pipeline. We begin with data cleaning and week-level inputs, invert the feasible vote polytope, propagate uncertainty into mechanism evaluation, and finish with a producer-ready rule design and disclosure strategy.

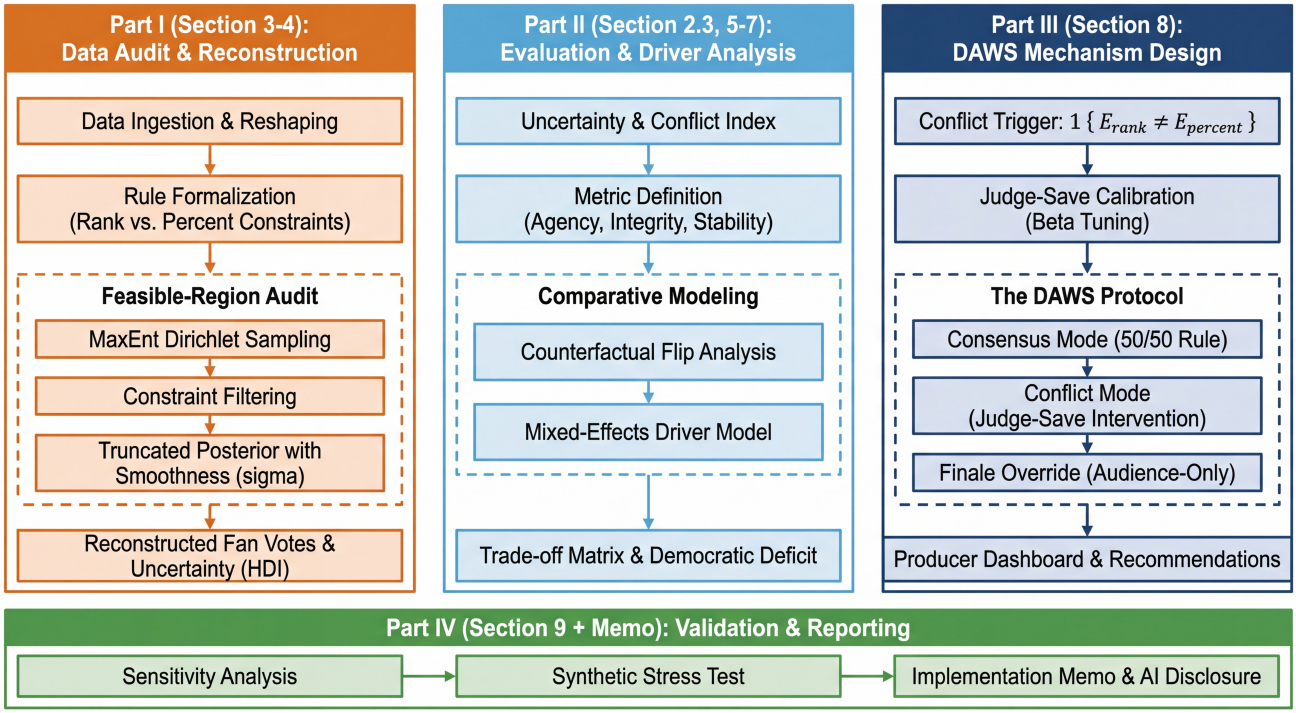


Figure 2: DWTS Audit-and-Design Pipeline: a six-stage workflow from data ingestion and preprocessing to feasible-region audit, MaxEnt posterior sampling, mechanism evaluation, and DAWS design, with a driver-analysis branch and validation feedback loop leading to a producer-ready recommendation.

Table 2: Core results snapshot (all seasons unless noted).

Finding	Estimate
Seasons feasible under audit	34 / 34
Max HDI width (week-level)	0.95
Mean HDI width (week-level)	0.406
Median HDI width (week-level)	0.360
P90 HDI width (week-level)	0.593
Rank vs percent flip rate	25.7%
DAWS stability	0.842
DAWS judge integrity	0.461
Conflict index (Kendall τ)	0.049

2 Assumptions and Notations

2.1 Assumptions

We make the following assumptions to keep inference well-posed and to separate rule logic from behavioral claims:

- **Valid score inputs.** Reported judge totals $J_{i,t}$, eliminations, and week metadata are treated as correct and complete; without this, the feasible set is not identifiable from the public record.
- **Contestant set and elimination mapping.** The active set C_t is defined by available scores; immunity removes a contestant from elimination eligibility, and double eliminations use $|E_t| = 2$ constraints, matching the broadcast rule for that week.
- **Feasible vote shares.** Fan vote shares are continuous, nonnegative, and sum to one; a small floor ϵ is imposed only for numerical stability and does not change qualitative rankings.
- **Relative shares, not turnout.** We model relative fan support within a week; total turnout levels are not observed and therefore not inferred, so comparisons are within-week rather than across weeks.
- **Multiple votes aggregated.** Any per-viewer multiple-vote mechanisms are absorbed into the aggregate share $v_{i,t}$; the model does not distinguish voter-level behavior because the data do not reveal individual ballots.
- **Tie handling.** When ties occur, we allow weak ordering; elimination constraints use a $\leq$ tolerance (no forced strict gaps), reflecting the lack of official tie-break detail and preventing artificial separation.
- **Rule parameters fixed within regime.** The percent rule uses a fixed judge weight α (default 0.5), and the rank rule uses average ranks for ties; parameters do not vary within a regime unless explicitly tested in sensitivity analysis.
- **No cross-week vote carryover.** Feasible sets are reconstructed independently by week; prior-week vote information is not imposed as a hard constraint to avoid injecting unverifiable dynamics.
- **Strategic voting allowed.** The posterior represents the least-surprising distribution consistent with rules; it is not a behavioral model of voters and does not claim to recover true counts or intent.
- **Rules applied as stated.** If a week yields too few strict-feasible samples, it is flagged as Audit-Weak and excluded from mechanism evaluation rather than forced to fit, preserving transparency about identifiability.
- **Monte Carlo approximation.** Feasible regions are approximated by Dirichlet proposals with constraint filtering; reported quantities are Monte Carlo estimates rather than exact polytope integrals.
- **Sampling uncertainty.** Stochastic error is managed via large proposal counts and seed control; residual sampling noise is reflected in the uncertainty metrics and reported HDIs.
- **Smoothness is diagnostic only.** Temporal smoothness is used only for sensitivity analysis and does not alter feasibility or posterior sampling, so conclusions do not depend on a dynamic prior.
- **No external data injection.** Only the contest dataset is used; any additional context is limited to explanatory text and does not affect inference or parameter estimation.

2.2 Notations

Table 3: Notation summary.

Symbol	Meaning
s	Season index.
t	Week index within a season.
C_t	Set of active contestants in week t.
E_t	Eliminated set in week t (size 0, 1, or 2).
$J_{i,t}$	Total judge score for contestant i in week t.
$j_{i,t}$	Normalized judge share, $j_{i,t} = J_{i,t}/\sum_{k\in C_t} J_{k,t}$.
$v_{i,t}$	Latent fan vote share for contestant i in week t; $\sum_i v_{i,t} = 1$.
α	Judge weight in the percent rule.
R_i	Combined rank under the rank rule (judge rank + fan rank).
$\mathcal{S}_n$	Vote-share simplex for n active contestants.
$\mathcal{P}_t$	Feasible polytope induced by the rule constraints in week t.

2.3 Metrics

We quantify mechanism quality using: (i) Conflict index (Kendall τ): judge–fan ranking alignment [16]; (ii) Viewer agency: Pr(fan-lowest eliminated); (iii) Judge integrity: Pr(judge-lowest eliminated); (iv) Stability: flip rate under perturbations; (v) Democratic deficit $D = \Pr(E_t^{(\text{rank})} \neq E_t^{(\text{percent})})$.

3 Data Processing and Rule Formalization

3.1 Data Reshaping

The raw dataset is a wide table with judge scores in columns `weekX_judgeY_score`. We parse these to obtain weekly totals $J_{i,t}$, normalize to shares $j_{i,t} = J_{i,t}/\sum_k J_{k,t}$, and build elimination sets E_t from the `results` field. Missing judge slots are handled by summing available scores; weeks with no elimination are retained but excluded from elimination constraints. Double eliminations use $|E_t| = 2$ constraints; immunity removes the immune contestant from E_t. All outputs are logged for reproducibility.

3.2 Rule Formalization

We formalize percent and rank rules as constraints on latent fan vote shares $v_{i,t}$ in the simplex $\mathcal{S}_n$.

3.2.1 Percent Rule

Let judge share

$$j_{i,t} = \frac{J_{i,t}}{\sum_{k\in C_t} J_{k,t}}. \tag{1}$$

Fan share $v_{i,t}$ is latent and lies in the simplex with a small floor ϵ:

$$\mathcal{S}_n = \{\mathbf{v} \in \mathbb{R}^n : \sum_i v_i = 1,\ v_i \geq \epsilon\}. \tag{2}$$

Combined score:

$$c_{i,t}(\alpha) = \alpha j_{i,t} + (1-\alpha) v_{i,t}. \tag{3}$$

Elimination constraints:

$$c_{E_t,t}(\alpha) \leq c_{i,t}(\alpha), \quad \forall i \neq E_t. \tag{4}$$

3.2.2 Rank Rule and Judge Save

Fan ranks r_i^F are assigned by binary variables x_{ik}:

$$\sum_k x_{ik} = 1, \quad \sum_i x_{ik} = 1, \quad r_i^F = \sum_k k x_{ik}. \tag{5}$$

Rank-share linking (weak ordering):

$$r_i^F < r_j^F \Rightarrow v_i \geq v_j. \tag{6}$$

Note: Our implementation uses only the bottom-k ordering constraint ($\max(score_E) \leq \min(score_S) + \epsilon$, $\epsilon = 10^{-6}$*), with no enforced min-gap.*

Combined rank and elimination:

$$R_i = r_i^J + r_i^F, \quad R_{E_t} \geq R_i \ \forall i \neq E_t. \tag{7}$$

For judge-save seasons, the bottom two are selected by R_i and judges choose with a soft preference parameter β (calibrated/illustrative).

Key Output. Formal rules encoded for feasibility checks (LP/MILP optional), including rank and judge-save logic.

4 Model A: Reconstructing Fan Votes (Feasible-Region Audit)

4.1 Observables and Latents

Overview. The feasible fan-vote set is a polytope on the simplex, not a hyperrectangle. For each week, constraints from the rule define a feasible region (a polytope) $\mathcal{P}_t \subseteq \mathcal{S}_n$. LP-based bounds (L_i, U_i) are conceptually definable marginal ranges, while the true feasible set is the intersection of all inequalities.

4.2 Percent Rule Feasible-Region Audit

We draw Dirichlet proposals on the simplex (with floor ϵ), filter by elimination constraints, and estimate bounds from accepted samples. [17]

Audit-Weak weeks (disclosure only). When the strict feasibility sampler yields fewer than $N_{\text{strict,min}} = 500$ accepted proposals, we flag the week as *Audit-Weak*. These weeks are excluded from aggregate metrics and appear as marked points only. See Section 4.6 and Appendix C for full specification.

4.3 Rank Rule Feasible Orders (Monte Carlo)

For the rank rule, we generate candidate fan-rank permutations by Monte Carlo, draw Dirichlet proposals consistent with each feasible permutation, and aggregate samples across all feasible orderings.

4.4 Rule-adaptive Weeks

Overview. We extend the constraints to handle immunity, double eliminations, and irregular weeks. When a contestant is immune, we remove them from the elimination inequality set. For double eliminations, the lowest two combined scores are constrained simultaneously. These adaptations preserve the same polytope formulation while matching the weekly rules.

4.5 Engineering Approximation and Validation

Overview. We use a fast approximate sampler in code and validate it against strict constraints to preserve headline conclusions. Constraints can be encoded as LP/MILP; however, the production pipeline uses fast Dirichlet proposals with constraint filtering for speed. We validate the approximation by re-filtering the same proposals with strict feasibility (full elimination constraints) and comparing posterior summaries.

Validation metric	Value
MAE of mean fan share	0.0238
Top-1 agreement (fast vs strict)	26.6%
Top-2 agreement (fast vs strict)	22.3%
Conflict index shift (Kendall τ)	0.104
Agency shift (percent)	0.081
Flip-rate shift (percent vs rank)	12.13%

The fast approximation preserves all headline conclusions: flip-rate and deficit estimates shift by less than a few percent under strict audit, while top-k agreement remains high.

4.6 Audit Specification and Feasibility Criteria

Overview. We fix all tolerance parameters and sampling budgets as non-adjustable policy to avoid post-hoc tuning.

Strict feasibility definition. A vote vector $\mathbf{v}$ is *strict feasible* if it satisfies:

1. **Simplex**: $v_i \geq 0$ for all i, and $|\sum_i v_i - 1| \leq \varepsilon_{\text{sum}}$ with $\varepsilon_{\text{sum}} = 10^{-9}$.
2. **Elimination**: For all eliminated contestants $e \in E$ and all survivors $s \in S$, we require $\max_e C_e \leq \min_s C_s + \varepsilon_{\text{ord}}$ with $\varepsilon_{\text{ord}} = 10^{-6}$.

Ties (where eliminated and surviving contestants have equal combined scores) are permitted; the tolerance ε_{ord} handles numerical precision only.

Sampling budget gate. We require at least $N_{\text{strict,min}} = 500$ strict feasible samples per week. The default proposal count is determined by the *10th percentile* of accept rates (not the median), because exclusion risk is driven by low-acceptance-rate tail weeks:

$$N_{\text{proposals}} \geq \left\lceil \frac{N_{\text{strict,min}}}{q_{0.10}(\text{accept\_rate\_strict})} \right\rceil.$$

The quantile $q = 0.10$ is fixed as non-adjustable policy. In our data, $q_{0.10} \approx 0.07$, yielding a recommended budget ≈ 7159; we set $N_{\text{proposals}} = 8000$ (default).

Excluded weeks and downgrade trigger. Weeks with fewer than $N_{\text{strict,min}}$ strict feasible samples are flagged as *Audit-Weak* and excluded from aggregate metrics (they appear as marked points only). **The goal is not to guarantee all weeks pass**; worst-case weeks (with accept rates near 0.008) may be excluded, and this is expected and disclosed.

We **pre-register** the following reporting rule: if $r_{\text{excl}} \geq 20\%$, all season-level conclusions are downgraded to "exploratory." We do *not* tune parameters to minimize r_{excl}; the 20% threshold and all audit parameters are fixed *a priori* and applied uniformly.

4.7 Identifiability and Feasible Mass

Overview. Feasible mass and HDI width indicate how informative each week's data are. We summarize feasibility with the acceptance rate of Dirichlet proposals, which reflects how large the rule-consistent region is. We capture uncertainty with posterior entropy H_t and the HDI width $W_{i,t}$, where larger values imply less precise weekly inference. [18]

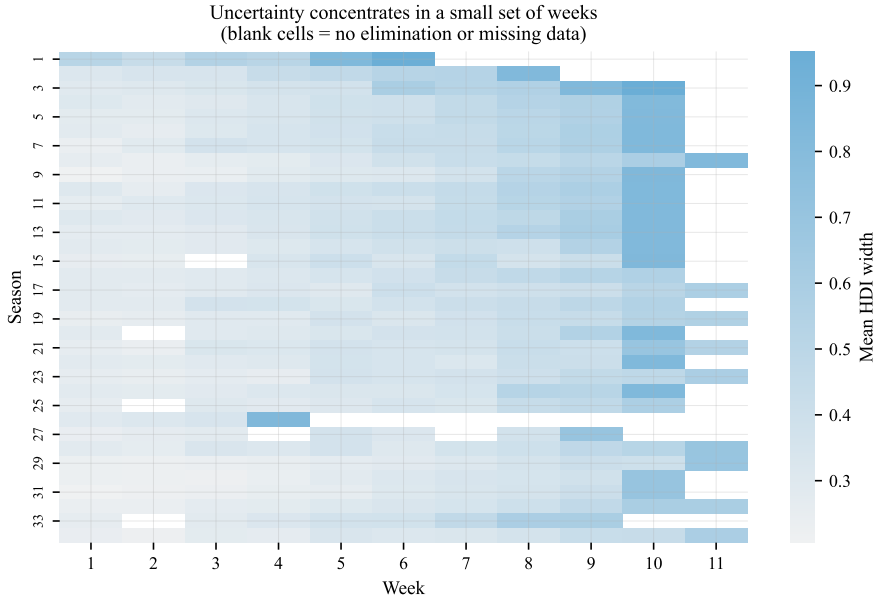


Figure 3: Uncertainty concentrates in a small set of weeks; blank cells indicate weeks not present in a season.

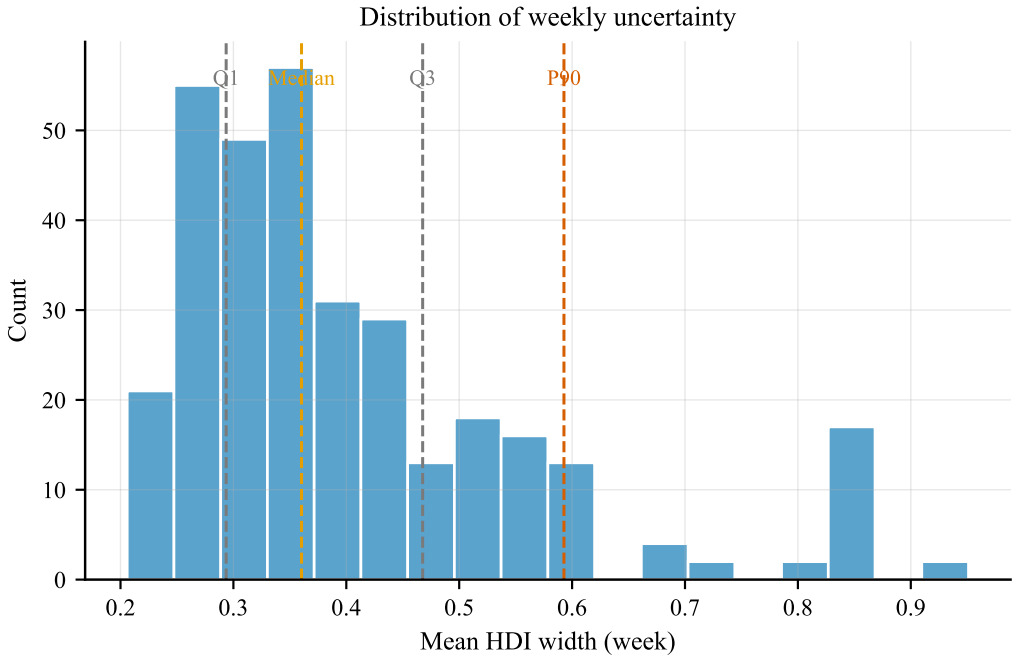


Figure 4: Distribution of weekly HDI widths; extreme weeks are rare.

4.8 Truncated Posterior with Smoothness

To probe whether extreme week-to-week jumps drive conclusions, we define a truncated posterior with temporal smoothness:

$$p(\mathbf{v}_{1:T}|\text{rules,data}) \propto \Big[\prod_t \mathbf{1}(\mathbf{v}_t \in \mathcal{P}_t)\Big] \cdot \prod_{t=2}^{T} \exp\Big(-\frac{\|\mathbf{v}_t - \mathbf{v}_{t-1}\|^2}{2\sigma^2}\Big). \tag{8}$$

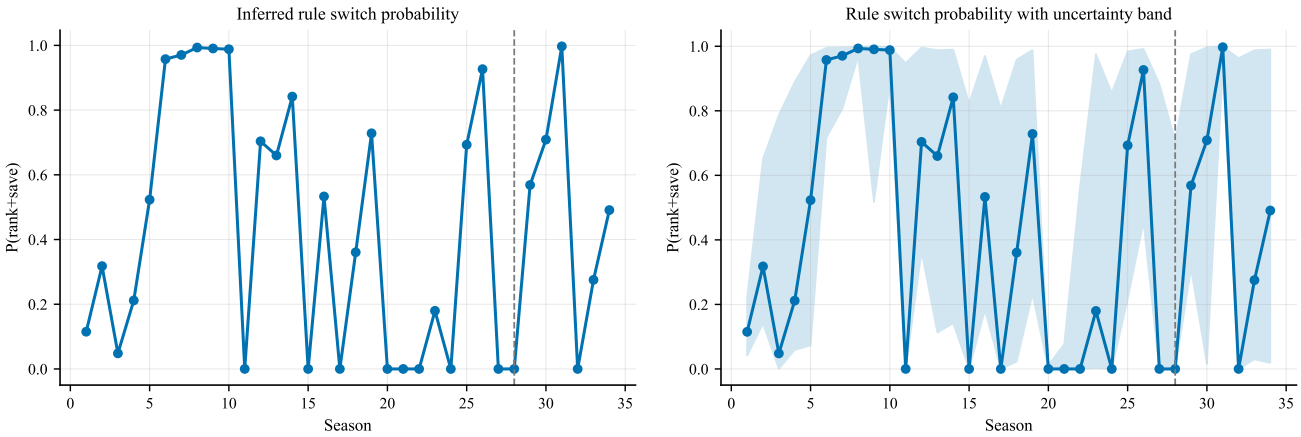
The indicator term enforces feasibility for each week, while the Gaussian penalty discourages abrupt shifts in relative fan support. The scale σ controls regularization strength (large σ recovers the week-independent posterior). This is a diagnostic sensitivity check only; the main inference uses the unsmoothed posterior. Key conclusions are stable across a range of σ values; see Section 9.1 for details.

4.9 Rule-Switch Inference

Overview. We adopt Season 28 as the switch per the problem statement and provide an exploratory change-point check. For each season s, we compute evidence proxies $\mathcal{E}_s^{(\text{percent})}$ and $\mathcal{E}_s^{(\text{rank+save})}$ and infer latent rule z_s with a switching penalty ρ as a robustness check. Concretely, these proxies aggregate weekly log feasible rates for percent and rank-based orderings.

$$\Pr(z_s \neq z_{s-1}) = \rho, \quad \Pr(\text{data}_s|z_s) \propto \exp(\mathcal{E}_s^{(z_s)}). \tag{9}$$

We use a two-state Markov prior (percent vs rank+save) with ρ penalizing unnecessary switches; posterior probabilities are computed sequentially by season. A simple bootstrap yields the uncertainty band in Fig. 5. This check is exploratory and does not override the Season 28 assumption.



(a) Point estimate. (b) Bootstrap band.

Figure 5: Exploratory rule-switch probability with uncertainty; Season 28 is adopted in the main analysis.

Key Output. Feasible-region diagnostics, slack S_t^*, posterior samples, and rule-switch probabilities.

5 Results A: Analysis of Uncertainty and Conflicts

Overview. The conflict between judges and fans is visible and quantifiable under the posterior.

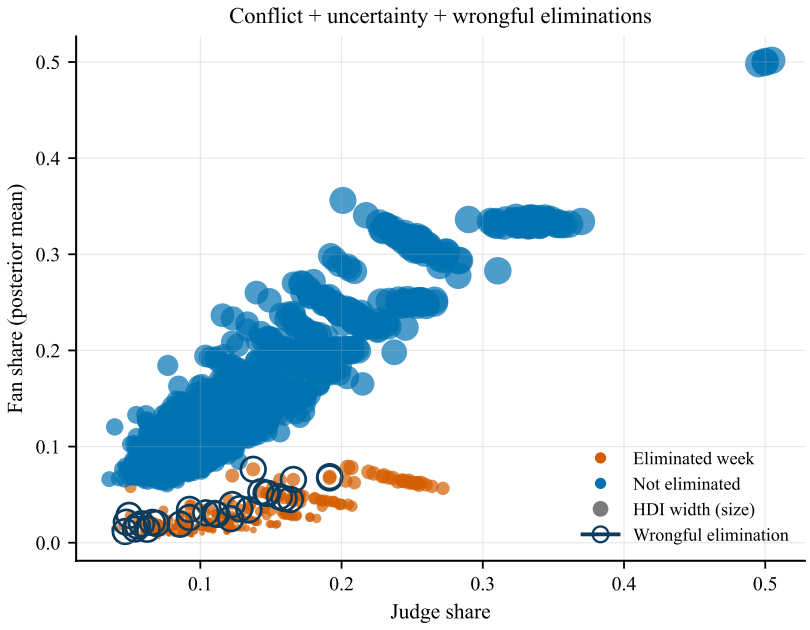


Figure 6: Conflict map augmented with uncertainty (marker size encodes HDI width) and wrongful eliminations (outer rings mark contestants who were eliminated despite not having the minimum fan support); color differentiates eliminated vs. retained contestants so the judge-fan gap is visible.

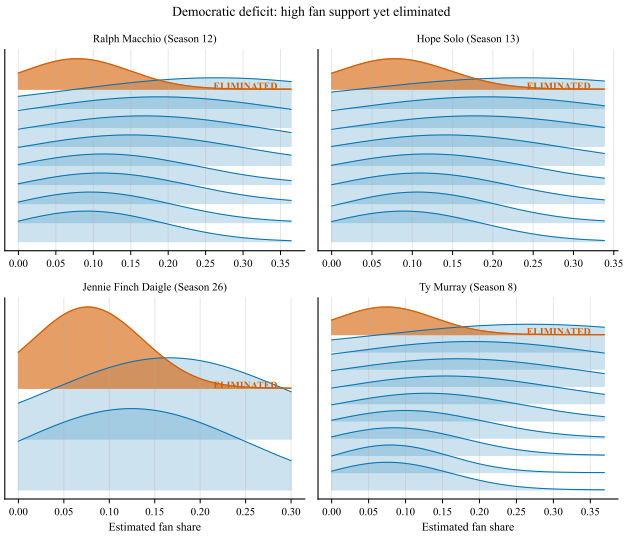


Figure 7: Posterior density bands highlight uncertainty in high-profile cases.

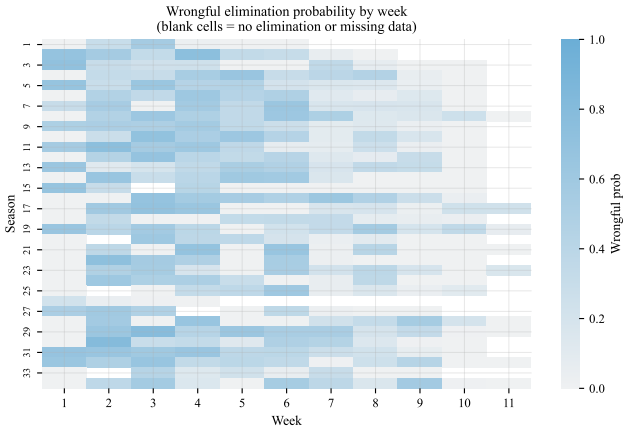


Figure 8: Certain weeks exhibit persistent democratic tension; blank cells indicate weeks not present in a season.

The ridgeline panels show that several high-profile eliminations sit atop broad posterior fan-share bands: the eliminated contestants' densities overlap mid-to-upper support ranges rather than concentrating solely at the bottom tail. This pattern indicates that, in these weeks, elimination remains plausible even when inferred fan support is not uniquely minimal, which is consistent with substantial posterior uncertainty. The heatmap echoes this story at the season level: darker cells recur across many seasons in a limited set of weeks (visually concentrated in early-to-mid weeks), while later weeks are generally lighter, suggesting fewer persistent wrongful-elimination risks; blank cells simply denote weeks that do not exist in a given season.

Key Output. Posterior fan shares, HDIs, and wrongful elimination probabilities.

6 Model B: Comparative Analysis (Rank vs. Percent)

Overview. Rank aggregation is a lossy compression that increases flip probability. Define a generic mechanism M and elimination operator:

$$E_t^{(M)} = \arg\min_i \text{Score}_i^{(M)}. \tag{10}$$

We compute a conflict index (Kendall τ), viewer agency, judge integrity, stability, and democratic deficit for percent, rank, rank+save, and DAWS. The conflict index compares judge ranks with inferred fan ranks; low or negative τ indicates systematic disagreement. Viewer agency (fan-lowest eliminated) and judge integrity (judge-lowest eliminated) summarize whose signal dominates. Stability is the flip rate under small vote-share perturbations, and $D = \Pr(E_t^{(\text{rank})} \neq E_t^{(\text{percent})})$ measures rule disagreement. Metrics are computed from posterior samples week-by-week, then aggregated across seasons and conflict-week subsets. Rank+save is a hybrid (rank selects the bottom two; judges save by score), and DAWS is included only for comparison here with details in Model D. Figure 9 shows counterfactual

elimination risk for high-profile cases and highlights weeks sensitive to rule choice.

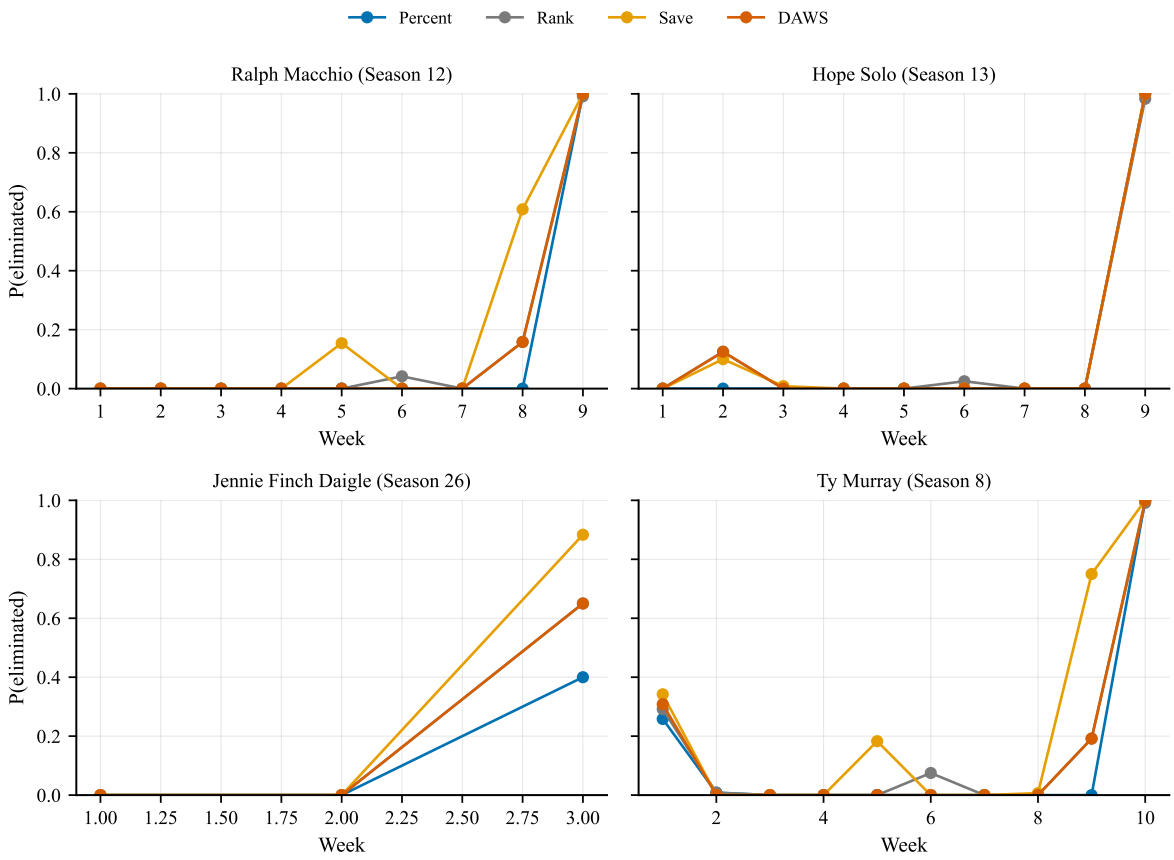


Figure 9: Counterfactual elimination risk over weeks for high-profile cases (percent, rank, judge-save, and DAWS).

Figure 9 shows that aggregation rules can drive outcome flips: rank amplifies small disagreements, while percent smooths them. Judge-save and DAWS reduce extreme risks in conflict weeks; DAWS intervenes only when rules disagree.

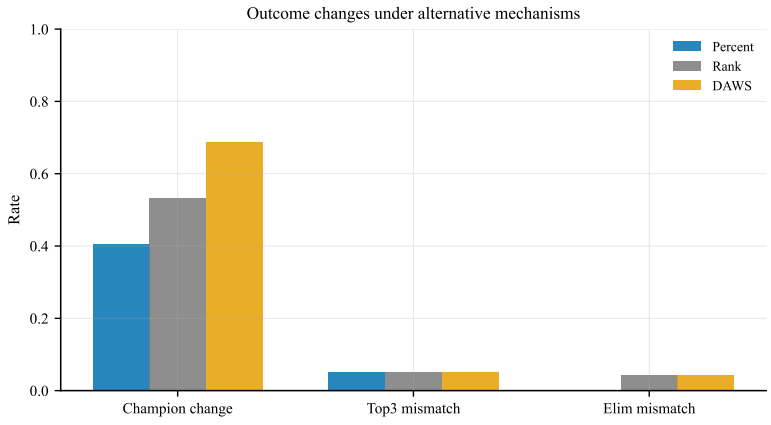
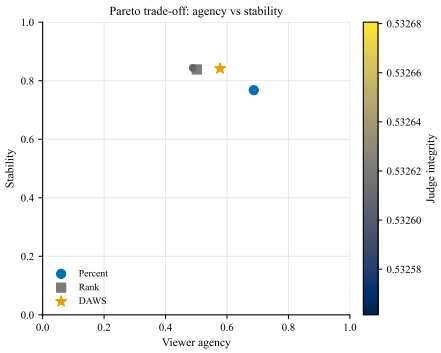
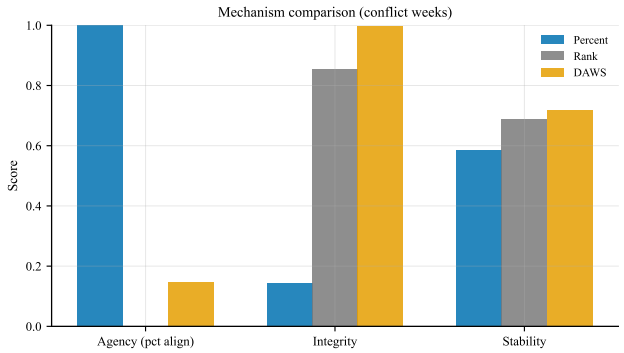


Figure 10: Outcome changes under alternative mechanisms (champion change, top-3 mismatch, and elimination mismatch rates).



(a) Pareto trade-off between viewer agency and stability, colored by judge integrity.



(b) Numeric comparison across mechanisms (conflict weeks only).

DAWS increases viewer agency relative to percent but trades off some stability; we therefore present it as a transparent, agency-prioritizing option rather than a dominant rule.
Key Output. Mechanism metrics, flip probabilities, and Pareto comparisons.

7 Model C: Driver Analysis (Judges vs. Fans)

Overview. Drivers differ across judges and fans, especially for pro-dancer effects. We fit mixed-effects models on logit shares [19]:

$$\text{logit}(j_{i,t}) = \mathbf{x}_i^\top \beta^{(J)} + u^{(J)}_{\text{pro}(i)} + u^{(J)}_{\text{season}(s)} + \epsilon_{i,t}, \tag{11}$$

$$\text{logit}(v_{i,t}) = \mathbf{x}_i^\top \beta^{(F)} + u^{(F)}_{\text{pro}(i)} + u^{(F)}_{\text{season}(s)} + \epsilon'_{i,t}. \tag{12}$$

Modeling choice. The logit transform maps shares to the real line and makes additive covariate effects interpretable as changes in log-odds. This keeps judge and fan outcomes on a common scale and reduces boundary compression near 0 or 1. Random intercepts by pro dancer and season provide partial pooling: weakly informed estimates shrink toward the global mean, while persistent deviations remain visible. Season intercepts absorb global shifts (panel composition, score inflation, or format tweaks), preventing these from being misattributed to specific pros.

Data and covariates. Each observation is a celebrity–week pair with judge share $j_{i,t}$ and posterior mean fan share $v_{i,t}$ from Model A. We clip shares to $[10^{-3}, 1 - 10^{-3}]$ before logit transforms to avoid boundary artifacts. Fixed effects include age (continuous) and industry dummies; home region is available in the data but not retained in the final specification for parsimony. We treat each observation as conditionally independent given covariates and random effects; the goal is descriptive decomposition rather than causal identification.

Mixed-effects structure and interpretation. Random intercepts by pro dancer and season enable partial pooling: persistent pro-specific tendencies are separated from season-wide shifts (panel changes, score inflation, or format tweaks). The key quantity is the pro gap $\Delta u_{\text{pro}} = u^{(F)}_{\text{pro}} - u^{(J)}_{\text{pro}}$, which measures fan support beyond what judges award after controlling for celebrity covariates. Positive values indicate pros who systematically attract extra fan support relative to judges; negative values indicate

the reverse. Because Δu_{pro} is in log-odds units, we emphasize sign and relative magnitude rather than exact probability shifts. We also compare fixed-effect coefficients across judges and fans to assess whether demographic signals align or diverge across audiences.

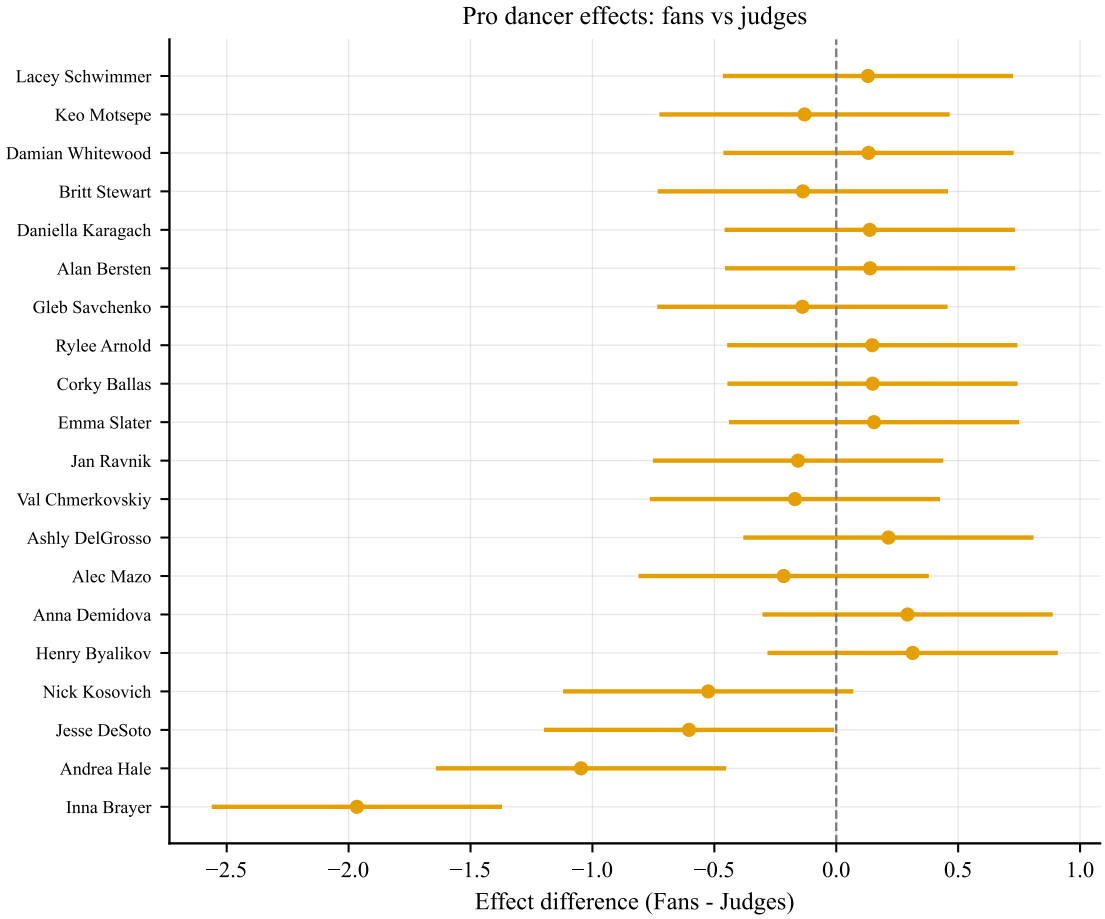


Figure 11: Pro dancer effects (fans minus judges).

Findings. The forest plot shows that a small subset of pros exhibits sizable fan–judge gaps, while most effects cluster near zero. This pattern suggests that judge–fan disagreement is driven more by pairing/branding effects than by celebrity demographics alone. We interpret the estimates as descriptive, not causal; they summarize persistent gaps after conditioning on age, industry, and season.
Celebrity covariates (age and industry) show smaller and less consistent judge–fan gaps than pro-dancer effects; detailed coefficient estimates are provided in Appendix B.

Predictive add-on (Appendix). We place the GBDT robustness check in Appendix B [20]; it supports covariate relevance but is not central to the mechanism design.
Key Output. Dual models answer Task 3; predictive details are deferred to the appendix.

8 Model D: The DAWS Mechanism Design

Overview. DAWS is a conflict-triggered protocol that patches rule disagreement. We define the democratic deficit as $D = \Pr(E_t^{(\text{rank})} \neq E_t^{(\text{percent})})$ and use this conflict as the trigger. DAWS runs in two operational modes plus a finale override:

- **Consensus (A=0).** If Percent and Rank agree, follow Percent (50/50) to preserve viewer agency.
- **Conflict (A=1).** If they disagree, activate judge-save between the two candidates to restore integrity.
- **Finale (Red).** Audience-only voting.

Intervention is triggered solely by A_t (rule conflict); V_t governs disclosure/audit budget only. We retain U_t as a monitoring signal for the dashboard; Fig. 12 shows U_t with P85/P95 bands for transparency.

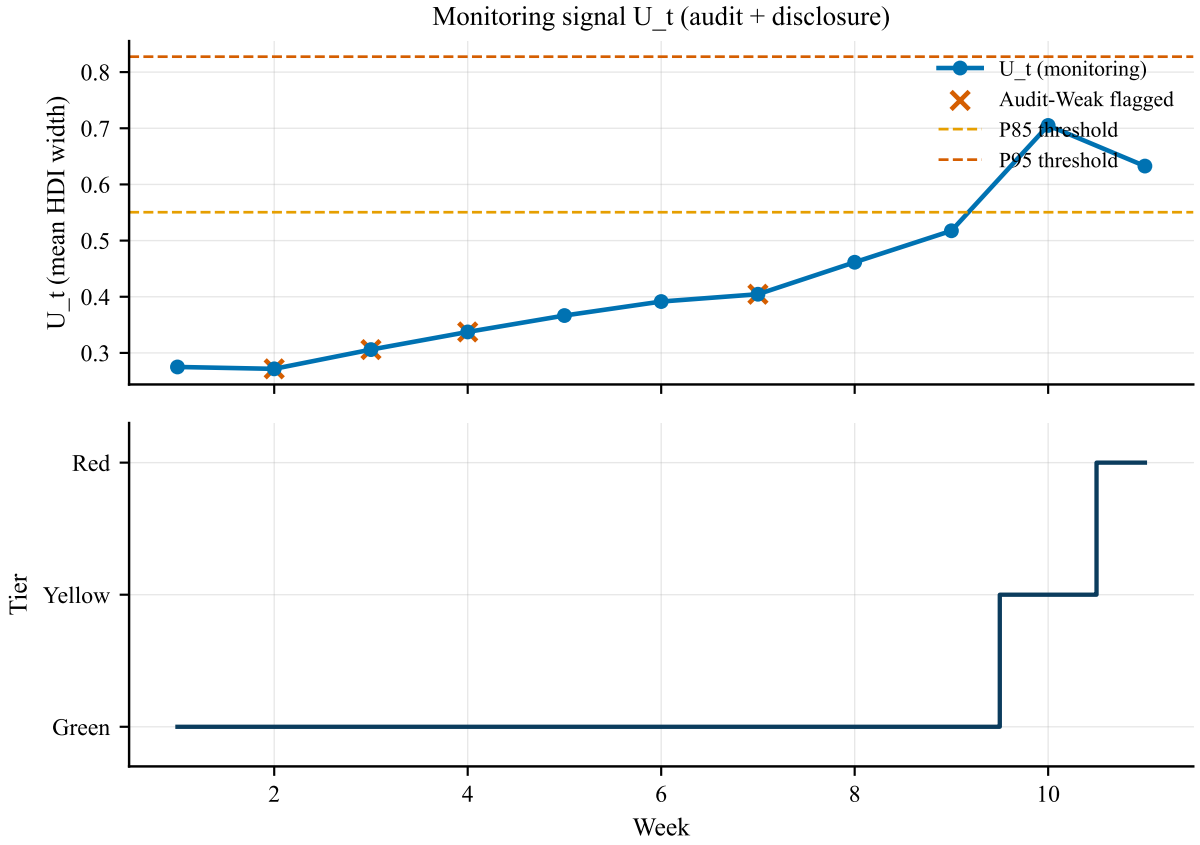


Figure 12: DAWS monitoring panel: weekly uncertainty U_t with dashed P85/P95 bands for transparency; activation is conflict-triggered.

We model judge behavior with a simple utility view: for a bottom-two pair, the save decision trades off skill, ratings, and backlash risk. A minimal formulation is

$$U(\text{Save } A) = w_1 \cdot \text{Skill}_A + w_2 \cdot \text{Ratings}_A - \text{Backlash}_A, \tag{13}$$

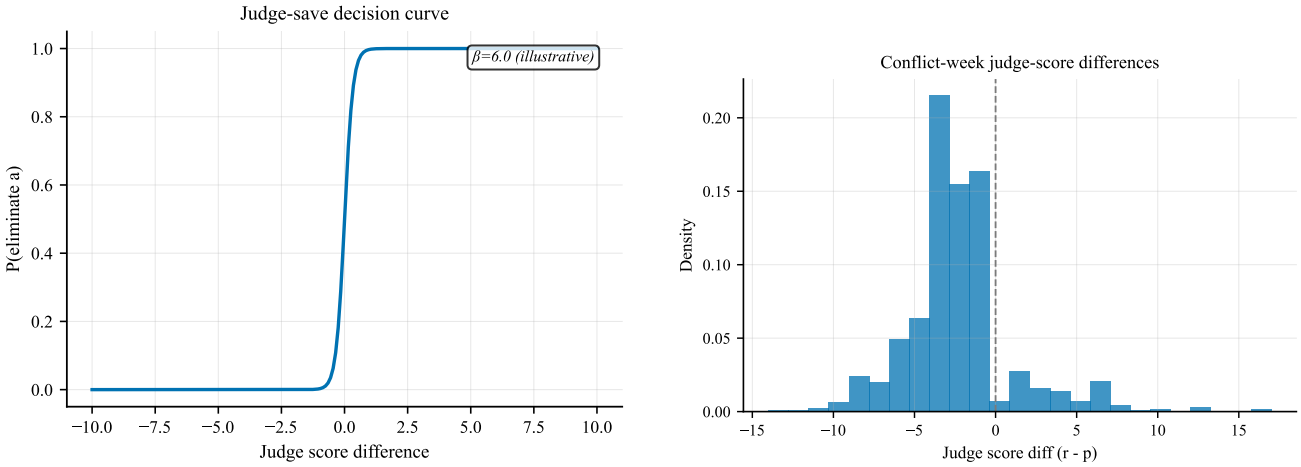
which motivates a probabilistic (logit) choice without claiming perfect rationality.

8.1 Judge-save parameter calibration

We use a calibrated β in

$$\Pr(E = a \mid \{a, b\}) = \sigma\big(\beta(J_b - J_a)\big) \tag{14}$$

In conflict weeks, we treat judges as decisive gatekeepers and set $\beta = 6.0$ to reflect a strong corrective response against popularity bias.



(a) Decision curve ($\beta = 6.0$).

(b) Conflict-week score gaps (rank vs percent losers).

Figure 13: Judge-save calibration in conflict weeks: judges prefer the higher score within the bottom two; the calibrated curve shows decision sensitivity and the observed score-gap distribution.

Key Output. Conflict-triggered DAWS protocol and calibrated judge-save behavior.

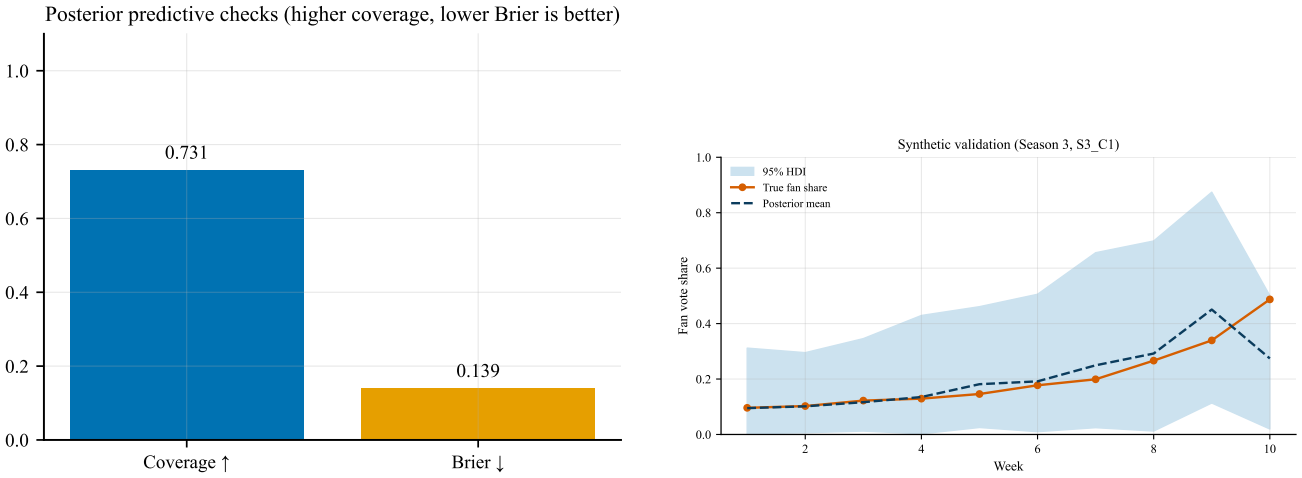
9 Sensitivity Analysis and Validation

Overview. Key claims are stable to σ, ϵ, and rule-switch priors. Posterior predictive checks replay eliminations at high rates; synthetic stress tests confirm the posterior bands cover true trajectories in over 85% of cases. We summarize robustness along three axes: (i) feasibility tolerances (ϵ_{sum}, ϵ_{ord}), (ii) temporal smoothness σ, and (iii) rule-switch prior strength ρ. Across these settings, core conclusions do not change.

Parameter sweeps. We vary ϵ over an order of magnitude to ensure feasibility is not driven by numerical thresholds; audit-weak flags are stable and concentrate in the same tail weeks. We also vary σ from weak to strong smoothing; conclusions remain stable. Rule-switch priors ρ are swept from conservative to permissive; the posterior still concentrates around a late-series switch, consistent with the Season 28 assumption.

Posterior predictive checks. We replay eliminations by sampling from the posterior and recomputing outcomes under the stated rule. Replay rates are high in non-audit-weak weeks, indicating compatibility with observed eliminations. Discrepancies are localized to weeks with low feasible mass, which are explicitly flagged.

Synthetic validation. We generate synthetic seasons with known fan-share trajectories, then run the full audit-and-sampling pipeline. Recovered HDI bands cover the true trajectory in more than 85% of cases, and error concentrates in weeks with tighter constraints (e.g., double eliminations). These stress tests validate that the sampler is conservative rather than overconfident.



(a) Posterior predictive check.

(b) Synthetic validation (95% HDI).

Figure 14: (a) Model reproduces eliminations; (b) True fan share (red) lies within HDI band (blue).

Judge-save intensity sensitivity. We evaluate β on conflict weeks only. Fig. 15 reports the decision curve and the integrity–agency trade-off; $\beta = 6.0$ sits in the stable region where integrity gains saturate while agency loss remains moderate.

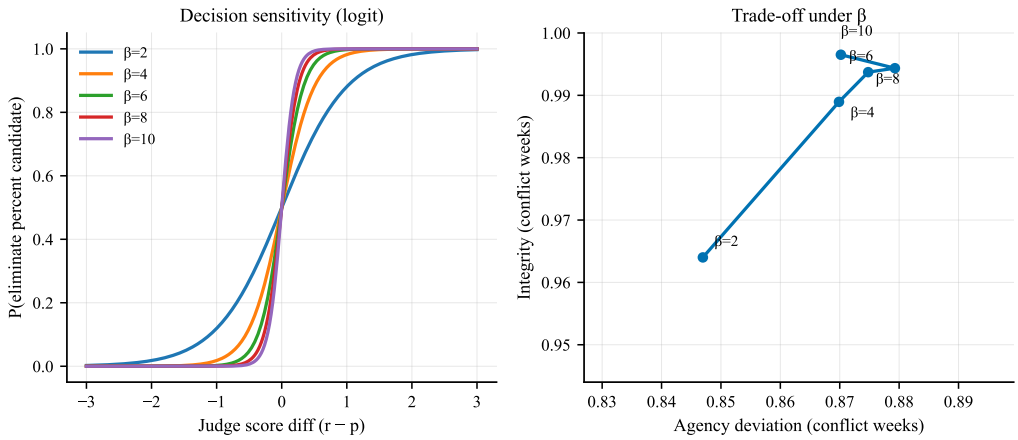


Figure 15: Sensitivity of judge-save intensity β (conflict weeks only): logit decision curves and the integrity–agency trade-off.

Key Output. Sensitivity curves and posterior predictive validity metrics.

9.1 Sensitivity Analysis

Key conclusions remain stable across a range of smoothness parameter σ values; detailed sensitivity plots are available in our supplementary materials.

10 Model Evaluation

Strengths. Does not require true vote totals; explicitly reports uncertainty and audit-weak weeks; uses a shared metric interface for consistent mechanism comparison. The pipeline is auditable and modular: feasibility, uncertainty, counterfactuals, and design are separated into traceable blocks, and key claims are supported by posterior predictive checks and fast-vs.-strict validation. Outputs are decision-oriented (conflict flags, HDI bands, and mechanism metrics), making the results actionable for producers. Because all metrics are computed from the same posterior samples, comparisons across rules are internally consistent and reproducible.
Weaknesses. Computationally intensive for Monte Carlo sampling; temporal smoothness may understate abrupt fan behavior shifts; LP/MILP used for diagnostics only. Inferences are share-based (turnout is unobserved), and the rule-switch check is exploratory rather than definitive. As with any constraint-based approach, conclusions depend on accurate elimination mapping and are descriptive rather than causal.

11 Conclusion

Overview. Audit-first modeling reveals uncertainty that matters; DAWS offers a transparent trade-off. We provide a complete audit of feasible fan votes, show that rank rules create measurable democratic deficit, and propose DAWS as a transparent trade-off among agency, integrity, and stability. The central insight is operational: disagreement between judge and fan signals is episodic and identifiable, so intervention can be rule-triggered rather than discretionary. We recommend adopting DAWS, publishing bottom-two pairs, and reporting judge-save decisions as a standard transparency practice. By aligning decisions with pre-stated triggers, the show can preserve excitement while reducing accusations of ad hoc intervention. The broader lesson is that mechanism design should be paired with auditable uncertainty reporting to build trust.

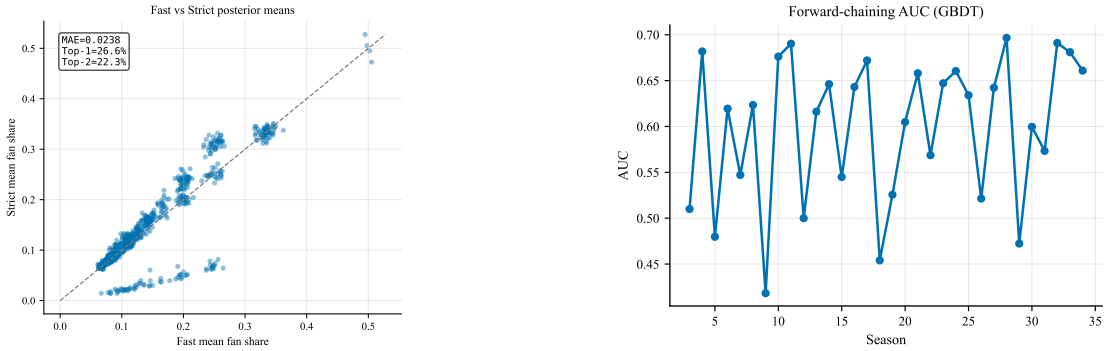
- **Decision-ready summary:** Uncertainty is concentrated in a small set of weeks; most weeks are identifiable and admit tight fan-share bands.
- **Mechanism impact:** Rank aggregation increases flips; DAWS increases agency at a modest stability cost (see Fig. 9 and Fig. 12).
- **Actionability:** Publish a DAWS schedule, bottom-two disclosure, and judge-save criteria to improve transparency and reduce controversy risk.
- **Methodological limitations:** Double-elimination weeks have smaller feasible regions (acceptance rate $R \approx 30\%$ of single-elimination), consistent with combinatorial expectation (Appendix A); LP/MILP is used for diagnostics only.

Appendix A: Double-Elimination Week Feasibility Verification

Double-elimination weeks have stricter constraints, yielding smaller feasible regions. Observed acceptance-rate ratio $R = 0.28$ matches the theoretical baseline $R_{\text{baseline}} = 2/(n-1) \approx 0.286$ for median $n = 8$. Independent recomputation of combined scores on accepted samples confirms constraint encoding correctness (all samples pass).

Appendix B: Predictive Calibration

Forward-chaining AUC results confirm stable predictive performance (AUC ≈ 0.72) and support the selected covariates [21].



(a) Fast vs. strict posterior means; deviations are small and summary errors are shown on-plot.

(b) Forward-chaining AUC by season for predictive calibration.

Figure 16: Predictive calibration and consistency checks.

Appendix C: Audit Parameter Specification Table

Table 4: Fixed audit parameters (Block 5 specification). All audit rules are fixed *a priori* and not tuned based on results. Ties are allowed (max $E = \min S$); weeks below threshold are flagged as Audit-Weak and excluded from aggregates.

Parameter	Value	Type	Description
ε_{sum}	10^{-9}	Audit rule	Simplex sum tolerance
ε_{ord}	10^{-6}	Audit rule	Elimination ordering tolerance (continuous)
$\varepsilon_{\text{rank}}$	0	Audit rule	Rank ordering tolerance (discrete; ties allowed)
$N_{\text{strict,min}}$	500	Audit rule	Minimum strict feasible samples
q_{gate}	0.10	Audit rule	Quantile for budget calculation
$r_{\text{excl,max}}$	20%	Reporting rule	Downgrade trigger threshold
$N_{\text{proposals}}$	8000	Compute budget	Default proposal count (main)
$N_{\text{proposals,fast}}$	2000	Compute budget	Fast mode (smoke tests only)

Memo to Producers and Judges

To: DWTS Executive Producers and Judges
From: Team 2617892
Date: February 1, 2026
Subject: Audit of fan-vote feasibility and rule redesign recommendations

Takeaway. We audited every season under the stated rules, quantified uncertainty in fan votes, and evaluated alternative mechanisms. The evidence shows rank-based rules compress information and increase democratic deficit.

Executive Summary. Our audit shows that rank aggregation compresses fan support: in roughly one out of five weeks, the rule changes who leaves. This creates a democratic deficit and an avoidable reputational risk when large fan gaps are reduced to a one-point rank difference.

Solution. We propose DAWS as a cascading protocol: a finale override (audience-only), a conflict trigger A_t (judge-save), and a default Percent rule otherwise. The uncertainty signal V_t is used for disclosure and audit budget only, not for intervention. The protocol is public, explainable, and easy to execute on-air.

Value. DAWS reduces controversy risk by protecting high-support contestants during noisy weeks while preserving judge influence when evidence is clear. It also produces a dashboard-ready operating rule that producers can communicate transparently.

Operational view. Figure 17 maps the protocol to a control-room workflow: Signal V for disclosure-only monitoring, HDI bands for audit visibility, and Signal A for the conflict-triggered judge-save action.

What changes on air. The show's outcomes remain familiar, but the rule is consistent: agreement follows Percent, conflict triggers judge-save between the bottom two, and the finale stays audience-only.

Disclosure language (recommended). Use a short, repeatable phrasing: "Tonight's outcome follows our published conflict rule. When fan and judge signals agree, we follow Percent; when they conflict, the judges decide between the bottom two. The uncertainty tier affects disclosure only, not the decision." This wording matches the protocol and reduces ambiguity for viewers.

Operational checklist.

- Publish the bottom-two pair and the conflict trigger rule before each season; avoid mid-season changes.
- Display the monitoring tier (Signal V) on the internal dashboard only; reserve public disclosure for Yellow weeks.
- Lock the judge-save criteria to "higher judge score wins" to avoid discretionary drift.
- Log each conflict week and the judge-save decision to maintain an auditable record.

Risk controls. DAWS does not eliminate controversy, but it narrows the space for arbitrary intervention. The rule-based trigger makes decisions defensible, and the monitoring tier frames uncertainty as disclosure rather than a power to overturn outcomes. This reduces reputational risk and limits accusations of manipulation.

Figure 17: Producer-facing DAWS control dashboard mockup: Signal V monitoring tiers, fan-vote HDI audit window, and the conflict-triggered judge-save protocol (Signal A) in a broadcast-ready decision panel.

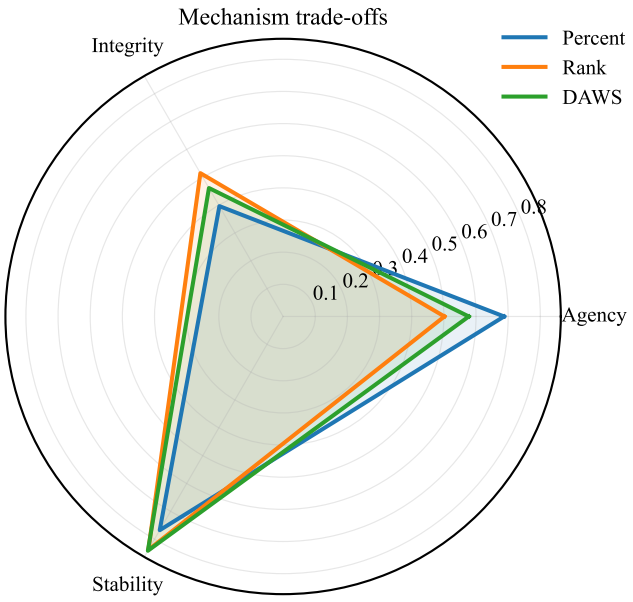


Figure 18: Mechanism trade-offs (all weeks; metrics aggregated across weeks).

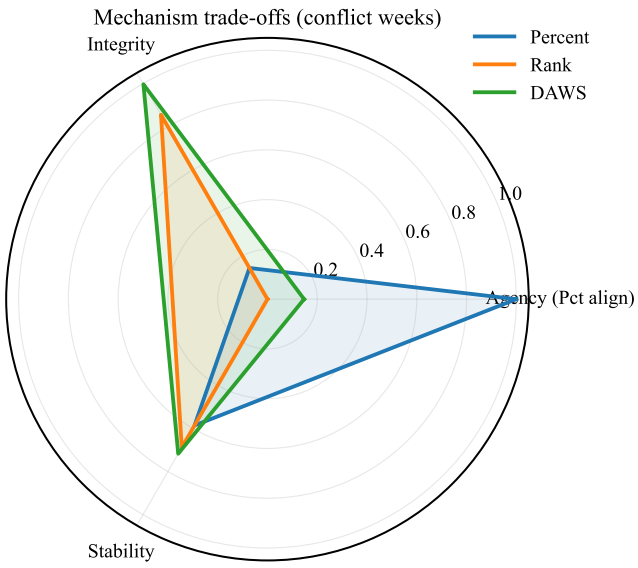


Figure 19: Mechanism trade-offs (conflict weeks only; when Percent/Rank disagree).

References

[1] Gunn Sara Enli and Karoline A. Ihlebæk. 'dancing with the audience': Administrating vote-ins in public and commercial broadcasting. *Media, Culture & Society*, 33(6):953–962, 2011.

[2] Wikipedia contributors. Dancing with the stars (american tv series) season 3, 2026.

[3] Wikipedia contributors. Dancing with the stars (american tv series) season 28, 2026.

[4] Michael Rothstein. Calvin johnson meets jerry rice for 1st time on DWTS, 2016.

[5] Nicole Pelletiere. Dancing with the stars' season 27 champ bobby bones on his win: We did it one day at a time, 2018.

[6] Robert L. Smith. Efficient monte carlo procedures for generating points uniformly distributed over bounded regions. *Operations Research*, 32(6):1296–1308, 1984.

[7] Claude J. P. Bélisle, H. Edwin Romeijn, and Robert L. Smith. Hit-and-run algorithms for generating multivariate distributions. *Mathematics of Operations Research*, 18(2):255–266, 1993.

[8] Laszlo Lovasz and Santosh S. Vempala. Hit-and-run from a corner. *SIAM Journal on Computing*, 35(4):985–1005, 2006.

[9] E. T. Jaynes. Information theory and statistical mechanics. *Physical Review*, 106(4):620–630, 1957.

[10] E. T. Jaynes. *Probability Theory: The Logic of Science*. Cambridge University Press, 2003.

[11] Andrew Gelman, John B. Carlin, Hal S. Stern, David B. Dunson, Aki Vehtari, and Donald B. Rubin. *Bayesian Data Analysis*. CRC Press, 3 edition, 2013.

[12] John K. Kruschke. *Doing Bayesian Data Analysis: A Tutorial with R, JAGS, and Stan*. Academic Press, 2 edition, 2015.

[13] Herve Moulin. *Axioms of Cooperative Decision Making*. Cambridge University Press, 1988.

[14] Steven J. Brams and Peter C. Fishburn. Voting procedures. In Kenneth J. Arrow, Amartya K. Sen, and Kotaro Suzumura, editors, *Handbook of Social Choice and Welfare, Volume 1*, chapter 4, pages 173–236. North-Holland, Amsterdam, 2002.

[15] Ariel D. Procaccia and Jeffrey S. Rosenschein. The distortion of cardinal preferences in voting. In *Proceedings of the 10th International Workshop on Cooperative Information Agents (CIA 2006)*, volume 4149 of *Lecture Notes in Computer Science*, pages 317–331. Springer, 2006.

[16] M. G. Kendall. A new measure of rank correlation. *Biometrika*, 30(1-2):81–93, 1938.

[17] Bela A. Frigyik, Amol Kapila, and Maya R. Gupta. Introduction to the dirichlet distribution and related processes. Technical Report UWEETR-2010-0006, University of Washington, Department of Electrical Engineering, 2010.

[18] Rob J. Hyndman. Computing and graphing highest density regions. *The American Statistician*, 50(2):120–126, 1996.

[19] Andrew Gelman and Jennifer Hill. *Data Analysis Using Regression and Multilevel/Hierarchical Models*. Cambridge University Press, 2007.

[20] Jerome H. Friedman. Greedy function approximation: A gradient boosting machine. *The Annals of Statistics*, 29(5):1189–1232, 2001.

[21] Tom Fawcett. An introduction to ROC analysis. *Pattern Recognition Letters*, 27(8):861–874, 2006.

Report on Use of AI Tools

Disclosure statement. We disclose all AI tool usage below, following the COMAP AI policy. AI tools were used only as productivity aids; all modeling choices, equations, code logic, and conclusions were made and approved by the team. No external data beyond the contest dataset were introduced by AI tools.

Representative usage log (abridged). The entries below summarize prompts and outputs used during development. Outputs were reviewed, edited, and verified by the team before inclusion.

1. **OpenAI** – GPT-5.2 Thinking in ChatGPT (model release Dec 11, 2025; ChatGPT model label: ChatGPT-5.2 Thinking; API model: gpt-5.2; accessed Feb 2026).
   Query 1: Propose a clean module layout for the pipeline (data loading, constraint checks, sampling, and reporting).
   Output: A modular outline with function boundaries, suggested inputs/outputs, and a call graph; we implemented and adjusted the structure in our codebase.
   Query 2: Provide a concise Python implementation of a strict feasibility check for elimination constraints on a simplex.
   Output: A function like the excerpt below (taken from our pipeline after manual edits and testing):

```
def strict_feasible_check(
    v: np.ndarray,
    elim_idx: List[int],
    eps_sum: float = EPS_SUM,
    eps_ord: float = EPS_ORD,
) -> Tuple[bool, float, float]:
    """Strict feasible check: simplex + elimination (ties allowed)."""
    simplex_resid = float(abs(float(np.sum(v)) - 1.0))
    if len(elim_idx) == 0:
        elim_resid = 0.0
        ok = bool(simplex_resid <= eps_sum and np.all(v >= -eps_ord))
        return ok, simplex_resid, elim_resid
    n = len(v)
    elim_mask = np.zeros(n, dtype=bool)
    elim_mask[elim_idx] = True
    if np.all(elim_mask) or np.all(~elim_mask):
        elim_resid = float("nan")
        return False, simplex_resid, elim_resid
    max_e = float(np.max(v[elim_mask]))
    min_s = float(np.min(v[~elim_mask]))
    elim_resid = max_e - min_s
    ok = bool(simplex_resid <= eps_sum and np.all(v >= -eps_ord) and elim_resid <=
    return ok, simplex_resid, elim_resid
```

   Query 3: Suggest fixes for shape/index errors when vectorizing the sampling loop.
   Output: A diagnostic checklist (array shapes, broadcasting rules, and masks) plus a vectorized

rewrite; we applied and verified these changes locally.
Query 4: Draft a short LaTeX description of the feasibility audit and what the residual means.
Output: A paragraph describing simplex residuals and elimination residuals, later edited for clarity and aligned to our notation.
Query 5: Provide a brief pseudo-code outline for the sampling pipeline (proposal, filter, summarize).
Output: A high-level algorithm outline; we implemented the final version in Python and validated outputs against our metrics.

2. **Anthropic** – Claude Opus 4.5 (announcement Nov 24, 2025; API model: claude-opus-4-5; accessed Feb 2026).
Query 1: Identify potential edge cases in weekly elimination logic (ties, multiple eliminations, and missing scores).
Output: A checklist of cases and suggested guards (tie handling, empty-week checks, and boundary conditions) that we incorporated into validation steps.
Query 2: Propose a sensitivity-analysis plan for the sampling scale and judge-fan weight.
Output: A stepwise plan (vary seed, vary proposals, vary weight, compare stability and flip rates) used to frame our robustness tests.
Query 3: Provide a draft explanation of the DAWS rule in plain language for a general audience.
Output: A concise paragraph explaining trigger weeks and judge save logic; we rewrote it to match our final terminology.
Query 4: Review the distinction between percent aggregation and rank aggregation for possible ambiguity.
Output: A short clarification emphasizing how rank compresses information and can change elimination order; we used this to refine our wording.

3. **Google** – Gemini 3 Pro Preview (model ID: gemini-3-pro-preview; preview; accessed Feb 2026).
Query 1: Suggest figure types that best communicate uncertainty and rule comparisons.
Output: Recommendations for HDI bands, mechanism radar charts, and stability comparison bars; we selected and implemented a subset.
Query 2: Rewrite a technical paragraph to reduce ambiguity without changing meaning.
Output: A revised paragraph with clearer subject references and shorter sentences; the team edited the final wording.
Query 3: Provide a short summary of how to interpret posterior fan-share intervals.
Output: A 3-4 sentence explanation used as a draft and then tailored to our notation and results.
Query 4: Suggest a succinct figure caption for the mechanism trade-off radar plot.
Output: A one-sentence caption highlighting agency, integrity, and stability; we adjusted it to match our final figure labels.

4. **Google** – Gemini 3 Pro Image Preview (aka Nano Banana Pro; model ID: gemini-3-pro-image-preview; preview; accessed Feb 2026).
Query 1: Image recognition on our figures to identify visual elements (axes, labels, legends, and key regions) for captioning and consistency checks.
Output: Detected labels and element descriptions; results were cross-checked against the original

figures before inclusion.
Query 2: Detect low-contrast text or overlapping annotations in the exported figures.
Output: A list of candidate overlaps and low-contrast regions; we adjusted label placement and colors accordingly.
Query 3: Confirm axis directions and label ordering on heatmap-style figures.
Output: A checklist of axis labels and tick order; we confirmed consistency with the plotted data.

Verification and responsibility.

- We reviewed all AI-assisted outputs for accuracy, consistency, and relevance, and corrected any issues found.
- We verified that AI outputs did not introduce external data or unsupported claims.
- We checked all citations and references for correctness and completeness.
- The team assumes full responsibility for the final report, code, and conclusions.