Problem Chosen	2026 MCM/ICM Summary Sheet	Team Control Number
C		2617610

Who manipulates the competition results? — A new method based on Weak Supervision Vote Estimation Model

Summary

Dancing with the Stars (DWTS) integrates judges' scores and audience votes to eliminate contestants under three distinct voting systems: percent, rank, and bottom-two with judges' save. While these mechanisms aim to balance performance quality and popularity, their fairness, stability, and uncertainty remain unclear. This study develops a unified modeling framework to reconstruct hidden fan votes, evaluate consistency and certainty, compare voting systems, and analyze key determinants of competition outcomes.

For Task 1, we propose a vote estimation model based on **baseline popularity**, **season-week offset**, **performance sensitivity**, and their **interactions** to infer weekly fan vote shares. We formulate **rule-based loss functions**, which can reduce the loss of percent, rank, and bottom-two systems in one loss function. For consistency, we calculate model's accuracy in different elimination system. The overall accuracy exceeds 90% and have violation rates ≈ 0, indicating accurate estimation with small distance of violation. For estimation's uncertainty over time, **we use margin which measures the safe distance for contestants who have not been eliminated. We find that season 3-27 (Percent) have very high certainty, and season 28-34 (Buttom-two) have higher uncertainty**. For overall uncertainty, we apply bootstrap to construct 95% confidence intervals in different system. **The CI lower bounds are greater than 0.85 for percent and bottom-two system. All the CI lower bounds are greater than 0.6.**

For Task 2, to prove the results produced by the percent and rank have difference, we create a new measurement called **agreement rate** which is the proportion of same eliminations for these two systems. **We find that percent and rank systems cause differences in over 22% of eliminations.** To analyze which method favors fan votes more than the other, we use compare the vote share distribution of their conflict eliminations, and also use Spearman correlation. **We find that the percent system favors audience votes more strongly, whereas the rank system places greater emphasis on judges' evaluations.** Case studies of controversial seasons further reveal that **judges' save mechanisms can significantly alter elimination risks near critical boundaries.**

For Task 3, we analyze the impact of dancers' and celebrities' characteristics using ensemble tree-based models. **Permutation importance, impurity-based importance, and information gain importance indicate that age and ballroom partner dominate judges' scoring, while partner and hometown exert stronger influence on audience voting.**

For Task 4, We propose a hybrid voting mechanism that employs **dynamic weight allocation across early, medium, and late stages.** Because we want to **preserve audience engagement, reduce extreme judge-fan conflicts, and highlights the fairness in last few weeks of seasons.** Overall, our framework provides a rigorous, interpretable, and transferable methodology for evaluating voting systems in competitive entertainment programs and supports more equitable decision-making in future seasons.

Keywords: Rule-based loss function; Margin; Agreement rate; Ensemble models; Feature importance

Contents

1 Introduction

Dancing with the Stars (DWTS) is a competitive reality show in which celebrity–pro partners perform weekly dances and face elimination. Each week, couples receive judges' scores (meant to reflect performance quality) and fan votes (reflecting popularity). The show combines these two signals to produce a weekly standing, and the lowest-ranked couple(s) are eliminated. Over time, DWTS has adjusted its aggregation rule in response to audience and fairness concerns, motivating a careful comparison of how different rules translate the same judges/fan information into elimination outcomes. For brevity, we refer to the three mechanisms below as *percent*, *rank*, and *bottom2*.

- **Percent:** The weekly standing is the sum of the judges' percentage share and the fan vote share. The eliminated contestant(s) are those with the lowest combined standing (bottom-1 or bottom-k).
- **Rank:** Contestants are ranked separately by judges' total score and by fan vote share, and the combined standing is the rank sum. The eliminated contestant is the one with the worst (largest) rank sum.
- **Bottom2 (Judges' Save):** A bottom-two group is first formed based on the combined standing (rank-sum), and the eliminated contestant must come from this group. Judges then choose which of the bottom two to eliminate.

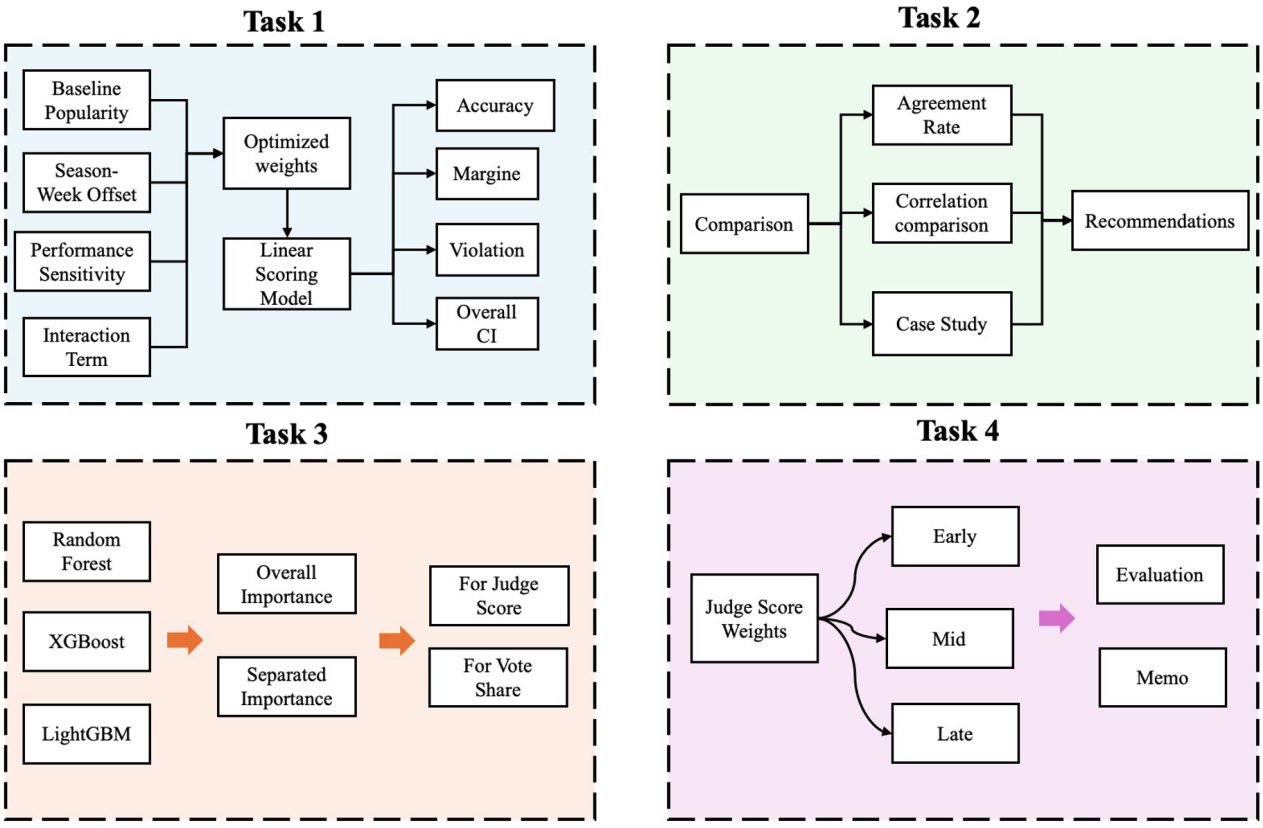


Figure 1: Our Work

2 Assumptions and Notation

2.1 Modeling Assumptions

To simplify the modeling process and ensure analytical feasibility, we make the following reasonable assumptions:

- The relative preferences of fans remain stable within each competition week, and short-term fluctuations do not significantly alter overall voting behavior.
- External factors such as media events and off-stage incidents have a limited short-term impact and are not explicitly modeled.
- There is only one parameter controlling how much a better performance turns into more fan votes, and a contestant's baseline popularity just amplifies or dampens that same effect.

2.2 Notation

Table 1: Notations

Symbol	Description
$\mathcal{I}_{s,t}$	Set of contestants still in the competition in season s, week t
$\mathcal{E}_{s,t}$	Set of contestants eliminated in season s, week t
$\mathcal{S}_{s,t}$	Set of contestants not eliminated in season s, week t
$k_{s,t}$	Number of contestants eliminated in season s, week t
$J_{i,s,t}$	Judges' total score for contestant i in season s, week t
$u_{i,s,t}$	Overall vote score for contestant i in season s, week t
$v_{i,s,t}$	Fan vote share for contestant i in season s, week t
$f_{i,s,t}$	Standardized weekly performance level for contestant i in season s, week t
a_i	Baseline popularity level for contestant i
$b_{s,t}$	Week effect for season s at week t
β	Strength of how performance relates to voting
$\mathcal{L}_{\text{percent}}(s,t)$	Model penalty for percent-rule weeks
$\mathcal{L}_{\text{rank}}(s,t)$	Model penalty for rank-rule weeks
$\mathcal{L}_{\text{bottom2}}(s,t)$	Model penalty for bottom2-rule weeks
$m_p,\ m_r,\ m_b$	Safety gaps used in the three penalty terms
$w_p,\ w_r,\ w_b,\ w_{\text{pl}}$	Weights that set the relative importance of each penalty term
I_j	Importance score of feature j from the permutation test

3 Data Preprocessing

- The dataset contains weekly records of judges' scores, voting outcomes, and elimination results across multiple seasons, with each record corresponding to one contestant

in a given season and week.

- Missing values, abnormal entries, and logical inconsistencies were examined, and incomplete records with insufficient key variables were removed or appropriately imputed.
- Judges' scores and performance-related variables were standardized using z-score normalization, while voting proportions were rescaled to satisfy probability constraints.
- Based on official competition regulations, each season was classified into percent, rank, and bottom-two voting systems, providing the foundation for rule-specific loss design.

4 Task 1: Estimating Weekly Fan Votes

4.1 Vote Estimation Model Construction

Each week, contestants receive judges' scores and fans cast votes; the show combines these to eliminate contestant(s). Fan vote totals are unobserved, so we infer them from weekly eliminations and judges' scores. The key difficulty is *non-identifiability*: multiple vote configurations can lead to the same elimination outcome. Thus, we build a rule-aware model that reproduces eliminations and supports uncertainty assessment.

The total weekly turnout is unknown and varies. Eliminations depend on relative standing; For each season-week (s, t), let $\mathcal{I}_{s,t}$ denote the *active set* of contestants still in competition, we estimate the score first and then convert it to a vote share as a proportion $v_{i,s,t}$ between [0, 1] and $\sum_{i\in\mathcal{I}_{s,t}} v_{i,s,t} = 1$. Finally, we use this proportion to multiply a base value to reflect the real vote numbers.

Step 1. Linear function of the vote score and rate

Based on the existing features and our understanding of the situation, we construct a linear model to estimate the score of votes we define as follows, and also added interaction terms as follows:

$$u_{i,s,t} = a_i + b_{s,t} + \beta f_{i,s,t} + a_i \beta f_{i,s,t}, \tag{1}$$

which includes some parameters. The parameters are initialized using a specific distribution or values, then forward-computed, and subsequently updated using backpropagation and gradient descent to obtain the final parameter estimates, providing parameters for different players in different weeks. The parameters include

- a_i is a basic popularity parameter. It captures baseline popularity for each dancer. We initialize it with $N(0, 0.05^2)$ using PyTorch Embedding function.
- $b_{s,t}$ is a season-Week offset parameter which measures week shocks.
- $f_{i,s,t} = \frac{J_{i,s,t} - \bar{J}_{s,t}}{\mathrm{sd}(J_{\cdot,s,t})}$ is a standardized weekly performance quantity to make scores comparable across weeks. J is the judgment score of several judges.
- β captures performance sensitivity, and the $a_i \beta f_{i,s,t}$ allows us to measure the interaction effects between the popularity and performance.

After we estimate the score $u_{i,s,t}$, we use softmax function to convert it to a proportion between 0 and 1 as follows

$$v_{i,s,t} = \frac{\exp(u_{i,s,t})}{\sum_{k \in \mathcal{I}_{s,t}} \exp(u_{k,s,t})}. \tag{2}$$

Step 2. Loss functions Design

In DWTS, there are three different systems (Percentage, Bottom-two with judges).

- **Percent system (Seasons 3–27):** $C_{i,s,t} = J^{\%}_{i,s,t} + v_{i,s,t}$ where $J^{\%}_{i,s,t} = \frac{J_{i,s,t}}{\sum_{k \in \mathcal{I}_{s,t}} J_{k,s,t}}$.
- **Rank system (Seasons 1–2):** combine judge-rank and vote-rank.
- **Bottom-two system (Seasons 28–34):** bottom two determined by combined standing, then judges decide which of the two is eliminated.

To fit our model, we design our loss function carefully to meet three above scenarios and minimize the error. The Rule-aware modeling prevents bias from imposing an incorrect rule on a subset of seasons. Based on our vote scoring linear model, we therefore train by weak supervision: we penalize configurations where the model-implied elimination under the season's rule disagrees with the observed elimination(s).

Let $\mathcal{E}_{s,t} \subseteq \mathcal{I}_{s,t}$ be the observed eliminated set (after removing withdrawals or disqualifications, which we treat as exogenous censoring), and define the safe set as $\mathcal{S}_{s,t} = \mathcal{I}_{s,t} \setminus \mathcal{E}_{s,t}$. Let $k_{s,t} = |\mathcal{E}_{s,t}|$ be the number eliminated in week (s,t); when $k_{s,t} = 0$ (no elimination) the system provides no direct ordering signal and we set the elimination loss to 0 (learning then relies on temporal smoothness/priors described later).

(L1) Percent-system pairwise hinge loss. The observed elimination implies that eliminated contestants should lie below the safe contestants under the combined score:

$$C_{e,s,t} \le C_{j,s,t} \quad \forall e \in \mathcal{E}_{s,t},\ \forall j \in \mathcal{S}_{s,t}. \tag{3}$$

We impose a positive margin $m_p > 0$ and soften the constraint using a pairwise hinge loss:

$$\mathcal{L}_{\text{percent}}(s,t) = \frac{1}{|\mathcal{E}_{s,t}||\mathcal{S}_{s,t}|} \sum_{e \in \mathcal{E}_{s,t}} \sum_{j \in \mathcal{S}_{s,t}} \max\{0,\ C_{e,s,t} - C_{j,s,t} + m_p\}. \tag{4}$$

This formulation *unifies* single- and multi-elimination weeks: when $k_{s,t} = 1$ it reduces to a bottom-1 separation constraint; when $k_{s,t} > 1$ it enforces that all eliminated contestants lie below all safe contestants (i.e., a bottom-k set constraint). The pairwise hinge directly encodes the show's decision boundary and yields stable training by enforcing a separation margin between eliminated and safe sets. Moreover, it provides natural *certainty diagnostics*: define the *week margin*

$$\text{margin}_{s,t} = \min_{e \in \mathcal{E}_{s,t},\, j \in \mathcal{S}_{s,t}} (C_{j,s,t} - C_{e,s,t}), \tag{5}$$

where larger values indicate a clear boundary and thus higher certainty, while negative values indicate boundary violations (the model would rank some eliminated contestant above some safe contestant). Similarly, the hinge terms quantify *constraint violation* magnitudes, allowing us to summarize uncertainty across weeks using statistics such as mean violation and upper quantiles.

(L2) Rank-system hinge loss (Seasons 1–2). The other loss functions have some similarity with L1, so we will mention them briefly. For L2, to avoid non-differentiable hard ranks during training, we use a differentiable soft-rank surrogate. For a vector $\mathbf{x} \in \mathbb{R}^n$ (larger is better), define

$$\mathrm{srank}_\tau(x_i) = 1 + \sum_{j \neq i} \sigma\left(\frac{x_j - x_i}{\tau}\right), \quad \sigma(z) = \frac{1}{1+e^{-z}} \tag{6}$$

where we use the same temperature τ for all sigmoid based surrogates (soft-rank and bottom-two membership) for simplicity. $\tau > 0$ controls smoothness (smaller τ approaches hard ranks). Define

$$r^J_{i,s,t} = \mathrm{srank}_\tau(J_{i,s,t}), \qquad r^V_{i,s,t} = \mathrm{srank}_\tau(u_{i,s,t}), \qquad R_{i,s,t} = r^J_{i,s,t} + r^V_{i,s,t}. \tag{7}$$

Larger $R_{i,s,t}$ indicates worse standing. Enforce eliminated worse than safe via

$$\mathcal{L}_{\mathrm{rank}}(s,t) = \frac{1}{|\mathcal{E}_{s,t}||\mathcal{S}_{s,t}|} \sum_{e \in \mathcal{E}_{s,t}} \sum_{j \in \mathcal{S}_{s,t}} \max\{0,\ R_{j,s,t} - R_{e,s,t} + m_r\}. \tag{8}$$

It matches the rank-based rule while keeping optimization smooth and stable.

(L3) Bottom-two membership loss (Seasons 28–34). In judges-save seasons, exact elimination identity is not fully determined by votes; the faithful constraint is that the eliminated contestant lies in the bottom two. Using the same combined standing proxy $R_{i,s,t}$, let $n = |\mathcal{I}_{s,t}|$ and define a soft count of how many contestants are better than eliminated e:

$$\mathrm{BetterCount}(e) = \sum_{j \in \mathcal{I}_{s,t} \setminus \{e\}} \sigma\left(\frac{R_{e,s,t} - R_{j,s,t}}{\tau}\right). \tag{9}$$

Intuitively, $\mathrm{BetterCount}(e)$ approximates the number of contestants ranked above e; requiring $\mathrm{BetterCount}(e) \geq n-2$ enforces that e lies among the two worst contestants under R. Being in the worst two means at least $n-2$ contestants are better: $\mathrm{BetterCount}(e) \geq n-2$. We impose a margin $m_b > 0$:

$$\mathcal{L}_{\mathrm{bottom2}}(s,t) = \frac{1}{|\mathcal{E}_{s,t}|} \sum_{e \in \mathcal{E}_{s,t}} \max\{0,\ (n-2) - \mathrm{BetterCount}(e) + m_b\}. \tag{10}$$

It avoids over-penalizing the model for judges' discretion while still enforcing the correct pre-save structure.

(L4) Placement regularizer (season-level stabilization). Weekly eliminations provide limited constraints early in the season; to prevent degenerate popularity orderings, we add a gentle season-level ranking regularizer on a_i. Let (i,j) be adjacent placement pairs with contestant i finishing better than j; then

$$\mathcal{L}_{\mathrm{place}} = \frac{1}{M} \sum_{(i,j)} \log\big(1 + \exp\big(-(a_i - a_j)\big)\big). \tag{11}$$

It improves stability and plausibility of inferred popularity without forcing a hard match.

Step 3. Final Loss Function & Hyperparameters

Our last step is to combine all the loss functions we defined above, we add several weights to each loss function term as hyperparameters. We tune hyperparameters via a coarse grid search and select a setting that yields consistently high elimination-consistency. We verified that conclusions are qualitatively robust across nearby settings. Mathematically, let $\mathcal{W}_p, \mathcal{W}_r, \mathcal{W}_b$ be the sets of (season,week) indices belonging to percent, rank, and bottom-two systems. We minimize the weighted objective

$$\mathcal{L} = w_p \sum_{(s,t)\in\mathcal{W}_p} \mathcal{L}_{\text{percent}}(s,t) + w_r \sum_{(s,t)\in\mathcal{W}_r} \mathcal{L}_{\text{rank}}(s,t) + w_b \sum_{(s,t)\in\mathcal{W}_b} \mathcal{L}_{\text{bottom2}}(s,t) + w_{\text{pl}}\mathcal{L}_{\text{place}}. \tag{12}$$

The final Hyperparameters are $w_p = 1.0$, $w_r = 2.0$, $w_b = 5.0$, $w_{\text{pl}} = 0.8$, $m_p = 0.01$, $m_r = 0.5$, $m_b = 0.5$, $\tau = 1.0$.

4.2 Consistency with Eliminations

We use several metrics to estimate the consistency of our model (Table 9). Some of them are straightforward in this setting and others are designed by ourselves.

- **single_elim_acc:** We compute accuracy on single-elimination weeks as the fraction of weeks where our predicted eliminated contestant matches the observed eliminated contestant. In Seasons 28–34 (bottom-two / judges-save), we use the relaxed criterion: if the observed eliminated contestant is in our predicted bottom two, we count the week as correct.

- **bottomk_elim_rate:** We compute set-level accuracy on multi-elimination weeks. If a week has k eliminations, we form our predicted bottom-k set (the k worst-ranked contestants). We count the week as correct only if all observed eliminated contestants are contained in our predicted bottom-k set. This is a strict criterion.

- **bottomk_elim_rate:** We compute an element-level hit rate for multi-elimination weeks, which is less strict than `bottomk_hit_rate`. Across all multi-elimination weeks, we measure the fraction of eliminated contestants that fall into our predicted bottom-k set (with k equal to the observed number of eliminations in that week).

- **judges_save_strict_acc:** We compute strict accuracy for the judge choice within the bottom two. After identifying the bottom two, we apply our deterministic tie-break rule (e.g., eliminating the one with the lower judge score) and compare it with the observed elimination.

- **violation_mean:** This metrics is from the part of our loss function. We quantify how much our predicted ordering violates the elimination constraint. Let $\mathcal{E}_{s,t}$ be the observed eliminated set and $\mathcal{S}_{s,t}$ be the safe set for week (s,t). We use percent-system as an example, let $S_{i,s,t}$ denote the system-specific standing score used for comparison

(for percent-system we use $S = C$ with higher better; for rank/bottom2 systems we use $S = -R$ so that higher is better). With margin $m > 0$, we define pairwise violations

$$\text{viol}_{e,j}(s,t) = \max\{0,\ S_{e,s,t} - S_{j,s,t} + m\}, \quad e \in \mathcal{E}_{s,t},\ j \in \mathcal{S}_{s,t}. \tag{13}$$

We then report `violation_mean` as the average of $\text{viol}_{e,j}(s,t)$ over all valid (e,j) pairs and over all elimination weeks in the group. Smaller values indicate better consistency; 0 means the separation margin is satisfied everywhere.

- **margin_mean:** We measure how clearly eliminated contestants are separated from safe contestants. We use percent-system as an example, using the same $S_{i,s,t}$ (higher is better), we define the week margin as

$$\text{margin}(s,t) = \min_{e \in \mathcal{E}_{s,t},\, j \in \mathcal{S}_{s,t}} \left(S_{j,s,t} - S_{e,s,t}\right). \tag{14}$$

We report `margin_mean` as the average of $\text{margin}(s,t)$ across the weeks in the group. Larger values indicate a clearer boundary (higher certainty), while negative values indicate that at least one eliminated contestant is ranked above at least one safe contestant under our score.

From Table 9, it's clear to see that our model have above 90% accuracy for strait metrics such as single_elim_acc, bottomk_elim_rate, bottom2_elim_rate. For violation_mean, our model prediction's violation mean is very close to 0. For margin_mean, it is also slightly greater than 0, indicating a very good consistency and certainty. We will focus more on certainty in next part.

Table 2: Consistency and certainty metrics by system

Metric	percent (S3–27)	rank (S1–2)	bottom2 (S28–34)	overall
single_elim_acc	0.9080	0.8000	0.9362	0.9091
bottomk_elim_rate	0.9167	–	1.0000	0.9394
bottom2_elim_rate	–	–	0.9362	0.9362
judges_save_strict_acc	–	–	0.5532	0.5532
violation_mean	5.6405e-04	1.5594e-02	2.6634e-02	6.6633e-03
margin_mean	9.9458e-03	5.0909e-01	7.0268e-01	1.0232e-02

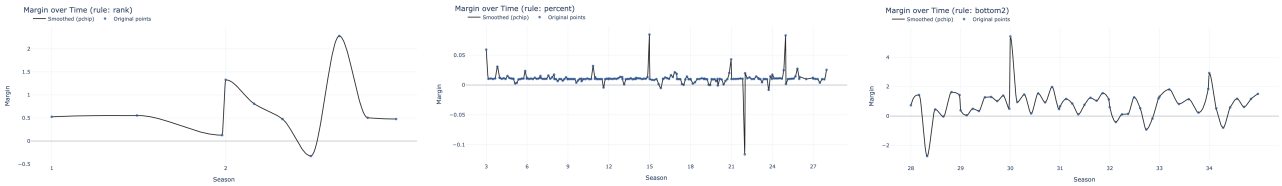
4.3 Certainty of Vote Estimates

To measure the model's certainty, we evaluate it with 2 different perspectives. First, we measure the certainty using the margin as we defined earlier and see the trend over time for different systems. Then, we also use a non-parametric statistical method, Bootstrap, to estimate the confidence interval (CI) to evaluate the prediction certainty.

4.3.1 Visualize Certainty Trend over Time

Due to the nature of Margin, when it is larger, our model's estimation becomes safer and determined. We use Margin defined earlier and visualize the trend over time for the three

system. We see that for rank and percent systems, most of the weeks' margins are greater than zero, suggesting a correct estimation, especially for percent. When the season comes to bottom2 system, our model's estimation increases the variance and becomes more unstable than previous seasons, but still most of them are above zero. We believe this is because the model requires a secondary judgment, and sometimes both individuals are eliminated, while other times both are retained, which increases the uncertainty to some extent.



(a) Rank (S1–S2) (b) Percent (S3–S27) (c) Bottom2 (S28–S34)

Figure 2: Margin Over Time under Different Elimination System

4.3.2 Measure overall Certainty with Bootstrap

In statistics, to capture the true parameter, people usually combines point estimate and confidence interval. Bootstrap is a statistical method to measure the certainty of our point estimation. It is a single unified approach without many assumptions (such as the distribution of votes), which is very proper for this context.

The core idea of bootstrapping is to subsample the existing data (participants weekly data) by sampling n points with replacement from the current sample $\{X_1, X_2, \ldots, X_n\}$, where each resample maintains the same size n as the original. Explicitly, this process is repeated multiple times to generate a sequence of bootstrap samples and their corresponding estimates as follows:

$$\begin{aligned} X_1^{(1)}, X_2^{(1)}, \ldots, X_n^{(1)} &\to \hat{\theta}^{(1B)}, \\ X_1^{(2)}, X_2^{(2)}, \ldots, X_n^{(2)} &\to \hat{\theta}^{(2B)}, \\ &\cdots . \end{aligned}$$

By doing so, the sampling distribution of $\hat{\theta}$ can be approximated by the distribution of these bootstrap estimates $\hat{\theta}^{(1B)}, \hat{\theta}^{(2B)}, \hat{\theta}^{(3B)}, \ldots$, and the standard deviation of $\hat{\theta}$ can be estimated by the standard deviation of this bootstrap distribution. Ultimately, these results can be used to construct bootstrap confidence intervals. We simply cut the upper and lower 2.5% bootstrap samples and construct our confidence interval (Table 3 & Figure 3).

We define the uncertainty based on the length of CI

$$Uncertainty = L_{CI} = CI_{upper} - CI_{lower}, \tag{15}$$

by using this, we divide the certainty with three levels (High/Medium/Low),

$$Very\ High\ Certainty : L_{CI} \leq 0.1$$

$$High\ Certainty : 0.1 < L_{CI} \leq 0.3$$

$$Medium\ Certainty : L_{CI} > 0.3$$

- **Very High certainty:** percent–single_elim (0.0853), bottom2–single_elim (0.0603), bottom2–bottom_2_judge_elim (0.0691). Their Lower bounds of CIs are all greater than 0.85. This suggests that our model performs good for single elimination system.
- **High certainty:** percent–bottom_k_elim (0.2490). The lower CI bound is greater than 0.7 and the upper CI bound is very close to 1, which is also stable and have a good balance of variance and bias.
- **Medium certainty:** bottom2–bottom_k_elim (0.3947), rank–single_elim (0.3254). For this two cases, we believe they have a medium certainty because their sample size is relative small. To be specific, Rank system only have 2 seasons data, and bottom_k_elim are also very rare). Thus, missing a few data points may cause larger variance in accuracy (i.e., high uncertainty).

Table 3: Certainty Estimation using Bootstrap with 95% confidence intervals

Group	Metric	Point	CI low	CI high	CI length
percent					
	single_elim	0.9080	0.8636	0.9489	0.0853
	bottom_k_elim	0.8750	0.7333	0.9823	0.2490
rank					
	single_elim	0.9000	0.6667	0.9921	0.3254
bottom2					
	single_elim	0.9787	0.9268	0.9871	0.0603
	bottom_k_elim	0.8889	0.6000	0.9947	0.3947
	bottom_2_judge_elim	0.9787	0.9268	0.9959	0.0691

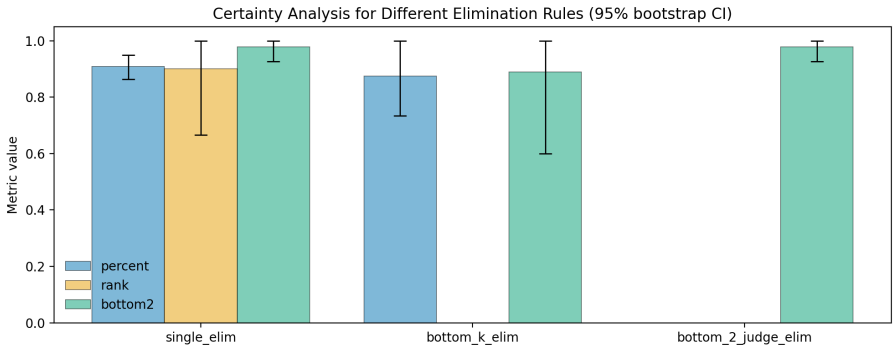


Figure 3: Certainty Estimation using Bootstrap with 95% confidence intervals

5 Task 2: Comparison of Voting Methods

5.1 Rank vs. Percent

5.1.1 Difference

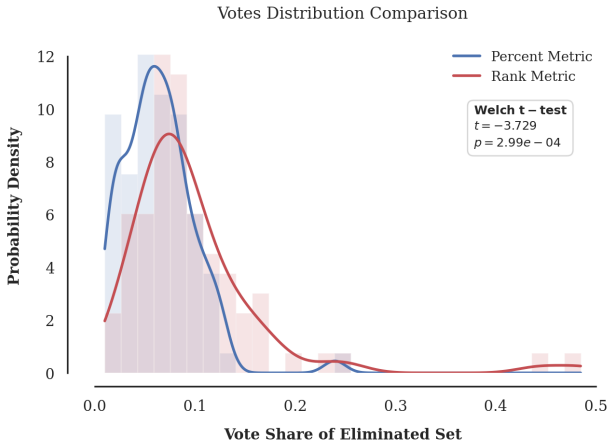
According to the problem's requirement, we applied both rank and percent system to all weeks elimination. In every week, if they results in the same elimination, we count it as 1 times, and calculate the proportion as the agreement rate

$$Agreement\ Rate = \frac{\#\ of\ same\ eliminations}{\#\ of\ elimination\ weeks} = 78.4090\%, \tag{16}$$

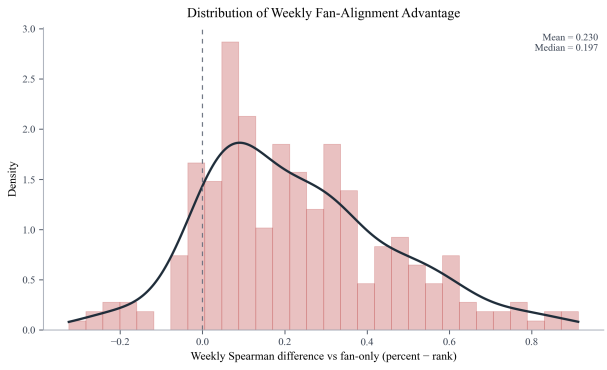
it turns out that, in the known elimination data (i.e., where the list of eliminated participants is already determined, and it is not possible to consider the elimination order of the two scenarios from beginning to end), two methods' agreement rate is about 78%. As the elimination process progresses, the two methods may show greater consistency in the weeks leading up to the final, but we believe that there are still significant differences in the contestants eliminated by each method (22% disagreement rate).

5.1.2 Which one favors fan votes more

We analyze two elimination methods with two perspectives. For the first perspective, we will identify the weeks where the two systems yield different results. Then compare your distributions of the vote share. As shown in Figure 4a, we use a histogram and add the density line with Kernel Density Estimation (KDE). We see that there is a location shift between them, and the rank system tends to have higher votes at the elimination week because it has a longer tail than the percent system. The two samples t-test (similar to the z-test here) also results in the same conclusion. It suggests that the rank system tends to eliminate participants with high vote shares, i.e., the percent system favors fan votes more than the ranking system.



(a) Vote share comparison



(b) Percent vs. Rank difference

Figure 4: Comparison of vote share and Spearman difference under different systems

We also measure the correlation between voting the rank of two systems. If one correlation is larger, it favors fan votes more than others. We use Spearman rank correlation

$$\rho = 1 - \frac{6\sum_{i=1}^{n} d_i^2}{n(n^2-1)}, \tag{17}$$

which can describe the monotonic relationship between two variables. We calculate spearman correlations between fan votes and the elimination rank for both systems each week (Figure 4b). We define the difference

$$\Delta\rho = \rho_{vote,rank} - \rho_{vote,percent}, \tag{18}$$

and the mean difference $\overline{\Delta\rho} = 0.23 > 0$. By our KDE method, $P(\Delta\rho > 0) = 0.8626$. These results suggest that the percentage system has a higher correlation with fan votes on average. Thus, percent system favors fan votes more than rank system. Both methods have the same conclusion.

5.2 Case Studies of Controversial Seasons

5.2.1 Method and rationalee

We evaluate controversy through a week-by-week counterfactual elimination test, using our estimated fan vote shares together with observed judges' totals. For each season-week, we apply alternative voting rules to the set of contestants who are observed to be active in that week, and we identify which contestant(s) would be eliminated under each rule. This design is motivated by a key data constraint: once a contestant is eliminated in the real show, their subsequent judges' scores and votes are recorded as zero. If we attempted to simulate an entire season forward after a counterfactual elimination, we would need to invent post-elimination performance and votes, which would add strong modeling assumptions not required by the prompt. Instead, our approach answers the question "when would this contestant first become the elimination target under rule X, given the information available at each elimination decision?" without extrapolating unobserved trajectories.

Concretely, we define the following evaluation objects for each controversial celebrity:

Step 1. Earliest elimination week under each aggregation rule

- **Percentage rule:** We compute a combined score using normalized weekly judge share and fan share; the contestant with the worst combined score is the week's predicted elimination.
- **Rank rule:** We convert judges and fan components into ranks and combine them; the contestant with the worst combined rank is the predicted elimination. We then record the earliest week in which the controversial celebrity is predicted as the elimination target. This "earliest week" is an interpretable risk metric: it captures the first point at which the contestant would have exited if that rule had been used, under the same week-level competitive context.

Step 2. Judges' choice within the bottom two (additional mechanism)

- **Percentage rule:** For each week, we first identify the bottom two couples under the same aggregation method. Then, under judges' choice, we eliminate the couple with the lower judges' total among those two. We again track the earliest week when the controversial celebrity would be eliminated by this modified rule.The rationale is that controversy is precisely about situations where fan support keeps a weakly judged contestant away from elimination; restricting judges' discretion to the bottom two makes the intervention limited and realistic, while still allowing us to test whether judge discretion would have changed that contestant's survival.

Importantly, this framework naturally allows an outcome of "no earliest elimination week" ("–"). This is not a computational failure: it means that, across all elimination weeks evaluated under that rule, the controversial contestant is never identified as the single worst candidate (or never selected for elimination within bottom2 when judges choose).

5.2.2 Results and interpretation

Applying the above procedure to the four controversy examples yields two key findings.

Table 4: Earliest elimination week under different competition formats

Season	Celebrity	Final result	Result 1	Result 2	Result 3	Result 4
2	Jerry Rice	2nd Place	–	–	–	7
4	Billy Ray Cyrus	5th Place	8	7	7	6
11	Bristol Palin	3rd Place	–	–	–	9
27	Bobby Bones	1st Place	–	–	–	–

Result 1: Earliest elimination week under the percent-based format
Result 2: Earliest elimination week under the rank-based format
Result 3: Earliest elimination week under the percent and judges' selection format
Result 4: Earliest elimination week under the rank and judges' selection format
"–" means the celebrity was not eliminated under this competition format.

First, the rank-based rule more frequently identifies controversial cases as vulnerable than the percentage-based rule, while some contestants are never selected as the single-week elimination target under either approach. In our elimination-week analysis, Billy Ray Cyrus is flagged by both methods, but the rank implies an earlier exit (Week 7) than the percentage (Week 8). In the same bottom-two framework, Jerry Rice and Bristol Palin are flagged under rank but not under percentage, whereas Bobby Bones never enters the bottom two, consistent with sufficiently strong inferred fan support keeping him away from the elimination margin.

This also clarifies why the earliest elimination week can be reported as "–": a contestant may exhibit substantial judge–fan disagreement yet never become the worst candidate in any evaluated week, particularly when fan vote share remains high enough to prevent last-place combined standings. Here, "–" indicates that the rule does not generate any week in which the contestant is the predicted elimination target.

Second, introducing "judges choose from the bottom two" strengthens judges' influence precisely at the margin—when contestants are near elimination—and therefore raises the elimination risk for those with persistently low judges' totals. For Billy Ray Cyrus, once placed in the bottom two, the judges' choice selects him for elimination, implying earlier exits than observed (Week 7 under percentage; Week 6 under rank). Similarly, for Jerry Rice and Bristol Palin, when the rank rule places them in the bottom two, judges' choice would eliminate them at the earliest such week (each one week earlier than observed). In contrast, Bobby Bones remains unaffected because he is not bottom-two in the evaluated weeks.

Overall, these results align with the intended role of the bottom-two judges' choice mechanism: it does not broadly override fan influence, but activates only when contestants are already marginal, concentrating its impact on cases repeatedly near the boundary. Consequently, while strong fan support can prevent some celebrities from ever becoming the single-week elimination target, the rank-based method can still dampen the effect of large fan-vote margins, and judges' choice provides a targeted channel that increases elimination risk for frequently low-ranked contestants with weak judges' scores.

5.3 Recommendation for future seasons

Based on our findings, we recommend that future seasons adopt the percentage-based approach for combining judges' scores and fan votes. The central reason is that the show's format relies on meaningful audience participation, and the percentage method preserves the magnitude of support expressed through fan voting. In contrast to rank-based aggregation—which compresses differences by treating narrow and landslide fan-vote margins similarly—the percentage method allows large gaps in fan support to translate into materially different combined outcomes.

At the same time, the controversial cases highlight an important governance issue: when judges and fans sharply disagree, a purely percentage-driven mechanism can amplify outcomes that may appear inconsistent with technical performance. For this reason, we do recommend including the additional "bottom two + judges' choice" mechanism, but explicitly as a limited safeguard rather than a season-long override of the voting process. Operationally, this structure retains the audience's role in determining the bottom two (via the combined score) while allowing judges to prevent extreme outcomes in the most contentious scenarios by selecting which of the bottom two couples to eliminate. This design targets precisely the setting where controversy arises—high judge–fan disagreement coupled with survival at the margin—without broadly diminishing the weight of fan votes throughout the season.

To maintain transparency and audience trust, we suggest constraining the judges' choice mechanism to clearly defined circumstances, such as late-stage weeks (e.g., late-stage weeks) or weeks where the combined results are especially close. Under these constraints, the show can preserve the core objective of strong audience agency through the percentage method while adding a transparent, narrowly tailored "safety valve" that reduces the likelihood of highly controversial eliminations. Overall, this hybrid recommendation best balances engagement and legitimacy: percentage aggregation ensures fan votes remain consequential, and judges' choice within the bottom two mitigates the most extreme judge–fan conflicts without fundamentally shifting the show away from a vote-driven competition.

6 Task 3: Impact of Dancers and Celebrity Characteristics

6.1 Model Used

For this task, we use several statistical tree-based regression models to analyze the impact factors. This is because we can get combined effects in regression, instead of letting other variables to float. And trees can capture non-linearity and consider the interaction between features. After we trained these models, we use Permutation score and the tree-based feature importance score to measure the impact of dancers and celebrity effects.

The first baseline model is random forest which belongs to the bagging methods. The core idea is to use bagging & random feature selection to reduce the variance and prevent overfitting. Other models are XGBoost and LightGBM, which are boosting tree methods. For boosting methods, we build many small trees sequentially, each new tree focuses on correcting the errors of the current model. Mathematically,

$$\hat{f}_M(x) = \sum_{m=1}^{M} \nu \, T_m(x) \tag{19}$$

where $T_m(x)$ is the prediction of tree m, ν is the learning rate (small step size), and M is the number of trees. This may allow us to further reduce bias.

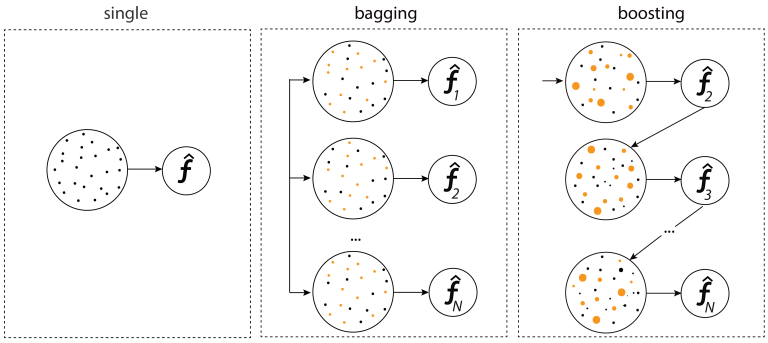


Figure 5: Bagging versus Boosting Intuition

6.2 Impact Factors for Judge Score Total

We only used features related to various professional dancers and the characteristics of the celebrities, with the aim of visually analyzing the magnitude of the impact. Since our goal is to find impact factors rather than prediction, we did not divide the data into a test set. The model's training set performance is in Table 5.

MAE is the Mean Absolute Error, MSE is the Mean Squared Error (our loss function optimization objective), and RMSE is the Root Mean Squared Error. MAE intuitively shows that our average model fitting error is around 3, and $R^2 = 0.58$ means that our models, using only the features of the dance partner and the contestant, can explain 58% of the variability in the judges' scores, which is acceptable. Those three models perform similarly in accuracy, so we think their feature importance scores all have reference value.

Table 5: Model performance for predicting judge score total

Model	MAE	MSE	RMSE	R^2
RandomForest	3.090925	16.575232	4.071269	0.585674
LightGBM	3.096321	16.625695	4.077462	0.584413
XGBoost	3.099003	16.659602	4.081618	0.583565

Once we make sure our fitted models are useful, we measure a feature's contribution by the permutation importance score. It is calculated by the drop in predictive performance when that feature is randomly permuted. Let $S(\cdot, \cdot)$ be a score (here we use R^2). For feature j, we compute

$$I_j = S\left(y, \hat{f}(X)\right) - S\left(y, \hat{f}(X^{\pi_j})\right), \tag{20}$$

where X^{π_j} is obtained by shuffling column j. We repeat the permutation R times and average I_j to reduce randomness. This method is model-agnostic and comparable across different learners. Note that this measurement is the overall feature importance score for each dimension.

In Figure 6, we see that a celebrity's age and ballroom partner have a high impact on judge score. Celebrity's home state and industry are medium-level important factors. A celebrity's home country or region has the lowest importance for the judge score.

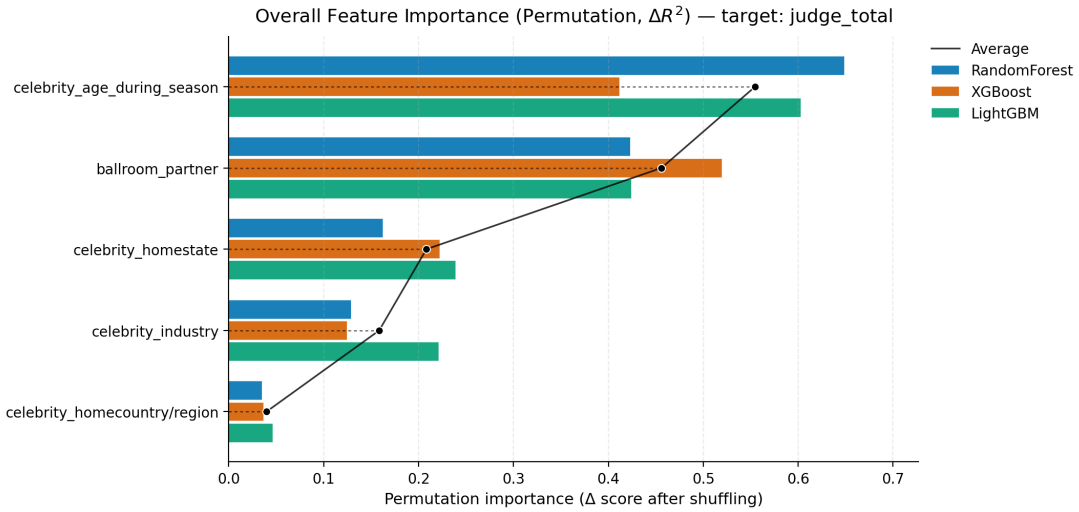


Figure 6: Permutation score for judge score (Overall Importance for each dimension)

To analyze separated (category-level) feature importance, we use impurity-based importance for Random Forest and gain (information gain) for the boosting models. For comparability across models, we normalize importances within each model so that they sum to 1, then average the normalized scores and select the top categories within each dimension. We do not include continuous variables (e.g., age) in this category-ranking table because split-based importances can be biased toward features with many potential split points or high cardinality; continuous variables are therefore analyzed via permutation importance. Finally, since we only report categories that appear among the overall top-30 most important

one-hot features, some dimensions may contain fewer than eight categories in the table. The separated feature importance is reported in Table 6.

Table 6: Separated Feature Importance (Top 30, for Judge Score). Each cell reports the category and the mean normalized importance across three models (in %).

Rank	ballroom_partner	celebrity_industry	celebrity_homestate	homecountry/region
1	Artem Chigvintsev (3.514)	TV Personality (2.043)	Michigan (2.022)	Russia (1.168)
2	Witney Carson (3.288)	Actor/Actress (1.977)	California (1.824)	United States (1.085)
3	Valentin Chmerkovskiy (2.717)	Social Media Personality (1.691)	Georgia (1.403)	–
4	Derek Hough (1.496)	Athlete (1.523)	New York (1.173)	–
5	Tony Dovolani (1.278)	Singer/Rapper (0.967)	Louisiana (0.839)	–

6.3 Impact Factors for Vote Share

For vote share, we use the same method to evaluate impact factors (Figure 7). We see that the permutation importance score has a clear difference from judge scores. The first-tier importance factors are the ballroom partner and the celebrity's homestate. The second-tier importance factors are the celebrity's age and industry. And the average score (line) drops significantly in the celebrity's homecountry, so this factor is not important for the vote share.

Table 7: Model performance for predicting vote share

Model	MAE	MSE	RMSE	R^2
RandomForest	0.030731	0.002933	0.054154	0.587407
LightGBM	0.030979	0.002945	0.054271	0.585632
XGBoost	0.031313	0.002976	0.054555	0.581278

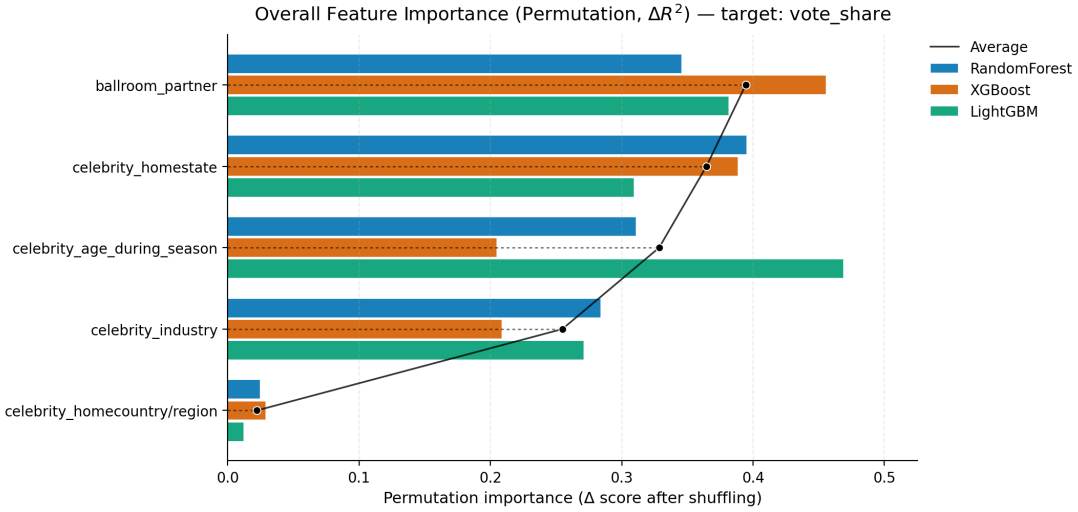


Figure 7: Permutation score for vote share (Overall Importance for each dimension)

Table 8: Separated Feature Importance (Top 30, for Vote Share). Each cell reports the category and the mean normalized importance across three models (in %).

Rank	ballroom_partner	celebrity_industry	celebrity_homestate	homecountry/region
1	Cheryl Burke (2.282)	Actor/Actress (2.260)	Mississippi (2.823)	–
2	Gleb Savchenko (1.719)	Athlete (2.242)	Georgia (2.340)	–
3	Britt Stewart (1.647)	TV Personality (1.141)	California (1.903)	–
4	Sasha Farber (1.636)	Motivational Speaker (1.132)	Utah (1.763)	–
5	Anna Trebunskaya (1.470)	Singer/Rapper (0.928)	Washington D.C. (1.583)	–

6.4 Comparison & Interpretation of Impact Factors for Judge Score & Vote Share

According to our findings, for ballroom partners and celebrities' home country or region, their importance is close to each other. But for celebrities' age, home state, and industry, they don't impact judge score and vote share in the same way (Figure 8). We analyze potential reasons based on the overall and separated feature importance scores.

- **celebrity's age**: Judges' scoring is more directly tied to *technical execution* (timing, posture, stamina, injury risk, and learning curve), all of which correlate with age. Older

contestants may face greater physical constraints and slower skill acquisition, which can translate into consistently lower technique-related subcomponents, while younger contestants may more easily deliver cleaner lines and higher-energy routines. By contrast, fan voting is less "mechanics-driven" and can be mediated by popularity, narrative, and emotional attachment, so age alone is a weaker standalone driver of vote share.

- **celebrity's home state**: Vote share is influenced by *audience concentration and mobilization*. Home-state effects can arise from local fan bases, regional pride, and geographically clustered communities (including alumni networks and local media exposure) that coordinate voting. Judges, however, are (in principle) evaluating the same performance criteria regardless of where a celebrity is from, so geography should matter much less for judge totals except indirectly through correlated factors (e.g., prior training opportunities), making the home-state signal comparatively stronger in votes than in scores.

- **celebrity's industry**: Industry primarily shapes *public familiarity and parasocial engagement*, which can drive fans' willingness to vote (e.g., performers with highly active online followings or strong media presence can mobilize audiences more effectively). Judges are less responsive to "who the celebrity is" and more to "what the celebrity does on the floor"; industry may only help insofar as it proxies prior relevant skills (e.g., stage experience, musicality), but those advantages are often partially offset by judges focusing on technique. Hence industry tends to have a larger impact on vote share than on judge scoring.

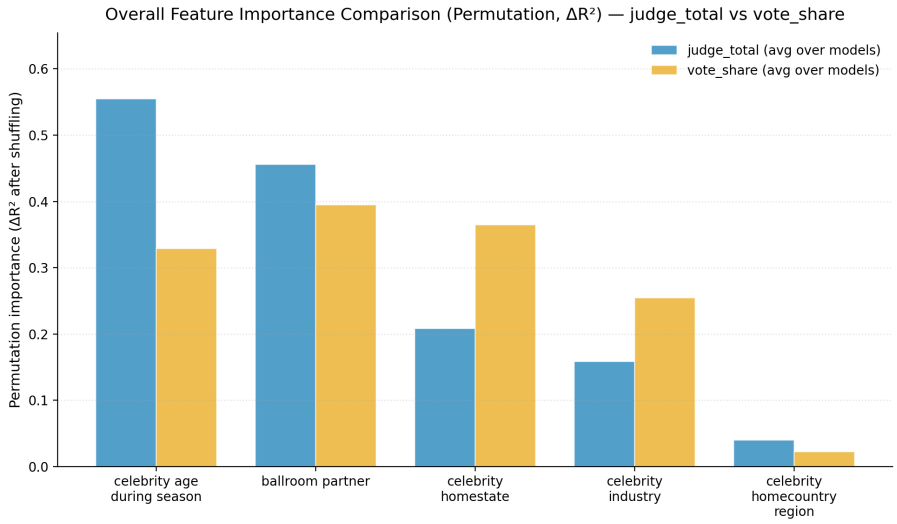


Figure 8: Feature importance comparison for Judge score & Vote share

7 Task 4: Proposal of a New Voting System

7.1 Proposed Voting System: Dynamic Stage-Dependent Weighting

To strengthen early-stage audience engagement while preserving competitive integrity in the later stage, we propose a dynamic voting system that adjusts the relative importance of fan votes and judges' scores as the season progresses. Let $J_{i,t}$ denote contestant i's normalized judges' score in week t, and $V_{i,t}$ denote the normalized fan vote share. We define the combined score as

$$S_{i,t} = w_J(t)\, J_{i,t} + w_V(t)\, V_{i,t}, \qquad w_J(t) + w_V(t) = 1, \tag{21}$$

where $w_J(t)$ and $w_V(t)$ are the week-dependent weights on judges and fans, respectively. Define season progress by $p_t = t/T$, where T is the total number of weeks in the season. The judges' weight follows a three-stage stepwise schedule:

$$w_J(t) = \begin{cases} 0.3, & p_t < \frac{1}{3}, \\ 0.5, & \frac{1}{3} \le p_t < \frac{2}{3}, \\ 0.7, & p_t \ge \frac{2}{3}, \end{cases} \qquad w_V(t) = 1 - w_J(t). \tag{22}$$

This design makes the early stage fan-dominant to enhance suspense and perceived voting impact, while gradually increasing judges' influence so that the late stage becomes more performance-driven and less sensitive to short-term popularity shocks. As a baseline, we consider a fixed-weight model $S^{\text{old}}_{i,t} = 0.5 J_{i,t} + 0.5 V_{i,t}$.

7.2 Simulation and Evaluation Metrics

We replay each season week-by-week using observed $(J_{i,t}, V_{i,t})$ and the recorded number of eliminations k_t. In week t, contestants are ranked by $S_{i,t}$ and the bottom k_t are eliminated. All computations are restricted to contestants alive in week t to avoid artifacts from post-elimination placeholder values. We report results by season stage (early/mid/late) according to $p_t = t/T$.

To quantify suspense, we measure the boundary margin

$$\Delta^{\text{score}}_t = S_{\text{lowest safe},t} - S_{\text{highest eliminated},t}, \tag{23}$$

where a smaller Δ^{score}_t indicates a tighter cutoff and thus more suspenseful weekly outcomes. To assess late-stage performance alignment, we compute the weekly Spearman correlation (among alive contestants)

$$\rho_t = \text{Spearman}(S_{\cdot,t}, J_{\cdot,t})\,. \tag{24}$$

We further examine popularity-bias control by tracking contestants who are simultaneously low on judges and high in popularity within the same week (defined by within-week percentiles $P^J \le 0.2$ and $P^V \ge 0.8$), and report their survival rates. Finally, to support practical adoptability, we compare predicted weekly elimination sets to historical eliminations using the exact match rate, reported stage-wise.

7.3 Results and Interpretation

Table 9: Stage-wise Mean Metrics (Baseline vs. Proposed)

Metric (mean)	Stage	Baseline (fixed 0.5/0.5)	Proposed (dynamic weights)
Boundary margin Δ_t^{score}	Early	0.096995	0.095868
	Mid	0.118011	0.120335
	Late	0.129373	0.219289
Spearman (S,J)	Early	0.694350	0.356857
	Mid	0.581729	0.573921
	Late	0.435141	0.936317
Survival rate of controversial individuals	Early	1.000000	1.000000
	Mid	1.000000	0.991935
	Late	0.845070	0.269231
Exact match	Early	0.778947	0.705263
	Mid	0.462963	0.444444
	Late	0.454545	0.424242

Table 9 summarizes the stage-wise mean performance of the baseline and the proposed dynamic-weight system. In the early stage, the proposed system yields a slightly smaller suspense margin (mean $\Delta_t^{\text{score}} = 0.0959$ vs. 0.0970), consistent with the goal of strengthening fan-perceived impact and weekly suspense. More importantly, the proposed system exhibits a strong late-stage shift toward performance alignment: the mean Spearman(S, J) rises to 0.9363 in the late stage under the proposed system, compared with 0.4351 under the baseline, indicating that late outcomes are far more consistent with judges' technical assessments. This shift is reinforced by the popularity-bias analysis: among "low-judge/high-popularity" contestants, the late-stage mean survival rate drops from 0.8451 (baseline) to 0.2692 (proposed), suggesting that contestants cannot rely on short-term popularity surges to persist deep into the season. While the proposed system is somewhat less consistent with historical eliminations (e.g., early-stage exact match mean 0.7053 vs. 0.7789), this divergence is expected because the objective is not to replicate baseline outcomes, but to provide a principled mechanism that is more engaging early and more performance-driven late. Notably, the late-stage increase in Δ_t^{score} (mean 0.2193) indicates a "cleaner" separation at the elimination boundary when judges carry more weight, which is consistent with reduced randomness in the decisive phase.

8 Strengths and Weaknesses

Strengths

- The model incorporates different elimination mechanisms into the loss function, enabling adaptation to rule changes and ensuring consistency with observed outcomes.
- Consistency analysis and bootstrap-based uncertainty estimation provide structural and statistical validation, improving estimation stability.
- The interaction between judges' scores, fan votes, and elimination results is explicitly modeled, ensuring good interpretability for fairness analysis.

Weaknesses

- Fan votes are inferred from elimination outcomes rather than directly observed, leading to inherent identifiability limitations.
- The model relies only on provided data and does not include external popularity indicators, limiting its ability to capture short-term behavioral changes.

References

[1] Brandt, F., Conitzer, V., Endriss, U., Lang, J., & Procaccia, A. D. (Eds.). (2016). *Handbook of computational social choice*. Cambridge University Press.

[2] Caragiannis, I., Chatzigeorgiou, X., Krimpas, G. A., & Voudouris, A. A. (2019). Optimizing positional scoring rules for rank aggregation. *Artificial Intelligence, 267*, 58–77.

[3] Ratner, A., Bach, S. H., Ehrenberg, H., Fries, J., Wu, S., & Ré, C. (2020). Snorkel: Rapid training data creation with weak supervision. *The VLDB Journal, 29*, 709–730.

[4] Fisher, A., Rudin, C., & Dominici, F. (2019). Model class reliance: Variable importance measures for any machine learning model class, from the "Rashomon" perspective. *Journal of Machine Learning Research, 20*(177), 1–81.

[5] Debeer, D., & Strobl, C. (2020). Conditional permutation importance revisited. *BMC Bioinformatics, 21*.

[6] Biau, G., & Scornet, E. (2016). A random forest guided tour. *TEST*.

[7] Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '16)*.

[8] Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., & Liu, T.-Y. (2017). LightGBM: A highly efficient gradient boosting decision tree. In *Advances in Neural Information Processing Systems*.

9 Memo

Dear Sir or Madam,

It is our great honor to present our data-driven analysis and recommendations regarding the voting mechanisms used in Dancing with the Stars (DWTS). Based on the historical competition data from Seasons 1 to 34, our team developed a comprehensive mathematical framework to estimate fan votes, evaluate elimination fairness, and analyze performance dynamics.

By integrating statistical modeling, optimization techniques, and machine learning methods, we constructed a multi-layer evaluation system that captures both judges' technical assessments and audience preferences. Our approach enables us to reconstruct hidden fan voting patterns, quantify uncertainty, and assess the stability of different elimination systems. To achieve these objectives, we carried out the following key tasks:

First, we established a vote estimation model that infers weekly fan vote shares from observed judges' scores and elimination outcomes. By formulating customized loss functions under rank-based, percentage-based, and bottom-two systems, we ensured that our estimated votes were consistent with historical results while maintaining numerical stability.

Second, we designed a series of consistency and certainty metrics, including elimination accuracy, violation measures, margin analysis, and bootstrap confidence intervals. These indicators allow us to systematically evaluate how reliably each voting method reflects true audience preferences.

Third, we conducted a comparative analysis of the three voting mechanisms across all seasons. Our results indicate that the percentage-based system generally provides higher stability and better representation of fan participation, while rank-based systems tend to reduce sensitivity to extreme voting behaviors. The judges' save mechanism effectively mitigates controversial eliminations in highly competitive weeks.

Fourth, we examined several historically controversial seasons using counterfactual simulations. By reconstructing alternative elimination paths, we demonstrated how different aggregation rules would have altered competition outcomes. These analyses provide quantitative evidence for understanding public dissatisfaction and judging fairness.

Furthermore, we employed tree-based machine learning models, including Random Forest, LightGBM, and XGBoost, to investigate the impact of professional dancers and celebrity characteristics. Our findings reveal that factors such as partner quality, age, and industry background significantly influence judges' scores and fan votes, though their effects differ across evaluation channels.

Based on our comprehensive analysis, we draw the following main conclusions:

- Fan voting behavior can be reliably inferred using constrained optimization and probabilistic modeling.
- Elimination outcomes are not random but follow interesting structural patterns.
- The percentage-based aggregation method most effectively balances fairness, transparency, and audience engagement.
- The judges' save mechanism serves as a critical safeguard in close competitions.
- Professional partnerships and demographic features play a decisive role in long-term performance.

Accordingly, we propose the following recommendations for future seasons:

- Adopt the Dynamic weighting mechanism voting system as the primary aggregation method to enhance transparency and audience trust.
- Monitor contestant momentum and popularity dynamics using real-time analytical indicators.
- Utilize performance profiling to support targeted training and casting strategies.

We believe that our data-driven framework provides a robust foundation for optimizing competition design and enhancing viewer experience. By integrating quantitative evaluation with domain expertise, DWTS can further strengthen its credibility and long-term appeal.

We sincerely hope that our findings and recommendations will be helpful. Should you require any further clarification, please feel free to contact us.

~

Warm regards,
The Modeling Team