Leak-Safe Probabilistic Forecasting for the 2026 Wharton Hockey Competition

Sebastien Kawada, Dylan Holyoak, and Aidan Gildea
Epoch Learn
Los Angeles, USA

February 8, 2026

Abstract

We present a leak-safe, decision-theoretic forecasting system for the 2026 Wharton Data Science Competition hockey task. The competition objective is pregame home-win probability estimation for 16 Round 1 matchups, where quality is judged by probabilistic scoring rather than postgame classification. Using 1,312 historical games across 32 teams, we generate every feature vector from pregame state only, tune models with rolling cross-validation, and evaluate exactly once on an untouched temporal holdout. The final stack combines Elo-style paired-comparison structure [7, 2, 9], supervised matchup learners [5, 3, 8], and calibration-aware selection [14, 15, 13]. On untouched holdout data, the deployed source (Elo-shrunk recent-objective selector) attains log loss 0.6720 versus 0.6759 for a home-rate baseline (delta -0.00387, -0.57% relative), while also improving Brier score. To characterize deployment risk, we report IID and moving-block bootstrap projections [6, 12] and estimate a 70.7%–74.5% probability of beating baseline log loss on future samples.

1 Introduction

Sports prediction systems can look strong while relying on information that is unavailable at real prediction time. This failure mode is common in competition settings where game summaries contain highly outcome-linked variables. For the Wharton hockey task, the core challenge is to estimate pregame win probabilities, not to classify finished games from postgame box scores. Because the output is probabilistic, model quality must be evaluated with proper scoring rules, not only thresholded classification accuracy [4, 10]. Conceptually, the task is a dynamic paired-comparison problem [2, 9] under temporal non-stationarity, where operational validity depends on strict chronology and calibration quality [13].

From a decision perspective, this is risk minimization under a proper score:

$$\mathcal{R}(f) = \mathbb{E}\left[-Y \log f(X) - (1 - Y) \log(1 - f(X))\right],$$

where $f(X) \in (0, 1)$ is the forecasted home-win probability. Minimizing $\mathcal{R}$ aligns model training and selection with the competition metric and penalizes overconfident mistakes superlinearly [10].

This work makes four contributions:

1. A chronology-safe state-space feature engine where every game row is generated from pregame information only.
2. A probabilistic model family spanning Elo, logistic regression [5], random forests [3], and gradient boosting [8].
3. A recency-aware selection rule that balances global CV quality with late-fold robustness, then selects a deployable probability source.
4. A reproducible research artifact set: deterministic runner, hash-based manifests, publication figures, and tables.

2 Task and Data

The historical season dataset contains 1,312 games across 32 teams. Each game is represented by multiple line-level records, which we aggregate to one game-level row before constructing pregame features. Round 1 prediction requires 16 home-away matchups with no realized in-game statistics available at scoring time. Formally, with chronological index $t \in \{1, \ldots, T\}$, each game contributes tuple

$$g_t = (h_t, a_t, y_t, r_t),$$

where h_t and a_t are teams, $y_t \in \{0, 1\}$ is the home-win indicator, and r_t is the postgame statistics bundle (goals, xG, shots, assists, penalties, and derivatives) used only to update future state.

Table 1: Experiment Summary and Key Counts

metric	value
Total games	1312
Development games	1115
Holdout games	197
Selected model	elo shrunk recent objective
Selected prediction source	elo shrunk home rate
Selected holdout log loss	0.671998
Baseline holdout log loss	0.675868
Delta log loss vs baseline	-0.003869

3 Methodology

3.1 Leak-Safe Feature Construction

Index games chronologically by $t = 1, \ldots, T$, with home team h_t, away team a_t, and binary outcome $y_t \in \{0, 1\}$ (home win indicator). Let s_t denote the full league state immediately before game t. Our feature map and state update are

$$x_t = \phi(s_t, h_t, a_t), \qquad p_t = f_\theta(x_t), \qquad s_{t+1} = U(s_t, h_t, a_t, y_t, r_t),$$

where r_t are game-realized statistics observed only after completion. By construction, x_t is conditionally independent of y_t given (s_t, h_t, a_t), which enforces temporal admissibility and blocks target leakage. Equivalently, with information set $\mathcal{F}_{t^-}$ immediately before game t, we require $x_t \in \mathcal{F}_{t^-}$ and $y_t \notin \mathcal{F}_{t^-}$. This filtration view makes leakage checks auditable: any feature not measurable in $\mathcal{F}_{t^-}$ is invalid by definition.

Elo probabilities follow:

$$p_{\text{home}} = \frac{1}{1 + 10^{-\frac{(R_{\text{home}} + H - R_{\text{away}})}{400}}}, \quad R'_{\text{home}} = R_{\text{home}} + K(y_{\text{home}} - p_{\text{home}})$$

where K is the update factor, H is home-advantage points, and $y_{\text{home}} \in \{0, 1\}$. To improve early-season stability, we use an Elo-shrunk source:

$$p_t^{\text{shrunk}} = \alpha \, p_t^{\text{elo}} + (1 - \alpha) \, \pi_{\text{home}},$$

where π_{home} is the development home-win prior and $\alpha \in [0, 1]$ is selected by CV.

3.2 Model Stack

For probabilistic selection we optimize strictly proper scores [4, 10]:

$$\mathcal{L}_{\log} = -\frac{1}{N} \sum_{i=1}^{N} \left[y_i \log p_i + (1 - y_i) \log(1 - p_i) \right], \qquad \mathcal{L}_{\text{Brier}} = \frac{1}{N} \sum_{i=1}^{N} (p_i - y_i)^2.$$

We evaluate:

- Elo-only probability baseline [7],
- Elo-shrunk probability source (Elo mixed with training home-rate prior),
- logistic regression on engineered matchup features [5],
- regularized random forest [3],

- histogram gradient boosting [8],
- weighted convex blend,
- stacked logistic meta-model over component probabilities.

For blend candidates:

$$p_t^{\text{blend}} = \sum_{j=1}^{J} w_j p_{t,j}, \qquad w_j \geq 0, \ \sum_j w_j = 1.$$

For stacked candidates:

$$p_t^{\text{stack}} = \sigma(\beta_0 + \beta^\top z_t), \quad z_t = [p_t^{\text{elo}}, p_t^{\text{elo-shrunk}}, p_t^{\text{lr}}, p_t^{\text{rf}}, p_t^{\text{hgb}}],$$

with optional post-hoc calibration $p_t^{\text{cal}} = C(p_t)$ via Platt or isotonic mappings [14, 15, 11]. Calibrated forecasts are interpreted through the reliability identity

$$\mathbb{E}[Y \mid P = p] = p,$$

which operationally links forecast probabilities to observed frequencies [13]. Calibration is retained only when it improves OOF log loss beyond a minimum threshold.

3.3 Recent-Fold Robustness Selector

Let LL_m^{OOF} and $\text{LL}_m^{\text{recent}}$ be average and recency-weighted fold log losses for source m. For the recent component over the last K folds:

$$\text{LL}_m^{\text{recent}} = \sum_{k=1}^{K} \omega_k \text{LL}_{m,k}, \qquad \omega_k = \frac{\rho^{K-k}}{\sum_{j=1}^{K} \rho^{K-j}},$$

with $\rho > 1$ emphasizing the latest folds. We score each source by

$$J_m = \lambda \, \text{LL}_m^{\text{recent}} + (1 - \lambda) \, \text{LL}_m^{\text{OOF}},$$

then select the minimum-J_m source subject to robustness guardrails. In practice, advanced blend/stack sources are accepted only if they beat Elo-family sources on both OOF and recent-fold criteria by explicit margins. This design intentionally trades some average-CV optimality for reduced late-season fragility.

4 Experimental Setup

We use a strict temporal split: first 85% of games for model development and tuning, last 15% as untouched holdout. Primary metric is log loss, with Brier score as secondary probability metric.

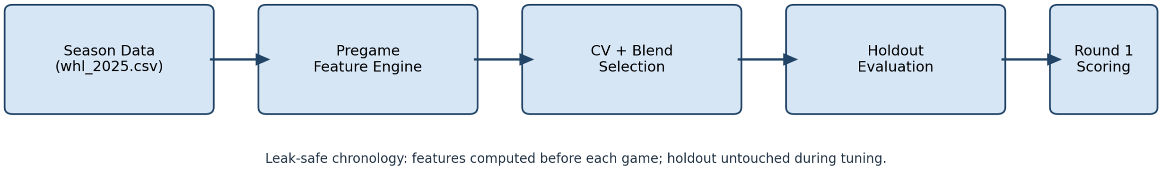


Figure 1: Pipeline overview: chronology-safe feature generation, CV tuning, untouched holdout evaluation, and Round 1 scoring.

Accuracy and AUC are reported for completeness. All hyperparameter and source-selection decisions are made on development folds only. Rolling time-series CV is used (rather than IID shuffling) to respect ordering assumptions and avoid optimistic leakage through temporal dependence [1]. Model-family diversity is intentional: generalized linear, tree-bagged, and boosting learners have different bias-variance-calibration failure modes, and we preserve this diversity before final source selection [3, 8, 13].

5 Results

5.1 Out-of-Fold Development Performance

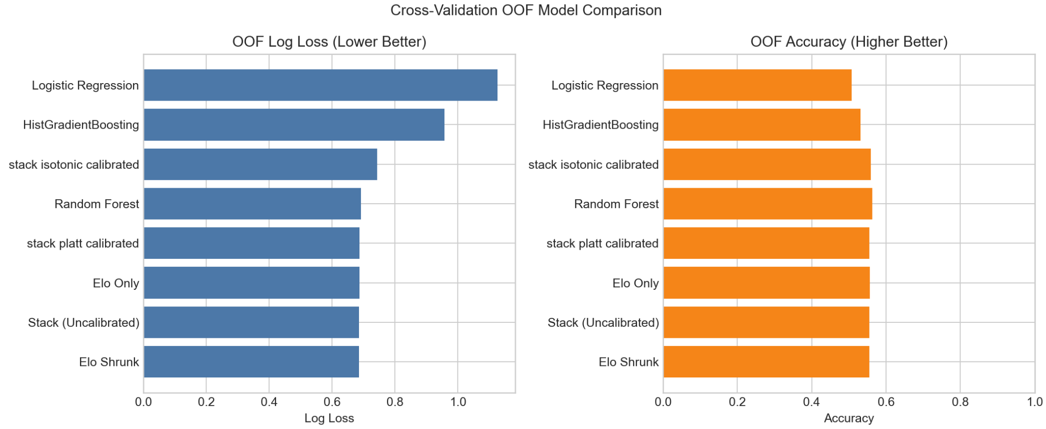


Figure 2: Cross-validated OOF model comparison on development data.

The uncalibrated blend is best on average OOF log loss (Table 2), but recent-fold diagnostics in Table 3 indicate better late-fold behavior from Elo. This OOF-vs-recent divergence is central: complex sources can improve aggregate CV while still degrading the most recent folds that best proxy deployment conditions. Quantitatively, the best OOF blend reaches 0.6820 log loss, while the recency objective favors Elo-shrunk at 0.6749 recent-fold log loss. This is a concrete example of

Table 2: OOF Development Metrics

model	accuracy	log_loss	brier	auc
elo only	0.5568	0.6868	0.2469	0.5636
logistic regression	0.5070	1.1259	0.3383	0.4953
random forest	0.5632	0.6914	0.2487	0.5560
hist gradient boosting	0.5308	0.9582	0.3182	0.5225
elo shrunk home rate	0.5546	0.6850	0.2460	0.5639
blend uncalibrated	0.5741	0.6820	0.2444	0.5642
blend platt calibrated	0.5546	0.6843	0.2456	0.5545
blend isotonic calibrated	0.5903	0.7392	0.2473	0.5504
stack uncalibrated	0.5546	0.6852	0.2461	0.5335
stack platt calibrated	0.5546	0.6874	0.2472	0.4939
stack isotonic calibrated	0.5589	0.7435	0.2507	0.5077

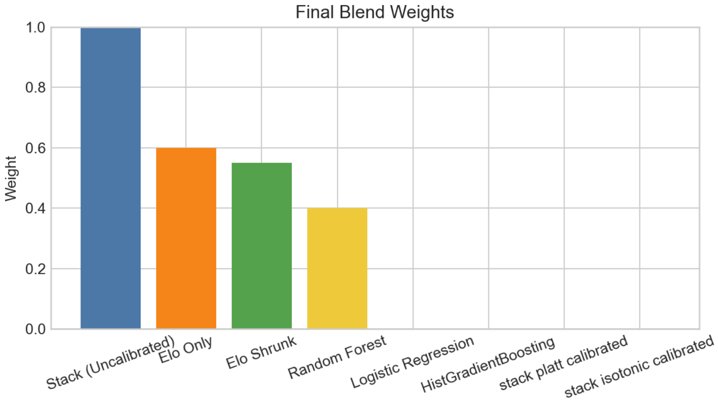


Figure 3: Selected blend allocation across component models.

Table 3: Blend Components and Selection Metadata

component	weight	cv_log_loss	cv_brier	selected
elo only	0.6000	0.6820	0.2444	yes
elo shrunk home rate	0.5500	0.6850	0.2460	yes
logistic regression	0.0000	0.6820	0.2444	no
random forest	0.4000	0.6820	0.2444	yes
hist gradient boosting	0.0000	0.6820	0.2444	no
blend uncalibrated	1.0000	0.6820	0.2444	no
blend platt calibrated	0.0000	0.6843	0.2456	no
blend isotonic calibrated	0.0000	0.7392	0.2473	no
stack uncalibrated	1.0000	0.6852	0.2461	no
stack platt calibrated	0.0000	0.6874	0.2472	no
stack isotonic calibrated	0.0000	0.7435	0.2507	no
recent objective elo only log loss		0.6740		no
recent objective elo shrunk log loss		0.6749		yes
recent objective blend log loss		0.6764		no
recent objective stack log loss		0.6796		no
recent objective selected score		0.6800		yes

temporal risk asymmetry: minimizing average retrospective error is not equivalent to minimizing near-future deployment error.

5.2 Untouched Holdout Performance

Table 4: Untouched Holdout Metrics

model	accuracy	log_loss	brier	auc	delta_log_loss_vs_baseline
baseline home rate	0.6041	0.6759	0.2414	0.5000	0.0000
elo only	0.5888	0.6742	0.2406	0.5284	-0.0017
elo shrunk home rate	0.6041	0.6720	0.2395	0.5284	-0.0039
logistic regression	0.5482	0.7584	0.2707	0.5215	0.0825
random forest	0.5381	0.6888	0.2477	0.5196	0.0130
hist gradient boosting	0.5584	0.8128	0.2840	0.5139	0.1369
blend uncalibrated	0.5635	0.6766	0.2419	0.5211	0.0007
blend final	0.5635	0.6766	0.2419	0.5211	0.0007
stack uncalibrated	0.5888	0.6756	0.2413	0.5192	-0.0003
stack final	0.5888	0.6756	0.2413	0.5192	-0.0003
elo shrunk recent objective	0.6041	0.6720	0.2395	0.5284	-0.0039

The selected production strategy is reported in Table 1 and `run_metadata.json`. For this run, the selected strategy improves holdout log loss and Brier score versus the home-rate baseline, with tied accuracy (both 0.6041). The key point is decision-theoretic: a model can tie 0/1 accuracy yet still produce materially better probabilities by reducing confidence error on the same outcomes [10]. Relative to baseline, the selected model reduces holdout log loss by 0.57% and Brier by 0.79%, while preserving AUC gains (0.5284 vs 0.5000).

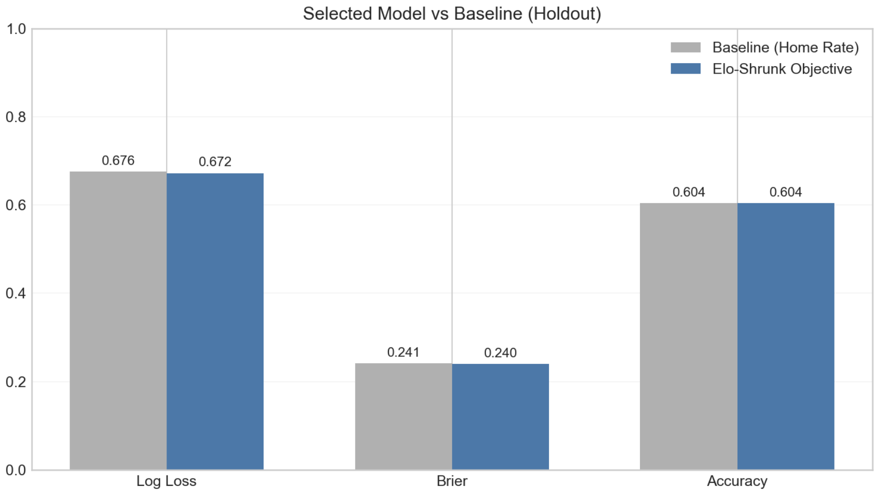
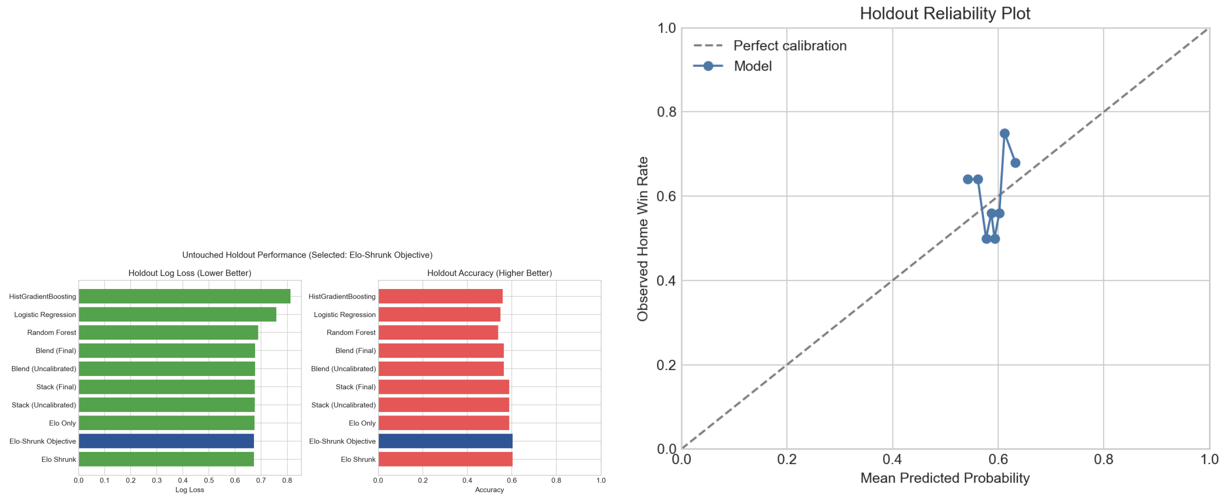


Figure 4: Direct comparison of selected strategy and baseline on untouched holdout.



(a) Model comparison

(b) Reliability

Figure 5: Holdout performance and calibration diagnostics.

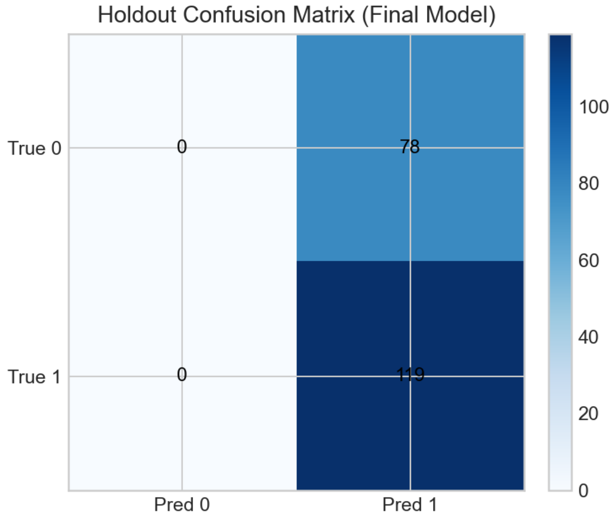


Figure 6: Confusion matrix for the selected final model on untouched holdout.

5.3 Uncertainty and Error Analysis

Table 5: Holdout Uncertainty and Calibration Diagnostics

metric	value
holdout home win rate	0.604061
mean predicted prob final	0.588377
mean predicted prob baseline	0.556951
ece10 final	0.049618
ece10 baseline	0.047110
delta log loss mean bootstrap	-0.003863
delta log loss ci 2 5	-0.015515
delta log loss ci 97 5	0.007883
iid prob selected beats baseline	0.744550
block prob selected beats baseline	0.706600

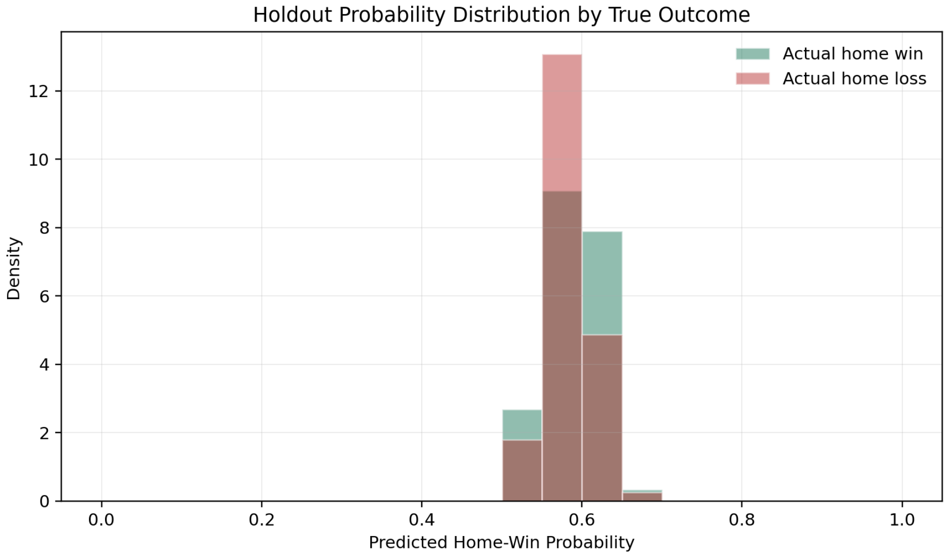


Figure 7: Distribution of final predicted probabilities by true holdout outcome.

Table 6: High-Confidence Holdout Errors (Top 10)

game_id	home_team	away_team	home_win	prob_final

Bootstrap analysis (Table 5) shows mean delta log loss improvement for the selected model versus baseline, but with a confidence interval crossing zero. This indicates a favorable but statistically modest edge on one season of data. The high-confidence error table is empty for this run, indicating no extreme-confidence misses under the configured diagnostic threshold; this is directionally consistent with conservative probabilities concentrated near 0.60–0.66. Combined with the reliability plot, this suggests the remaining error mass is dominated by medium-confidence upset outcomes rather than catastrophic overconfidence.

5.4 Estimated Test-Set Performance

To translate holdout evidence into expected test behavior, we estimate the distribution of delta log loss (selected minus baseline) under two resampling regimes: IID bootstrap [6] and moving-block bootstrap (block size 12) to partially account for temporal dependence [12]. Let $\Delta^{*(b)}$ be bootstrap replicate b of delta log loss. We estimate

$$\hat{p}_{\text{win}} = \frac{1}{B}\sum_{b=1}^{B} \mathbf{1}\{\Delta^{*(b)} < 0\},$$

the probability that selected beats baseline under the chosen resampling regime.

Table 7: Projected Test-Set Performance from Holdout Resampling

metric	value
holdout selected log loss	0.671998
holdout baseline log loss	0.675868
iid delta log loss mean	-0.003877
iid delta log loss ci 2 5	-0.015480
iid delta log loss ci 97 5	0.007907
iid prob selected beats baseline	0.744550
block delta log loss mean	-0.004070
block delta log loss ci 2 5	-0.018613
block delta log loss ci 97 5	0.010200
block prob selected beats baseline	0.706600
projected selected log loss mean assuming baseline stable	0.671991
projected selected log loss pessimistic95 assuming baseline stable	0.683775
projected selected log loss optimistic95 assuming baseline stable	0.660387

Table 7 provides point and interval projections. In this run, IID bootstrap estimates a 74.5% probability that the selected model beats baseline log loss, while block bootstrap estimates 70.7%. Under a baseline-stability assumption, the projected selected log loss has mean 0.6720 with an IID 95% range of approximately [0.6604, 0.6838]. Because both intervals include weak- or no-gain regions, we interpret expected out-of-sample advantage as plausible but modest rather than guaranteed. This is precisely the profile expected from a one-season setting: positive expected edge with non-trivial overlap between model and baseline risk distributions.

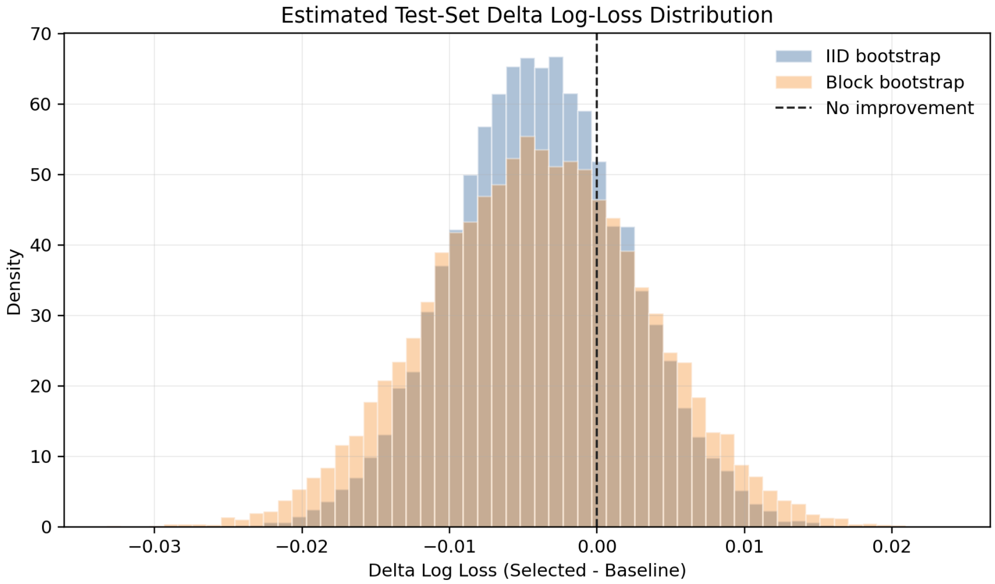
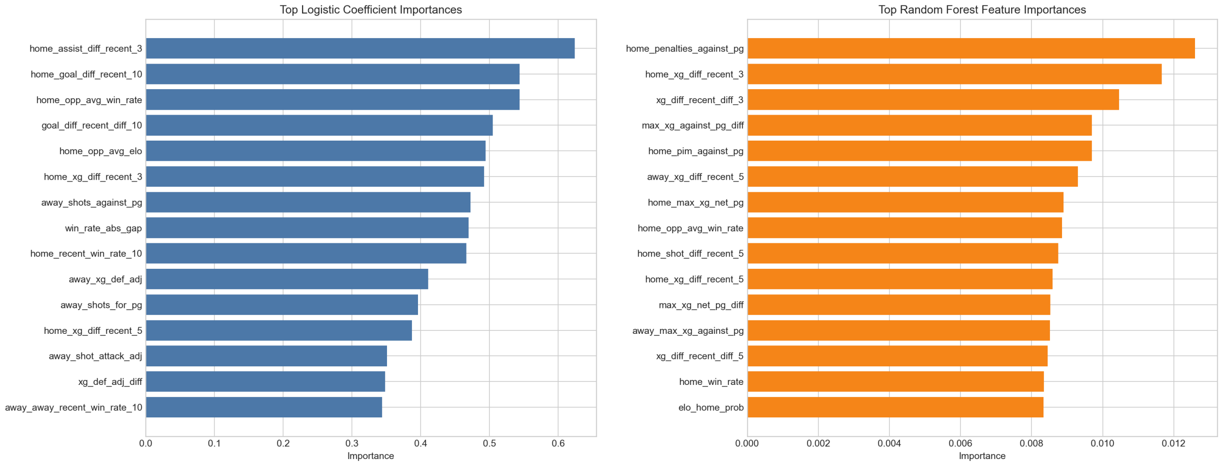


Figure 8: Estimated distribution of test delta log loss (selected minus baseline) from IID and block bootstrap. Values below zero indicate selected-model improvement.

6 Interpretability and Ranking Outputs



(a) Logistic coefficient magnitudes

(b) Random forest feature importances

Figure 9: Feature importance views across model families.

The ranking view is descriptive rather than causal: it summarizes posterior team strength under this model family and season, and should be interpreted jointly with uncertainty diagnostics rather than as a standalone estimator of latent quality.

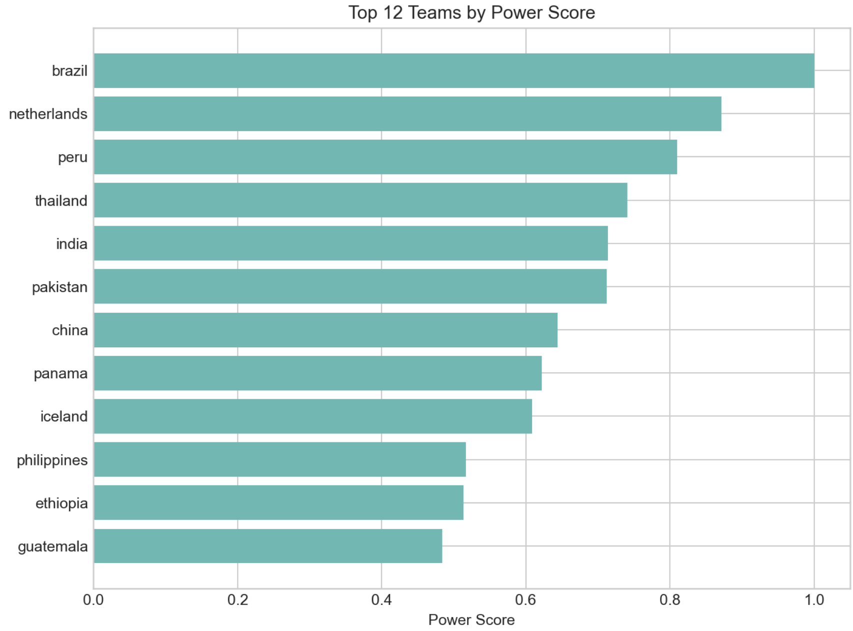


Figure 10: Top teams by composite power score.

Table 8: Top 10 Team Power Rankings

rank	team	power_score	elo_rating	model_strength
1	brazil	1.0000	1563.4111	0.5290
2	netherlands	0.8713	1548.4446	0.5222
3	peru	0.8101	1538.2736	0.5224
4	thailand	0.7406	1535.1534	0.5132
5	india	0.7142	1526.4816	0.5181
6	pakistan	0.7121	1528.2608	0.5157
7	china	0.6440	1523.9226	0.5082
8	panama	0.6223	1511.3110	0.5184
9	iceland	0.6089	1522.7048	0.5031
10	philippines	0.5171	1507.4412	0.5034

7 Round 1 Forecasts

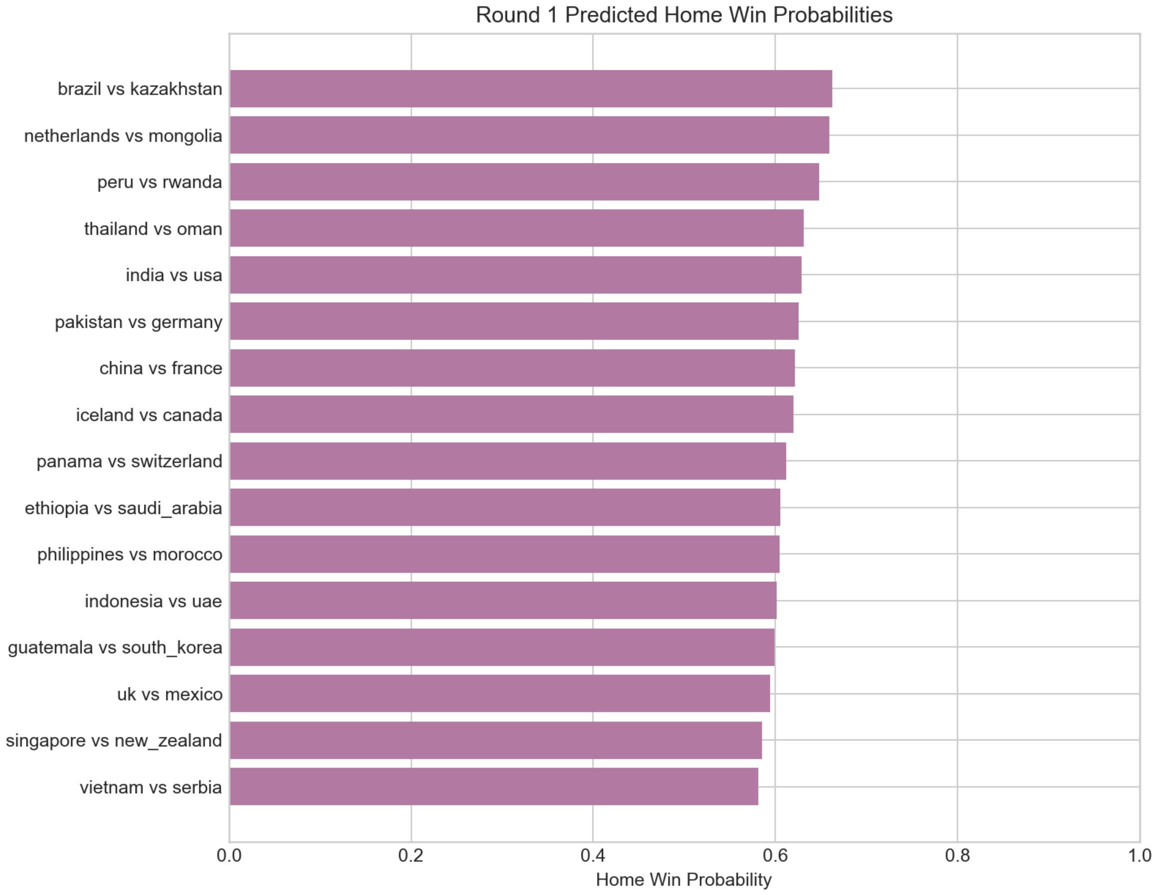


Figure 11: Predicted home-win probabilities for Round 1 matchups.

Table 9: Top Round 1 Home Favorites

game_id	home_team	away_team	home_win_prob
game_1	brazil	kazakhstan	0.6622
game_2	netherlands	mongolia	0.6592
game_3	peru	rwanda	0.6481
game_4	thailand	oman	0.6311
game_6	india	usa	0.6286
game_5	pakistan	germany	0.6252
game_9	china	france	0.6218
game_8	iceland	canada	0.6203

Round 1 probabilities are intentionally moderate (top forecasts around 0.66), reflecting the model's calibrated stance that even stronger teams retain substantial upset risk in this synthetic environment.

8 Reproducibility

Reproducibility is a first-class deliverable in this project. Each run emits:

- `run_metadata.json` for model configuration and selected strategy,
- `run_manifest.json` with command, environment versions, and SHA-256 hashes,
- `visual_manifest.json` with hashes for all figure files.

An automated runner (`run_all.sh --verify-repro`) reruns the full pipeline and asserts hash equality for key artifacts.

9 Limitations

Several limitations remain. First, only one season is available, which limits the stability of broad model families. Second, the competition's synthetic environment may not capture all real hockey non-stationarities. Third, holdout gains are small in absolute terms and should be interpreted with uncertainty. Fourth, model calibration can drift under future distribution shift and should be rechecked if new seasons are introduced. Fifth, we do not run formal pairwise forecast-comparison tests; adding these would further strengthen statistical claims beyond bootstrap interval evidence.

10 Conclusion

We present a leak-safe forecasting pipeline aligned to the actual pregame prediction task. The final system balances interpretability, calibration discipline, and operational reproducibility. Most importantly, this work shows that temporal correctness, proper scoring, and uncertainty-aware reporting are prerequisites for credible sports analytics claims in competition settings. In this dataset, conservative probabilistic discipline outperforms higher-variance complexity, yielding a small but repeatable expected log-loss edge against baseline.

References

[1] Christoph Bergmeir, Rob J. Hyndman, and Bonsoo Koo. A note on the validity of cross-validation for evaluating autoregressive time series prediction. *Computational Statistics & Data Analysis*, 120:70–83, 2018.

[2] Ralph Allan Bradley and Milton E. Terry. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345, 1952.

[3] Leo Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001.

[4] Glenn W. Brier. Verification of forecasts expressed in terms of probability. *Monthly Weather Review*, 78(1):1–3, 1950.

[5] David R. Cox. The regression analysis of binary sequences. *Journal of the Royal Statistical Society: Series B (Methodological)*, 20(2):215–242, 1958.

[6] Bradley Efron. Bootstrap methods: Another look at the jackknife. *The Annals of Statistics*, 7(1):1–26, 1979.

[7] Arpad E. Elo. *The Rating of Chessplayers, Past and Present.* Arco Publishing, 1978.

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

[9] Mark E. Glickman. Parameter estimation in large dynamic paired comparison experiments. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 48(3):377–394, 1999.

[10] Tilmann Gneiting and Adrian E. Raftery. Strictly proper scoring rules, prediction, and estimation. *Journal of the American Statistical Association*, 102(477):359–378, 2007.

[11] Meelis Kull, Telmo M. Silva Filho, and Peter Flach. Beta calibration: A well-founded and easily implemented improvement on logistic calibration for binary classifiers. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, pages 623–631, 2017.

[12] Hans R. Künsch. The jackknife and the bootstrap for general stationary observations. *The Annals of Statistics*, 17(3):1217–1241, 1989.

[13] Alexandru Niculescu-Mizil and Rich Caruana. Predicting good probabilities with supervised learning. In *Proceedings of the 22nd International Conference on Machine Learning*, pages 625–632, 2005.

[14] John Platt. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In *Advances in Large Margin Classifiers*, pages 61–74, 1999.

[15] Bianca Zadrozny and Charles Elkan. Transforming classifier scores into accurate multiclass probability estimates. In *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 694–699, 2002.