Trinity College Dublin
Coláiste na Tríonóide, Baile Átha Cliath
The University of Dublin

LAIDLAW SCHOLARS

LAIDLAW SCHOLARS

Undergraduate Leadership and Research Programme · Research Report

Prediction Markets vs Machine Learning

Evidence from NBA Game Forecasting Across Pre-Game Horizons

Anna Demasure

Supervised by Alessio Benavoli

11 Sept 2026

Abstract

This study compares the probabilistic forecasting performance of prediction markets and machine-learning models using 1,019 NBA games from the 2025/26 season. Polymarket prices are evaluated against Logistic Regression and XGBoost forecasts at eight horizons from five days to one hour before tip-off. Models are trained chronologically using historical team performance, efficiency, injury and lineup information, with all features restricted to information available at each forecast cutoff. Both ML models significantly outperform Polymarket in Brier score five days before tip-off, but this advantage declines as the game approaches and is no longer statistically distinguishable from approximately three days onward. Brier decomposition suggests that Logistic Regression benefits from stronger calibration, XGBoost from stronger resolution, while Polymarket produces sharper forecasts closer to tip-off. The results suggest that forecasting performance depends less on a universally superior method than on which information source is most valuable at each horizon.

Table of Contents

1. Introduction

Is it possible to predict the future? Advances in statistical modelling and artificial intelligence have made forecasting increasingly sophisticated, yet sport remains an environment where uncertainty is difficult to eliminate. Injuries, tactical changes, individual performance and random variation can all alter a game's outcome. Still, the ambition to predict sports has grown from a niche gambling activity into a major global industry. In 2025 alone, bettors legally wagered $166.94 billion through state-regulated US sportsbooks (Surendran Shwetha), a figure reflecting not just appetite for risk but appetite for prediction.

One development reshaping this industry is the rise of prediction markets. Unlike traditional sportsbooks, where a bookmaker sets odds and builds in a margin, prediction markets let participants trade contracts directly with one another. A contract pays out if an event occurs, so its price, ranging between 0 and 1, can be read as the market's estimated probability of that event. Because traders profit from spotting mispriced contracts, they are incentivised to incorporate new information and private beliefs into prices, producing a continuously updating, crowd-generated forecast.

At the same time, machine learning (ML) has become an important tool in sports forecasting. ML models are algorithms trained on large quantities of historical data to identify relationships between characteristics and outcomes and generate predictions. Their ability to consider many variables simultaneously and capture complex, non-linear interactions allows them to extract predictive information that traditional statistical methods may miss. However, they remain restricted to the structured inputs they receive, whereas prediction markets can draw on a broader range of information, including news, judgement, rumours and private beliefs. This raises the question of whether prediction markets provide predictive value beyond what ML models can extract from structured data.

This study investigates that question by comparing market probabilities with independently trained ML models across multiple pre-game horizons, examining whether structured models or crowd-based forecasts perform better as new information becomes available closer to tip-off.

2. Related Works

Research on sports forecasting has largely developed along two main strands. The first examines whether betting and prediction markets efficiently aggregate information. Wolfers and Zitzewitz (2004) argue that prediction markets combine dispersed private beliefs into prices that can be interpreted as probabilistic forecasts. In sport this is further shown by Spann and Skiera (2009) which found that prediction markets and betting odds performed strongly in German football and even outperformed expert tipsters. However, market efficiency is not constant over time. Page and Clemen (2013) show that prediction-market calibration improves as events approach resolution, highlighting the importance of evaluating forecasts across multiple time horizons.

The second strand focuses on machine-learning models. Wang (2025) and Horvat (2023) both report NBA prediction accuracies of approximately 66–78%, demonstrating the strong forecasting potential of ML models. Specific ML feature selection is also important, as Thabtah, Zhang and Abdelhamid (2019) show that it can improve forecasting accuracy by 2–4%. Despite evidence supporting both approaches, few studies directly compare ML-generated probabilities with prediction market prices. This study addresses that gap by comparing their performance in sports forecasting.

3. Data and Methods

3.1 NBA

The NBA was selected as the domain for this study. Its structured gameplay and abundant historical records make it well suited to ML model training, while its global popularity generates the trader volume necessary to analyse market prices meaningfully. With no draws, game outcomes are strictly binary, making them ideal for both classification modelling and direct comparison against yes/no prediction market contracts. Critically, the NBA's fast-moving information environment, where new games are played continuously and injury reports constantly emerge, makes it particularly valuable for studying how forecast accuracy evolves across horizons. The 2025/26 NBA season, containing 1,230 games, serves as the out-of-sample test period, while earlier seasons are used for model development and calibration.

3.2 Market Data

Market data was sourced from Polymarket, selected for its high trader volume and early market opening relative to competitors. Of the 1,230 NBA games played, Polymarket listed 1,071 settled markets. However, not every game had price history available far in advance of tipoff. A fixed cohort was therefore established at the earliest horizon that retained a sufficiently large sample while still allowing meaningful price development to be analyzed. This point was five days before tipoff, yielding 1,019 games (82.8% coverage).

Markets were identified using Polymarket's public Gamma API, while contract prices were retrieved through the CLOB price-history API. For each game, every change in the home-team contract price was recorded with its timestamp. At each minute from five days to one hour before tipoff, the latest price observable at that moment was selected, creating a real-time price path without using future information (see Appendix I). The same fixed cohort was then tracked across all subsequent checkpoints, so performance changes reflected evolving prices rather than shifts in sample composition.

3.3 ML Models

3.3.1 Logistic Regression

Logistic Regression was selected as a baseline model, known for its simplicity and efficiency in capturing smooth relationships without overfitting to noise. For a game with input feature vector X, it estimates the probability of a home win using the sigmoid function:

$$P(y = 1 \mid X) = \frac{1}{1 + e^{-(\beta_0 + \beta^\top X)}}$$

Here β represents the estimated coefficients learned from historical data. The output is a probability between zero and one, making it well-suited for comparison against Polymarket's market-implied prices.

3.3.2 XGBoost

XGBoost was selected as a more flexible classification model capable of capturing nonlinear relationships and interactions between features. It builds an ensemble of decision trees sequentially, with each tree improving on the errors of the previous trees. For binary classification, the model produces a raw score from the sum of these trees, which is then transformed into a predicted home-win probability using a logistic function:

$$P(y_i = 1 \mid X_i) = \sigma\left(\sum_{k=1}^{K} f_k\,(X_i)\right)$$

where K represents the number of trees, f_k represents each tree and σ the logistic function.

Full specifications and hyperparameters of both models are reported in Appendix IX.

3.4 ML Features and Data Construction

The modelling dataset spans the 2018/19 to 2025/26 seasons and contains four feature groups, covering team form and schedule, basketball efficiency, injury and availability, and lineup or rotation proxies. Sources include nba_api, NBA Stats, Basketball Reference and official NBA injury reports (Appendix II). Matchup variables were constructed as team differentials and oriented, where practical, so positive values represented an advantage to the home team. When current-season history did not yet exist, prior season values were used temporarily until same-season observations became available. Appendix III lists all ML features and their descriptions.

Injury reports required additional processing because official statuses are categorical rather than numerical. Historical reports from 2018/19 to 2024/25 were therefore matched to subsequent box scores to estimate the empirical probability that a rotation player listed as Out, Doubtful, Questionable, Probable or Available actually played (shown in Appendix IV). Restricting the calculation to genuine rotation players reduced the risk of treating coach's-decision DNPs as injury absences. Expected lost minutes were then weighted by the player's most recently available pre-game Box Plus/Minus (BPM) score which estimates a player's overall contribution and value

to their team. This helped produce injury-value features that combine availability risk, normal playing time and estimated player impact. The full calculations are shown in Appendix V.

3.5 Model training and leakage control

Forecasts were generated at five days, three days, two days, one day, twelve hours, six hours, three hours and one hour before tip-off. The dataset was structured as a game-horizon panel, meaning that each game appeared once at every cutoff (example shown in Appendix VI). Only information available at or before the relevant cutoff was used. Team-form, efficiency and rotation features could change when either team completed another game (see example in Appendix) while injury features changed as new official injury reports were released (see example in Appendix). The target game's own lineup and minutes, released shortly before play, were never used.

The models were not retrained separately at each horizon. Instead, one Logistic Regression model and one XGBoost model were fitted to pooled historical horizon observations and then supplied with the feature snapshot available at each cutoff. Changes in a game's forecast therefore reflect new information entering the feature set rather than changing model parameters. Polymarket probabilities were excluded from all ML inputs and used only as an external benchmark.

The data were split chronologically to prevent leakage. Seasons 2018/19 to 2023/24 were used for training, 2024/25 was reserved only for probability recalibration using Platt scaling, and the 1,019 matched 2025/26 games were used for final testing. Model specifications were fixed before evaluating the test season, and the calibration season was not used for feature selection or hyperparameter tuning. This preserves 2025/26 as a genuinely out-of-sample evaluation.

3.6 Forecast Evaluation

Forecast performance was evaluated using classification accuracy and the Brier score. Accuracy measures the proportion of games where the predicted winner was correct. It is intuitive, but limited because it treats all correct predictions equally, regardless of confidence. The Brier score addresses this by evaluating probabilistic accuracy:

$$\text{Brier Score} = \frac{1}{N}\sum_{i=1}^{N}(p_i - o_i)^2$$

where N is the total number of predictions, p_i is the predicted probability of a home win and o_i is the observed outcome. Lower scores indicate better probabilistic accuracy, with $o_i \in \{0, 1\}$. A score of 0 represents a perfect forecast and 0.25 an uninformative 50% forecast.

Because the same games were forecast by each method, differences were evaluated using paired game-level bootstrap 95% confidence intervals with 10,000 resamples. Games, rather than individual horizon observations, were resampled to preserve the paired comparison between Polymarket and the ML models. Values above zero indicate that the model produced a lower Brier score than Polymarket for the same games.

Following Murphy (1973), Brier scores were decomposed into reliability, resolution and uncertainty. Reliability measures calibration error, resolution measures separation between likely winners and uncertain games, and uncertainty reflects the base difficulty of the sample. Sharpness was also reported separately as the average distance from 50%.

4. Results

4.1 Market vs ML accuracy and Brier score

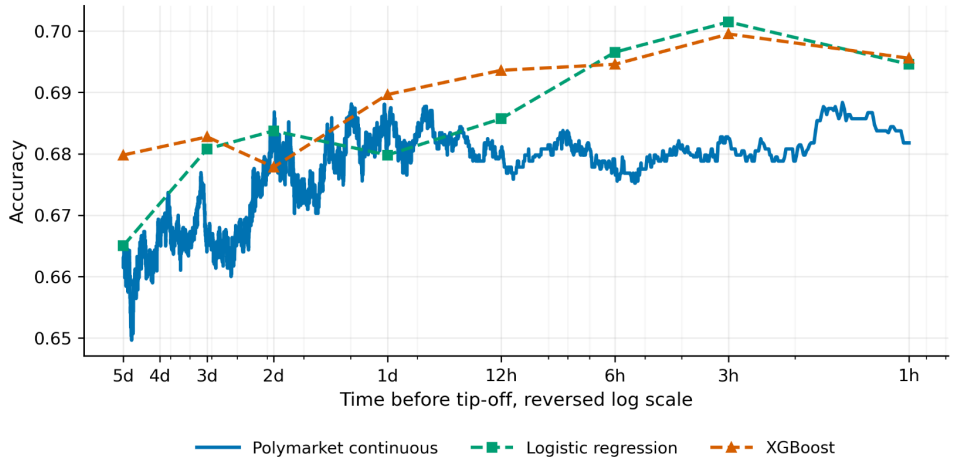

Figure 1. Forecast accuracy by pre-game horizon for Polymarket, Logistic Regression and XGBoost.

The graph shows that prediction accuracy generally improves as game time approaches, but the three methods improve in different ways. Polymarket accuracy increases noticeably from roughly 0.66, five days before tip-off, toward 0.68 as new information is incorporated. After about two days, however, its accuracy largely plateaus. In contrast, both ML models continue improving as game time approaches. Logistic regression shows the strongest late improvement, catching up to XGBoost around six hours before tip-off and briefly becoming the highest point estimate, peaking above 0.70 near three hours. XGBoost is smoother and consistently strong.

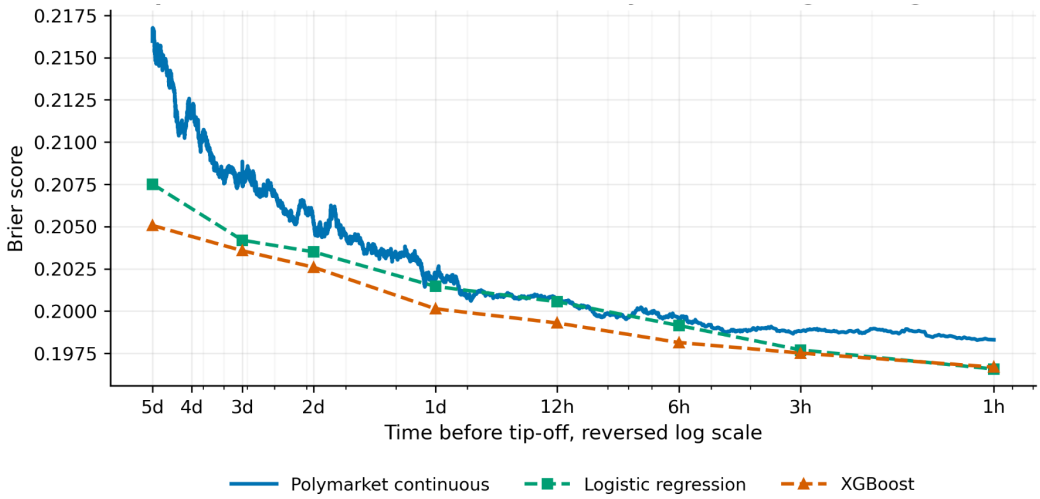

Figure 2. Brier score by pre-game horizon for Polymarket, Logistic Regression and XGBoost.

The Brier score is more informative because it evaluates the probability itself rather than only the predicted winner. All three forecasters improve toward tip-off. Polymarket falls the most from roughly 0.216 to 0.202 by one day out, while both ML models begin lower and continue improving. XGBoost has the lowest point estimate through much of the period, with Logistic Regression catching it near tip-off. All three remain below the 0.25 score of a constant 0.5 forecast, though none approaches 0, reflecting the irreducible randomness of a single NBA game.

4.2 Statistical Testing

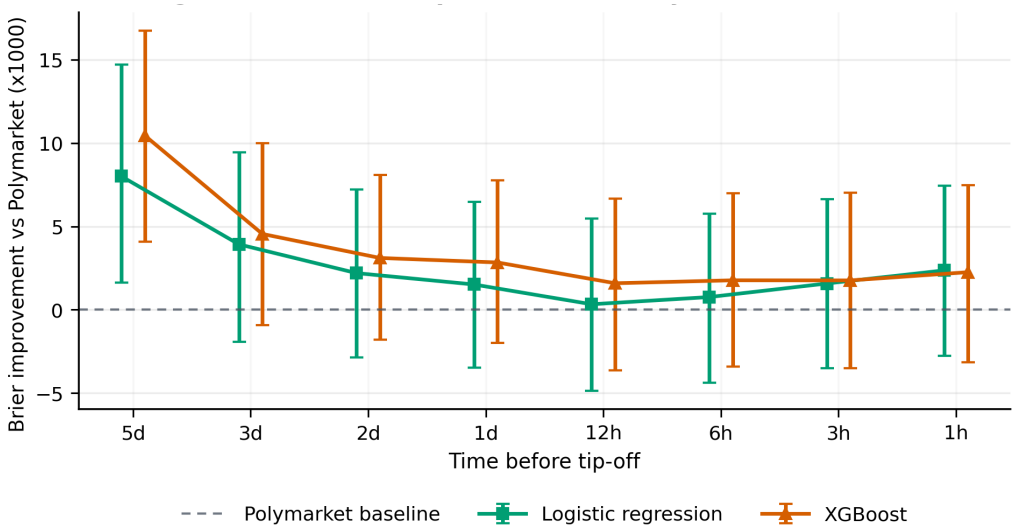

Figure 3. Paired game-level Brier-score improvement of the ML models relative to Polymarket, with 95% bootstrap confidence intervals.

At five days, the confidence intervals for both ML models lie above zero, indicating a statistically significant Brier-score advantage over Polymarket. This advantage, however, declines as the game approaches, and from around three days onward the confidence intervals overlap zero, meaning neither model is clearly superior to Polymarket. The small rebound in improvement after 12 hours suggests that late structured information, especially injury and availability data, helps the ML models refine their probabilities. However, because the confidence intervals remain wide, this late improvement should be interpreted as suggestive rather than statistically decisive.

4.3 Brier Decomposition

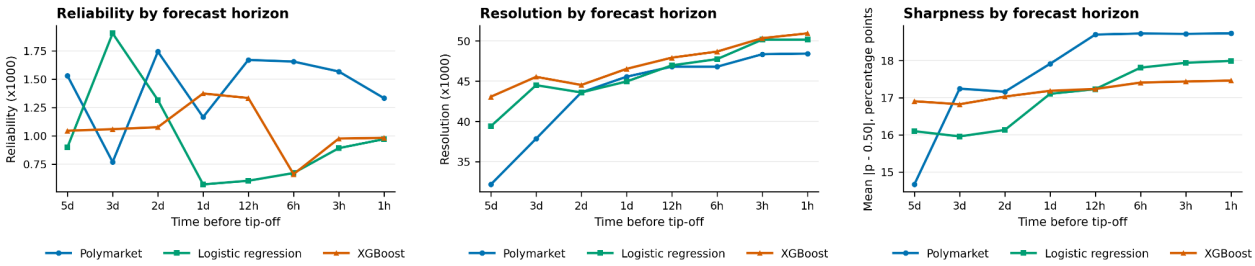

Figure 4. Brier-score decomposition and sharpness across forecast horizons.

These graphs help explain why the ML models achieve lower Brier scores than Polymarket. Logistic regression generally has lower reliability error close to tip-off, indicating better calibration, while XGBoost consistently achieves strong resolution, meaning it is better at identifying which games have a strong likely winner and which are genuinely difficult to predict. Polymarket, by contrast, becomes the sharpest forecaster, producing more extreme probabilities, particularly from 12 hours onward. However, sharpness is only valuable when those confident probabilities are well calibrated. Polymarket's relatively high and volatile reliability error suggests that some of this confidence is misplaced. Thus, the ML models' Brier-score advantage appears to come not from making more aggressive forecasts, but from balancing discrimination with calibration more effectively, whereas prediction markets provide sharper but somewhat noisier probability estimates.

4.4 Exploratory Performance Across Game Types

To examine whether markets and models respond differently across different game situations, forecasts were compared for three pre-defined groups: star-player injury shocks, clear favourites that won, and clear favourites that unexpectedly lost. Clear favourites were games where the reference team had at least an 80% Polymarket win probability five days before tip-off, producing 141 favourite wins and 30 favourite losses. Star-player injury shocks were games where the opponent suffered a late increase in unavailable player value from a high-value player listed Out or Doubtful, producing 36 games.

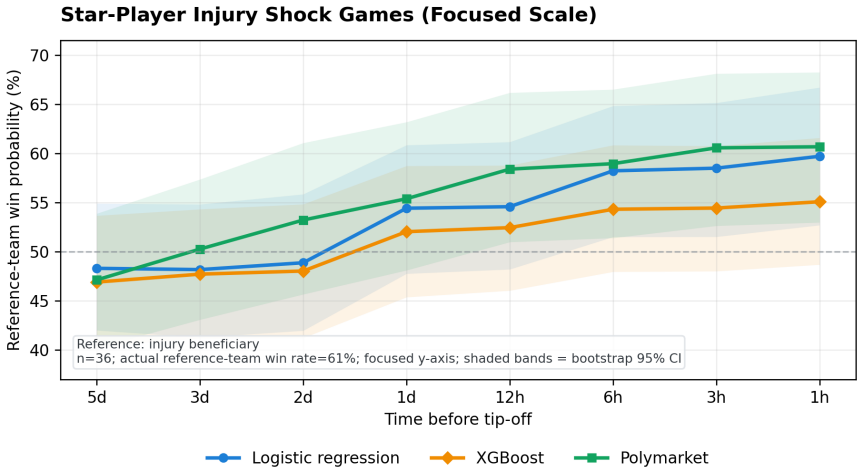

Figure 5. Forecast probability for the eventual injury-beneficiary team in star-player injury-shock games (n = 36).

In star-player injury shock games, all three methods initially place the eventual injury beneficiary below 50%, before increasing its win probability as tip-off approaches. Polymarket reacts earliest and most consistently, crossing 50% around three days out and reaching approximately 61% by one hour, suggesting rapid incorporation of injury-related information. Logistic Regression responds similarly, rising from 48% to 60%, indicating that updated injury features capture much of this change. XGBoost adjusts more cautiously, ending near 55%. However, the wide, overlapping 95% confidence intervals and small sample of 36 games mean Polymarket's apparent advantage should be interpreted cautiously

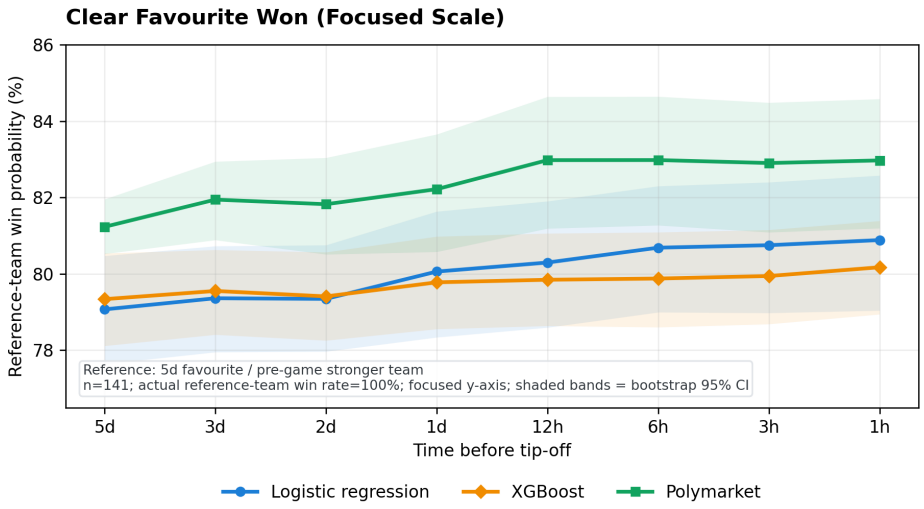

Figure 6. Forecast probability in games where the clear favourite wins.

For games where the clear favourite won, Polymarket consistently assigns higher win probabilities than both ML models, suggesting stronger conviction in obvious team-strength advantages. Logistic regression gradually increases its confidence as tip-off approaches, while XGBoost remains more conservative and relatively flat. Although all three sit below the sample's 100% realized win rate.

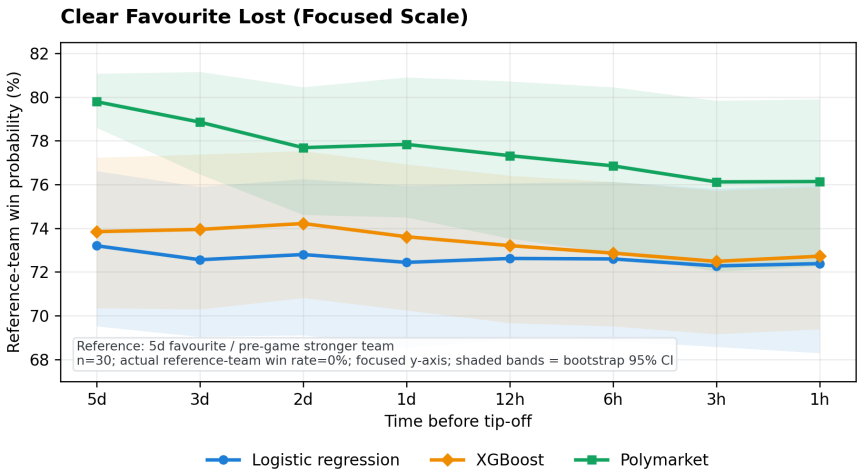

Figure 7. Forecast probability in games where the clear favourite unexpectedly loses

This graph shows games where the clear pre-game favourite ultimately lost to the underdog. In these cases, all three methods continued to favour the expected winner close to tip-off, despite the underdog eventually winning. Polymarket's confidence becomes a weakness here, since being further from 50% leads to a bigger Brier penalty when the upset happens. These games are rare though, so the occasional large error is outweighed by sharper predictions in the far more common favourite win cases. Polymarket also softens its position the most as tip off nears, suggesting traders adjust to late signals more flexibly than the stable ML models. Even so, none of the methods can fully predict these upsets, which reflects the natural uncertainty of sport.

4.5 Predictive contribution of different data sources

The game-type results allude to the fact that different games may need different types of information. Games with a clear favourite can be predicted well early, using mainly team strength, while close or disrupted games may rely more on recent injury and player availability updates. To identify which data matters most, both models were retrained using NBA API data, efficiency metrics, injury information, lineup proxies, and the full dataset. Polymarket remains an external benchmark.

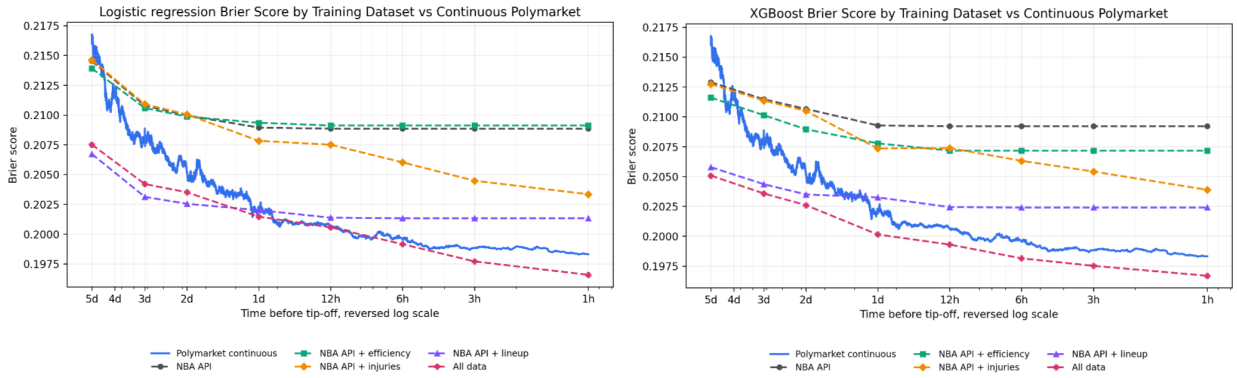

Figure 8. Brier score by feature-group availability for Logistic Regression and XGBoost, with Polymarket as an external benchmark.

These graphs show that richer training data substantially improves Brier scores for both Logistic Regression and XGBoost. NBA API data alone performs the weakest, while adding efficiency measures produces only modest gains. Injury information improves forecasts further but it is lineup proxies that deliver a much larger individual reduction in Brier score. The strongest performance comes from the full dataset. Accuracy shows the same pattern (Appendix X), where using all available data increases accuracy by 2% for both models compared to using structural features alone.

5. Discussion

The central finding is that neither machine learning nor prediction markets are consistently superior. Their relative performance depends on when the forecast is made, what information is available, and how that information is translated into a probability.

Five days before tip off, both ML models achieve a statistically significant Brier score advantage over Polymarket. This can be expected because structured inputs such as team strength, rest and rotation history are already available well in advance, while prediction markets may still have lower liquidity and fewer informed traders actively pricing the game. As tip off approaches, this early ML advantage narrows. From around three days onward, the confidence intervals around the Brier score differences increasingly overlap zero, meaning neither approach can be identified as reliably superior. This suggests that both the ML models and Polymarket gradually converge on the same broad information, and the natural uncertainty of sport imposes a limit neither can forecast past.

Interestingly, the Brier decomposition shows that similar overall performance can come from different forecasting behaviour. Polymarket is the sharpest forecaster, assigning more extreme probabilities, particularly from twelve hours onward. This is rewarded when clear favourites win but heavily penalised when favourites unexpectedly lose, a confidence for accuracy trade off rather than a straightforward edge. Logistic Regression performs strongly on calibration, since its linear structure allows the whole probability scale to be adjusted smoothly during recalibration. XGBoost instead performs strongly on resolution, since tree splits can separate clear favourites, close matchups and injury disrupted contests more distinctly than a linear model can. Neither ML model's advantage comes from one algorithmic strength alone. It comes from balancing discrimination against calibration more evenly than a market that is more willing to make bold probability claims.

The injury shock results point to a specific area where markets may hold a genuine edge. Across the thirty six star player injury games, Polymarket reacts earlier and more steadily toward the team benefiting from the opponent's injury, crossing fifty percent nearly two days before the ML models catch up. This suggests markets may have an advantage when information first appears informally, through rumours or journalist reports, before it becomes an official injury report feature. Though, this should be interpreted cautiously, given the small sample and wide confidence intervals.

The data ablation results reinforce the importance of timely information, showing that predictions improved as richer data became available. Lineup and rotation proxies emerged as among the most valuable features, contrasting with Thabtah, Zhang and Abdelhamid (2019), who identified efficiency measures as most important. This difference may reflect feature design. Efficiency measures capture how well a team has performed, whereas our lineup features capture whether the players responsible for that performance are expected to form the current rotation. Because NBA performance is highly player-dependent, expected rotation strength may therefore better represent a team's true game-day strength, helping explain its greater predictive value in our study.

Overall, the main forecasting challenge is not simply choosing a more sophisticated algorithm, but identifying which information source matters most at each horizon. This also helps explain why XGBoost, with extra flexibility, does not reliably outperform Logistic Regression. Alves and Barbosa found a similar result, with Logistic Regression even marginally outperforming XGBoost (66.90% vs 65.42%). It is clear that structured ML models provide stable forecasts well before tip-off, while prediction markets are possibly more responsive as new information begins to arrive. A natural extension is therefore a hybrid model that combines market probabilities with ML features. Sports outcomes will always carry genuine uncertainty, since even with excellent information an underdog can still win. The goal of forecasting is therefore not to remove that uncertainty, but to quantify it as accurately as possible.

6. References

Alves, J. M., & Barbosa, R. S. (2025). Machine learning for basketball game outcomes: NBA and WNBA leagues. Computation, 13(10), 230–230. https://doi.org/10.3390/computation13100230

Horvat, T., Job, J., & Logozar, R. (2023). A Data-Driven Machine Learning Algorithm for Predicting the Outcomes of NBA Games. MDPI. https://doi.org/10.3390/sym15040798

Murphy, A. H. (1973). A new vector partition of the probability score. *Journal of Applied Meteorology and Climatology*, 12(4), 595–600

Myers, D. (2020, February). About box plus/minus (BPM). Basketball Reference. https://www.basketball-reference.com/about/bpm2.html

Page, L. and Clemen, R.T. (2013). Do prediction markets produce well-calibrated probability forecasts? The Economic Journal, 123(568), pp.491–513. doi:10.1111/j.1468-0297.2012.02561.x.

Shwetha Surendran (2026). Sports betting hits record $16.96 billion in revenue in 2025. [online] ESPN.com. ESPN. Available at: https://www.espn.com/espn/betting/story/_/id/48045855/sports-betting-hits-record-1696-billion-revenue-2025

Spann, M. and Skiera, B. (2009). Sports Forecasting: A Comparison of the Forecast Accuracy of Prediction Markets, Betting Odds and Tipsters. Journal of Forecasting, 28. doi:10.1002/for.1091.

Thabtah, F., Zhang, L. and Abdelhamid, N. (2019). NBA Game Result Prediction Using Feature Analysis and Machine Learning. Annals of Data Science, 6. doi:10.1007/s40745-018-00189-x.

Wang, Y. (2025). Comparative evaluation of machine learning models for NBA game outcome prediction. Journal of Computer and Communications, 13(11), pp.41–62. doi:10.4236/jcc.2025.1311004.

Wolfers, J. and Zitzewitz, E. (2004). Prediction markets. Journal of Economic Perspectives, 18(2), pp.107–126. doi:10.1257/0895330041371321.

7. Appendices

Appendix I: Polymarket Price Extraction

Table I.1. Polymarket home-price extraction at forecast checkpoints for CHI at ORL, 2025-10-25 23:00.

Check point	Checkpoint time UTC	Home price used	Source price-change time UTC	Minutes since last update
5 day	2025-10-20 23:00	0.500	2025-10-20 23:00	0
3 day	2025-10-22 23:00	0.565	2025-10-22 17:06	354
2 day	2025-10-23 23:00	0.605	2025-10-23 20:05	175
1 day	2025-10-24 23:00	0.650	2025-10-24 22:20	40
12 hour	2025-10-25 11:00	0.685	2025-10-25 10:39	21
6 hour	2025-10-25 17:00	0.675	2025-10-25 15:39	81
3 hour	2025-10-25 20:00	0.675	2025-10-25 15:39	261
1 hour	2025-10-25 22:00	0.685	2025-10-25 21:17	43

Appendix II: Raw data sources and Collection

Table II.1. Raw data sources and collection methods by feature group.

Feature group	Raw data collected	Source	Collection method
Team form and schedule	Game ID, season, game date/time, home/away teams, final scores, wins/losses, team game logs, rest gaps, recent games played	nba_api, NBA _ Stats	API calls using ScheduleLeagueV2, TeamGameLogs, LeagueGameLog
Team efficiency	Offensive rating, defensive rating, net rating, pace, EFG%, TS%, FG%, 3P%, FT rate, turnover rate, rebound rates, assist ratio, steal/block/foul rates, points	nba_api team _ advanced/base logs	API calls to NBA Stats team game logs, using only completed games before each cutoff
Injuries and availability	Official injury-report PDFs, report timestamp, game ID, team, player name, injury status, previous status, injury category/reason	Official NBA injury reports from ak-static.cms.nba.com	PDF URL generation/download, then PDF text parsing into player-status rows
Player value	Player BPM, minutes, games, team, monthly/season player-value snapshots	Basketball Reference	Web scraping with requests and pandas.read_html. _
Lineup / rotation proxies	Box-score player rows, player minutes, starter indicator, team, game ID, safe availability timestamp	NBA.com/NBA Stats boxscore pages	Scraped/downloaded boxscore HTML files.

Appendix III: Machine-Learning Features

Table III.1. Machine-learning features and definitions.

Group	Feature	Description
NBA API / form and schedule	home_advantage	Constant home-team indicator (predictions are expressed as home win probability)
NBA API / form and schedule	rest_days_diff	Home rest days minus away rest days (capped at 10 days)
NBA API / form and schedule	back_to_back_diff	Away back-to-back flag minus home back-to-back flag.
NBA API / form and schedule	games_last_7_diff	Home games played in the previous seven days minus away.
NBA API / form and schedule	games_last_14_diff	Home games played in the previous fourteen days minus away.
NBA API / form and schedule	current_or_prior_win_pct_l5_diff	Home recent last-5 win percentage minus away. Falls back to prior-season value only before same-season data exists.
NBA API / form and schedule	current_or_prior_win_pct_l10_diff	Home recent last-10 win percentage minus away, with same current-or-prior rule.
NBA API / form and schedule	current_or_prior_season_win_pct_diff	Home season-to-date win percentage minus away, with prior-season fallback for early games.
NBA API / form and schedule	current_or_prior_avg_margin_l10_diff	Home average scoring margin in last ten games minus away, with prior-season fallback when needed.
Team efficiency	current_or_prior_net_rating_l5_diff	Home last-5 net rating minus away.
Team efficiency	current_or_prior_net_rating_l10_diff	Home last-10 net rating minus away.
Team efficiency	current_or_prior_season_net_rating_diff	Home season-to-date net rating minus away.
Team efficiency	current_or_prior_off_rating_l10_diff	Home last-10 offensive rating minus away.
Team efficiency	current_or_prior_def_rating_l10_diff	Away last-10 defensive rating minus home, so positive values favour the home team.
Team efficiency	current_or_prior_pace_l10_diff	Home last-10 pace minus away.

Group	Feature	Description
Team efficiency	current_or_prior_efg_l10_diff	Home last-10 effective field-goal percentage minus away.
Team efficiency	current_or_prior_ts_l10_diff	Home last-10 true-shooting percentage minus away.
Team efficiency	current_or_prior_fg_pct_l10_diff	Home last-10 field-goal percentage minus away.
Team efficiency	current_or_prior_three_pct_l10_diff	Home last-10 three-point percentage minus away.
Team efficiency	current_or_prior_ft_rate_l10_diff	Home last-10 free-throw rate minus away.
Team efficiency	current_or_prior_turnover_rate_l10_diff	Away turnover rate minus home, so positive values favour the home team.
Team efficiency	current_or_prior_oreb_pct_l10_diff	Home last-10 offensive rebounding percentage minus away.
Team efficiency	current_or_prior_dreb_pct_l10_diff	Home last-10 defensive rebounding percentage minus away.
Team efficiency	current_or_prior_assist_ratio_l10_diff	Home last-10 assist ratio minus away.
Team efficiency	current_or_prior_steal_rate_l10_diff	Home last-10 steal rate minus away.
Team efficiency	current_or_prior_block_rate_l10_diff	Home last-10 block rate minus away.
Team efficiency	current_or_prior_foul_rate_l10_diff	Away foul rate minus home, so positive values favour the home team.
Team efficiency	current_or_prior_points_l10_diff	Home last-10 points scored minus away.
Injury and availability	unavailable_starter_count_diff	Away unavailable starters minus home; positive values favour the home team.
Injury and availability	unavailable_rotation_count_diff	Away unavailable top-8 rotation players minus home (positive values favour the home team).
Injury and availability	injury_value_lost_diff	Away team value lost minus home team value lost (positive values favour the home team).
Injury and availability	star_out_diff	Away maximum individual injury value lost minus home maximum individual loss.
Injury and availability	questionable_value_diff	Away uncertain-status value lost minus home uncertain-status value lost.

Group	Feature	Description
Lineup / rotation proxy	previous_starters_returning_diff	Home previous starters not Out/Doubtful minus away.
Lineup / rotation proxy	top5_minutes_share_diff	Home top-5 recent-minute share minus away.
Lineup / rotation proxy	top8_minutes_share_diff	Home top-8 recent-minute share minus away.
Lineup / rotation proxy	starter_minutes_l10_diff	Home previous starters' last-10 minutes minus away.
Lineup / rotation proxy	top5_bpm_value_diff	Home minutes-weighted BPM value for top-5 recent-minute players minus away.
Lineup / rotation proxy	top8_bpm_value_diff	Home minutes-weighted BPM value for top-8 recent-minute players minus away.
Lineup / rotation proxy	expected_rotation_strength_diff	Home expected BPM-minutes rotation strength minus away.

Appendix IV: Injury Status Estimation Weights

Official injury statuses were converted into empirical participation weights using historical reports from 2018-19 to 2024-25. Each game-player observation was matched to the corresponding box score. Participation was defined as:

$$\text{Played} = 1 \text{ if minutes} > 0$$

$$\text{Played} = 0 \text{ if minutes} = 0$$

To avoid treating fringe DNPs as injury absences, the sample was restricted to rotation players with at least one pre-game rotation signal: recent or season-to-date 15+ minute usage, starter/top-eight minutes history, or a prior-season 15+ minute role.

For each status s, the participation weight is:

$$w_{\text{status}} = P(\text{Played} = 1 \mid \text{Status} = s) = \frac{\text{number of players with that status who played}}{\text{total observations with that status}}$$

For example:

$$w_{\text{Out}} = \frac{147}{22174} = 0.0066$$

Table IV.1. Empirical participation weights by official injury status.

Status	Observations	Played >0 min	Participation weight
Out	22,174	147	0.0066
Doubtful	840	50	0.0595
Questionable	5,791	2,928	0.5056
Probable	2,315	1,997	0.8626
Available	1,489	1,291	0.8670

Appendix V: Injury-Value Feature Calculation

The status weights estimate how likely a listed player is to participate, but they do not measure how damaging that absence is to the team. The injury features therefore combine three pieces of information: the player's normal playing time, the probability that those minutes will be unavailable, and the value of the player occupying those minutes.

Normal playing time is estimated from the team's previous ten games before the forecast cutoff, rather than from the player's previous ten appearances. This keeps the window tied to the team's current rotation.

For a player i:

$$\text{Expected Minutes}_i = w_{\text{status},i} \times \text{Average Minutes}_{i,\text{last ten team games}}$$

The corresponding expected minutes lost are:

$$\text{Lost Minutes}_i = \text{Average Minutes}_{i,\text{last 10 team games}} - \text{Expected Minutes}_i$$

Each player's lost minutes were weighted by his most recently available Basketball Reference Box Plus/Minus (BPM) (measures a player's estimated contribution per 100 possessions relative to a league-average player) before the game.

The model uses adjusted value $\text{BPM}_i + 2.0$, where +2.0 treats roughly -2 BPM as replacement level, so losing a replacement-level player creates little or no value loss while losing a high-BPM player creates a larger penalty (Myers, 2020).

$$\text{ValueLost}_i = \frac{(\text{BPM}_i + 2.0) \times \text{LostMinutes}_i}{240}$$

The denominator of 240 represents the total number of regulation player-minutes available to one NBA team (240 = 5 players × 48 regulation minutes). Dividing by 240 converts the player's expected lost minutes into his share of total team playing time.

Team-level injury loss is then calculated by summing across all players listed on that team's most recent available injury report:

$$\text{TeamValueLost} = \frac{\sum_i (\text{BPM}_i + 2.0) \times \text{LostMinutes}_i}{240}$$

The sum is taken over every relevant player on the team's latest injury report.

$$\text{InjuryValueLostDiff} = \text{ValueLost}_{\text{away}} - \text{ValueLost}_{\text{home}}$$

Appendix VI: Feature Values Across Forecast Horizons

Table VI.1. ML feature values by forecast horizon for MIA at CHI, 2026-01-29.

Group	ML feature	5d	3d	2d	1d	12h	6h	3h	1h	Changed
Team form and schedule	home_advantage	1	1	1	1	1	1	1	1	No change
Team form and schedule	rest_days_diff	0.0208	0.0208	0.0208	0.0208	0.0208	0.0208	0.0208	0.0208	No change
Team form and schedule	back_to_back_diff	0	0	0	0	0	0	0	0	No change
Team form and schedule	games_last_7_diff	0	0	0	0	0	0	0	0	No change
Team form and schedule	games_last_14_diff	0	0	0	0	0	0	0	0	No change
Team form and schedule	current_or_prior_win_pct_l5_diff	0.4000	0.2000	0.2000	0.2000	0	0	0	0	3d, 12h
Team form and schedule	current_or_prior_win_pct_l10_diff	0.1000	0.1000	0.1000	0.1000	0.1000	0.1000	0.1000	0.1000	No change
Team form and schedule	current_or_prior_season_win_pct_diff	-0.0111	-0.0208	-0.0319	-0.0319	-0.0315	-0.0315	-0.0315	-0.0315	3d, 2d, 12h
Team form and schedule	current_or_prior_avg_margin_l10_diff	9.3000	6	6.3000	6.3000	6	6	6	6	3d, 2d, 12h
Team efficiency	current_or_prior_net_rating_l5_diff	17.2000	8.9600	7.6200	7.6200	-0.6200	-0.6200	-0.6200	-0.6200	3d, 2d, 12h
Team efficiency	current_or_prior_net_rating_l10_diff	8.4700	5.7600	6.0300	6.0300	6	6	6	6	3d, 2d, 12h
Team efficiency	current_or_prior_season_net_rating_diff	-2.6798	-3.2986	-3.4884	-3.4884	-3.3031	-3.3031	-3.3031	-3.3031	3d, 2d, 12h
Team efficiency	current_or_prior_off_rating_l10_diff	7.3400	5.6500	6.5900	6.5900	4.8800	4.8800	4.8800	4.8800	3d, 2d, 12h
Team efficiency	current_or_prior_def_rating_l10_diff	1.1400	0.1100	-0.5500	-0.5500	1.1200	1.1200	1.1200	1.1200	3d, 2d, 12h
Team efficiency	current_or_prior_pace_l10_diff	-6.7000	-6.7500	-6.1000	-6.1000	-4.9500	-4.9500	-4.9500	-4.9500	3d, 2d, 12h
Team efficiency	current_or_prior_efg_l10_diff	0.0491	0.0590	0.0689	0.0689	0.0562	0.0562	0.0562	0.0562	3d, 2d, 12h
Team efficiency	current_or_prior_ts_l10_diff	0.0426	0.0454	0.0551	0.0551	0.0454	0.0454	0.0454	0.0454	3d, 2d, 12h
Team efficiency	current_or_prior_fg_pct_l10_diff	0.0362	0.0388	0.0470	0.0470	0.0413	0.0413	0.0413	0.0413	3d, 2d, 12h
Team efficiency	current_or_prior_three_pct_l10_diff	0.0275	0.0546	0.0565	0.0565	0.0275	0.0275	0.0275	0.0275	3d, 2d, 12h

Group	ML feature	5d	3d	2d	1d	12h	6h	3h	1h	Changed
Team efficiency	current_or_prior_ft_rate_l10_diff	-0.0487	-0.0913	-0.0809	-0.0809	-0.0655	-0.0655	-0.0655	-0.0655	3d, 2d, 12h
Team efficiency	current_or_prior_turnover_rate_l10_diff	0.0178	0.0060	0.0049	0.0049	0.0123	0.0123	0.0123	0.0123	3d, 2d, 12h
Team efficiency	current_or_prior_oreb_pct_l10_diff	-0.0224	-0.0398	-0.0435	-0.0435	-0.0585	-0.0585	-0.0585	-0.0585	3d, 2d, 12h
Team efficiency	current_or_prior_dreb_pct_l10_diff	0.0189	0.0173	0.0456	0.0456	0.0601	0.0601	0.0601	0.0601	3d, 2d, 12h
Team efficiency	current_or_prior_assist_ratio_l10_diff	2.9800	3.3300	3.6200	3.6200	2.9600	2.9600	2.9600	2.9600	3d, 2d, 12h
Team efficiency	current_or_prior_steal_rate_l10_diff	-0.3644	-0.3065	-0.0525	-0.0525	-0.3403	-0.3403	-0.3403	-0.3403	3d, 2d, 12h
Team efficiency	current_or_prior_block_rate_l10_diff	2.5622	2.8782	2.4295	2.4295	1.8730	1.8730	1.8730	1.8730	3d, 2d, 12h
Team efficiency	current_or_prior_foul_rate_l10_diff	0.1578	1.0002	0.4111	0.4111	-0.0847	-0.0847	-0.0847	-0.0847	3d, 2d, 12h
Team efficiency	current_or_prior_points_l10_diff	0.5000	-1.9000	-0.2000	-0.2000	-1	-1	-1	-1	3d, 2d, 12h
Injuries and availability	unavailable_starter_count_diff	0	0	0	-1	-1	0	0	-1	1d, 6h, 1h
Injuries and availability	unavailable_rotation_count_diff	0	0	0	-2	-2	-2	-2	-2	1d
Injuries and availability	injury_value_lost_diff	0	0	0	-0.7093	-0.6732	-0.5281	0.3429	0.0368	1d, 12h, 6h, 3h, 1h
Injuries and availability	star_out_diff	0	0	0	-0.5456	-0.5107	-0.3165	0.2756	0.0813	1d, 12h, 6h, 3h, 1h
Injuries and availability	questionable_value_diff	0	0	0	0	0	-0.3399	-0.1845	0	6h, 3h, 1h
Lineup / rotation proxy	previous_starters_returning_diff				-1	-1	0	0	-1	1d, 6h, 1h
Lineup / rotation proxy	top5_minutes_share_diff	0.0490	0.0062	0.0059	0.0059	0.0047	0.0047	0.0047	0.0047	3d, 2d, 12h
Lineup / rotation proxy	top8_minutes_share_diff	0.1114	0.0941	0.0967	0.0967	0.0717	0.0717	0.0717	0.0717	3d, 2d, 12h
Lineup / rotation proxy	starter_minutes_l10_diff	8.6179	6.5068	7.0596	7.0596	6.5881	6.5881	6.5881	6.5881	3d, 2d, 12h
Lineup / rotation proxy	top5_bpm_value_diff	-1.4938	-1.6449	-1.8452	-1.8452	-2.1528	-2.1528	-2.1528	-2.1528	3d, 2d, 12h
Lineup / rotation proxy	top8_bpm_value_diff	-1.5138	-1.3743	-1.3620	-1.3620	-1.3490	-1.3490	-1.3490	-1.3490	3d, 2d, 12h
Lineup / rotation proxy	expected_rotation_strength_diff				-1.1872	-1.1803	-1.3137	-1.2949	-1.2963	1d, 12h, 6h, 3h, 1h

Contextual information

> For pure injury-loss features, no report before the cutoff is treated as no known injury penalty yet, so the value is 0. In cases where we need an actual availability snapshot, values are left blank before a valid report exists, showing that they are not computable yet.

Appendix VII: Season Win Percentage Updates Across Horizons

Table VII.1. Examples of changes in current_or_prior_season_win_pct_diff as newly completed games enter the information set.

update	Target game	Previous game newly included	Season win pct _ _ update
5d->3d	CLE at NYK 2025-12-25	NYK beat MIA on 2025-12-21	NYK rises from 0.704 to 0.714
3d->2d	DAL at GSW 2025-12-25	GSW beat ORL and DAL lost to NOP on 2025-12-22	GSW rises from 0.483 to 0.500 and DAL falls from 0.379 to 0.367
2d->1d	OKC at CLE 2026-01-19	OKC lost at MIA on 2026-01-17	OKC falls from 0.833 to 0.814
1d->12	LAL at MIA 2026-03-19	LAL beat HOU on 2026-03-18	LAL rises from 0.632 to 0.638

Appendix VIII: Injury-Value Updates Across Horizons

Table VIII.1. Changes in injury_value_lost_diff across forecast horizons for MIA at CHI, 2026-01-29.

Update	Latest report used	PDFs available before cutoff	Injury value lost _ diff	What changed
2d->1d	Jan 28, 8:00 PM	81	-0.709	CHI had larger unavailable value
1d->12h	Jan 29, 8:00 AM	129	-0.673	Similar CHI injury burden
12h->6h	Jan 29, 2:00 PM	146	-0.528	Some statuses updated
6h->3h	Jan 29, 5:00 PM	158	0.343	Balance shifts toward more MIA loss
3h->1h	Jan 29, 7:00 PM	166	0.037	Final report nearly balances the injury loss

> Note: The model does not use all available PDFs as separate features. At each horizon it counts all reports, but the injury variables are only computed from the latest valid report snapshot before that cutoff.

Appendix IX: Model Specifications and Hyperparameters

Table IX.1. Logistic Regression model specifications and hyperparameters.

Setting	Value Used	Purpose
Software	sklearn.linear model.LogisticRegression _	Probabilistic linear classifier
Preprocessing	Median imputation + standardisation	Fills missing values and scales continuous features
Penalty	L2 regularisation	Reduces overfitting by shrinking coefficients
Regularisation strength	C = 0.5	Moderate shrinkage; smaller C means stronger regularisation
Solver	lbfgs	Stable optimiser for logistic regression
Maximum iterations	5000	Ensures convergence
Random state	42	Makes results reproducible
Calibration	Platt scaling on 2024/25	Improves probability calibration before testing

Table IX.2. XGBoost model specifications and hyperparameters.

Setting	Value Used	Purpose
Software	xgboost.XGBClassifier	Gradient-boosted decision-tree classifier
Objective	binary:logistic	Predicts home-win probability
Evaluation metric	logloss	Optimises probabilistic prediction quality
Number of trees	320	Builds many small sequential improvements
Maximum depth	2	Keeps trees shallow to reduce overfitting
Learning rate	0.025	Uses small updates for stable learning
Row subsampling	0.9	Uses 90% of rows per tree to improve generalisation
Feature subsampling	0.9	Uses 90% of features per tree to reduce over-reliance on one variable
Minimum child weight	25	Prevents splits based on very small/noisy groups
L2 regularisation	reg lambda = 8.0	Penalises overly complex trees
L1 regularisation	reg alpha = 0.3	Encourages simpler split structure
Tree method	hist	Efficient histogram-based tree construction
Missing values	Native NaN handling	Learns default directions for missing values
Random state	42	Makes results reproducible
Calibration	Platt scaling on 2024/25	Improves probability calibration before testing

Appendix X: Prediction accuracy by feature-group availability

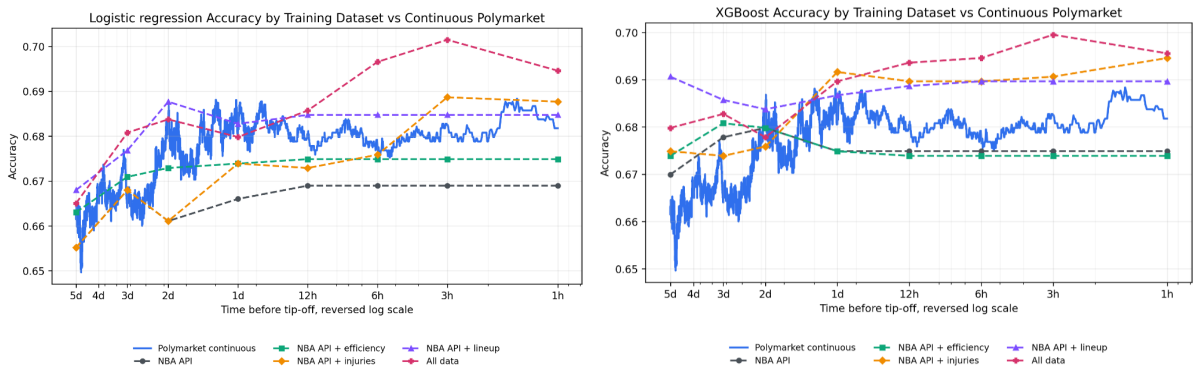

Figure X.1. Prediction accuracy by feature-group availability for Logistic Regression and XGBoost, with Polymarket as an external benchmark.