Bayesian Oracle Design for Sequential E-Sports

Hitansh Shah

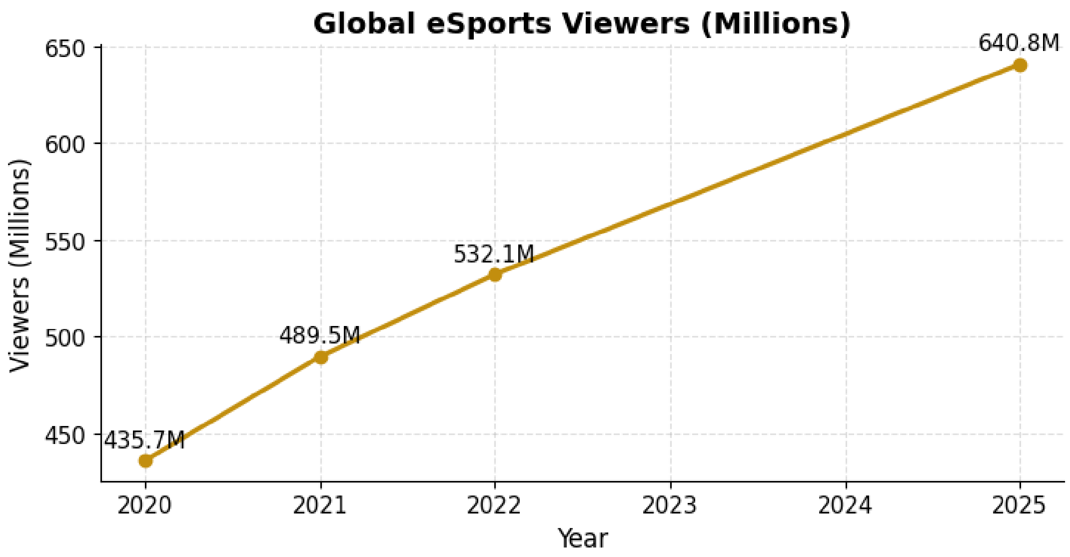
Nov, 25, 2025

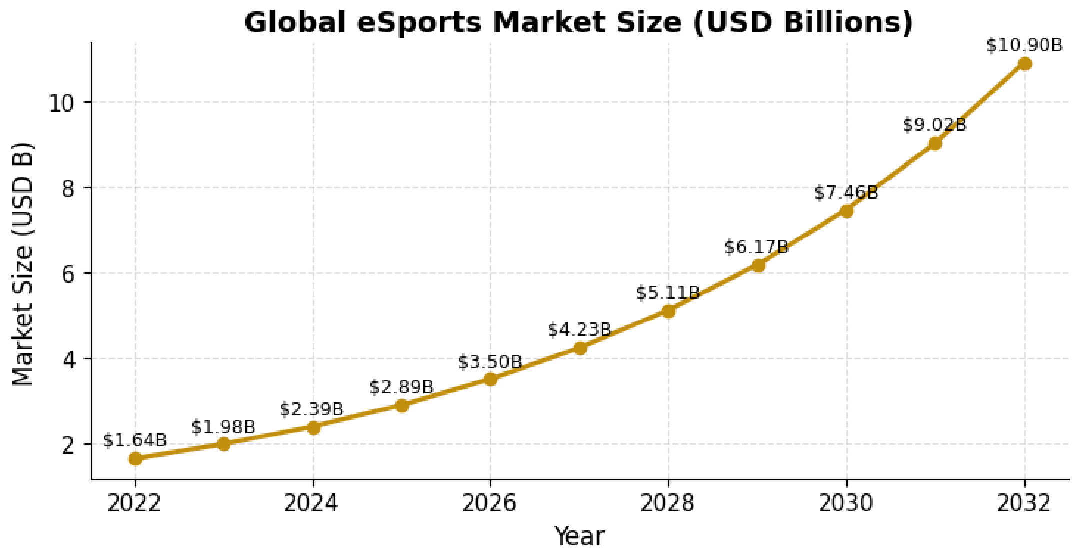
PURDUE UNIVERSITY®

Motivation

e-Sports are on the rise

- Between 2020-2025, total number of e-Sports viewers grew from 435.7M to 640.8M viewers [1]
- Between 2022-2025, the market value of e-Sports has risen from USD 1.64bn to USD2.89bn[2]
- Marketing, sponsorships, tournaments, prize pools, reach multi-million dollar figures (eg: $70Mn at EWC Riyadh)
- Aided by pervasiveness of digital media and digital-first viewership platforms such as Twitch, YouTube, Kick, gives it a global, instant, highly-engaged viewership.
- With professional organizations, teams and environments, e-Sports boasts viewership comparable to traditional sports.





PURDUE UNIVERSITY

Motivation

Popularity of prediction markets

- Prediction markets have exploded in visibility, driven by platforms like Kalshi and Polymarket and the rise of event-based trading.
- The global decentralized prediction-market segment was valued at ≈ **US $1.4 billion in 2024**, with projections to reach ~US $95.5 billion by 2035 (CAGR ~46.8 %). [1]
- The global esports betting market was valued at **US $18.4 billion in 2024**, with a forecast of US $47.2 billion by 2033 (CAGR ~11.2 %) [1]
- Despite rapid growth, most markets lack professional market-makers and sophisticated models leaving structural inefficiencies unchallenged.

Kalshi

Polymarket

Motivation

Counter-Strike 2 as a viable market

- I chose Counter-Strike 2 because of 6+ years of domain experience.
- Its round-based structure creates clean, discrete states that are easy to model.
- Features like equipment value, economy, and scoreline translate directly into predictive inputs.
- The game is stable and skill-driven, which improves signal quality and reduces randomness.
- CS2 markets on Kalshi are liquid, with S-Tier matches often exceeding $100K in traded volume.
- This makes CS2 a data-rich, structured environment that's ideal for testing trading models.

Cover art, depicting the game's main teams: the Counter-Terrorists (left) and Terrorists (right)

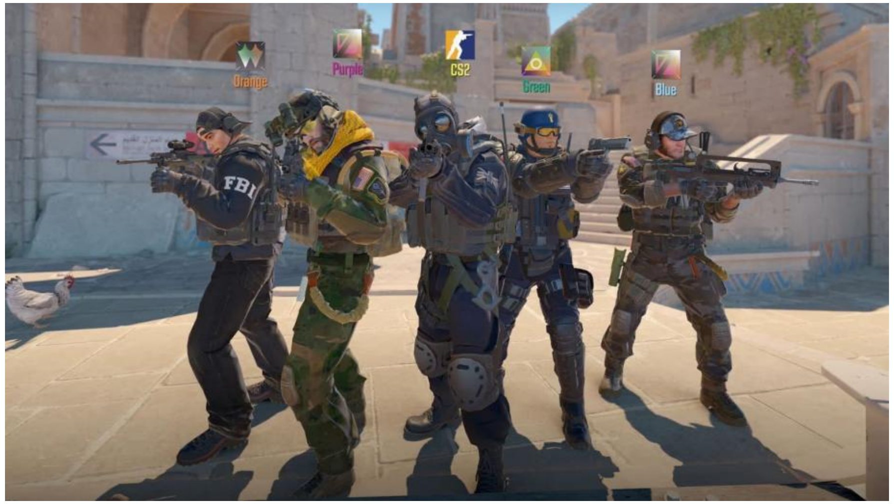
Introduction

PURDUE
UNIVERSITY

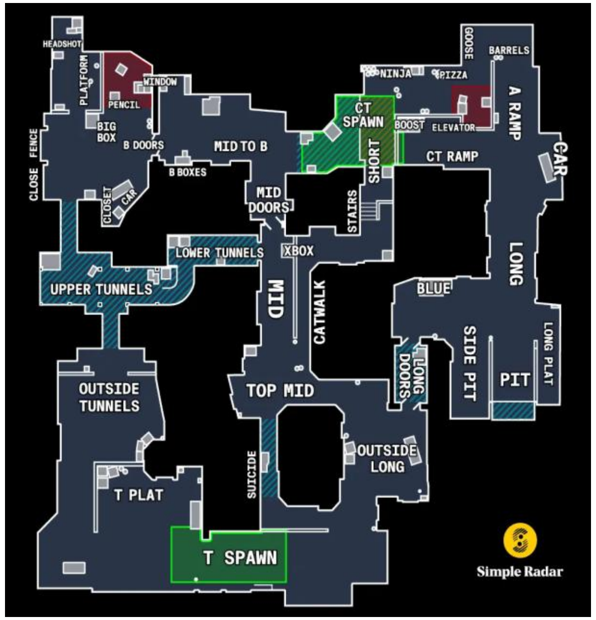
Introduction

Introduction to Counter Strike 2

- Counter-Strike 2 is played between two teams of five players across a sequence of independent rounds.
- Each round ends when one side eliminates the other or completes/denies an objective (bomb plant).
- A map is typically won by the first team to reach 13 rounds (in regulation) or via overtime if tied.
- Teams switch sides (T ↔ CT) halfway through regulation, producing side-specific dynamics.
- Each round yields a binary outcome, enabling round-level probabilistic modelling.



A CS team is composed of 5 players (CT shown here)



Dust II, the most famous and most played maps in Counter-Strike

Introduction

Introduction to Counter Strike 2

- Each team maintains an economic balance which constrains weapon and utility purchases per round.
- Round outcomes affect subsequent economic capacity, introducing state dependence across rounds.
- Weapon and equipment compositions vary substantially across rounds and affect win probabilities.
- Observable information accumulates discretely after each round, enabling sequential model updates.
- This structure provides a repeated series of partially dependent Bernoulli trials under evolving state.

CS2 Buy menu. Each item has a different use case and cost associated to it

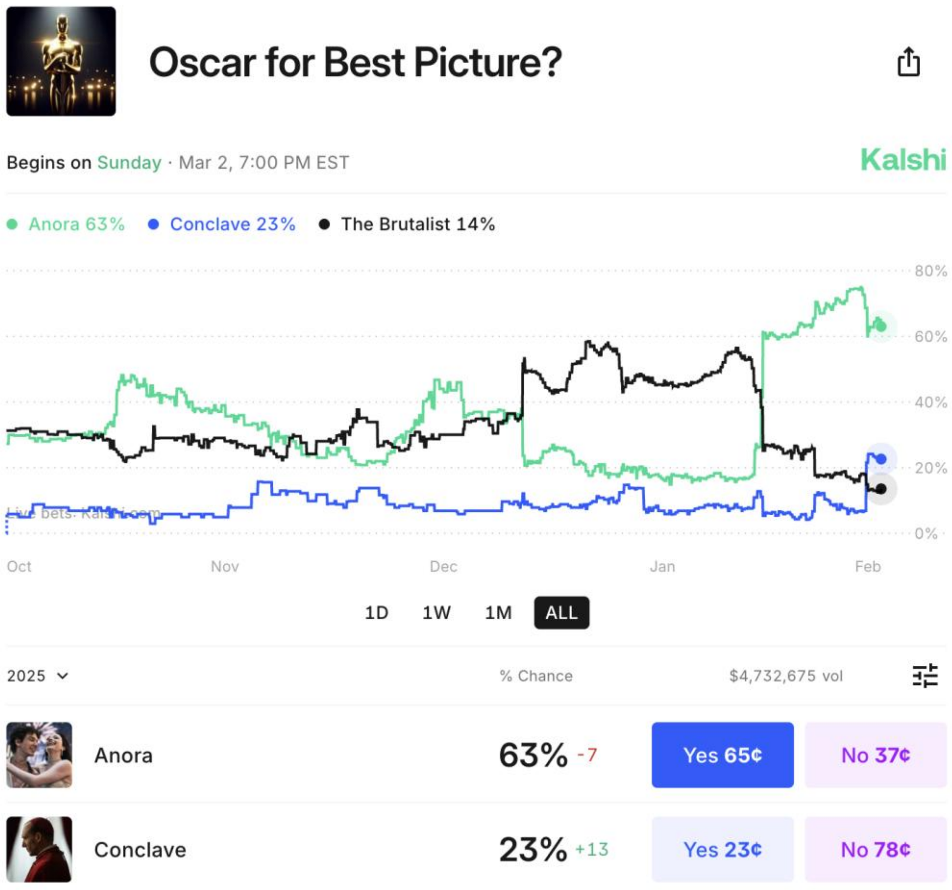
A snapshot of the HUD providing necessary economic and equipment information

PURDUE UNIVERSITY

Introduction

Introduction to Prediction Markets

- Prediction markets are exchange-like platforms that facilitate trading on the outcomes of future events.
- Participants buy and sell contingent claims (contracts) whose value depends on a verifiable future outcome.
- Market prices emerge from the aggregation of dispersed beliefs, information, and expectations held by traders.
- Prices in liquid markets are often interpreted as real-time, probabilistic forecasts of event outcomes.

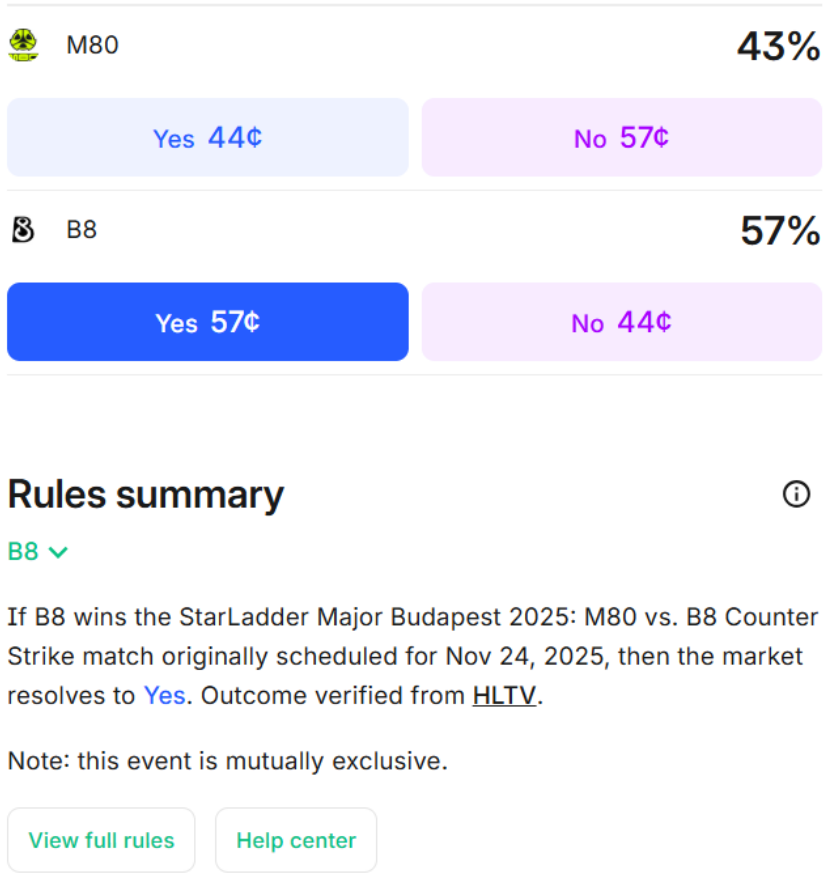


An example screen of a Kalshi market

Introduction

Introduction to Prediction Markets

- Operate via centralized venues (e.g., regulated exchanges) or decentralized platforms (e.g., on-chain protocols).
- Contracts are listed with clearly defined settlement criteria and maturity conditions.
- Trading mechanisms include limit order books, Limit, IOC, GTC or scheduled orders.
- Continuous price discovery occurs through active order flow and liquidity provision.
- Settlement occurs when the outcome is resolved by an authorized adjudicator or oracle.
- Regulatorily, entities with power to influence decision or access to MNPI about the market are not allowed to trade on the market/series.



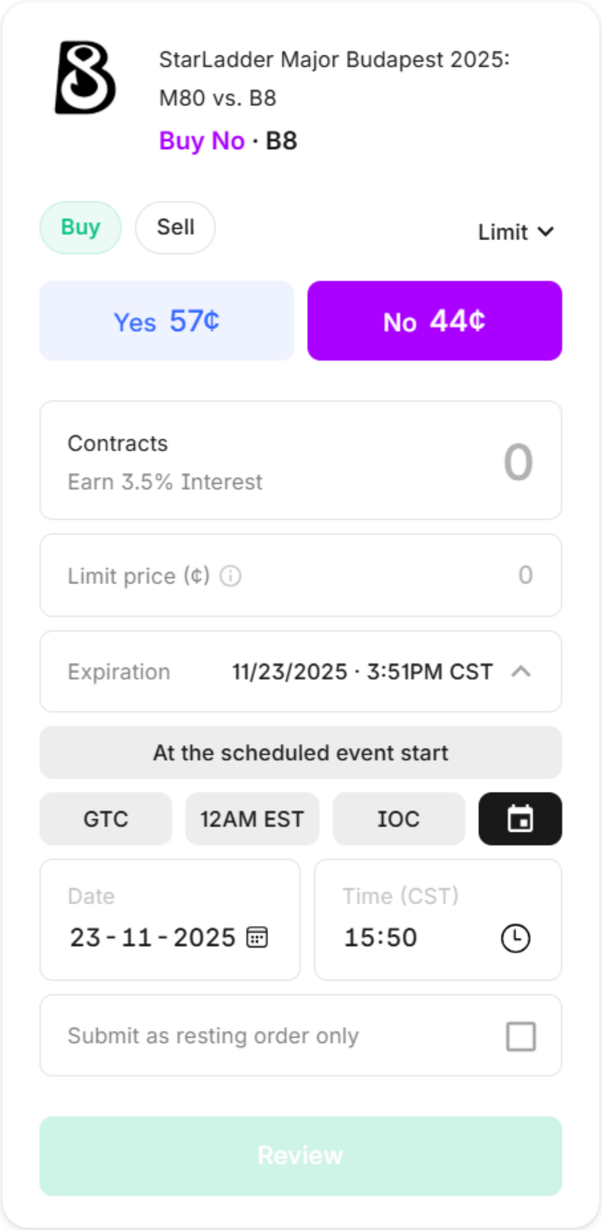
An example of rules for resolution

PURDUE UNIVERSITY

Introduction

Introduction to Prediction Markets

- Core instrument: binary contract paying $1 if the specified outcome occurs, and $0 otherwise.
- Price lies between $0 and $1, representing the implied market probability of the outcome.
- Contracts can represent political outcomes, macroeconomic indicators, sports results, corporate events, or scientific milestones.
- On the underlying buying a ‘YES’ and selling a ‘NO’ contract is essentially the same contract and is treated as such. i.e. There is no arbitrage between buying a ‘YES’ and selling a ‘NO’. Yes/No bifurcation is merely a convenience tool.
- Empirical literature shows that prediction markets often outperform expert surveys and polling in accuracy and responsiveness.



Buy screen showing different order types

PURDUE UNIVERSITY

Literature Review

Literature Review

General Review

- Most papers related to the topic deal with match level odds prediction
  - Tennis betting: Can statistics beat bookmakers? (Lisi & Zanella, 2013) *– Inconclusive Results, Similar odds to bookmakers, but cannot beat them. Importantly, showed Logistic Regression as a viable ML model for bookmaking.*
  - Predicting Counter-Strike Matches: Using Machine Learning Models (Broms & Nordansjo, 2024) *– uses logistic regression and random forest models trained on team-level differences of player performance statistics from professional Counter-Strike: Global Offensive matches, and finds modest predictability (≈ 60 % accuracy) but does not outperform bookmaker benchmarks.*
  - Graph Neural Networks to Predict Sports Outcomes (Xenopoulos & Silva, 2022) *– Predicts pre-match winners by modeling teams as nodes in a relational sports graph and applying a GNN to learn team-strength embeddings. The GNN method outperforms traditional ML baselines, showing that graph structure improves match-outcome prediction. Predicts odds before-hand, no updates, can be a useful prior to our model.*

PURDUE UNIVERSITY®

Literature Review

Precision Under Fire: Analysis and Predictive Modeling in Counter-Strike: Global Offensive, Xia et al. (2025)

- Paper builds structured datasets from CS:GO pro matches
- Predicts round winners and player deaths from single-state snapshots
- Uses features like HP, armor, economy, equipment value, positions, and events – Consistent with my model
- Finds simple ML (Such as Logistic Regression) models can capture core gameplay dynamics
- Shows economy matters most when predicting round level winners
- Confirms CS:GO has stable, predictable structure that models can learn
- Useful baseline reference for my project

	Default Parameters			Tuned Parameters		
Model	Accuracy	F1 (Class 0)	F1 (Class 1)	Accuracy	F1 (Class 0)	F1 (Class 1)
Logistic Regression	0.858	0.702	0.907	0.859	0.703	0.908
Decision Tree	0.797	0.606	0.863	0.841	0.656	0.896
XGBoost	0.850	0.690	0.901	0.859	0.709	0.907
Random Guess	0.500	0.500	0.500	0.500	0.500	0.500
Majority Class	0.740	0.000	0.850	0.740	0.000	0.850

TABLE II: Performance comparison for round result prediction between default models and tuned models.

Literature Review

Precision Under Fire: Analysis and Predictive Modeling in Counter-Strike: Global Offensive, Xia et al. (2025)

Dimension	Precision Under Fire (2025)	This Project
Game Version	CS:GO (Dust 2 only), demo-parsed offline dataset	CS2 (all maps), real-time scoreboard-derived data
Prediction Tasks	Round-result prediction & player-death prediction	Continuous map-win probabilities + BO3 series EV
Feature Type	Rich telemetry: HP, armor, distances, weapons, econ, spatial metrics	Lightweight round/score features + prior beliefs + live evidence
Modeling Approach	Logistic Regression, Decision Trees, XGBoost	Bayesian Softmax + Beta-prior blending + Monte-Carlo series engine
Update Frequency	Snapshot-based (e.g., at bomb plant)	Updates every round; designed for live markets
Output	Class labels, accuracy, F1 metrics	Probability curves, credible intervals, fair-value odds
Intended Use	Coaching, strategy, player analytics	Live-betting, market-making, pricing accuracy

Methodology

PURDUE UNIVERSITY

Methodology

Problem Statement

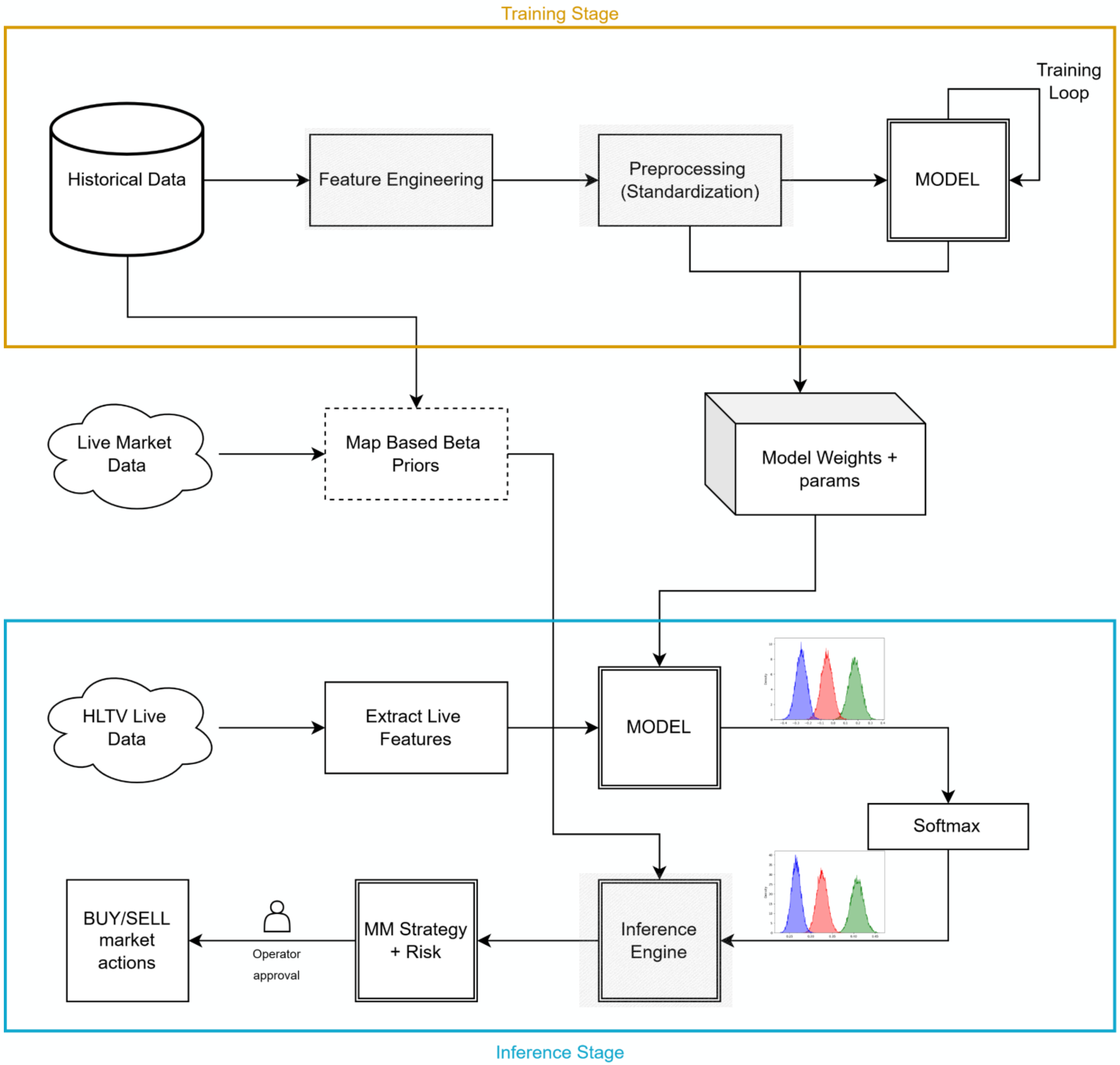
Core Problem Statement:

Create a model incorporating current game state (Round differential, Equipment difference, game stage etc.), prior market beliefs, and current match state (map number, scoreline) to create an oracle for live probability distribution of odds of a team winning a given series (match).

- Some heuristics we would like to see
  - When little or no evidence is available – Fall back to prior beliefs
  - When we have strong belief about the outcome based on current game state, favor those odds.
  - Propagate map level beliefs to Bo3 series

Methodology

Overview



PURDUE UNIVERSITY

Methodology

Data Sources

This project needs two types of data sources

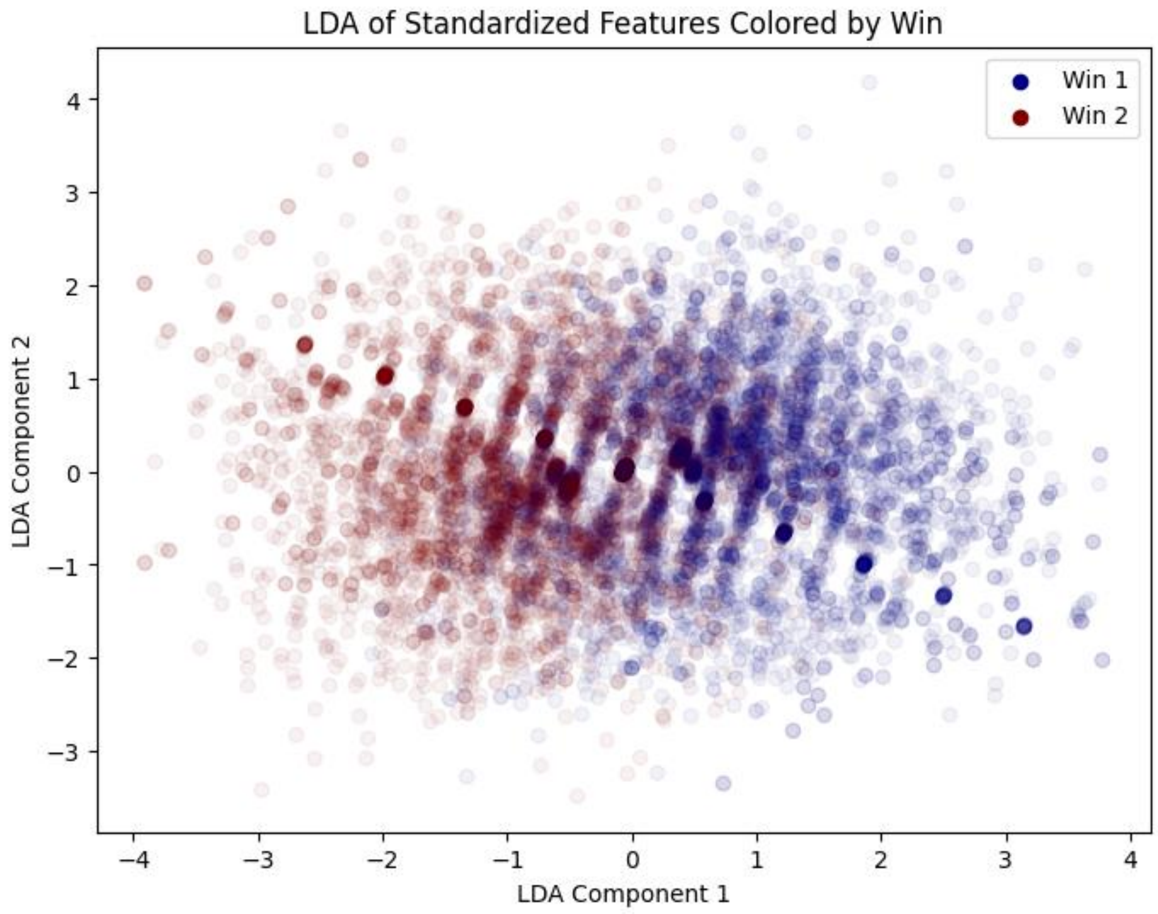
- Historical Data for training and fitting
  - Counter-Strike games are recorded tick-by-tick (64 ticks/sec)
  - Recordings are saved as demo (.dem) files, that store in-game actions rather than a video
  - Stored and available on HLTV Database
  - However, scraping is not easy as it is protected by CloudFlare DNS and severely rate-limited.
  - Created a custom multi-threaded scraping script to bypass CloudFlare's rate-limits
  - Collected and curated a dataset of 11K rounds ~500 maps across 2 years since CS2's release.
  - Each demofile is then parsed using a custom script to distil it to round start snapshots (i.e. the tick after the round ends and money is added to each team)
  - Importantly demo files do not have wall-clock/UTC time recorded. Thus, it is difficult to create a co-timed historical data of odds vs gamestate unless collected live over a long time.
- Live Data for live odds and market making
  - Live scoreboard similarly scraped from HLTV
  - Live data pulled from Kalshi, recording is optionally available.

Methodology

Feature Engineering

- Selected Features
  - While demo files collect all available actions taken in game, not every action or game-state at every tick is an important factor.
  - In fact, from previous literature review and domain expertise the most important features are:
    - Round Difference ($\mathrm{diff\_current\_rounds}$)
    - Total Rounds Played ($\mathrm{total\_rounds\_played}$)
    - Difference in Current Equipment value ($\mathrm{diff\_current\_equip}$)
    - Difference in Balance cash available ($\mathrm{diff\_balance}$)
  - Target: map outcome (win)
  - LDA of selected features shows linear separation

total_rounds_pl...	diff_current_eq...	diff_balance	diff_current_ro...	win	demo_id
0	-650	50	0	1	iem-katowice-2025-...
1	-7750	50	-1	1	iem-katowice-2025-...
2	18400	3700	0	1	iem-katowice-2025-...
3	26250	8550	1	1	iem-katowice-2025-...
4	5250	31400	2	1	iem-katowice-2025-...
5	26850	19050	3	1	iem-katowice-2025-...
6	200	4150	2	1	iem-katowice-2025-...
7	1800	8250	3	1	iem-katowice-2025-...



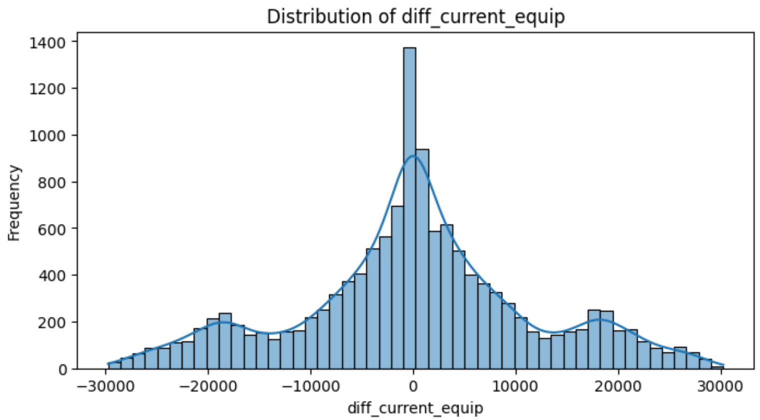
Methodology

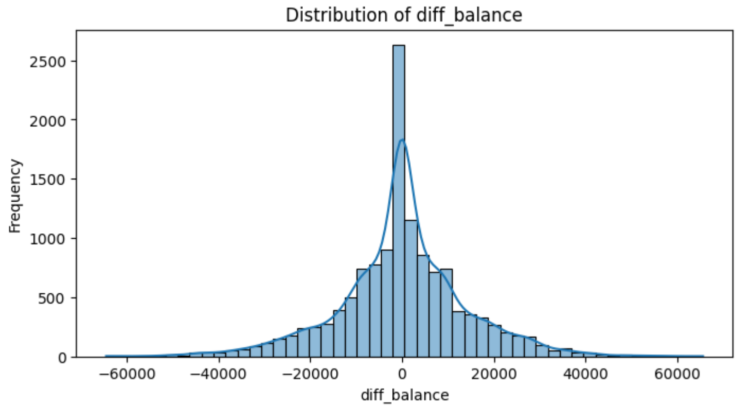
Feature Engineering

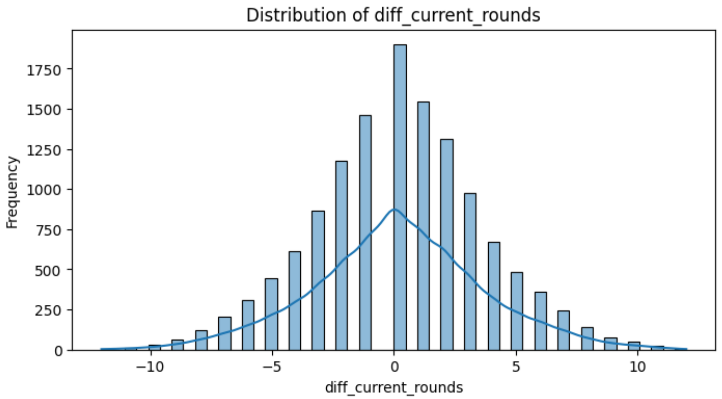
- **Why use $\mathrm{round\_diff}$ and $\mathrm{total\_rounds}$ over simply scoreline of the game such as 7-5, 6-8 and so on …**
- Difference is: $\mathrm{a,b\ vs\ (a+b),(a\text{-}b)}$ (Linear Transformation)
- This representation separates two distinct dimensions of information:
  - who is ahead (magnitude & direction of advantage)
  - how far into the map we are (progression / remaining rounds)
- This yields a more interpretable and intuitive coordinate system for the model:
  - $\mathrm{round\_diff}$ is symmetric around 0 (Almost cauchy-esque distribution)
  - $\mathrm{total\_rounds}$ is monotonically increasing
- Intuitively similar to how we decompose price processes into a drift term + some random walk with $\mu = 0$

$$dS_t = \mu dt + \sigma dW_t$$

- Matches intuitive isotropic Gaussian priors on weights







PURDUE UNIVERSITY

Methodology

Pre Processing

- Each feature is standardized before use. μ, σ are calculated from historical dataset and stored for inference

$$\tilde{x}_i = \frac{x_i - \mu}{\sigma}$$

- Final data setup

$$x_i \in R^{\mathbb{D}}$$
$$y_i \in \{0, 1, 2\}$$

$$y_i = 0 \Rightarrow \text{Overtime},$$
$$y_i = 1 \Rightarrow \text{Team 1 wins},$$
$$y_i = 2 \Rightarrow \text{Team 2 wins}$$

Model and Training

Methodology

Model Training

- At its core, model assumes:

$$y_i \mid x_i, W, b \sim \mathrm{Categorical}(\pi_{i0}, \ldots, \pi_{i,K-1})$$

- Each class(k) gets a logit:

$$\eta_{ik} = x_i^\top W_{\cdot k} + b_k$$

- Which is then softmaxed:

$$P(y_i = k \mid x, W, b) = \pi_{ik} = \frac{\exp(\eta_{ik})}{\sum_j \exp(\eta_{ij})}$$

- Each W_i is initialized with Bayesian Prior:

$$W_{dk} \sim \mathcal{N}(0,1), \qquad b_k \sim \mathcal{N}(0,1).$$

Methodology

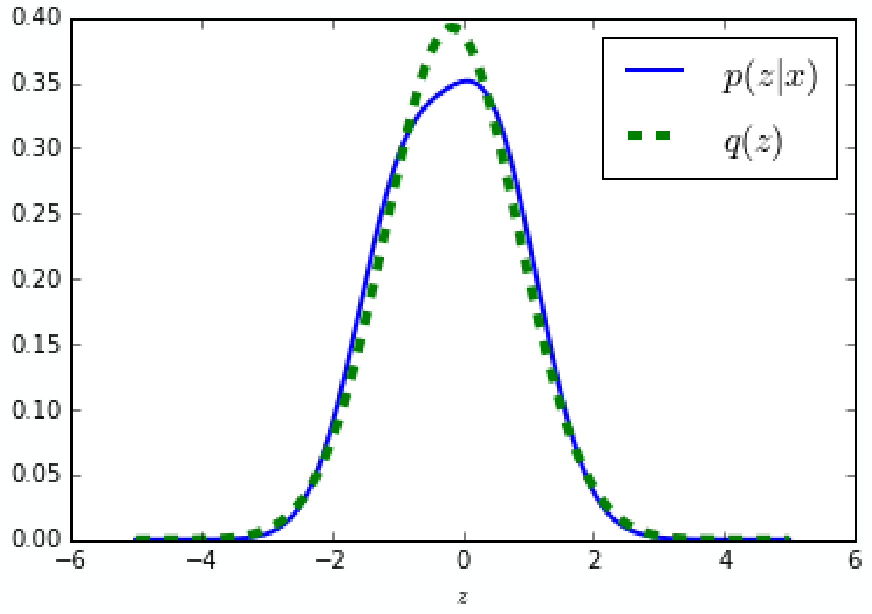
Model Training

- The following is the Bayesian Posterior Objective

$$p(W, b \mid \text{data}) = \frac{\overbrace{\boxed{p(\text{data} \mid W, b)}}^{\text{Likelihood}}, \overbrace{\boxed{p(W, b)}}^{\text{Prior}}}{p(\text{data})}$$

- The denominator p(data) is intractable for Multinomial Logistic Regression.
- Posterior cannot be computed exactly.
- Thus approximate with parametric approximation of posterior

$$q(W, b) = \prod_{d,k} \mathcal{N}(W_{dk}; \mu_{dk}, \sigma^2_{dk}) \prod_{k} \mathcal{N}(b_k; \mu_{b_k}, \sigma^2_{b_k}).$$



Methodology

Model Training: Variational Inference

- Now, Ideally,

$$q^* = \arg\min_q \mathrm{KL}\big(q(W,b)||p(W,b \mid \mathrm{data})\big)$$

- Obviously, $\mathrm{p(W,b \mid data)}$ is intractable
- However we can rewrite KL Divergence as

$$\mathrm{KL}\big(q(W,b)||p(W,b \mid \mathrm{data})\big) = log(p(data) - ELBO(q,p)$$

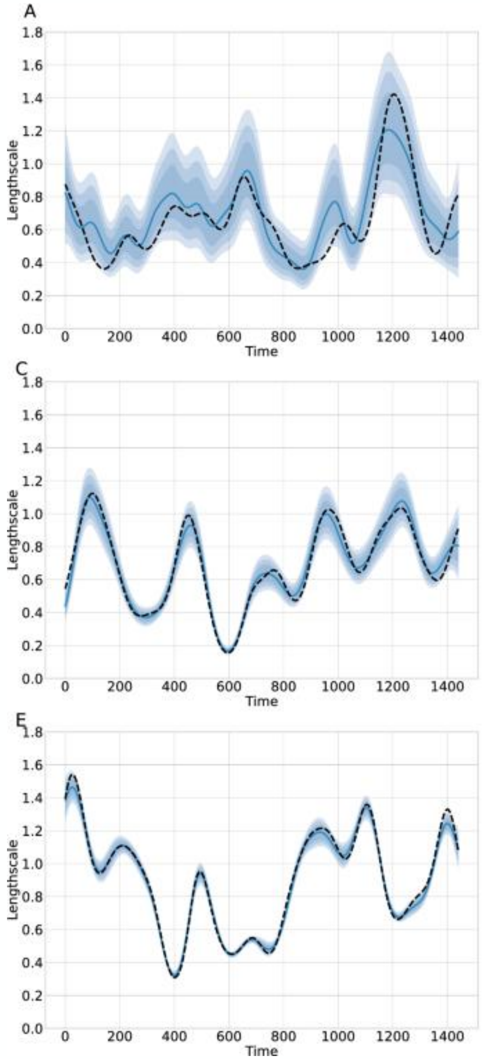
$$\mathrm{ELBO}(q,p) = \int q(W,b) \log \frac{p(W,b,data)}{q(W,b)} dW\,db$$

$$= \mathbb{E}[\log p(\mathrm{data}, W, b)] - \mathbb{E}[\log q(W,b)]$$

- Drop $\mathrm{log(p(data))}$ since we want to find KL over the space of $\mathrm{(W,b)}$

$$\implies \arg\min_q \mathrm{KL}\big(q(W,b), |, p(W,b \mid \mathrm{data})\big) \leftrightarrow \arg\max_q \mathrm{ELBO}(q)$$

- Optimize using MC sampling using Pyro (python)



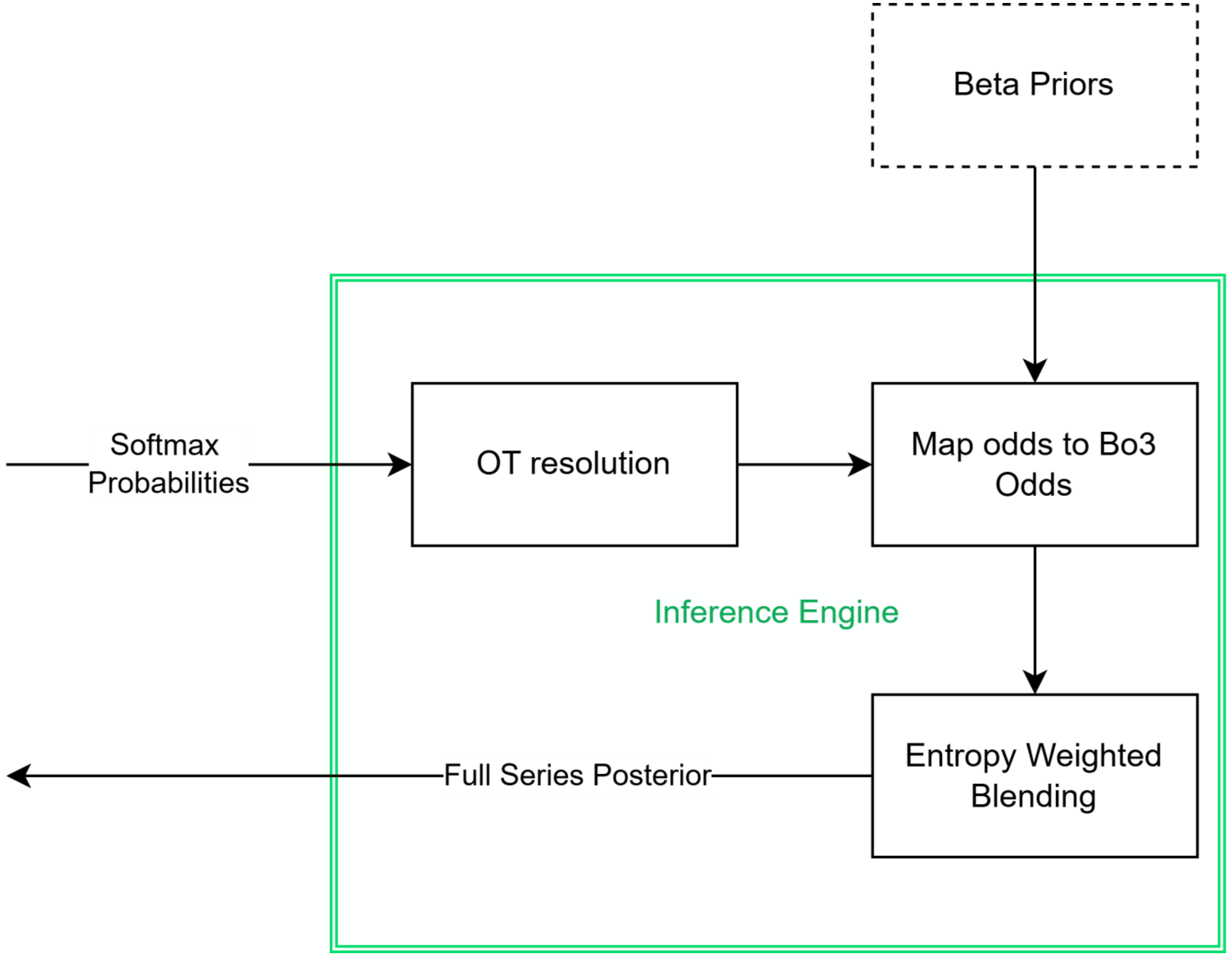
Sample SVI training (Not ours)

The Inference Engine

PURDUE UNIVERSITY®

Methodology

Inference Overview



PURDUE UNIVERSITY

Methodology

Inference

- Draw S samples of $W^{(s)},\ b^{(s)}$ from q
- Given a new standardized row x, compute logits:

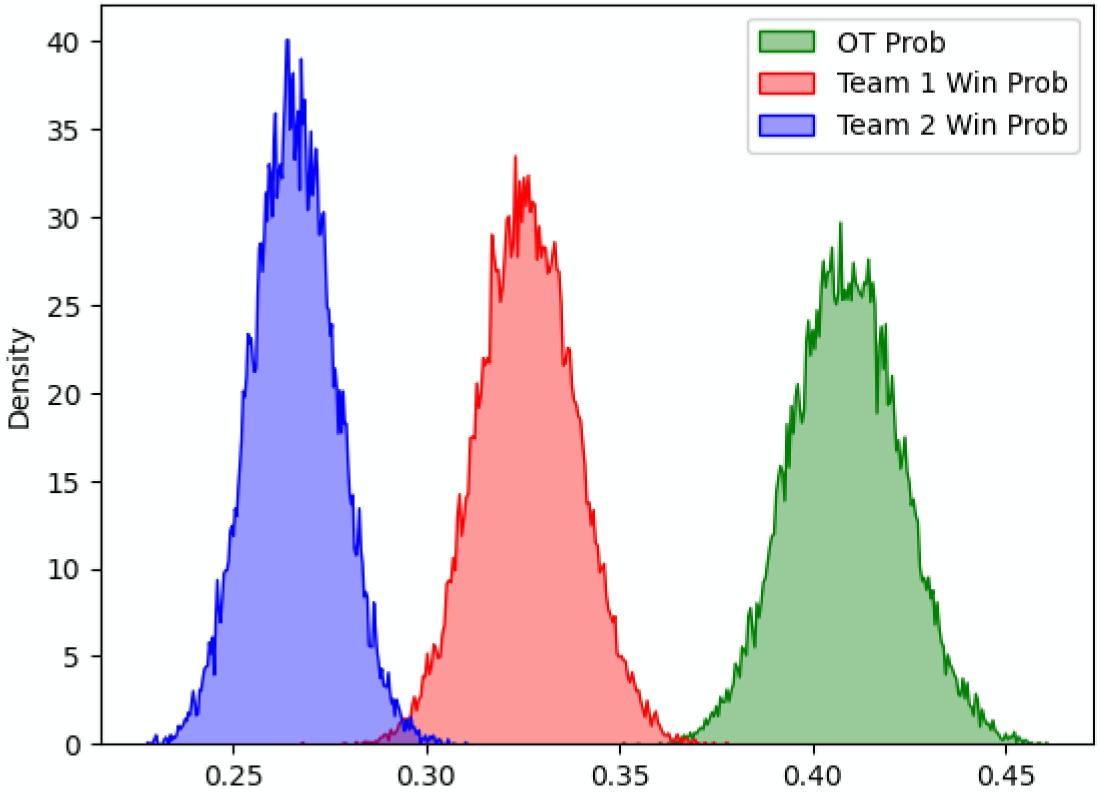
$$\eta_k^{(s)} = x^\top W_k^{(s)} + b_k^{(s)}$$

- Apply categorical softmax

$$\pi_k^{(s)} = \frac{\exp(\eta_k^{(s)})}{\sum_j \exp(\eta_j^{(s)})}$$

- This yields:

$$p_{\text{live}} = (p_{\text{OT}}, p_{\text{T1}}, p_{\text{T2}})$$



PURDUE UNIVERSITY

Methodology

Inference – OT Resolution

- Since OTs must be resolved, we collapse the OT probability as follows

$$p^*_{\mathrm{T1,\ live}} = p_{\mathrm{T1}} + \omega_{\mathrm{OT}}, p_{\mathrm{OT}}, \qquad p^*_{\mathrm{T2,\ live}} = 1 - p^*_{\mathrm{T1,\ live}}$$

- Where OT weights

$$\omega_{\mathrm{OT}} = \begin{cases} \dfrac{t_2}{t_1}, & t_1 > t_2, \\ 1 - \dfrac{t_1}{t_2}, & t_1 < t_2, \\ 0.5, & t_1 = t_2. \end{cases}$$

PURDUE UNIVERSITY

Methodology

Inference – Beta Priors

- We use Beta priors to encode beliefs about the match such as
  - T1 has a stronger roster than T2
  - Map choice bias
  - Market favors T1/T2 to win
- These priors are encoded as μ and κ

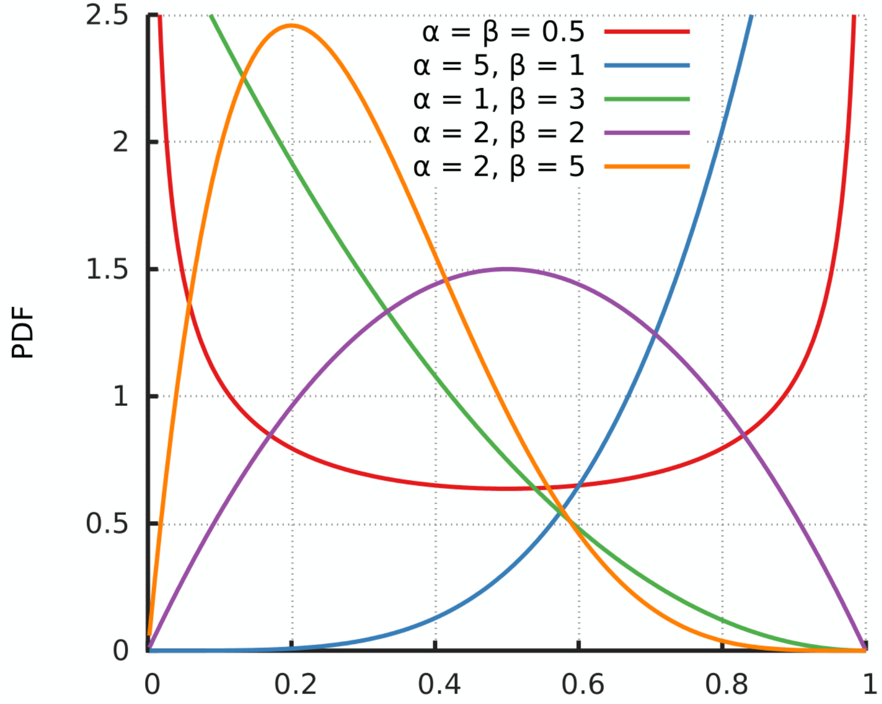
$$p_m \sim \mathrm{Beta}(\alpha_m, \beta_m),$$
$$\alpha_m = \mu_m \kappa_m, \quad \beta_m = (1 - \mu_m)\kappa_m$$

- A series level prior is defined as

$$\frac{\mu_1 + \mu_2 + \mu_3}{3}$$

- And Monte-Carlo draws

$$p_1^{(i)}, p_2^{(i)}, p_3^{(i)} \sim \mathrm{Beta}(\alpha_m, \beta_m), \quad i = 1, \dots, N$$



PURDUE UNIVERSITY

Methodology

Inference – Convert map probability to Bo3 Outcome

- Model only predicts current map win probabilities, mixed with prior we arrive at series probs.
  - If map 1 is being played (0-0)

$$P_{\mathrm{T1}}^{(i)} = p_L, p_2^{(i)} * p_L(1 - p_2^{(i)})p_3^{(i)} * (1 - p_L)p_2^{(i)}p_3^{(i)}$$

  - If map 2 is being played (0-1/1-0)

$$P_{\mathrm{T1}}^{(i)} = p_L + (1 - p_L)p_3^{(i)}$$

$$P_{\mathrm{T1}}^{(i)} = p_L p_3^{(i)}$$

  - If map 3 is being played (1-1)

$$P_{\mathrm{T1}}^{(i)} = p_L$$

PURDUE UNIVERSITY

Methodology

Inference – Entropy weighted Blending

- Idea: When the model is unsure we want to err more towards the prior as dictated by the market and when model is surer (or "screaming" a more interesting result), use the model's prediction.
- Metric for model surety: Information Gain!
- Let Predictive Entropy H:

$$H = -\mathbb{E}\left[p \log p + (1-p)\log(1-p)\right]$$

- Then IG:

$$I = \log 2 - H$$

- Use as weight, we empirically find $\tau = 10.0$ works best

$$w = e^{-\tau I}$$

- Blend on logit scale and sigmoid to return to probability space:

$$L^{(i)} = (1-w)\,\mathrm{logit}\left(P_{\mathrm{T1}}^{(i)}\right) + w\,\mathrm{logit}\left(\bar{p}_{\mathrm{prior}}\right) \qquad \tilde{P}_{\mathrm{T1}}^{(i)} = \sigma(L^{(i)})$$

Methodology

Inference – Final Bo3 Posterior

- Finally, we calculate Bo3 posterior probability as:

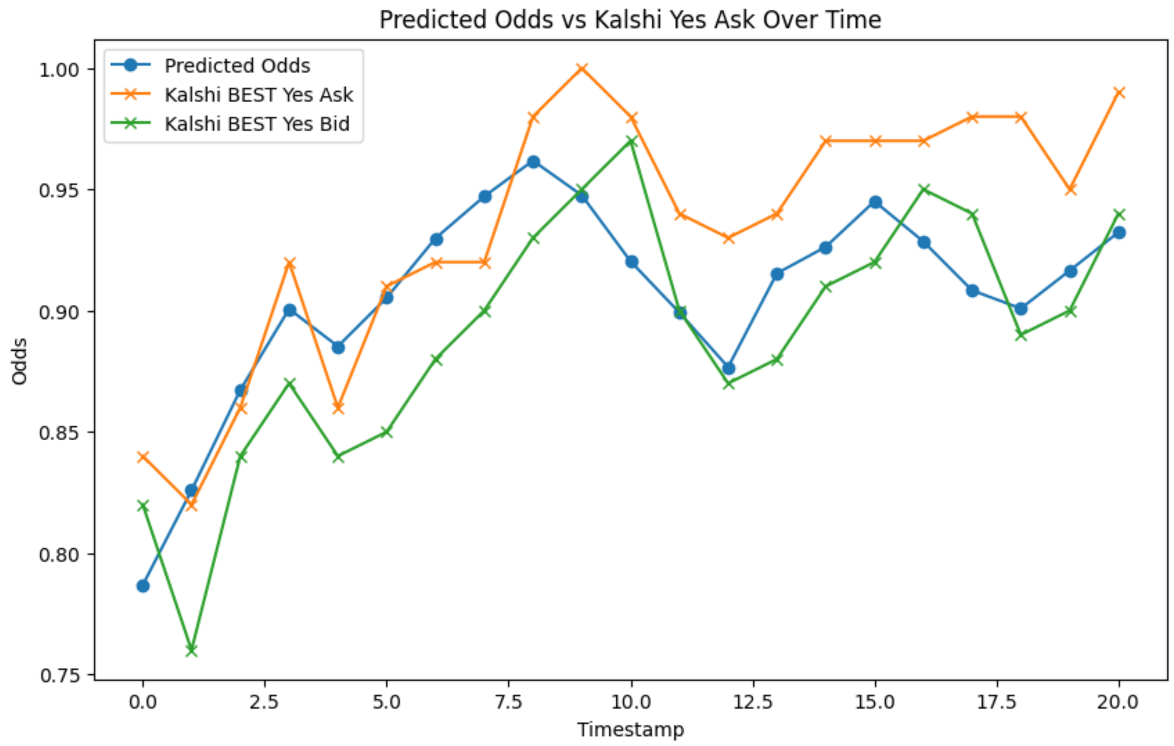
$$\boxed{P_{\mathrm{T1}} = \frac{1}{N}\sum_{i=1}^{N} \tilde{P}_{\mathrm{T1}}^{(i)}, \qquad P_{\mathrm{T2}} = 1 - P_{\mathrm{T1}}}$$

- With Corresponding CIs as

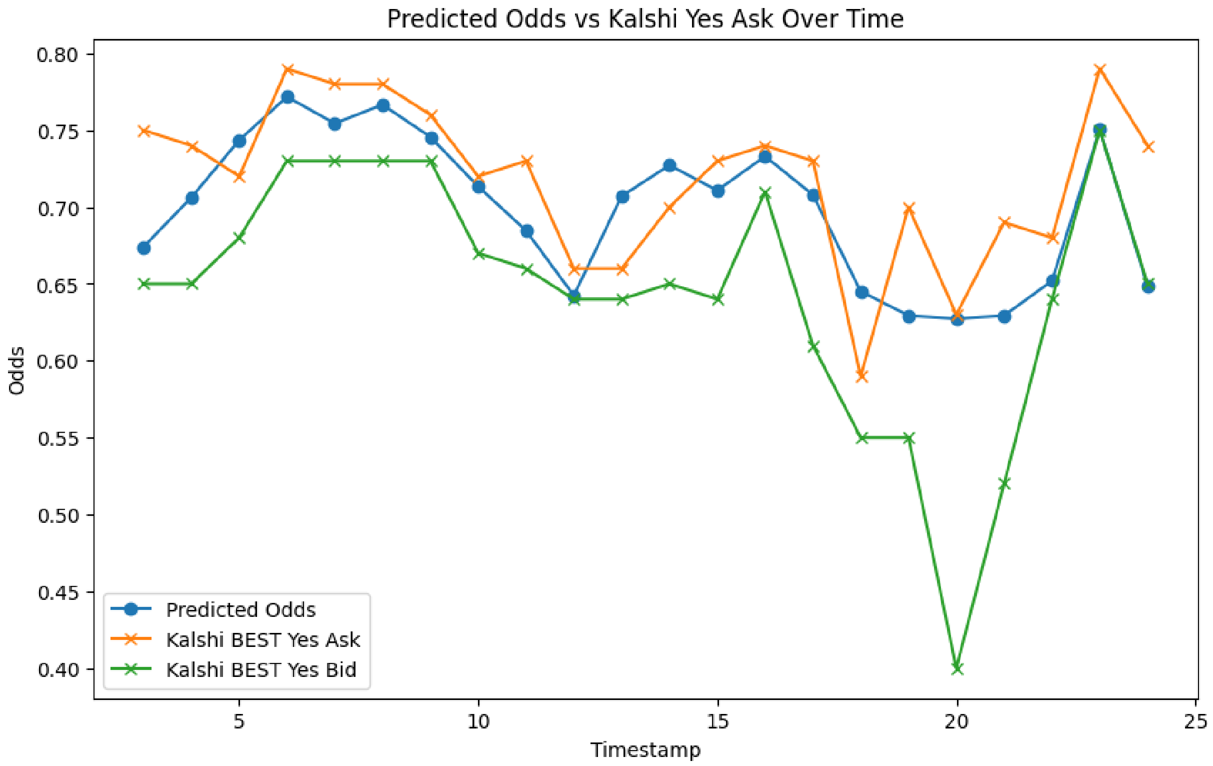
$$\mathrm{CI} = \left[\mathrm{Quantile}_{0.025}(\tilde{P}_{\mathrm{T1}}^{(i)}), \mathrm{Quantile}_{0.975}(\tilde{P}_{\mathrm{T1}}^{(i)})\right]$$

Results

Results

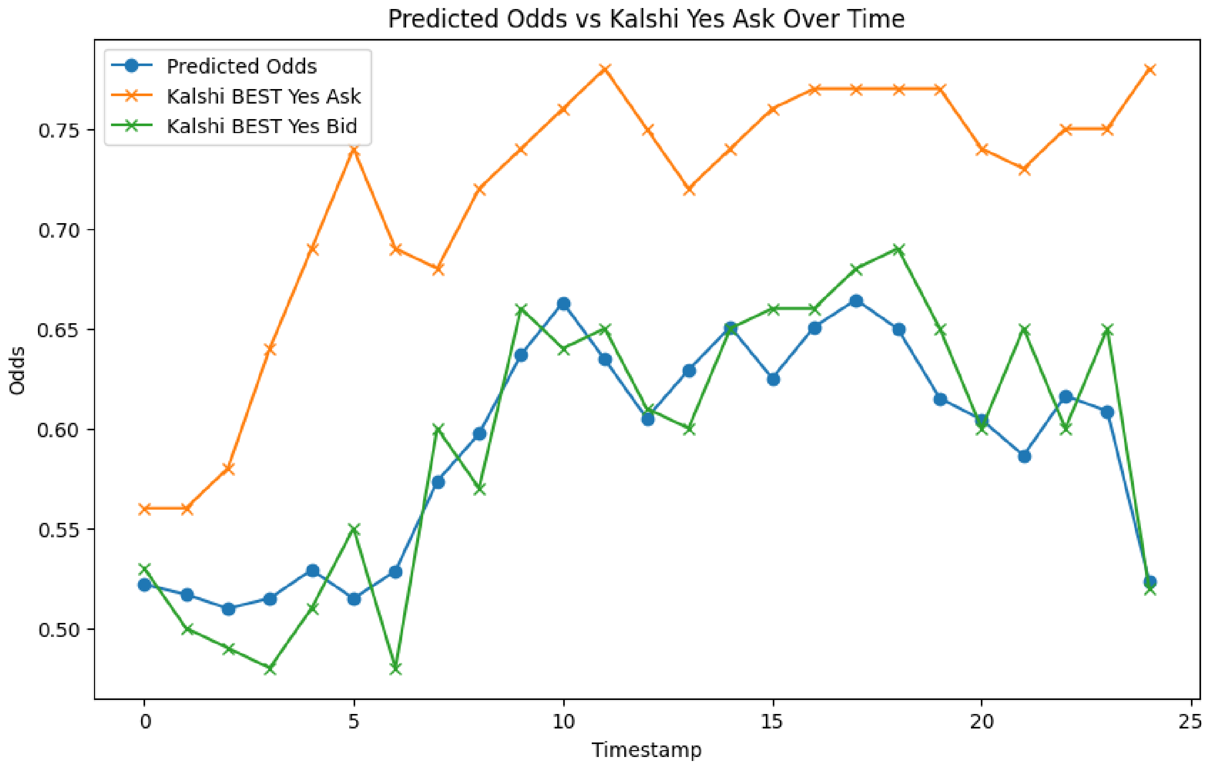


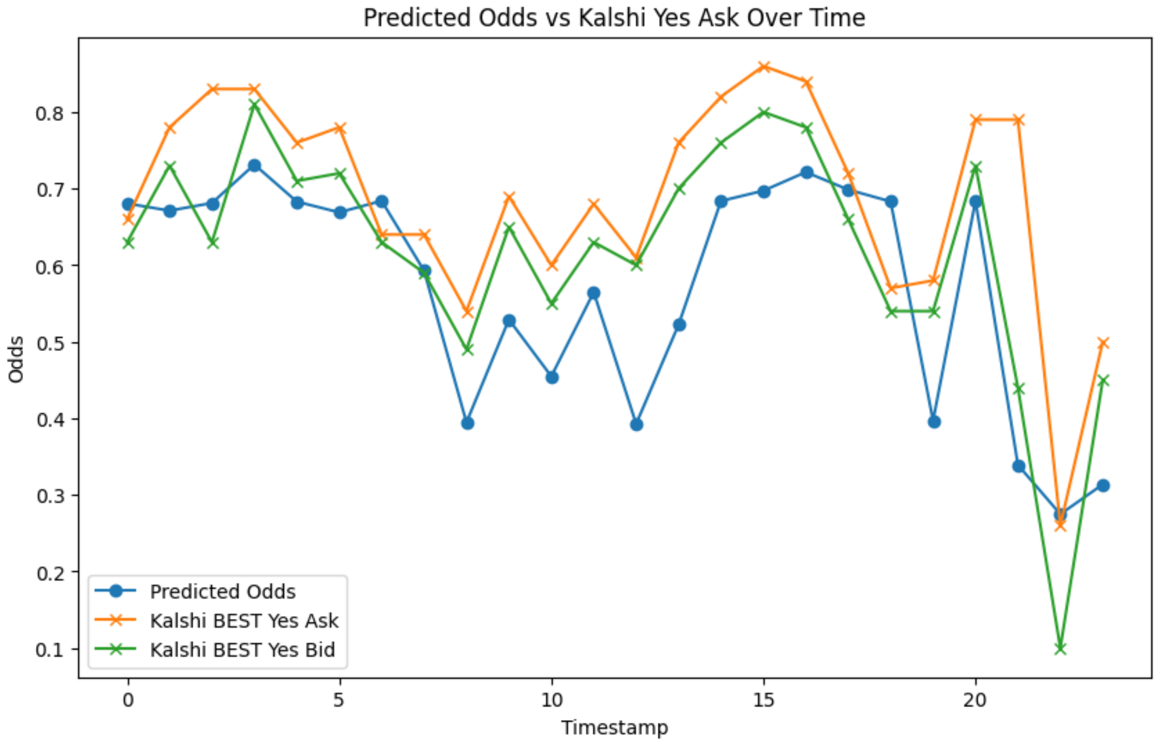
Predicted Odds vs Kalshi Yes Ask Over Time
Predicted Odds
Kalshi BEST Yes Ask
Kalshi BEST Yes Bid
Odds
Timestamp
1.00
0.95
0.90
0.85
0.80
0.75
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
20.0



Predicted Odds vs Kalshi Yes Ask Over Time
Predicted Odds
Kalshi BEST Yes Ask
Kalshi BEST Yes Bid
Odds
Timestamp
0.80
0.75
0.70
0.65
0.60
0.55
0.50
0.45
0.40
5
10
15
20
25

Results





PURDUE UNIVERSITY

Results

Segment	N	MSE	RMSE	MAE	Bias	Pearson ρ	Directional Accuracy	SSAE
1	25	0.018238	0.135047	0.126766	−0.126766	0.7058	100.0%	0.5800
2	21	0.001745	0.041770	0.035900	−0.028732	0.8021	100.0%	0.9509
3	22	0.001707	0.041318	0.033882	−0.019974	0.7283	100.0%	0.5147
4	24	0.029368	0.171370	0.143921	−0.135641	0.7129	79.17%	2.9864

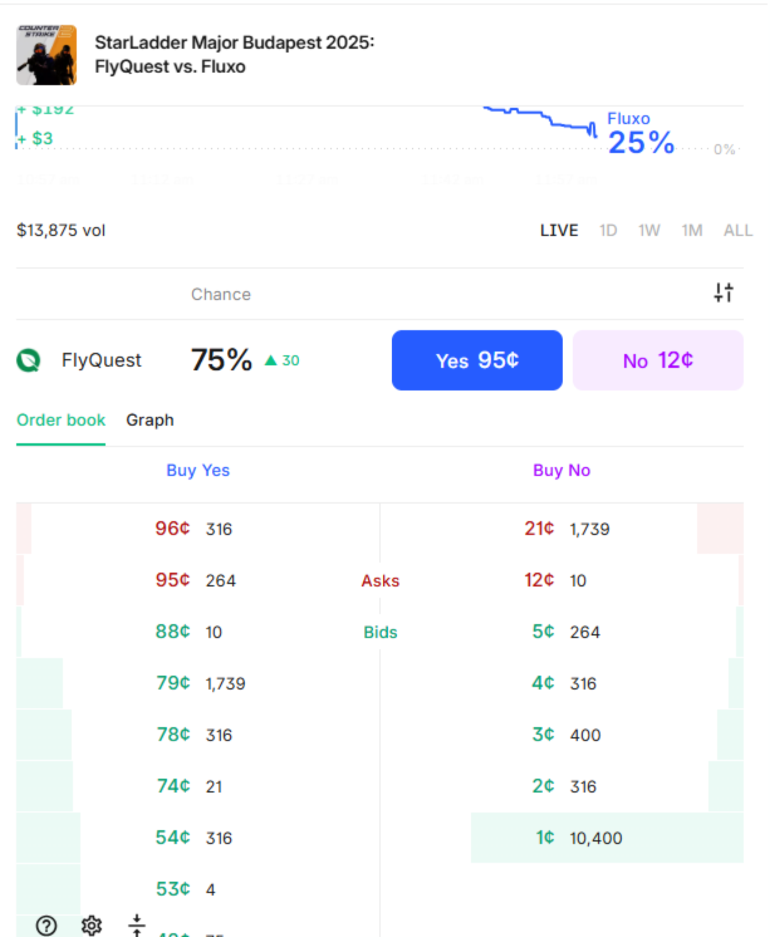
Risks, Limitations, and Future Directions

PURDUE UNIVERSITY®

Risks, Limitations and Future Directions

Structural Risk and Execution Risk

- **Risk:**
  • Prediction markets have thin books, shallow depth, and wide spreads.
  • Orders may sit unfilled for minutes; fills may occur after the midprice has moved.
  • Maker quotes may get "picked off" by latency-sensitive actors or conditional orders.
- **Mitigation:**
  • Quote **wider safety spreads** proportional to real-time volatility of your posterior.
  • Use **short time-in-force** (TIF) or small **iceberg-style split orders** to reduce pickoff.
  • Use **midprice + protective skew** when uncertain, and **cancel/replace** on every state tick.
  • Maintain **internal fair-value bands** (e.g., ±1σ of posterior samples) as no-trade zones.



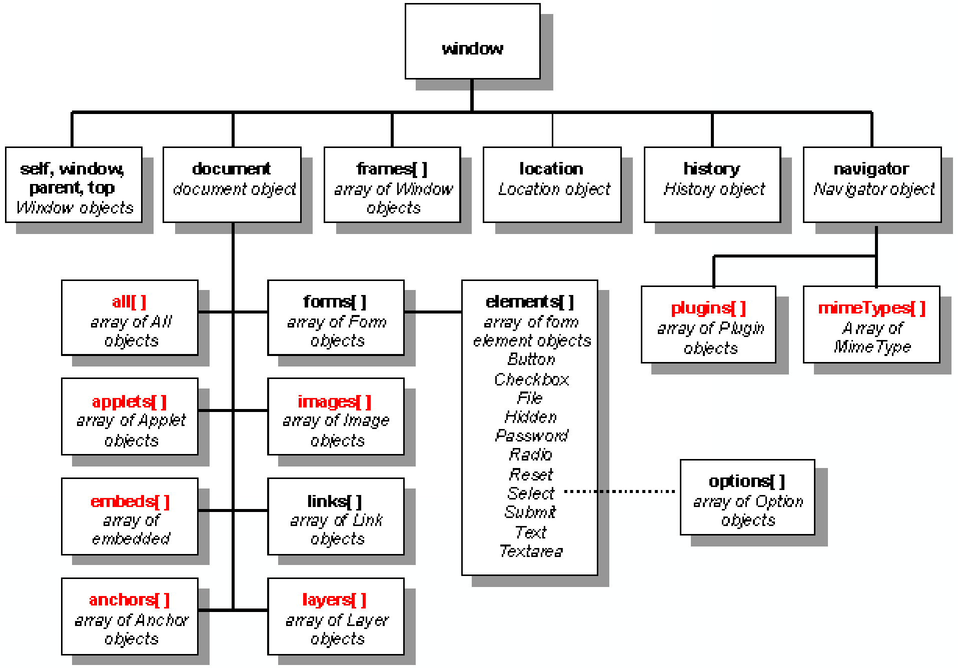
Spread of 5-10% of total market width are common in relatively illiquid markets

PURDUE UNIVERSITY

Risks, Limitations and Future Directions

Latency Risk

- **Risk:**
  - Scoreboard polling (Playwright) is slower than exchange tick updates.
  - You may price based on “t–1 round” while the market is at “t”.
  - BO3 transitions may be based on stale map scores.
- **Mitigation:**
  - Aggressive **domcontentloaded + reduced rendering** in Playwright.
  - Strip page to **minimal DOM capture**; avoid re-querying static elements.
  - Add **timestamp deltas** to every prediction; discard anything older than threshold.
  - Background thread to keep map_no, scoreline, economy synced at sub-250ms intervals.’
  - Ideally, pay for a low-latency data provider



Fully parsing a DOM is a computing intensive task

Conclusion

PURDUE
UNIVERSITY®

Conclusion

- This work demonstrates that **live prediction markets can be priced with principled Bayesian machinery**, not heuristics.
- The model integrates **state-dependent round dynamics**, **map-specific priors**, and **continuous posterior updating**, producing probabilities that are statistically coherent and operationally stable.
- By combining **variational Bayesian softmax inference** with a **Monte-Carlo series engine**, the framework delivers **fair-value odds**, **credible intervals**, and **entropy-controlled prior blending** in real time.
- The resulting system behaves like a true **market-making core**: calibrated, adaptive to new evidence, and robust to early-round noise.
- Empirically, it produces **smooth, defensible prices** that align with microstructure constraints of thin prediction markets while preserving the interpretability demanded by risk teams.
- The outcome is not merely a model it is a **deployable pricing architecture** that scales across maps, matches, and market conditions.
- This establishes a foundation for **systematic trading**, **liquidity provision**, and **automated risk management** in esports prediction markets, filling a methodological gap in both academia and industry.

THANK YOU!

Please ask if you'd like to see a live demonstration of this project!

PURDUE UNIVERSITY