Horse Race Outcome Modeling with PyMC-BART

Homa Aghababaei

May 2025

Objective

This project aims to predict race outcomes and estimate the probability of a horse finishing **first** using a Bayesian Additive Regression Tree (BART) model. We prepared both training and prediction datasets using historical XML records and raw BRISNET files.

1. Training Dataset Construction

1.1. Source and Parsing

Historical race data (Churchill Downs, May 2023) was collected from XML files:

- **Past Performance (PP)**: Contained individual horse records.
- **Race Results**: Contained true finish positions.

Each horse's record was parsed and merged with race results using normalized `RaceID` and `HorseName`.

1.2. Feature Engineering

We engineered features reflecting horse form, race suitability, and jockey/race strength:

- `AvgPastFinishPosition`, `WinRate`, `DistanceSuitability`
- `JockeyWinRate`, `FieldStrength`, `SurfaceWinRate`

`Surface` was one-hot encoded (`Surface_Dirt`, `Surface_Turf`), and `Distance` was normalized. Missing values were imputed appropriately.

1.3. Output

The cleaned dataset (`selected_features_dataset.csv`) included all features and the target variable `FinishPosition`, ready for supervised learning.

2. Prediction Dataset Preparation

2.1. Raw Input

A BRISNET file (`CDX0515.csv`) with no headers was mapped manually to extract:

- Basic info: `HorseName`, `Distance`, `Surface`, `RaceNumber`, `Jockey`
- Past performance columns (e.g., `PP1_FinishPosition`, `PP1_Surface`, etc.)

2.2. Processing and Features

The same feature engineering logic from the training set was applied:

- `AvgPastFinishPosition`, `WinRate`, `DistanceSuitability`
- Jockey metrics derived from prior finishes
- `FieldStrength` computed per `RaceID`

2.3. Output

The processed dataset (`CDX0515_processed_for_prediction.csv`) was standardized and aligned with training data for seamless inference.

3. Modeling with PyMC-BART

3.1. Model Structure

We trained a regression model using the following PyMC structure:

- $\mu = \texttt{BART}(X, Y)$: Mean predicted finish position.
- $\sigma \sim \texttt{HalfNormal}(1.0)$: Noise term.
- $y \sim \mathcal{N}(\mu, \sigma)$: Likelihood.

Sampling used MCMC with 4000 draws, 3000 tuning steps, 10 chains, and a target acceptance rate of 0.95.

3.2. Evaluation

On a hold-out test set:

- Accuracy of exact position (rounded): e.g., 16%.
- RMSE and MAE were computed.
- A scatter plot visualized true vs. predicted ranks.

4. First-Place Probability Estimation

4.1. Approach

To compute the probability of each horse finishing first:

1. Posterior samples were drawn for all horses.
2. Within each race, we identified the horse with the lowest predicted finish per draw.
3. Each horse's **first-place probability** is the proportion of draws in which it ranks first.

4.2. Output

A sorted table is created showing:

- `HorseName`, `RaceNumber`, `RaceID`
- `FirstPlaceProb` (e.g., 0.22 = 22% chance to win)

This output highlights top contenders for each race and enables ranking based on win probabilities.