Project Report: Bayesian BART Modeling for Horse Race Prediction Using PyMC

Shi Chen

Overview

This project implements a Bayesian Additive Regression Tree (BART) model using the pymc-bart module to predict outcomes of horse races based on historical performance data. The goal is to infer complex patterns from XML race data and quantify predictive uncertainty through Bayesian inference. The process includes extensive data extraction, merging, probabilistic model construction, and performance evaluation.

Detailed Process

1. Data Extraction and Feature Engineering
Two sets of XML files were processed:

- Past Performance Data (pastPerformanceData/): Each file contains details about horses, including attributes like HorseName, Trainer, WeightCarried, Odds, and more. The parse_past_performance function extracts this information into a structured DataFrame.

- Results Data (resultsData/): These files record actual outcomes, such as OfficialFinish, SpeedRating, and DollarOdds, parsed using the parse_results function.

Each horse's performance record was joined with the corresponding race result using RaceNumber and HorseName as keys. This combined dataset was then preprocessed to handle missing values and convert numeric fields.

2. Modeling with PyMC-BART
After data preparation, the following modeling strategy was applied:

- Target Variable: OfficialFinish (the horse's placement in the race).
- Predictor Variables: A selection of numerical features such as WeightCarried, Odds, and SpeedRating.

A PyMC model was constructed where:
- A BART model served as a non-parametric prior over the regression function.
- The likelihood assumed Gaussian noise around the true finish rank.

Example code:

```
with pm.Model() as model:
   X_shared = pm.Data("X", X)
   Y_shared = pm.Data("Y", Y)
   f = BART("f", X=X_shared, Y=Y_shared)
   sigma = pm.HalfNormal("sigma", sigma=1)
```

```
    y_obs = pm.Normal("y_obs", mu=f, sigma=sigma, observed=Y_shared)
    trace = pm.sample(1000, tune=1000, cores=2)
```

Inference and Uncertainty Quantification

- Posterior Sampling: Conducted using MCMC to obtain samples from the posterior distribution of predictions.
- Posterior Predictive Checks: Evaluated model fit and ensured that predicted outcomes match real observations.
- Prediction: The posterior mean of the finish position was used for ranking horses, while the standard deviation of the posterior captured uncertainty.

Results Interpretation

- The BART model identified key nonlinear effects—such as the influence of weight and odds—without requiring manual feature engineering.
- The predictive intervals from the posterior distribution provided a clear uncertainty estimate for each horse’s expected performance.
- Example: A horse predicted to finish 2nd had a posterior 95% interval spanning ranks 1 to 4, indicating moderate uncertainty despite high confidence.

Conclusion

This project demonstrates a full Bayesian modeling pipeline applied to real-world race data. The use of pymc-bart allowed automatic detection of interactions and nonlinearity, while maintaining interpretability through posterior uncertainty.

Future improvements could include:

- Encoding categorical variables (like Trainer) using embeddings or indicators.
- Modeling ordinal nature of the target explicitly.
- Including time or track-specific covariates.