Big Data Derby

Horse racing through statistics

William Deng, Sabrina Mei, Fenglin Wang, Leo Wang   Project Advisor: Ron Yurko

Research Question & Context

GOAL: Analyze the relationships between spatial-temporal data and the outcome of horse races

Case Context:
- Kaggle competition hosted by the New York Racing Association (NYRA) in the style of the NFL Big Data Bowl
- Observe spatial positioning of horses every 0.25 seconds across 2000 races
- Open-ended competition with the prospect of leveraging complex spatiotemporal data to gain insights about horse racing

Data

Our dataset contains 2000 races and 22 variables. We identify the useful variables and split them into 3 sections:

Spatial Temporal	Race Condition	Jockey Information
• Time frame (0.25 sec) • Horse ID • Raw longitude • Raw latitude	• Race ID • Race location • Race date • Type of race • Type of track	• Jockey name • Finish position • Weight carried • Betting odds

Using the variables: “time frame”, “raw longitude”, and “raw latitude”, we derived the following variables based on Haversine distance formula.

$$\text{haversin}\left(\frac{d}{r}\right) = \text{haversin}(\phi_2 - \phi_1) + \cos(\phi_1)\cos(\phi_2)\,\text{haversin}(\lambda_2 - \lambda_1)$$

Modeling Predictor Variables:

$\vec{V}$ Velocity (m/s) of each horse at each 0.25 sec interval

\# Nearest horse at each 0.25 sec interval

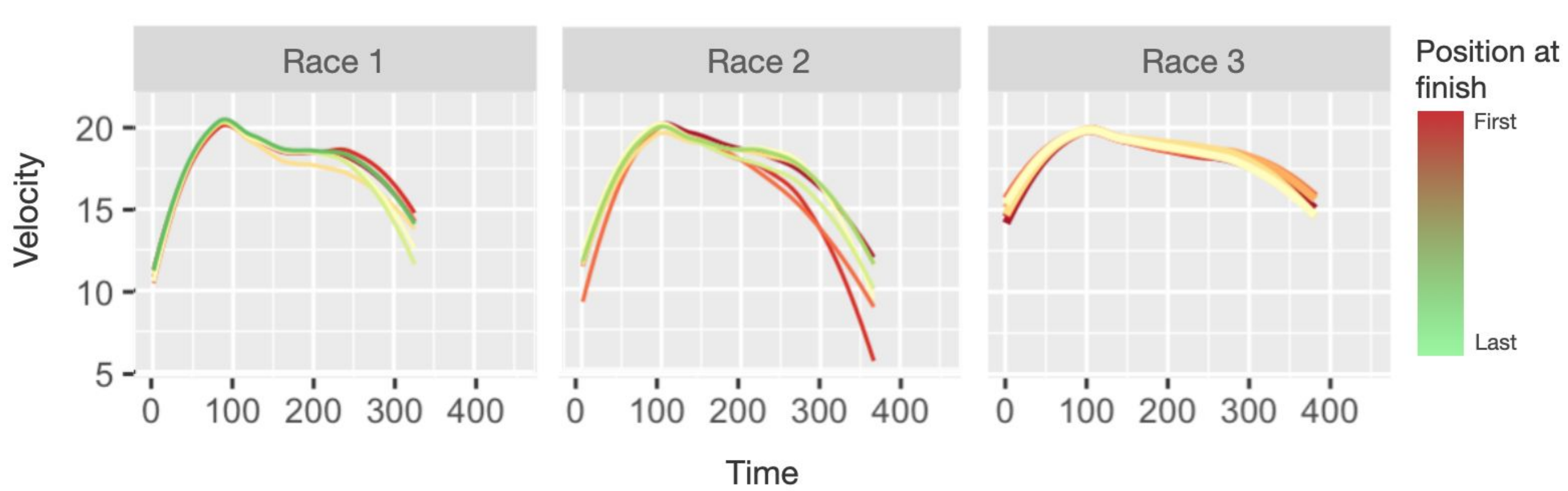
$\vec{D}$ Distance to the nearest horse (m) at each 0.25 sec interval

P Inferred probability of winning by odds of betting

Modeling Response Variable:

Probability of winning for each horse at time t

The graph below shows an example of the change of velocity over time of different horses in three races. The color of the curve shows the final position of the horse (red color shows the horse that ranked first).

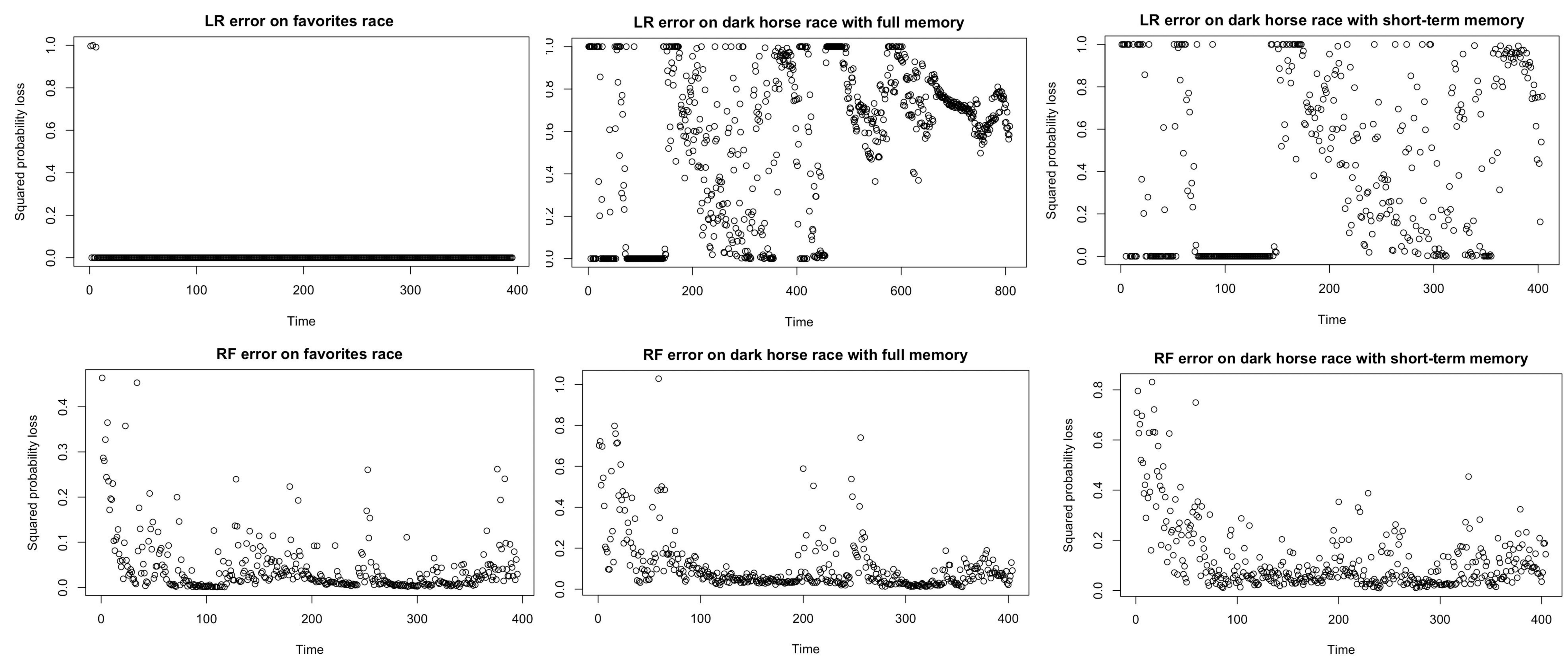


Methods & Analysis

Goal: Predicting a winning probability distribution for each horse at any time t.

To reduce the complexity during the training and improve the explainability of the model, we only feed the data from to the most current 20% of the time. In other words, we make our model independent of the more previous observations. This in fact filters out the noise and improve the accuracy (see graph for comparison)

- The loss function is defined as the squared difference between estimate distribution and true distribution.
  The metrics of optimization is based on the Brier Score:
  $$BS = \frac{1}{N}\sum_{t=1}^{N}(f_t - o_t)^2$$ (o and f are predicted probability and true outcome)
- The model of logistic regression (LR) is compared with probabilistic random forest (RF) with short-term memory on 2 types of races. We predict a distribution on every time snapshot of the race and sum the probability losses.
- Dark-horse race: The horse of best inferred prob of winning is not the winning horse.
  Sum of Squared loss for LR = 282.78, RF = 50.94, Brier Score of LR=0.6999, RF=0.1185.
- Favorites race: The horse of best inferred prob of winning is the winning horse.
  Sum of Squared loss for LR = 3.073,RF = 16.46, Brier Score of LR= 0.00756,RF=0.0452.



It turns out that the LR model tends to make a definite decision too early in the match (one horse's winning probability converge to 1) and assign almost all weights to the initial probability of winning, which is a result of overfitting on the normal race, since it happens more frequently. And for a “dark horse” race, our RF model appears to be more predictive and robust in all scenarios.Although our RF model tend to converge faster with full memory, the short term memory model reduces the maxima of possibly wrong probability predictions, thus increasing its robustness under different types of matches.

Conclusion

- RF with short term memory has significantly more accurate predictions than LR model in “Dark horse” scenarios (0.11-0.15 Brier Score vs. 0.6 or more) .
- RF with short term memory has less “accurate” predictions than LR model in favorites races (loss difference 0.05 on avg), Converges slower than the LR models.
- RF with short term memory predicts most accurately compared to all models in a tight match.
- RF model assigns 0-probability to horses too late in the game, which can be optimized to decrease prediction loss.

Discussion & Next Steps

Normalization of track and racing data
- Coordinates and track orientation were not uniform
- Oval shape or the track makes it hard to determine real-time pacing/positioning of horses
- Lack of standardized starting and ending points
- Accounting for DNFs throughout the races

Incorporation of other features
- Introduce GBDT or Time-series model that either takes better use of categorical features or captures the temporal structure of data.
- External data source for track information
- Weather data/track conditions
- Betting odds and applications to sports betting