The Korean Journal of Applied Statistics
2024, Vol. 37, No. 2, 239–253

DOI: https://doi.org/10.5351/KJAS.2024.37.2.239
Print ISSN1225-066X / Online ISSN 2383-5818

Horse race rank prediction using learning-to-rank approaches

Junhyoung Chung[a], Donguk Shin[a], Seyong Hwang[a], Gunwoong Park[1,a]

[a]Department of Statistics, Seoul National University

Abstract

This research applies both point-wise and pair-wise learning strategies within the learning-to-rank (LTR) framework to predict horse race rankings in Seoul. Specifically, for point-wise learning, we employ a linear model and random forest. In contrast, for pair-wise learning, we utilize tools such as RankNet, and LambdaMART (XGBoost Ranker, LightGBM Ranker, and CatBoost Ranker). Furthermore, to enhance predictions, race records are standardized based on race distance, and we integrate various datasets, including race information, jockey information, horse training records, and trainer information. Our results empirically demonstrate that pair-wise learning approaches that can reflect the order information between items generally outperform point-wise learning approaches. Notably, CatBoost Ranker is the top performer. Through Shapley value analysis, we identified that the important variables for CatBoost Ranker include the performance of a horse, its previous race records, the count of its starting trainings, the total number of starting trainings, and the instances of disease diagnoses for the horse.

Keywords: horse race, lambdamart, learning-to-rank, ranknet, rank prediction

1. 서론

Learning-to-rank (LTR) 기법은 정보 검색 시스템, 추천 시스템, 온라인 광고, 문서 분류 등 다양한 분야에서 순위 예측을 위해 사용되는 기계 학습 기법으로, 주어진 데이터를 학습하여 관찰되지 않은 쿼리 내 아이템의 점수를 예측한 뒤 이를 기반으로 순위를 예측한다 (Liu, 2009). 데이터는 순위 예측의 대상이 되는 아이템의 특징벡터(feature vector) 및 쿼리의 특징벡터가 설명변수로, 아이템의 점수(score)가 반응변수로 구성된다. 여기서 점수로는 아이템의 순위 또는 평점이 사용될 수 있다.

점수를 정확히 예측하는 것이 목표인 기존의 회귀 모형과는 달리, LTR 기법은 적절한 손실함수를 정의하여 아이템 간의 상대적인 순위를 학습할 수 있다는 측면에서 의의가 있다. 예를 들어 정보 검색의 경우, 관련순으로 웹페이지를 정렬하는 데에 있어 중요한 것은 정확한 점수보다도 웹페이지 간 점수의 대소 관계이다. 이와 유사하게 경주경기에서도 실제 경주기록보다 순위를 예측하는 것이 더 중요한 문제일 수 있는데, 이 경우 쿼리와 아이템이 각각 경기와 선수에 대응되어 설명변수로는 선수 정보 및 경기 정보, 반응변수로는 선수의 경주기록을 사용함으로써 LTR 기법을 통한 접근이 가능하다 (Kholkine 등, 2021).

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (NRF-2021R1C1C1004562 and RS-2023-00218231). Additionally, this work was supported by the New Faculty Startup Fund from Seoul National University.

[1]Corresponding author: Department of Statistics, Seoul National University, 1 Gwanak-ro, Gwanak-Gu, Seoul 08826, Korea. E-mail: gwpark23@snu.ac.kr

Published 30 April 2024 / journal homepage: http://kjas.or.kr

LTR 기법은 크게 point-wise, pair-wise, list-wise 세 가지로 나뉜다. Point-wise learning은 각 아이템의 점수를 개별적으로 학습하는 방식으로 회귀, 분류, 순서형 회귀 등 다양한 방법들을 유연하게 적용 가능하지만, 개별 아이템의 점수를 예측하는 것에만 집중하고 쿼리 내 아이템 간의 순위는 학습에서 고려하지 못한다는 문제점이 있다. List-wise learning은 쿼리 내 아이템 사이의 가능한 순서를 모두 고려하여 학습하는 방식으로 순위 예측에서 가장 직관적인 방법이지만, 모든 경우의 수를 계산하며 최적화하는 방향으로 확장할 경우에는 적합에 매우 긴 시간이 소요된다는 한계가 존재한다. 예를 들어, 본 연구에서 순위를 예측하고자 하는 서울 경마경기는 대체로 한 경기에 10마리 이상의 경주마가 참가하기 때문에 모든 가능한 순서(10! = 3628800)를 고려할 경우 방대한 계산량으로 인해 사용하기 어렵다. 또한 쿼리 내에 이상치나 결측치 등의 문제로 인해 모든 데이터를 관찰하기 어려운 경우에도 list-wise learning을 활용하기 어렵다. Pair-wise learning은 임의의 두 아이템을 쌍(pair)으로 묶어 점수를 비교하며 순위를 학습하는 방법으로, point-wise learning과 달리 아이템 간의 상대적인 순서를 학습과정에서 고려할 수 있다. 또한 list-wise learning에 비해 적은 계산량이 요구되고, 이상치 및 결측치의 문제로부터 비교적 자유롭다는 장점이 있다.

본 연구에서는 point-wise learning에 해당하는 선형 회귀와 랜덤 포레스트, pair-wise learning에 해당하는 RankNet (Burges 등, 2005)와 LambdaMART (XGBoost Ranker, LightGBM Ranker, CatBoost Ranker) (Burges, 2010)를 활용하여 서울 경주마 순위 예측을 수행하였다. 그 결과 LambdaMART 모형이 point-wise learning에 비해 유의미하게 높은 예측 성능을 보여주었고, 특히 CatBoost Ranker는 여섯 개의 모형 중 가장 뛰어난 예측력을 보여주었다. 이러한 결과는 LambdaMART가 아이템 간의 상대적인 순위를 더 정밀하게 학습한다는 것을 의미하며, 이를 통해 아이템 간 순서 정보를 반영하는 pair-wise learning의 장점을 확인할 수 있었다.

또한 서울 경주마 예측을 수행한 Choe 등 (2015)에서 간과되었던 마필 출발훈련 정보와 마필 진료기록 등의 중요 데이터도 분석에 포함하였으며, 섀플리 값(Shapley value)을 활용하여 변수 중요도 및 각 변수별로 예측치에 주는 영향을 확인함으로써 모형의 해석력을 확보하였다. 이러한 종합적인 분석은 경주마 순위 예측 분야에서 개선된 접근법을 제시하는 동시에 기존 연구의 한계를 극복하는 데 중요한 발판을 제공했다고 할 수 있다.

다음 2장에서는 분석에 활용한 point-wise learning과 pair-wise learning 각각에 대해 더 구체적으로 설명하고 그에 해당하는 모형들을 제시한다. 3장에서는 서울 경마 데이터를 소개하고 반응변수를 포함한 주요 변수들 간의 상관관계를 파악하며, 4장에서는 모형 학습 과정과 그 결과를 설명한다. 5장에서는 결론과 추후 연구 방향을 제시한다.

2. 모형

각 쿼리 $q \in \{1, 2, \ldots, m\}$에 대해 설명변수 $\mathbf{X}_q = [\mathbf{x}_q^1, \mathbf{x}_q^2, \ldots, \mathbf{x}_q^{n_q}]^T \in \mathbb{R}^{n_q \times p}$ 와 반응변수인 점수 $\mathbf{y}_q = [y_q^1, y_q^2, \ldots, y_q^{n_q}]^T \in \mathbb{R}^{n_q}$ 가 있을 때 데이터 $(\mathbf{X}, \mathbf{y})$는 다음과 같이 표현될 수 있다.

$$\mathbf{X} = \left[\mathbf{X}_1^T, \mathbf{X}_2^T, \ldots, \mathbf{X}_m^T\right]^T \in \mathbb{R}^{N \times p}, \quad \mathbf{y} = \left[\mathbf{y}_1^T, \mathbf{y}_2^T, \ldots, \mathbf{y}_m^T\right]^T \in \mathbb{R}^N.$$

이때 p는 설명변수의 수, n_q는 q 번째 쿼리에 속한 아이템의 개수이며, $N = \sum_{q=1}^m n_q$은 전체 데이터의 개수이다.

LTR 기법은 적절한 손실함수 $\mathcal{L}$을 정의하여 이를 최소화하는 예측 함수 $\hat{f} : \mathbb{R}^p \to \mathbb{R}$를 찾는 것을 목표로 하며, 예측 함수는 아이템의 설명변수를 입력값으로 받아 점수의 예측값을 출력한다. 편의상 임의의 함수 $f : \mathbb{R}^p \to \mathbb{R}$에 대해 $f(\mathbf{X_q}) = [f(\mathbf{x}_q^1), f(\mathbf{x}_q^2), \ldots, f(\mathbf{x}_q^{n_q})]^T$, $f(\mathbf{X}) = [f(\mathbf{X}_1)^T, f(\mathbf{X}_2)^T, \ldots, f(\mathbf{X}_m)^T]^T$로 간략히 표현할 때, 예측 함수 $\hat{f}$와 q 번째 쿼리에 속한 i 번째 아이템의 예측 점수는 아래의 식을 통해 구할 수 있다.

$$\hat{f} = \arg\min_f \ \mathcal{L}\left(f(\mathbf{X}), \mathbf{y}\right), \quad \hat{y}_q^i = \hat{f}\left(\mathbf{x}_q^i\right).$$

q 번째 쿼리 내 아이템 간 실제 순위를 $\pi_q = [\pi_q^1, \ldots, \pi_q^{n_q}]^T$, 예측 순위를 $\hat{\pi}_q = [\hat{\pi}_q^1, \ldots, \hat{\pi}_q^{n_q}]^T$라 할 때, $\hat{\pi}_q$는 $\hat{\mathbf{y}}_q = [\hat{y}_q^1, \ldots, \hat{y}_q^{n_q}]^T$ 에 기반하여 계산한다. 반응변수가 평점과 같이 높을수록 순위가 높아지는 경우에는 $\hat{\pi}_q^i = \sum_{j=1}^{n_q} I(\hat{y}_q^j \geq \hat{y}_q^i)$로 계산하며, 반대로 반응변수가 경주기록처럼 낮을수록 순위가 높아질 때에는 $\hat{\pi}_q^i = \sum_{j=1}^{n_q} I(\hat{y}_q^j \leq \hat{y}_q^i)$로 예측한다. 해당 식은 $\hat{\mathbf{y}}_q$ 내에서 동점이 없다는 전제 하에 정의되었으며, 만약 동점이 존재한다면 평균순위, 최소순위 등의 방식을 통해 처리할 수 있다.

앞서 언급한 세 가지 학습방식의 차이는 손실함수의 형태에서 기인하며, 이후부터는 분석 과정에서 활용한 point-wise learning과 pair-wise learning 각각에 대한 구체적인 개념과 모형 몇 가지를 소개한다. 전체 데이터에서 분할되는 훈련 데이터와 평가 데이터는 각각 $(\mathbf{X}_{train}, \mathbf{y}_{train})$, $(\mathbf{X}_{test}, \mathbf{y}_{test})$로 표현한다.

2.1. Point-wise learning

Point-wise learning은 각 아이템의 점수를 순위와 상관없이 개별적으로 예측하는 방식이다. 따라서 point-wise learning은 LTR 알고리즘 중 가장 간단한 형태의 손실함수를 가지고 있으며 주로 평균 제곱 오차(mean squared error; MSE)를 사용한다. Point-wise learning은 개별 아이템을 독립적으로 예측하는 구조로 인해 데이터 처리가 효율적이다. 특히 대규모 데이터셋에서도 학습 속도가 빠르고 다양한 방법들을 유연하게 적용 가능하여 다양한 선행연구에서 활용되어 왔다. 예를 들어 Choe 등 (2015)과 Park 등 (2017)은 회귀 및 분류 알고리즘을 활용하여 각각 경마경기와 경륜경기의 순위를 예측하는 연구를 진행하였다.

하지만 point-wise learning은 다음과 같은 한계점을 가지고 있다. 우선 아이템의 순서보다는 점수 예측에만 집중하는 손실 함수를 사용하고 있기 때문에 순위 예측 문제에 적합하지 않을 수 있다. 이에 따라 쿼리 내 아이템 간의 순위관계를 학습에 고려하지 않게 되어 정확한 순위 예측이 어려울 수 있다. 또한 분류 알고리즘을 사용하기 위해 특정 순위의 범위에 속하는 아이템과 그렇지 않은 아이템으로 나누어 이진분류를 수행할 경우에는 데이터가 편향되어 불균형 문제가 발생할 수 있다. 순서형 회귀를 사용하게 된다면 이진분류에서 자연스럽게 문제를 확장할 수 있지만, 반응변수의 등분산 가정에서 자유롭지 못하다는 문제점이 있고 이를 해결하기 위해서는 반응변수의 분산에 영향을 줄 수 있는 설명변수의 탐색이 추가적으로 필요하다 (Christensen, 2018). 그리고 쿼리 내 아이템의 순위는 서로 독립이 아님에도 이를 해소하기 위한 별도의 가정이 없다는 한계점이 있다.

선형 회귀는 가장 대표적인 point-wise learning 모형으로 반응변수와 설명변수 사이의 관계를 선형으로 가정한 뒤, 학습을 통해 적절한 선형계수를 찾아 예측을 수행한다. 선형 회귀는 평균 제곱 오차를 손실함수로 사용할 경우, 이를 최소화하는 선형계수를 닫힌 형태(closed form)로 구해 예측 함수에 적용할 수 있다는 점에서 효율적이다. 훈련 데이터$(\mathbf{X}_{train}, \mathbf{y}_{train})$를 통해 예측 함수 $\hat{f}$의 선형계수 $\hat{\beta}$를 계산하고, 평가 데이터의 설명변수 $\mathbf{X}_{test}$를 이용하여 점수 $\mathbf{y}_{test}$를 예측하는 과정을 다음과 같이 식으로 나타낼 수 있다.

$$\hat{\beta} = \underset{\beta \in \mathbb{R}^p : f(\mathbf{X}) = \mathbf{X}\beta}{\arg\min} \left(\mathbf{y}_{train} - f(\mathbf{X}_{train})\right)^T \left(\mathbf{y}_{train} - f(\mathbf{X}_{train})\right). \tag{2.1}$$

$$\hat{f}(\mathbf{X}_{test}) = \mathbf{X}_{test}\hat{\beta} = \mathbf{X}_{test}\left(\mathbf{X}_{train}^T \mathbf{X}_{train}\right)^{-1} \mathbf{X}_{train}^T \mathbf{y}_{train}.$$

선형 회귀는 다른 모델들에 비해 해석력이 뛰어나며, 통계적 추론 및 변수선택법을 이용하여 예측에 중요한 변수를 찾을 수 있는 장점이 있다. 또한 릿지(Ridge), 라쏘(Lasso) 등의 벌점화 방법을 이용하여 고차원 데이터 분석에도 활용 가능하다. 하지만 반응변수와 설명변수 사이의 관계가 비선형일 경우 예측력이 떨어질 수 있으며, 선형계수를 추정하는 과정에서 이상점의 영향을 크게 받을 수 있다는 한계점도 있다.

Point-wise learning 중 또 하나의 대표적인 모형인 랜덤 포레스트(random forest) (Breiman, 2001)는 회귀 및 분류 분석 등에 폭넓게 활용되는 배깅(bagging)을 기반으로 한다. 구체적으로는 일반적인 배깅과 마찬가지로 부트스트랩(bootstrap)을 통해 훈련 데이터를 여러 번 재표집하고 의사 결정 트리(decision tree)를 학습하는 과정을 반복하여 다수의 의사 결정 트리로부터 예측값을 결정한다. 하지만 랜덤 포레스트는 일반적인 배깅과

Table 1: Description of the data

Horse race data			
Race	Jockey	Horse	Trainer
Result	Total record	Total record	Total record
Track		Detailed information	Detailed information
Weather		Diagnostic record	
		Start training record	

달리 각각의 의사 결정 트리를 학습하는 과정에서 일부 변수만을 활용한다는 점에서 차이가 있다. 즉, 임의의 $b \in \{1, 2, \ldots, B\}$에 대하여, 훈련 데이터를 복원 추출 n번을 통해 재표집한 데이터 $(\mathbf{X}_{train}^{(b)}, \mathbf{y}_{train}^{(b)})$를 통해 의사 결정 트리 f_b를 학습한 뒤, $\mathbf{X}_{test}$에 대해 다음과 같이 예측값을 구한다.

$$\hat{f}_b = \underset{f:\text{Tree}}{\arg\min} \left(\mathbf{y}_{train}^{(b)} - f\left(\mathbf{X}_{train}^{(b)}\right)\right)^T \left(\mathbf{y}_{train}^{(b)} - f\left(\mathbf{X}_{train}^{(b)}\right)\right). \tag{2.2}$$

$$\hat{f}\left(\mathbf{X}_{test}\right) = \frac{1}{B}\sum_{b=1}^{B} \hat{f}_b\left(\mathbf{X}_{test}\right).$$

랜덤 포레스트는 의사 결정 트리를 사용하기 때문에 설명변수 간의 상호작용 및 비선형성을 다루기 용이하여 예측력이 우수하다는 것이 장점이다. 하지만 선형 회귀에 비해 해석력이 떨어지고, 크기가 큰 데이터셋의 경우에는 학습에 많은 시간이 소요될 수 있다는 단점이 있다.

2.2. Pair-wise learning

Pair-wise learning은 임의의 두 아이템에 대해 점수를 비교하여 순위가 더 높은 아이템의 점수를 크게 예측하는 방식이다. 따라서 point-wise learning과 달리 아이템 간의 상대적인 순서를 학습과정에서 고려할 수 있다는 의의가 있다 (Liu, 2009). Pair-wise learning에서의 손실함수는 아이템 간 순위를 반영하며 주로 pair-wise loss (Burges 등, 2005)를 사용하고, 그 밖에도 YetiRank, ApproxNDCG 등 다양한 손실함수를 사용할 수 있다.

Pair-wise learning 관련 모형으로는 RankNet (Burges 등, 2005), LambdaRank (Burges 등, 2006), LambdaMART (Burges, 2010) 등이 있으며, 특히 LambdaMART 모형은 Yahoo! Learning-to-Rank 챌린지에서 우승했을뿐만 아니라 전반적으로 좋은 성능을 나타낸다고 알려져 있다 (Burges, 2010; Li 등, 2019; Hu 등, 2019). 이에 따라 Kholkine 등 (2021)과 Soldaini와 Goharian (2017)은 LambdaMART를 활용하여 각각 사이클 경주의 순위 예측과 건강 관련 상품을 검색하는 소비자들에게 보여줄 관련 웹페이지 노출 순서를 정렬하는 연구를 진행하였다.

RankNet은 pair-wise learning 방식에서 가장 많이 사용되는 초기에 개발된 인공신경망 모형으로, 다음 손실함수를 사용한다. 표현의 편의를 위해 모든 q에 대해 $\pi_q = [1, \ldots, n_q]^T$로 가정한다.

$$\hat{f} = \underset{f}{\arg\min} \sum_{q} \sum_{1 \le i < j \le n_q} \log\left(1 + \exp\left(-\sigma\left(f\left(\mathbf{x}_q^i\right) - f\left(\mathbf{x}_q^j\right)\right)\right)\right), \text{ where } \sigma > 0. \tag{2.3}$$

식 (2.3)에는 반응변수가 직접적으로 표현되어 있지는 않지만, 반응변수의 순위관계를 학습에서 활용하고 있다.

RankNet은 인공신경망을 활용하기 때문에 반응변수와 설명변수 간의 복잡한 비선형 관계를 학습하는 데에 유리하다. 하지만 계산이 복잡하여 학습을 하는 데에 많은 시간이 소요되고, 과적합 우려가 있으며 이미지나 텍스트와 같은 비정형 데이터와 달리 정형 데이터의 경우에는 인공신경망 모형이 트리 기반 모형보다 상대적으로 예측력이 뒤떨어진다고 알려져 있다 (Grinsztajn 등, 2022).

Table 2: Description of the variables

	Variable	Description	Type
Input	waterPercent	The humidity(%) at the time of the race	Numerical
	rcDist	The distance(m) that the racehorse must run during the race	
	dusu	# of horses participating in a single race	
	chulNo	The number assigned to each racehorse for the race, representing the lane or position the horse must run in	
	age	The age(years) of the racehorse	
	wgBudam	The additional weight(kg) deliberately imposed on the racehorse	
	wgHr	The weight(kg) of the racehorse	
	wgHrDelta	The change in weight(kg) of the racehorse compared to the previous race	
	str_rcTimeLag1	The standardized race time of the racehorse recorded in the previous race	
	startTrainingCntT	# of training sessions the racehorse has received for starts	
	startTrainingCnt_diff	# of training sessions for starts the racehorse received between the previous race and the current race	
	ord1CntT_hr	# of times the racehorse has recorded a first-place	
	ord2CntT_hr	# of times the racehorse has recorded a second-place	
	ord3CntT_hr	# of times the racehorse has recorded a third-place	
	rcCntT_hr	The total number of races in which the racehorse has participated	
	performance_hr	$\frac{\sum_{n=1}^{5}(6-n)\times \text{ord } n \text{ CntT\_hr}}{15\times \text{rcCntT\_hr}}$	
	clinicCntT_hr	# of times the racehorse has been diagnosed with disease	
	ord1CntT_jk	# of times the jockey has recorded a first-place	
	ord2CntT_jk	# of times the jockey has recorded a second-place	
	ord3CntT_jk	# of times the jockey has recorded a third-place	
	rcCntT_jk	The total number of races in which the jockey has participated	
	performance_jk	$\frac{\sum_{n=1}^{5}(6-n)\times \text{ord } n \text{ CntT\_jk}}{15\times \text{rcCntT\_jk}}$	
	trCareer	The years of experience of the trainer	
	ord1CntT_tr	# of times the trained racehorses have achieved a first-place	
	ord2CntT_tr	# of times the trained racehorses have achieved a second-place	
	ord3CntT_tr	# of times the trained racehorses have achieved a third-place	
	rcCntT_tr	The total number of races in which the trained racehorses have participated	
	performance_tr	$\frac{\sum_{n=1}^{5}(6-n)\times \text{ord } n \text{ CntT\_tr}}{15\times \text{rcCntT\_tr}}$	
	weather	The weather at the time the race was hel	Categorical
	track	The track condition at the time the race was held	
	sex	The sex of racehorse	
	name	The place of birth of the racehorse	
	rank	The grade of class of the racehorse	
Output	str_rcTime	The standardized race time of the racehorse	Numerical
	ord	The race ranking of the racehorse	

LambdaRank는 식 (2.3)의 손실함수를 직접 계산할 필요 없이 미분값만으로도 충분히 학습할 수 있다는 발견을 통해 RankNet을 한단계 발전시킨 모형으로, 식 (2.3)에 임의의 변화분 $|\Delta Z_{ij}|$을 곱한 형태의 손실함수를 사용한다 (Burges, 2010).

$$\hat{f} = \arg\min_{f} \sum_{q} \sum_{1\le i<j\le n_q} |\Delta Z_{ij}| \log\left(1+\exp\left(-\sigma\left(f\left(\mathbf{x}_q^i\right)-f\left(\mathbf{x}_q^j\right)\right)\right)\right), \text{ where } \sigma>0. \tag{2.4}$$

Table 3: The number of observations and games by distance

Distance (m)	1000	1100	1200	1300	1400	1600	1700	1800	1900	2000	2300
# of observations	8,452	420	21,835	18,382	14,509	434	8,821	7,307	1,077	1,298	146
# of games	1,111	49	2,040	1,691	1,267	45	810	651	94	124	12

이 때 $|\Delta Z_{ij}| = 1$로 두면 식 (2.3)과 동일한 것을 확인할 수 있다. 만약 $|\Delta Z_{ij}|$로 NDCG (normalized discounted cumulative gain)의 변화분 (Burges, 2010)을 사용한다면 순위 정보뿐만 아니라 반응변수까지 손실함수에 포함할 수 있게 되는데, NDCG에 대한 수식적인 설명은 4.2절에 기술되어 있다. NDCG의 변화분은 쿼리 내 아이템 i와 j의 예측 순위가 서로 바뀌면서 나타나는 NDCG 변화량의 크기를 말하며, NDCG의 변화분을 사용하게 되면 NDCG의 하한을 최적화하는 것이 알려져 있다 (Wang 등, 2018).

LambdaMART는 LambdaRank에 그래디언트 부스팅 회귀 트리(MART)를 결합한 모형으로, 순위 예측 모형 중 상대적으로 가장 좋은 성능을 가진다고 알려져 있으며, 특히 비선형 관계와 이상치에 대한 강한 대응 능력이 있다는 강점이 있다. 대표적인 MART 알고리즘으로는 XGBoost (Chen과 Guestrin, 2016), LightGBM (Ke 등, 2017), CatBoost (Prokhorenkova 등, 2018)가 있고, 이들 중 하나를 LambdaRank에 결합하여 LambdaMART로 활용하게 된다. 하지만 대부분의 LambdaMART 기반 선행 연구는 XGBoost를 주로 활용하였으며, LightGBM이나 CatBoost의 활용 사례는 상대적으로 더 적다. LightGBM은 level-wise하게 트리를 학습하는 XGBoost, CatBoost와 달리 leaf-wise하게 트리를 학습한다는 차이가 있다. 또한 CatBoost는 범주형 변수를 처리하는 데에 강점이 있는데, 원-핫 인코딩뿐만 아니라 ordered TS (target statistic) 및 feature combination 등의 방법을 다양하게 적용함으로써 범주형 변수를 더 효율적으로 처리하고 있다 (Prokhorenkova 등, 2018). 이를 통해 설명변수의 차원이 과도하게 늘어나는 문제를 방지함과 동시에 더욱 풍부한 정보를 활용할 수 있게 된다. 따라서 본 연구에서는 XGBoost, LightGBM, CatBoost 기반의 LambdaMART를 모두 분석에 사용하였으며, 각 LambdaMART 알고리즘은 모두 파이썬(Python)에서 xgboost, lightgbm, catboost 라이브러리를 통해 구현되어 있다.

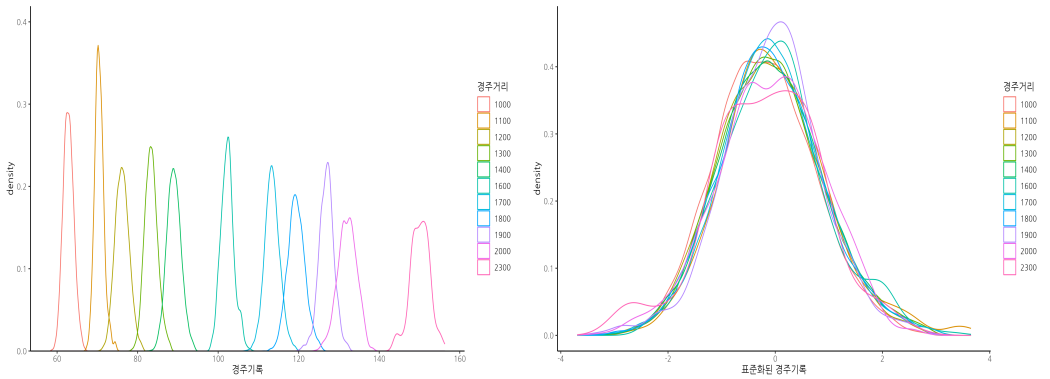
3. 데이터

3.1. 데이터 소개

본 연구에서 사용한 데이터는 한국마사회(www.kra.co.kr)에서 제공하는 경기 정보(경주성적, 경주로, 날씨), 기수 정보(통산기록), 마필 정보(통산기록, 상세정보, 진료기록, 출발훈련), 조교사 정보(통산기록, 상세정보)로 구성되어 있으며, Table 1에는 사용한 10개 데이터셋이 정리되어 있다. 병합하는 과정에서 결측치나 실격 등의 문제로 인해 비정상적으로 측정된 경주기록은 모두 제거하였고, 본 연구에서 설명변수로 활용하고 있는 직전 경주기록이 존재하지 않는 신마 역시 분석대상에서 제외하였다. 그 결과 2013년 1월부터 2023년 7월까지 총 82,681개의 자료($N = 82681$), 7,894개의 경기($m = 7894$)를 분석에 사용하였다. 이 때 설명변수는 총 33개($p = 33$)로, 5개의 범주형 변수와 28개의 수치형 변수로 이루어져 있다. 각 변수의 설명과 자료형은 Table 2에 정리되어 있다.

3.2. 탐색적 자료분석

Choe 등 (2015)은 경주기록이 경주거리에 따라 상이하다는 문제점을 해결하기 위해 경주거리별로 모형을 달리 적합하였다. 하지만, 이러한 방법은 자료가 적은 경주거리의 경우에 모형 적합이 어려울 수 있다는 단점이 있다. Table 3는 경주거리별 관측치의 개수와 경주경기의 개수를 나타내고 있는데, 해당 표를 보면 2,300m의 경우에는 경기 수가 12개로 1,200m에 해당하는 경기 수가 2,040개인 것과 비교했을 때 상당히 적은 수인



(a) Original race record (b) Standardized race record

Figure 1: *Distributions of race records by distance: Original race record (left) and standardized race record (right).*

것을 확인할 수 있다. 이러한 데이터의 불균형은 모형 적합에 방해될 수 있다. 따라서, 본 연구는 경주거리별로 표준화한 경주기록을 하나의 모형으로 예측함으로써 충분한 관측치를 확보하였다.

경주거리 d에 해당하는 쿼리들의 집합 S_d를 정의할 때, 임의의 $q \in S_d$에 대해 표준화된 경주기록 y_q^j는 아래와 같다.

$$y_q^j = \frac{\tilde{y}_q^j - \bar{\tilde{y}}_d}{(1/(N_d-1))\sum_{l\in S_d}\sum_{i=1}^{n_l}\left(\tilde{y}_l^i - \bar{\tilde{y}}_d\right)^2}, \text{ where } N_d = \sum_{l\in S_d} n_l, \quad \bar{\tilde{y}}_d = \frac{1}{N_d}\sum_{l\in S_d}\sum_{i=1}^{n_l}\tilde{y}_l^i,$$

여기서 $\tilde{y}_q^j$는 표준화 이전의 경주기록을 말하며, 별다른 언급이 없는 한 이후부터 반응변수는 모두 경주거리별로 표준화된 경주기록이라고 가정한다.

Figure 1은 경주기록이 경주거리에 따라 어떻게 변하는지 표준화 전과 후의 분포를 보여주고 있다. Figure 1(a)를 보면 경주거리가 길어질수록 경주기록의 분포가 오른쪽으로 이동하는 것을 볼 수 있다. 반면에 Figure 1(b)에서는 경주거리에 따라 표준화된 경주기록의 분포를 확인할 때, 경주거리마다 큰 차이가 없는 것을 확인할 수 있다. 즉, 표준화를 통해 경주거리가 경주기록에 미치는 효과를 제거했다고 볼 수 있다.

Figure 2는 Choe 등 (2015)에서 다루지 않았던 마필 출발훈련 정보, 마필 진료기록 정보 등을 추가적으로 활용하여 추출한 새로운 변수들과 반응변수인 표준화된 경주기록 사이의 상관관계를 나타내고 있다. 표본상관계수의 t-검정을 통해 유의미하지 않은 상관관계로 파악된 경우(p-value ≥ 0.01)에는 공백으로 표시되어 있다.

표준화된 직전 경주기록과 표준화된 경주기록은 높은 양의 상관관계를 보여주고 있다. 이는 이전 경기에서 좋은 경주기록을 기록한 마필이 역시 좋은 경주기록을 가져갈 확률이 높다는 것을 시사한다. 경주마의 통산 1위횟수부터 5위횟수까지를 가중평균하여 생성한 경주마의 성적 변수는 Pudaruth 등 (2013)에서 제시한 변수로, 경주마의 능력치를 나타낸다고도 해석 가능하다. 해당 변수는 표준화된 경주기록과 음의 상관관계를 보이고 있는데, 이전까지 좋은 성적을 보여준 경주마가 더 짧은 경주기록을 달성할 수 있음을 의미한다는 점에서 자연스러운 결과이다. 같은 방법으로 생성한 조교사의 성적과 기수의 성적 변수에서도 표준화된 경주기록과 음의 상관관계를 보이는 현상이 동일하게 나타나고 있다. 이는 조교사와 기수 역시 경주경기에 충분히 영향을 미칠 수 있다는 것을 의미한다.

경주마의 출발훈련 횟수와 누적 출발훈련 횟수는 표준화된 경주기록과 음의 상관관계를 가지고 있다. 두 변수는 모두 경주마가 좋은 경주기록을 달성하기 위해 연습한 정도를 나타내며, 더 높은 훈련량을 소화한 마

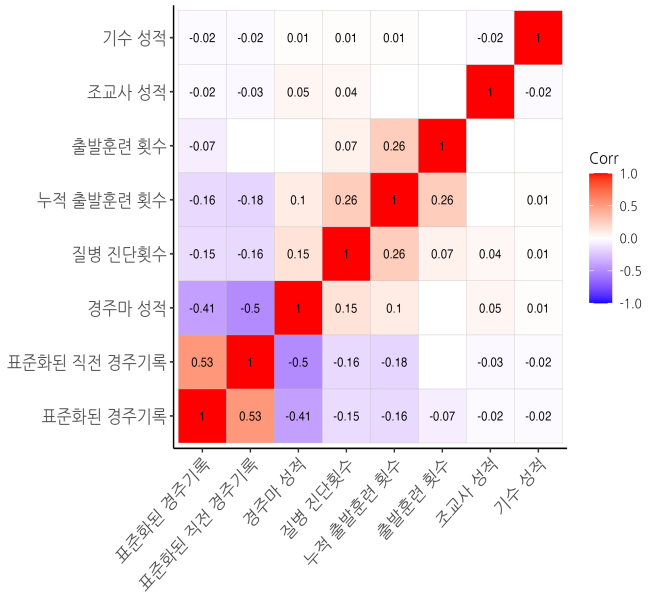


Figure 2: *Correlation plot for the standardized race record and the newly added variables.*

필이 통상적으로 좋은 경주기록을 가져간다고 해석할 수 있다. 질병 진단횟수는 표준화된 경주기록과 음의 상관관계를 보이고 있다는 점에서 다소 직관적이지 않지만, 해당 변수를 마필이 얼마나 잘 관리받고 있는지에 대한 척도로 이해한다면 납득 가능한 결과이다.

4. 결과

4.1. 모형 학습

본 연구에서는 경주 순위를 예측하기 위해 point-wise learning에 해당하는 선형 회귀(2.1)와 랜덤 포레스트(2.2), pair-wise learning에 해당하는 RankNet(2.3), LambdaMART(2.4) 중 XGBoost Ranker, LightGBM Ranker, CatBoost Ranker까지 총 6개의 모형을 활용하였다. 반응변수로는 표준화된 경주기록을 사용하였으며, 설명변수 중 범주형 변수의 경우 CatBoost Ranker는 자체적인 임베딩 기능을 활용하였고, 나머지 모형은 원-핫 인코딩을 통해 임베딩하였다. 특히, point-wise learning 방법으로 사용된 선형 회귀와 랜덤 포레스트는 Choe 등 (2015)이 제시한 경마 순위 예측 방법이다.

4.2. 모형 평가방법

훈련 및 평가 데이터를 무작위로 분할하게 되면, 각 경기의 데이터가 섞이게 되어 LTR 학습에 부정적인 영향을 줄 수 있다. 본 연구에서는 이러한 문제를 방지하기 위해 한 경기의 데이터가 훈련 및 평가 데이터 사이에 중복되지 않도록 경기 기준으로 그룹화하여 분리하였다. 따라서 총 66,034개의 자료(6,315개의 경기)를 훈련 데이터로, 16,647개의 자료(1,579개의 경기)를 평가 데이터로 활용하였다. 각 모델에 대한 성능 평가는 데이터 분할을 동일한 비율로 랜덤하게 100번 반복한 후, 평가지표의 평균을 통해 비교하였다.

성능 평가는 k-승식(k-winrate) 우승확률, 스피어만 상관계수, 켄달의 타우, NDCG의 평가 지표를 사용하였다. 먼저 k-승식 우승확률은 경마경기에서 흔히 이루어지는 단승식(1-winrate), 복승식(2-winrate), 삼복승식

Table 4: Means and standard errors for each metric by models, with the best value for each metric highlighted in bold

Metric	Point-wise learning		Pair-wise learning			
	Linear regression	Random forest	RankNet	XGBoost Ranker	LightGBM Ranker	CatBoost Ranker
Single winrate	0.2744	0.2598	0.2600	0.2733	0.2803	**0.3049**
	(0.0117)	(0.0102)	(0.0124)	(0.0102)	(0.0110)	(0.0107)
Quinella winrate	0.1155	0.1070	0.1087	0.1142	0.1225	**0.1345**
	(0.0066)	(0.0074)	(0.0090)	(0.0069)	(0.0065)	(0.0075)
Trifecta winrate	0.0643	0.0579	0.0585	0.0625	0.0701	**0.0771**
	(0.0058)	(0.0062)	(0.0067)	(0.0060)	(0.0052)	(0.0059)
Spearman correlation	0.4080	0.3859	0.3844	0.4037	0.4284	**0.4474**
	(0.0077)	(0.0074)	(0.0157)	(0.0070)	(0.0071)	(0.0075)
Kendall's tau	0.3129	0.2951	0.2935	0.3083	0.2935	**0.3439**
	(0.0062)	(0.0058)	(0.0126)	(0.0058)	(0.0126)	(0.0061)
NDCG	0.7025	0.7001	0.6901	0.7002	0.6901	**0.7149**
	(0.0040)	(0.0043)	(0.0071)	(0.0042)	(0.0071)	(0.0042)

(3-winrate) 우승확률을 계산한다. 단승식은 1등으로 들어올 경주마 1마리를 적중하는 베팅, 복승식은 2등 안으로 들어올 경주마 2마리를 순서와 상관없이 적중하는 베팅, 삼복승식은 3등 안으로 들어올 경주마 3마리를 순서와 상관없이 적중하는 베팅이다. 이 때, k-승식(단승식, 복승식, 삼복승식) 우승확률은 아래와 같이 구한다.

$$S_{k-winrate} = \frac{1}{m}\sum_{q}\prod_{j:\pi_q^j \le k} I\left(\hat{\pi}_q^j \le k\right).$$

스피어만 상관계수는 데이터를 순위로 변환한 뒤 피어슨 상관계수와 동일한 방법으로 계산한다. 본 연구에서는 스피어만 상관계수를 경기별로 구한 뒤 평균을 내어 평가지표로 사용한다.

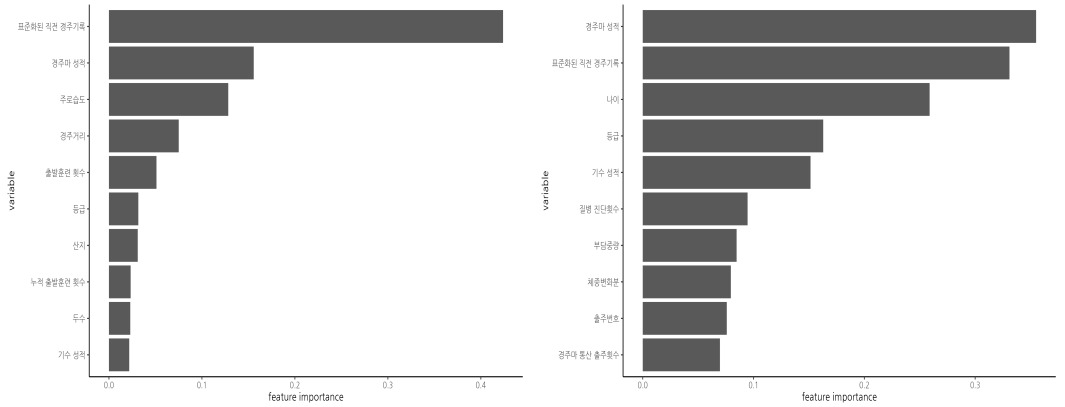
$$S_{spearman} = \frac{1}{m}\sum_{q} r_q, \text{ where } r_q = \frac{\sum_j \left(\pi_q^j - \bar{\pi}_q\right)\left(\hat{\pi}_q^j - \bar{\hat{\pi}}_q\right)}{\sqrt{\sum_h \left(\pi_q^h - \bar{\pi}_q\right)^2 \sum_l \left(\hat{\pi}_q^l - \bar{\hat{\pi}}_q\right)^2}}, \quad \bar{\pi}_q = \frac{1}{n_q}\sum_{j=1}^{n_q}\pi_q^j, \quad \bar{\hat{\pi}}_q = \frac{1}{n_q}\sum_{j=1}^{n_q}\hat{\pi}_q^j .$$

켄달의 타우는 이변량 데이터에서 한 변수가 증가(감소)하면 다른 변수도 증가(감소)하는 관계에 놓여있는지를 평가하는 측도이다. 본 연구에서는 경기별 켄달의 타우의 평균을 평가지표로 사용한다.

$$S_{kendall} = \frac{1}{m}\sum_{q}\tau_q, \text{ where } \tau_q = \frac{1}{\binom{n_q}{2}}\sum_{i<j}\left[I\left(\left(y_q^i - y_q^j\right)\left(\hat{y}_q^i - \hat{y}_q^j\right) > 0\right) - I\left(\left(y_q^i - y_q^j\right)\left(\hat{y}_q^i - \hat{y}_q^j\right) < 0\right)\right].$$

NDCG는 정보 검색 및 추천 시스템 평가에서 사용되는 지표로, 주로 정확하게 순위를 예측했는지 평가하기 위해 사용된다 (Järvelin과 Kekäläinen, 2017). NDCG를 계산하기 위해서는 우선 DCG (discounted cumulative gain)를 계산해야 하는데, DCG는 더 높은 순위를 정확하게 예측했을 때 더 높은 가중치가 부여되는 방식으로 계산된다. q 번째 경기의 DCG는 아래와 같다.

$$\mathrm{DCG}_q = \sum_j \frac{2^{-y_q^j} - 1}{\log\left(\hat{\pi}_q^j + 1\right)} .$$



(a) Random Forest (b) CatBoost Ranker

Figure 3: *Feature importance of Random Forest (left) and CatBoost Ranker (right).*

NDCG는 DCG에 해당 경기에서 가질 수 있는 DCG의 최댓값인 IDCG (ideal discounted cumulative gain), 즉 순위가 모두 정확하게 예측되었을 때의 DCG의 역수를 곱해주어 지표가 0에서 1사이의 값을 가질 수 있도록 한다. 본 연구는 경기별 NDCG를 구한 뒤 평균을 내어 평가지표로 사용한다.

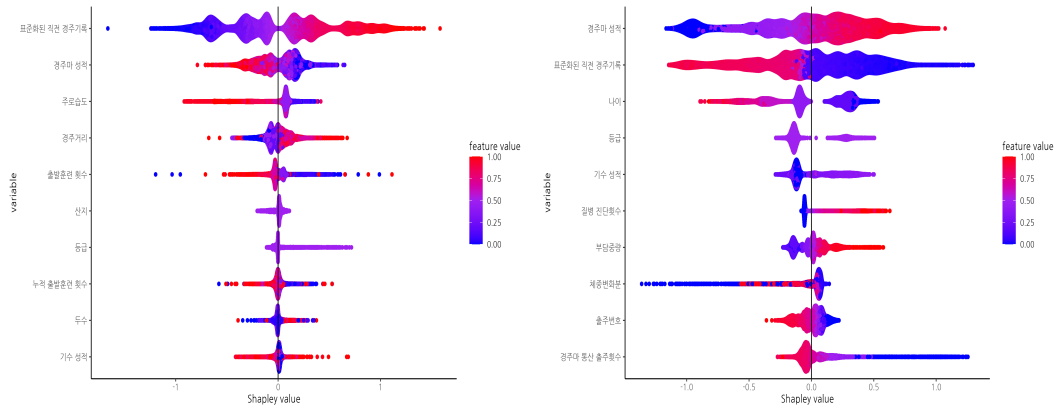
$$S_{NDCG} = \frac{1}{m}\sum_{q} \mathrm{NDCG}_q, \text{ where } \mathrm{NDCG}_q = \frac{\mathrm{DCG}_q}{\mathrm{IDCG}_q}.$$

단, $\mathrm{IDCG}_q = \sum_j (2^{-y_q^j} - 1)/\log(\pi_q^j + 1)$이다. 이 때, 반응변수로 사용하고 있는 표준화된 경주기록은 값이 작을수록 높은 순위를 가지게 되기 때문에 통상적인 DCG의 정의와는 다르게 분자의 지수부분에 음수를 취하고 있다.

4.3. 최종 결과

Table 4는 평가 데이터로부터 구한 모형별 성능 평가지표의 평균과 표준오차를 보여주고 있다. 우선 pair-wise learning 중에서도 LambdaMART에 속하는 XGBoost Ranker는 point-wise learning에 해당하는 랜덤 포레스트에 비해 전반적으로 우수한 성능을 보여주고 있고, LightGBM Ranker, CatBoost Ranker는 모든 평가지표에서 선형 회귀와 랜덤 포레스트를 앞서고 있다. 특히 CatBoost Ranker는 모든 평가지표에서 가장 뛰어난 성능을 보여주고 있다. 이는 앞서 설명한 이론적 성질에 부합하는 결과로 CatBoost Ranker는 pair-wise learning을 통해 아이템 간의 순위를 학습할 수 있을뿐만 아니라, 데이터에 범주형 변수들 (e.g., 경주마의 성별)이 포함되어 있기 때문에 범주형 변수의 임베딩 과정에서 더 많은 정보를 활용한다는 특징 덕분에 우수한 예측 성능을 달성할 수 있었다.

LightGBM Ranker 역시 대부분의 평가지표에서 point-wise learning에 비해 뛰어난 예측력을 보여주지만, 켄달의 타우와 NDCG 지표 기준으로는 선형 회귀에 비해 성능이 떨어진다. 스피어만 상관계수는 모든 순위에 대해 예측력을 동등하게 평가하는 지표인 반면, NDCG는 높은 순위에 대한 예측력을 더 높게 평가하는 지표이다. 이를 고려할 때 LightGBM Ranker가 전반적인 순위 예측은 잘 하였지만 높은 순위의 경주마들을 정확하게 예측하는 것에는 다소 제한적인 방법이라고 할 수 있다. XGBoost Ranker의 경우 LightGBM Ranker와는 반대로 켄달의 타우와 NDCG 측면에서는 비교적 우수한 성능을 보여주고 있다. 이 결과는 XGBoost Ranker가 전체 순위 예측은 잘 못하더라도 높은 순위의 경주마들은 정확한 예측하는 특징이 있음을 보여준다.



(a) Random Forest (b) CatBoost Ranker

Figure 4: *Beeswarm Shapley plot of Random Forest (left) and CatBoost Ranker (right).*

마지막으로 RankNet은 point-wise learning에 해당하는 선형 회귀와 랜덤 포레스트에 비해 오히려 성능이 떨어지는 것을 확인할 수 있다. 이 결과는 앞서 언급한 RankNet의 한계점과 부합한다. 분석에 사용된 경마 데이터에서 반응변수와 설명변수 사이의 관계는 인공신경망으로 모델링하기 부적절했기 때문에 이와 같은 결과가 나타났다고 볼 수 있다.

Figure 3는 point-wise learning에 해당하는 랜덤 포레스트와 pair-wise learning 중에서 성능이 우수한 CatBoost Ranker 각각에 대해 변수별로 섀플리 값에 절대값을 취한 뒤 구한 평균으로 도출한 변수 중요도를 각각 보여주고 있다. 랜덤 포레스트에서는 표준화된 직전 경주기록이 다른 변수들에 비해 큰 변수 중요도를 보여주고 있는 반면, CatBoost Ranker의 경우 경주마의 성적이 가장 높은 변수 중요도를 보이고 있고, 다른 변수들 역시 랜덤 포레스트와는 달리 예측에 어느 정도 기여하고 있는 것을 확인할 수 있다. 이는 point-wise learning과 pair-wise learning의 차이를 보여줄 수 있는 결과로, 표준화된 경주기록만을 정확히 예측하는 것이 목적인 point-wise learning의 경우 표준화된 직전 경주기록만으로도 충분할 수 있지만 두 경주마 사이의 순위를 예측해야 하는 pair-wise learning의 경우에는 이를 넘어 더 많은 변수들을 종합적으로 고려해야 한다는 것을 의미한다. 특히 랜덤 포레스트는 한 경주경기에서 공통으로 가지게 되는 주로습도, 경주거리, 두수가 상위 10개 중요 변수로 선정되었지만, CatBoost Ranker는 한 경주경기 안에서 경주마 간의 편차를 보여주는 경주마의 나이, 질병 진단 횟수, 부담중량, 경주마의 체중변화분, 출주번호, 경주마의 통산 출주횟수가 상위 10개 중요 변수에 포함되었다. 이를 통해 순위 예측에 있어서 더욱 중요한 것은 경주마 간의 상대적인 능력치를 나타내는 변수들일 것이라고 해석 가능하다. 또한, 표준화된 직전 경주기록, 경주마의 출발훈련 횟수, 누적 출발훈련 횟수, 질병 진단횟수는 기존 연구에서 고려되지 않은 변수들임에도 불구하고 높은 중요도를 보이고 있어 예측력 향상에 기여하였다.

Figure 4는 앞서 살펴본 상위 10개 중요 변수에 대해 그린 beeswarm Shapley plot을 랜덤 포레스트와 CatBoost Ranker로 나누어 각각 보여주고 있다. 각 자료들이 변수들에 대해 가지는 섀플리 값이 점들로 표시되어 있다. 한 점에 대해 섀플리 값이 음수일 때에는 점에 대응되는 자료에서 해당 변수가 예측치를 산출하는 데에 음(−)의 영향을 미친다는 것을 의미하고, 섀플리 값이 양수일 때에는 반대로 양(+)의 영향을 미친다는 것을 의미한다. 그리고 모든 점들은 각 변수 내에서 대응되는 자료의 값이 상대적으로 클수록 빨간색으로 표시되며, 크기를 가지지 않는 범주형 변수의 경우에는 보라색으로 표시되어 있다. 따라서 만약 한 변수에 대해 x축에서 0을 기준으로 왼쪽 부분이 파란색, 오른쪽 부분이 빨간색으로 나타난다면 해당 변수의 값이 클수록

예측치를 더욱 크게 예측하는 경향이 있다고 해석할 수 있다.

랜덤 포레스트는 표준화된 직전 경주기록이 작을수록 표준화된 경주기록을 작게 예측하고 있고, 이는 상관계수 그림(Figure 2)에서도 확인할 수 있다. 또한 경주마의 성적, 출발훈련 횟수, 누적 출발훈련 횟수는 큰 값을 가질수록 표준화된 경주기록을 작게 예측하는 경향이 있어 이 역시 탐색적 자료분석 결과와 일치한다. 주로습도도 마찬가지로 값이 높을수록 표준화된 경주기록의 예측값은 작게 나타나고 있다. Choe 등 (2015)에 따르면 주로상태가 포화일 경우 빠른 경주기록이 측정되는 경우가 많기 때문에 주로습도와 표준화된 경주기록이 반비례하는 현상은 타당한 결과이다.

CatBoost Ranker 역시 랜덤 포레스트와 비슷하게 경주마의 성적이 좋을수록, 표준화된 직전 경주기록이 좋을수록 점수를 크게 예측하는 경향이 있다. 이때 CatBoost Ranker에서 예측하는 점수는 그 값이 클수록 높은 순위로 예측한다. 반면에 경주마의 나이, 통산 출주횟수는 예측 점수와 반비례하고 있다. 통산 출주횟수와 경주마의 나이 모두 노화와 관련있는 변수라는 점을 생각한다면, 이는 다른 조건이 모두 동일한 상황에서 경주마의 노화는 순위를 예측하는 데에 있어 부정적인 영향을 미친다는 것을 의미한다. 이러한 결과는 Choe 등 (2015)에서 보인 경주마의 나이가 많을수록 3위 안으로 들어오는 비율이 낮다는 분석결과와 일치한다. 또한 출주번호가 클수록, 즉 경주로의 바깥에서 달릴수록 순위가 낮게 예측되고 있다. 이는 같은 조건의 경주마라면 더 낮은 출주번호를 부여받는 것이 경주에서 유리하다는 것을 의미하고, 이 역시 Choe 등 (2015)에서 보인 분석결과와 일치한다.

경주마의 질병 진단횟수가 예측 점수와 비례하는 결과 역시 탐색적 자료분석에서 보인 상관계수와 일치하지만, 앞서 언급한 것처럼 해당 변수에는 생략된 효과들이 존재할 수 있기 때문에 해석에 유의해야 한다. 부담중량이 클수록 점수가 높게 예측되는 현상은 다소 의외이지만, 부담중량은 경주마 간의 능력차를 인위적으로 조정하기 위해 추가하는 무게이기 때문에 높은 부담중량을 부여받는 경주마가 능력치는 오히려 더 뛰어나다는 점을 감안한다면 설명 가능한 결과일 수 있다. 하지만, 일반적으로 부담중량은 경주에 부정적인 영향을 주는 것으로 알려져 있으므로 이에 대해서는 조금 더 면밀한 분석이 필요하다.

5. 결론

본 연구는 서울 경마 순위 예측을 위해 대표적인 point-wise 및 pair-wise learning 방법들을 적용하였다. Point-wise learning에는 선형 회귀와 랜덤 포레스트를, pair-wise learning에는 RankNet과 LambdaMART (XGBoost Ranker, LightGBM Ranker, CatBoost Ranker)를 활용하였다. 결과적으로, 아이템 간의 순서를 고려하는 pair-wise learning 방법이 point-wise learning 방법보다 더 우수한 순위 예측 성능을 보였다. 특히, LambdaMART 중 CatBoost Ranker는 모든 평가 지표에서 가장 뛰어난 성능을 보였다. 섀플리 값에 기반하여 구한 랜덤 포레스트와 CatBoost Ranker의 변수 중요도 분석 결과, 랜덤 포레스트는 다른 변수들에 비해 표준화된 직전 경주기록이 훨씬 높은 중요도를 나타내었다. 반면에 CatBoost Ranker에서는 경주마의 성적과 표준화된 직전 경주기록이 높은 중요도를 보였지만, 다른 변수들도 상대적으로 높은 중요도를 갖는 것으로 나타났다. 이는 pair-wise learning의 특성상 두 경주마의 순위를 학습하는 과정에서 경주마 간의 차이를 나타내는 다양한 변수들을 종합적으로 고려하기 때문인 것으로 해석할 수 있다.

본 연구는 선행연구에서 크게 주목받지 않았던 pair-wise learning 방법론을 활용하여 예측 성능의 향상을 이루었다는 점에서 중요하다. 뿐만 아니라 경주거리에 따른 경주기록을 표준화하여 데이터 불균형을 해결하였고, 경기 정보, 기수 정보, 마필 정보, 조교사 정보 등 다양한 데이터를 통합하여 더욱 다양한 변수를 고려하였다. 특히, 경주마의 성적, 표준화된 직전 경주기록, 경주마의 출발훈련 횟수, 누적 출발훈련 횟수, 그리고 질병 진단횟수 등의 변수는 모형의 예측력을 향상시키는 데 크게 기여하였다.

추후 연구를 통해 list-wise learning을 포함한 다양한 LTR 접근 방법으로 경주마 순위 예측 데이터에 적용하면, 각 LTR 기법의 독특한 특성을 파악하는 기회가 될 것으로 기대한다. 또한 pair-wise learning은 두 아이템

쌍만을 고려하는 접근법이지만, 3개 혹은 그 이상의 아이템 그룹을 동시에 고려하는 k-wise learning 접근법의 개발은 흥미로운 후속 연구 주제가 될 것이다. 특히 k-wise learning의 개발은 순위 예측의 정확도와 계산 효율성 사이의 균형을 더욱 최적화할 수 있을 것으로 예상된다. 더불어, 심층 학습과 다양한 네트워크 구조를 활용하여 순위 예측 알고리즘의 표현력을 향상시키는 연구 방향도 고려할 수 있다. 이러한 방법들을 통해 경주마 순위 예측의 복잡한 특성과 다양한 요소들을 효과적으로 반영하는 정확한 모델을 찾을 수 있을 것으로 기대한다. 마지막으로, LTR 방법을 단순 순위 예측에만 국한시키지 않고, 데이터 프라이버시를 위한 데이터 생성 과정에서도 순위를 고려하는 방식으로 접목하는 것은 매우 흥미로운 연구 주제가 될 것으로 기대한다.

References

Breiman L (2001). Random forests, *Machine Learning*, **45**, 5–32.

Burges C, Shaked T, Renshaw E, Lazier A, Deeds M, Hamilton N, and Hullender G (2005). Learning to rank using gradient descent. In *Proceedings of the 22nd International Conference on Machine Learning. Association for Computing Machinery*, New York, NY, 89–96.

Burges C, Ragno R, and Le Q (2006). Learning to rank with nonsmooth cost functions, *Advances in Neural Information Processing Systems*, **19**.

Burges CJ (2010). From ranknet to lambdarank to lambdamart: An overview, *Learning*, **11**, 81.

Chen T and Guestrin C (2016). Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, San Francisco, CA, USA, 785–794).

Choe H, Hwang N, Hwang C, and Song J (2015). Analysis of horse races: Prediction of winning horses in horse races using statistical models, *The Korean Journal of Applied Statistics*, **28**, 1133–1146.

Grinsztajn L, Oyallon E, and Varoquaux G (2022). Why do tree-based models still outperform deep learning on typical tabular data?, *Advances in Neural Information Processing Systems*, **35**, 507–520.

Hu Z, Wang Y, Peng Q, and Li H (2019). Unbiased lambdamart: An unbiased pairwise learning-to-rank algorithm. In *Proceedings of The World Wide Web Conference. Association for Computing Machinery*, New York, NY, USA, 2830–2836.

Järvelin K and Kekäläinen J (2017). IR evaluation methods for retrieving highly relevant documents, *ACM SIGIR Forum*, **51**, 243–250.

Ke G, Meng Q, Finley T, Wang T, Chen W, Ma W, Ye Q, and Liu TY (2017). Lightgbm: A highly efficient gradient boosting decision tree, *Advances in Neural Information Processing Systems*, **30**.

Kholkine L, Servotte T, De Leeuw AW, De Schepper T, Hellinckx P, Verdonck T, and Latré S (2021). A learn-to-rank approach for predicting road cycling race outcomes, *Frontiers in Sports and Active Living*, **3**, 714107.

Li P, Qin Z, Wang X, and Metzler D (2019). Combining decision trees and neural networks for learning-to-rank in personal search. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, Anchorage, AK, USA, 2032–2040.

Liu TY (2009). Learning to rank for information retrieval, *Foundations and Trends® in Information Retrieval*, **3**, 225–331.

Park G, Park R, and Song J (2017). Analysis of cycle racing ranking using statistical prediction models, *The Korean Journal of Applied Statistics*, **30**, 25–39.

Prokhorenkova L, Gusev G, Vorobev A, Dorogush AV, and Gulin A (2018). CatBoost: Unbiased boosting with categorical features, *Advances in Neural Information Processing Systems*, **31**.

Pudaruth S, Medard N, and Dookhun ZB (2013). Horse racing prediction at the champ de mars using a weighted probabilistic approach, *International Journal of Computer Applications*, **72**, 39–42.

Soldaini L and Goharian N (2017). Learning to rank for consumer health search: A semantic approach. In *Advances in Information Retrieval: 39th European Conference on IR Research, ECIR 2017, Aberdeen, UK, April 8-13, 2017, Proceedings 39* (pp. 640-646). Springer International Publishing.

Wang X, Li C, Golbandi N, Bendersky M, and Najork M (2018). The lambdaloss framework for ranking metric optimization. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, Torino, Italy, 1313–1322.

Received September 24, 2023; Revised November 11, 2023; Accepted November 27, 2023

Learning-to-rank 기법을 활용한 서울 경마경기 순위 예측

정준형[a], 신동욱[a], 황세용[a], 박건웅[1,a]

[a]서울대학교 통계학과

요 약

본 연구는 learning-to-rank (LTR) 기법 중 point-wise와 pair-wise learning을 적용하여 서울 경마경기 순위 예측을 수행하였다. Point-wise learning으로는 선형 회귀와 랜덤 포레스트를 pair-wise learning으로는 RankNet, LambdaMART (XGBoost Ranker, LightGBM Ranker, CatBoost Ranker)을 활용하였다. 또한 데이터 불균형 문제를 해결하기 위해 전처리 과정에서 경주기록을 경주거리에 따라 표준화하는 방식을 채택하였으며, 모형의 예측 능력 향상을 위해 경기 정보, 기수 정보, 마필 정보, 조교사 정보 등의 다양한 데이터를 사용하였다. 그 결과 아이템 간의 순위관계를 학습할 수 있는 pair-wise learning이 point-wise learning보다 전반적으로 더 뛰어난 예측력을 보이는 것을 확인하였다. 특히 CatBoost Ranker는 제시된 모형들 중 가장 뛰어난 예측 성능을 보였다. 마지막으로 섀플리 값을 통해 CatBoost Ranker에서 경주마의 성적, 직전 경주기록, 경주마의 출발훈련 횟수, 누적 출발훈련 횟수, 질병 진단횟수 등이 상위 10개 중요 변수에 포함된 것을 확인하였다.

주요용어: 경마, 람다마트, 순위학습기법, 랭크넷, 순위예측

이 성과는 정부(과학기술정보통신부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임 (NRF-2021R1C1C1004562 and RS-2023-00218231). 또한, 이 연구는 서울대학교 신임교수 연구정착금으로 지원되는 연구비에 의하여 수행되었음.

[1]교신저자: (08826) 서울시 관악구 관악로 1, 서울대학교 통계학과. E-mail: gwpark23@snu.ac.kr