Technical Report
Indycar Rank Prediction

Bo Peng

12/01/2019

1. Introduction

Racing is a highly dynamic process. Many different events happen along the time, including rank change, pitstop, mechanical failures, and crashes. Prediction in the racing sport can potentially help to improve the safety of the game, which is one of the most critical aspects of the concerns. It can also potentially help the teams to improve their performance by selecting better strategies. Prediction is also important to the fans. Before the race event each year, people will make all kinds of predictions of the race which including the winner(rank1 at the final lap) and rank for each team. It is part of the sports betting/gambling industry.

A traditional approach to build prediction models for racing events is simulation models. In a simulation method, both the relevant factors and the function equations for the target variable should be given. The accuracy of the model is determined by how accurate the factors and equations can refect the reality of the dynamics of the race, highly depends on the domain knowledge of the domain experts.

Machine learning provides another approach to solve the problem of building a model. In machine learning models, the function to map the input factors to the output target variable is no longer limited to what we can easily understand. It can be as simple as polynomials and can be non-linear as decision trees and neural networks that are capable of approximating any continuous function to any degree of accuracy. The learning algorithm can learn the optimal function and its parameters from the data.

This project aims to build rank prediction models for the IndyCar racing dataset by a machine learning approach. The learned model can further help to explore performance-related factors and strategies.

2. Related Work

2.1. Simulation-based method

A simulation model for the lap time in Formula 1 is well explained in [1]. Most of the features are statistics on historical data, including those from the qualification tests; metrics include lap time, speed, DNF(Did not finish) event, Pit Event. Because of the uncertainties, the output is a distribution of lap time for each driver. The following eight features are included:

Table 3.1 Features used in an F1 simulation model

Feature	Description
Qualifying position	Drivers who qualify further down the grid start the race at a time disadvantage.

Start bonus/penalty	The average number of positions gained/lost on the first lap. These two are the metrics to evaluate the impacts of the start position over lap time.
Maximum speed	Straight-line speed is important for determining how easily one car can pass another.
Pace on long runs	lap times in the 1.5-hour FP2 session, the driver's lap time in clean air under race conditions.
Lap-time variability	These three are the speed metrics for each driver
Pit strategy	"we need to *guess* the strategy each driver will use, based on which strategies are optimal. "
DNF(Did Not Finish) probability	This metric represents the uncertainty of crash, mechanical failure, etc.
Tyre degradation	This metric evaluates the degradation ratio for different types of tyres.

The basic model of lap time is a simple linear regression model including the influences of fuel load and tyre degradation on the lap time:

ForecastedPrimeTime = FP2time + 0.5(QualifyingDelta – FP2Delta).
LapTime = ForecastedPrimeTime + Random + TyreDeg + FuelAdj.

Where TyreDeg is the lap-time difference for the current tyres as a function of laps into the stint, the type of tyre, and the driver's tyre degradation multiplier. FuelAdj is the change in lap time per lap of fuel burn.

To model the interactions between the cars, especial the overtaking by DRS, the model contains the conditions, threshold, and gains and penalty for an overtaking event. The drag reduction system (or DRS) is a form of driver-adjustable bodywork aimed at reducing aerodynamic drag in order to increase top speed and promote overtaking in motor racing.

From the work of these simulation models, we learn that the model is built upon specific knowledge of the race. For example, the tyre degradation, pit strategy of tyre selection, and DRS are all special to F1. IndyCar uses one type of tyre on all the paved tracks, and therefore there is no tyre selection in pit strategy. Instead of DRS, the Push-to-Pass button is adopted, which provides the driver with the ability to increase the car's power for short periods to make overtaking easier and hence make the sport more exciting to watch. However, the actions of Push-to-Pass button are currently not included in the timing and scoring log.

2.2. Machine learning-based method

Paper[4][5] is a series of work forecasting the decision-to-decision loss in rank position for each racer in NASCAR. [4] describes how they leveraged expert knowledge of the domain to produce a real-time decision system for tire changes within a NASCAR race. Major results and findings include:

- ‘‘This work started with the hypothesis that a data-driven prediction engine operating in real-time may be able to assist team captains in making these critical tire change decisions.’’
- ‘‘Unfortunately, we cannot perform randomized controlled trials in order to measure the effect of a decision; we are limited by what we can do with the historical data.’’
- ‘‘our evidence suggests that it is possible to generalize across races; that is, we can borrow strength from the data of similar races to make improved predictions.’’
- ‘‘we can see that the machine learning methods are significantly better than the baseline methods.’’
- ‘‘expert commentaries that are typically stated either before or after the race can also be used to qualitatively validate the inferences of our modeling approach.’’

The first important idea in this work is to use the “**stage**” as the unit of input data. As in Fig below, a race is segmented into serval stages, which are split by pitstop or caution laps(yellow flags). The timing of each sub-segments is determined by the performance of either pit crew or driver and car. By using a long stage as input, the influences of those artificial rank position changes caused by pitstop or caution lap temporarily can be mitigated.

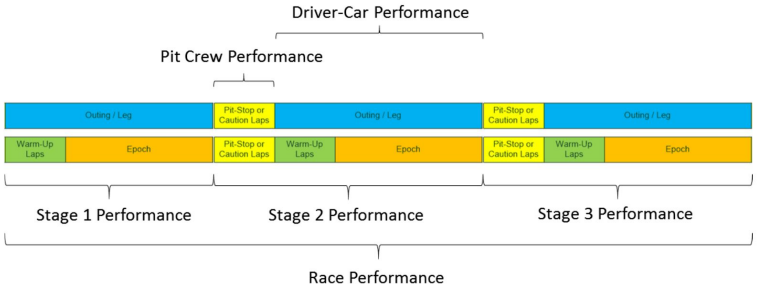

Fig.2.1 Composition of Race Performance based on Leg, Driver-Car, and Pit Crew based on a hypothetical Race comprising 3 stages[5]

Secondly, the way of problem setup in this work is inspiring. They chose to model the **change in rank position** and not the other functions. They also avoid predicting the rank position directly since it is complicated due to its dependency on the timing of other racers’ pit stops. Thirdly, they developed over 100 features with the help of domain experts. Some of them are explained in the paper; therefore, it can be used if they are available in the IndyCar dataset.

The model predicts the rank position change for the next stage before a car enters into it. Baselines include the predictor using current rank or average rank. As shown in the following figure, the accuracy of the sign of prediction(rank increase or decrease) over different models, machine learning models can achieve accuracy between 50%-60%, significantly better than baselines(20%-50%). Since the tire-change strategy is one of the features of the model, once the model is trained, the prediction result can then be used to evaluate which strategy should be chosen for a racer during the race.

2.3. Prediction in Industry

It is difficult to predict the final result of a coming race. According to one sports betting website[3], the odd of the final winner of 2018, Will Power, is the highest one and is around 7/1 before Indy500 2019. It means that if one bets 1$ on Will Power, he can earn 7$ after the race. It also means that the betting company believes the probability of money spend on Will Power will be less than 1/(1+7)=0.125. An underestimated probability of correctly winner prediction will cause financial failure for the company. Roughly, we can regard 12.5% as the state-of-the-art upper bound of this type of prediction. Simulation methods are adopted behind the predictions of these betting services.

Racing software may also support some capability of prediction. RaceTool[6] is one of the most popular software adopted by all IndyCar teams. It supports the “Live Section Times” that providing what cars are doing on their current lap, projected lap times, and rank. From its manual, the software uses the leaders’ last three green laps to project the maximum number of laps possible with the time remaining and compares it with the number of laps to go for a distance race and displays the lower of the two in the To-Go field. It utilizes a relatively simple model based on historical average lap time to predict lap times in the future.

3. The Problem

3.1. Definition of ‘Rank’

According to the protocols document[7], we have the following information on rank:

- "Race results are based on laps led and crossing order." (Results protocol definition)
- "Rank based on best time or race position, updated at each timeline during the race" ($O overall results)

We have verified that the following assumptions are correct:

- Rank is calculated by the elapsed time when the car crosses the boundary of sections(timelines). The order of the cars for the same lap number is its rank.
- Indy500 has multiple sections; therefore, the rank of a car may change during one lap.

Although real-time position/rank is provided through the LED display on all the racing cars, there are no logged into the log file. We can use a lap-based or timeline/section-based rank only. In this work, lap-based rank data are used.

3.2. Rank prediction for the final lap

The problem of rank prediction of the final lap, i.e., the winner, is interesting and useful, but difficult at the same time due to the dynamic nature of the racing. For a machine learning method, another problem of the final winner prediction task is that it can **NOT** be evaluated with limited data. In order to draw a conclusion that a model achieves accuracy of 1/8, it should predict one time correctly on at least eight different races. In practice, more data are needed to get a statistically effective result. To achieve an accuracy of one decimal, 1000 samples of test data are needed, at least. Alternatively, we can adjust the task to predict the rank in shorter future D laps instead of the final. In this way, we have more test data to evaluate.

3.3. Rank prediction for each lap

Rank prediction for each lap is more practical than the winner prediction because there are more data observed. However, due to the dynamics of a race, the ranks of the cars change abruptly in the laps with pitstops, as in Fig.3.1. The rank of a car then is correlated with its team's strategy of pitstop and also other teams strategies. It is still very complex to build a model to predict rank directly. We observed overfitting in training a very simple LSTM model in this task.

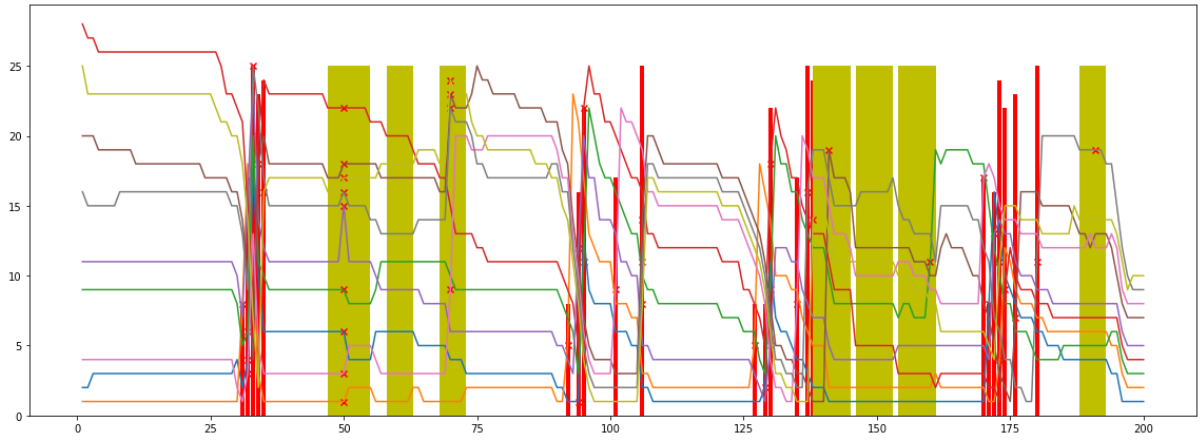

Fig.3.1 Ranks evolving along with the laps. The Final top 10 cars in Indy500 2018. The red bar and the cross marker are pitstops. Yellow bars are caution laps.

3.4. Rank prediction for stages

Inspired by work in [4][5], we define our task to be the rank change prediction on stage dataset. In IndyCar, 'stint' refers to the period of driving between pit stops, similar to the term 'stage.' A stage is defined as the laps between pit stops, adding a warmup period at the beginning denoted as 'trim.' We do not add separate stages for caution laps, as they do not have many influences on rank.

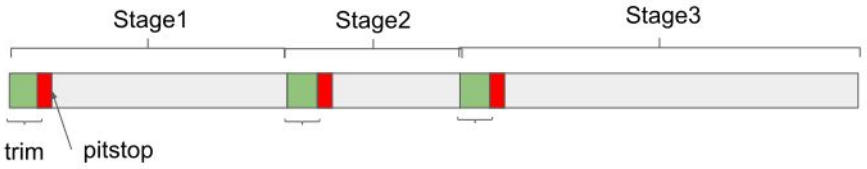

Fig.3.2 A race split into stages, defined as the laps between pit stops, adding a warmup period at the beginning denoted as 'trim'.

The task is to predict the rank change for each stage. When a car goes into the pit lanes, the model can predict how well the car will perform in the next stage, whether it would lose or gain in rank. With the support of this prediction model, it potentially enables the team to evaluate different strategies, such as when to pit stop.

4. Experiments

4.1. IndyCar Dataset

We have the timing and score log file of 2018 IndyCar series, which contains 17 events. Six of them are racing on the oval speedway, as Sfc=P in the following Table 4.1.

Table 4.1 Summary of 2018 IndyCar series

#	Date	Site	Cars	Winner(s)	St	C/E/T	Len	Sfc	Miles	Purse	Pole	Cau	Laps	Speed	LC
1	03/11/18	St. Petersburg	24	Sebastien Bourdais	14	DHF	1.800	S	198		105.085	8	25	86.207	11
2	04/07/18	Phoenix	23	Josef Newgarden	7	DCF	1.022	P	256		188.539	2	23	147.395	12
3	04/15/18	Long Beach	24	Alexander Rossi	1	DHF	1.968	S	167		106.454	4	17	88.622	6
4	04/23/18	Birmingham	23	Josef Newgarden	1	DCF	2.300	R	189		122.773	2	14	93.335	4
5	05/12/18	Indianapolis G.P.	24	Will Power	1	DCF	2.439	R	207		125.761	2	8	113.318	9
6	05/27/18	Indianapolis	33	Will Power	3	DCF	2.500	P	500		229.618	7	41	166.935	30
7	06/02/18	Belle Isle	23	Scott Dixon	2	DHF	2.350	S	165		113.024	2	10	99.285	6
8	06/03/18	Belle Isle	23	Ryan Hunter-Reay	10	DHF	2.350	S	165		90.661	1	3	105.176	6
9	06/09/18	Fort Worth	22	Scott Dixon	7	DHF	1.500	P	372		220.613	3	29	177.250	9
10	06/24/18	Elkhart Lake	23	Josef Newgarden	1	DCF	4.014	R	221		140.020	0	0	132.101	2
11	07/08/18	Iowa	22	James Hinchcliffe	11	DHF	0.894	P	268		182.391	2	17	149.737	4
12	07/15/18	Toronto	23	Scott Dixon	2	DHF	1.786	S	152		108.068	3	12	93.898	9
13	07/29/18	Mid-Ohio	24	Alexander Rossi	1	DHF	2.258	R	203		125.677	0	0	116.957	5
14	08/19/18	Pocono	22	Alexander Rossi	3	DHF	2.500	P	500		219.511	2	10	191.304	11
15	08/25/18	Gateway	21	Will Power	4	DCF	1.250	P	310		NTT	2	16	155.644	10
16	09/02/18	Portland	25	Takuma Sato	20	DHF	1.967	R	207		123.292	4	18	102.971	9
17	09/16/18	Sonoma	25	Ryan Hunter-Reay	1	DHF	2.385	R	203		110.605	1	5	99.440	5

Extract from these events, we get a stage dataset with 805 stage records.

Table 4.2 Stage Dataset of Six Paved Racing in 2018 IndyCar series

Event	Records
Phoenix	114
Indy500	225
Texas	127
Iowa	109
Pocono	126
Gateway	104

4.2. Data Analysis

In work [4], typical racers in a NASCAR race demonstrate a sawtooth shape of lap time distribution over time. A strong influence of tire wear happens in these races. Moreover, a visible waved pitstops pattern can be observed in NASCAR. Most of the cars will adopt pitstop early in the caution laps. This kind of regularity helps to make the idea of ‘stage’ work because of fewer impacts of the different timing of pit stops on the rank position.

While in IndyCar, tire wear seems to be not an issue. As in Fig.4.1 (a), no consistent pattern can be observed. Instead of modeling the trend of lap time, we extract mean and std as features of the lap time within a stage. IndyCar is more challenging to deal with the dynamics, as shown in Fig. 4.1(b). We use ‘trim=4’ in order to mitigate the dynamic of the pitstops.

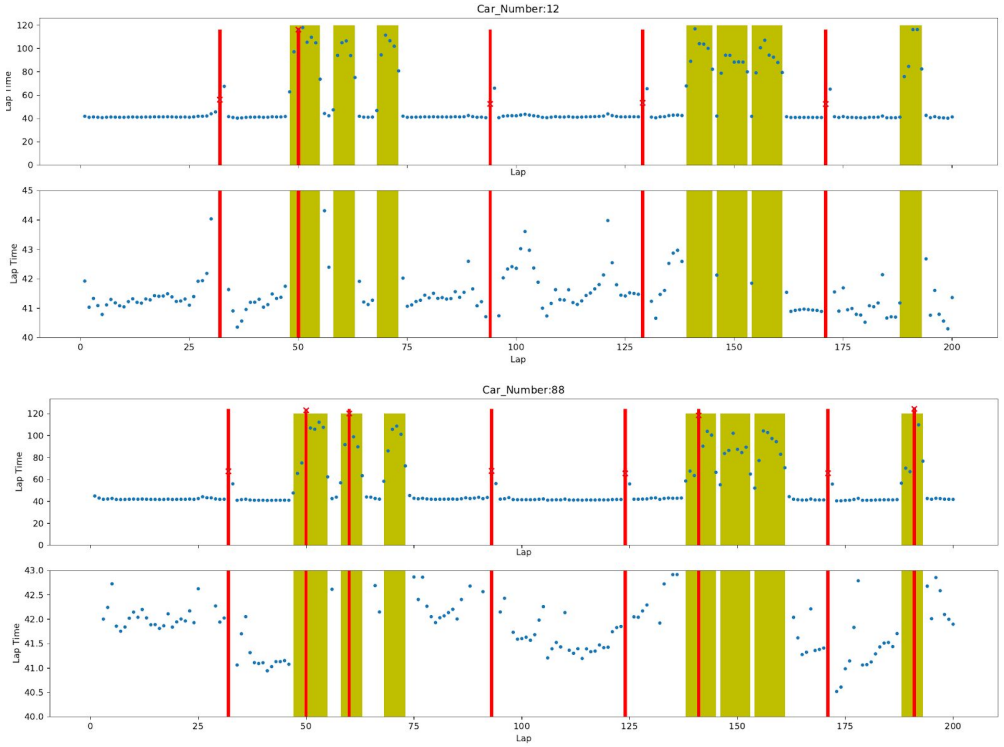

Fig. 4.1(a) IndyCar. Car#12 finished in rank 1 and #88 finished in rank 14. Yellow bar indicates caution laps and red bar indicates pit stop.

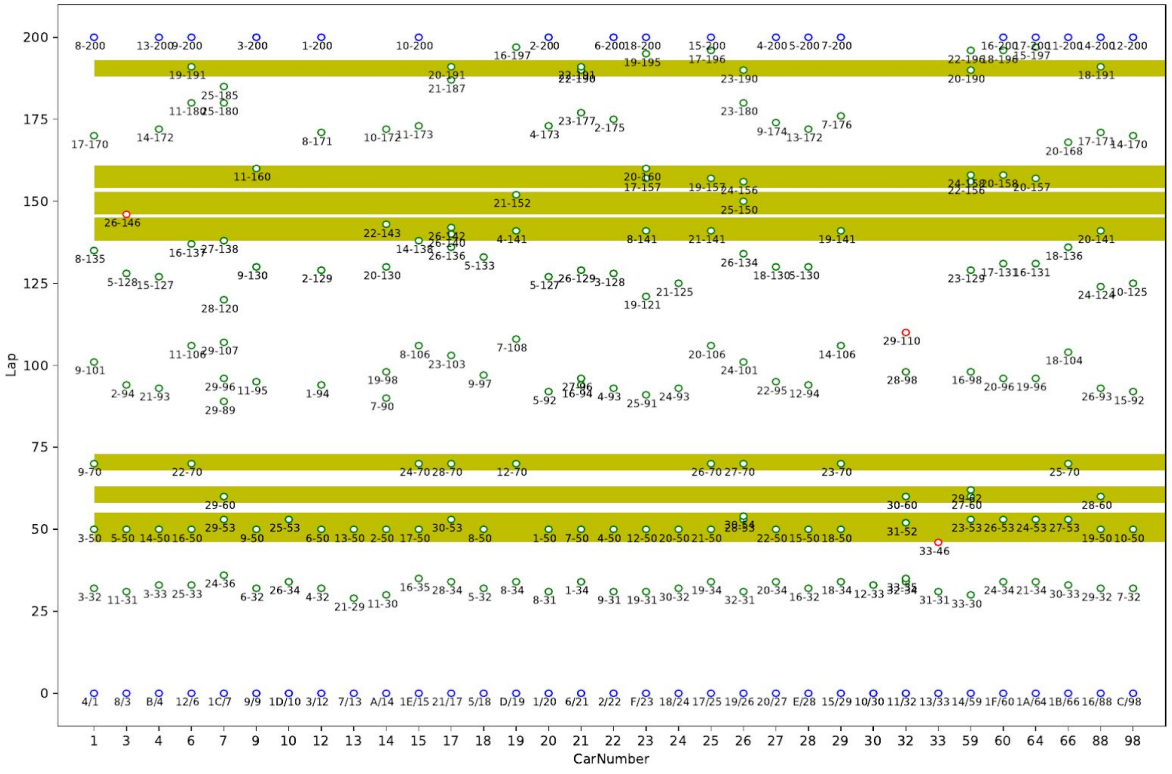

Fig. 4.1(b)IndyCar (2018-Indy500, green circle denotes pitstop labeled with rank-lap, blue circles are the start/end position)

4.3. Features

Features for each stage including the initial start position, start rank, change of rank, rate of changes, lap time, pitstop related features, and information of its neighbors.

Table 4.3 Features in Stage Dataset

Type	Feature	Meaning
gobal info	stageid	
	firststage	1/0
	pit_in_caution	1/0
	start_position	#
0 order of #rank	start_rank	#rank
	start_rank_ratio	#rank/carnum
	top_pack	top5 1/0
	bottom_pack	bottom5 1/0
	average_rank	previous stage
	average_rank_all	all previous stages
1 order of #rank	change_in_rank	previous stage
	change_in_rank_all	all previous stages
2 order of #rank	rate_of_change	previous stage
	rate_of_change_all	all previous stages
neighbors	prev_nb0_change_in_rank	previous car
	prev_nb1_change_in_rank	
	prev_nb2_change_in_rank	
	follow_nb0_change_in_rank	following car
	follow_nb1_change_in_rank	
	follow_nb2_change_in_rank	
Laptime	laptime_green_mean_prev	mean and std of the lap time in green laps of previous stage
	laptime_green_std_prev	
	laptime_green_mean_all	mean and std of the lap time in green laps before
	laptime_green_std_all	
	laptime_mean_prev	mean and std of the lap time in all laps of previous stage
	laptime_std_prev	
	laptime_mean_all	mean and std of the lap time in all laps before
	laptime_std_all	
Pitstop	laps_prev	lap number of the previous stage
	pittime_prev	pit time of previous pitstop

4.4. Baseline

First, we have two types of tasks:

- Predict the sign of rank change for a stage, which is a classification problem.

- Predict the value of rank change for a stage, which is a regression problem.

We have three baselines:

- CurRank: predict the end rank with the current start rank of the stage, i.e., means change is always zero.
- AvgRank: predict the end rank with the average rank changes in previous stages.
- Dice: predict the rank change by randomly throw dice, which follows the distribution of the training data. For the classification task, it is a three facets dice(+,0,-).

We have two ways to split the dataset into training and test set.

- split by event: select 5 events to train, the other 1 to test
- split by stage: select the beginning stages to train, the left stages to test

4.5. Experiment Results

Sign of rank change prediction: We build four models in the task of the sign of rank change prediction, including logistic regression(lr), linear SVM(lsvc), random forest(rf), and gradient boosting tree(xgb). Moreover, use feature selection to select the most prominent ten features.

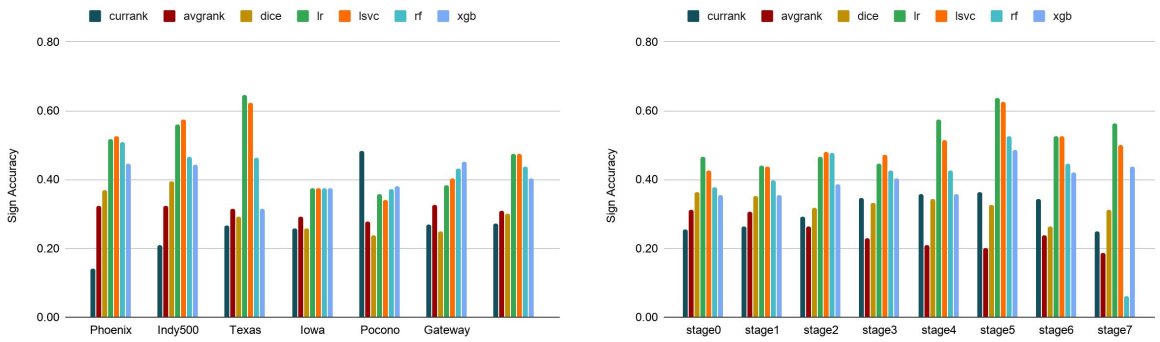

Fig.4.2 Sign of rank change prediction.(a) split_by_events. (b)split_by_stages

As shown in Fig. 4.2, machine learning-based models are significantly better than the baselines. In Fig.4.2(a), there is one exception that CurRank is the best in Pocono, where more than 50% of stages have no rank changes. In Fig.4.2(b), learning models are significantly better than the baselines, and the gap increase when trained with more stages. On average, there is a 50% improvement in accuracy compared with the baselines.

LR and LSVC models are stable and even get better performance with feature selection. However, the accuracy of RF and XGB models drop with feature selection. In the case of the task on split_by_stages dataset, RF is even worse than baselines, with the accuracy drops from 0.44 to 0.06 when feature selection applied. It implies that the salient features for different stages can be different, and the tree models can select them more effectively than a separate feature selection step.

Value of rank change prediction: We build four models in the task of the value of rank change prediction, including linear regression(lasso), SVM(svr), random forest(rf), and gradient boosting tree(xgb).

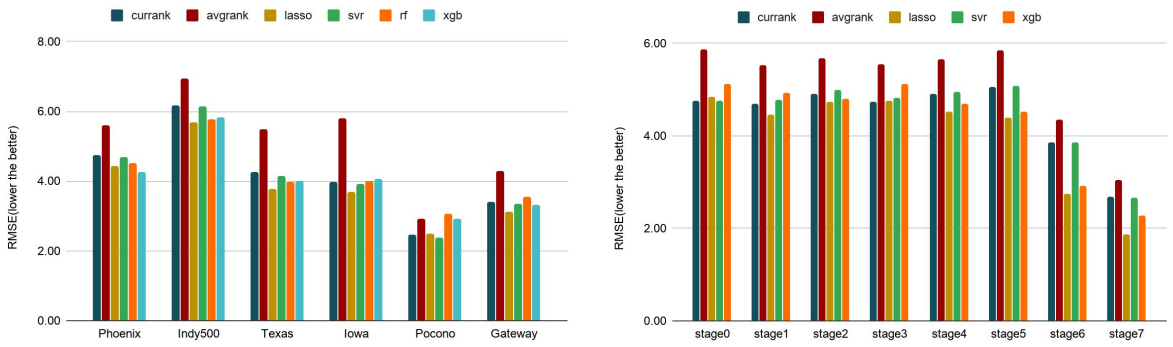

Fig.4.3 Value of rank change prediction.(a) split_by_events. (b)split_by_stages

As shown in Fig. 4.3, machine learning-based models are significantly better than AvgRank and consistently better than CurRank. In Fig.4.2(a), CurRank on Pocono achieves the closest RMSE. In Fig.4.2(b), learning models are similar to CurRank in stage0, and the gap increase when trained with more stages. The best model, LASSO, has a 25% improvement in RMSE compared with AvgRank and about 6% improvement compared with CurRank.

Feature Analysis: The top 20 most salient features selected on the sign of rank change task are:
start_rank, pit_in_caution, average_rank, stageid, start_rank_ratio,
bottom_pack, firststage, top_pack, eventid, laptime_std_all,
average_rank_all, laptime_green_mean_all, laptime_green_std_all,
follow_nb2_change_in_rank, car_number,
rate_of_change_all, pittime_prev, prev_nb0_change_in_rank, laps_after_last_pitstop,
laptime_mean_prev,

From the list above, we find out that features in all the categories in Table 4.3 are included. The features of neighbors are expected to express the interactions among the racing cars; however, current neighbor features extracted are not as important as expected. Further investigation of these features can be our future work.

5. Conclusions

In this work, we investigate the possibility of building machine learning models for rank prediction tasks on the IndyCar dataset. When directly learning the model for winner prediction and rank prediction for each lap is difficult, we find out that machine learning methods are feasible on the task of rank change predictions based on the stage. Similar to the concept of ‘stint,’ stage is defined as consecutive laps split by pitstops. On stage dataset, a model can be learned to predict the sign or value change of the rank between the beginning and the end of a stage. Compared with simple prediction baseline models, machine learning models can achieve significantly better performance, about 1.5x speedup on accuracy in sign prediction task, and 1.3x speedup on RMSE in value prediction task.

6. Future work

Most of the importance, we would like to collaborate with domain experts to find the potential applications to utilize the result of rank change prediction. It might be useful to help to evaluate the strategies of pitstop timing. It might be useful to improve the projection functions in the state-of-the-art racing software.

One major difficulty in applying machine learning or deep learning to the prediction problem on the IndyCar dataset lies in the limitation of the size of the dataset. Considering to add extra information about the drivers from social media, such as tweets, news, could be another direction promising to explore.

References

[1] Mathematical and statistical insights into Formula 1, https://f1metrics.wordpress.com/2014/10/03/building-a-race-simulator/

[2] The intelligentF1 Model, https://intelligentf1.wordpress.com/the-intelligentf1-model/

[3] 2019 Indianapolis 500: Sportsline has Surprising Picks and Predictions, https://www.sportsline.com/insiders/50883891/2019-indianapolis-500-sportsline-has-surprising-picks-and-predictions/

[4]T. Tulabandhula and C. Rudin, “Tire changes, fresh air, and yellow flags: challenges in predictive analytics for professional racing,” Big data, vol. 2, no. 2, pp. 97–112, 2014.

[5]C. L. W. Choo, “Real-time decision making in motorsports: analytics for improving professional car race strategy,” PhD Thesis, Massachusetts Institute of Technology, 2015.

[6]RaceTools: Timing and Scoring Data Analysis and Strategy Software, https://racetools.com/

[7]IndyCar Timing and Scoring Document, V2018.1