Learning a Racing Strategy from the Geometric Features of a Racetrack

Catherine Weaver, Sara Shonkwiler, Jose Zavala

Objective

The objective of this project is to determine a function that describes human racing strategies based on the geometric features of a path of the car. Human racecar divers have a specific strategies for navigating a racetrack, with the ultimate goal of minimizing the overall laptime. These strategies include choosing an appropriate corning speed, braking distance from a turn, or acceleration into and out of a turn. Track geometry greatly influences these racing strategies; for example, length of the curve, maximum turn curvature, and the proceeding track segment influence the optimal racing strategy.

Often the planning of an optimal trajectory around the track is done in two steps: path curvature minimization and velocity maximization subject ot the dynamics of the vehicle []. The second step of this method requires an accurate model of the vehicle dynamics. In this work we aim to explore the determination of an appropriate velocity profile for a track given the geometric features of a path.

Related Work

Pomerleau [3] presented a neural network that related images from a camera and laser range finder to the direction the vehicle should travel to follow the road. Bojarski et al. [1] trained a convolution neural netowrk to map raw pixels from a front-facing camera directly to steering commands. Significant progress has been made using deep reinforcement learning, such as the work presented by Fuchs et al [2] which determined the racing policy (throttle, braking, and steering) based on a set of features including the velocity, acceleration, orientation of the car, as well as a range of distance finders to the edge of the track and the curvature of the future track using data collected from Sony's Gran Turismo Sport racing game.

Data

We will analyze human expert data collected from successful (in terms of low lap time) trials in Sony's Gran Turismo Sport (GTS) racing game. The GTS game has a high-fidelity vehicle dynamics model that simulates a true racing scenario. The data consists of the states (location, speed, yaw rate, etc.) of a car of different human players in a time trial race around various racetracks. Our goal is to predict the the speed of the car at a given waypoint from the next l waypoints of the path taken by the driver. We consider only one vehicle to eliminate the affect of different vehicle dynamics.

The data consists of 31 trials around a single track, and a total of 178262 waypoints. We will split the data into a training and validation set, and time permitting validate the learned function with data recorded from second track.

Problem Formulation

We aim to determine the weight vector v that minimizes

$$\min_{w} \|y - \Phi v\|_2^2, \tag{1}$$

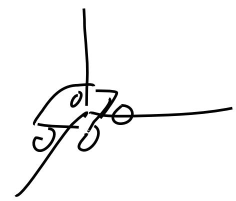
WR V

where $y = \left[y_1, \ldots, y_n\right]^T$ consists of the measurements of the speed $y = V_k$ at the current waypoint k. The set of feature vectors, Φ, consists of the features of the next l waypoints of the path as shown below:

$$\Phi = \begin{bmatrix} \phi_1^T \\ \vdots \\ \phi_n^T \end{bmatrix}, \quad \text{where} \quad \phi_i = \left[\kappa_k, \ldots, \kappa_{k+l}, x_k, \ldots, x_{k+l}, y_k, \ldots, y_{k+l}, w_k, \ldots, w_{k+l}\right]^T. \tag{2}$$

Here each feature vector consists of the following geometric features of the curve on the racetrack:

- κ_i: The path curvature at waypoint i
- (x_i, y_i): The position of the car at waypoint i → ΔS , ΔZ
- w_i: The width of the track at waypoint i



Methods

rot = $[\psi, \theta, \kappa]$

Our project method consists of the following steps:

Step 1: *Formulate problem and construct feature vectors (1.5 weeks, done by Oct 31)*
We have already begun to construct the data sets y and Φ from the raw track data. We will analyze the relationship between the features and the function y to determine the best featurization to preform the regression analysis, including the best number of look-ahead features l.

Step 2: *Analyze which features are important to regression function (1 week, by Nov 4)*
We will analyze which features are significant (have large singular values) by using techniques such as principle component analysis (PCA).

Step 3: *Perform various regression techniques to learn weight vector w (2.5 weeks, by Nov 20)*
We will perform regression using applicable methods taught in class, such as ordinary least squares (OLS), ridge regression to regularize w, LASSO if there appear to be features that are not significant (sparse w). We will compare the performance of these methods.

underline**Step 4:** *Calculate predication error with a validation set (1 week, by Nov 28)*
We will analyze the prediction error for each of the methods used in Step 3 using the separated validation set from the original data. Time permitting, we will test the function on data collected from a different track.

Evaluation

We predict that the majority of effort on this project will be spent on steps 1 and 3, and our evaluation should be based on our analysis of the data and exploration of which methods are most useful for this scenario. The significant contribution of this work is an analysis of which geometric track features play a significant role in the braking distance of a curve, as well as a function to describe this relationship.

References

[1] Mariusz Bojarski, Davide Del Testa, Daniel Dworakowski, Bernhard Firner, Beat Flepp, Prasoon Goyal, Lawrence D. Jackel, Mathew Monfort, Urs Muller, Jiakai Zhang, Xin Zhang, Jake Zhao, and Karol Zieba. End to End Learning for Self-Driving Cars. *arXiv:1604.07316 [cs]*, April 2016. arXiv: 1604.07316.

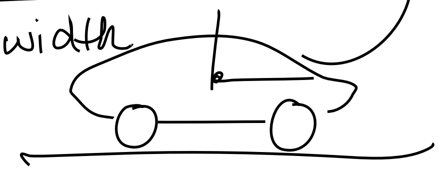
[2] Florian Fuchs, Yunlong Song, Elia Kaufmann, Davide Scaramuzza, and Peter Duerr. Super-Human Performance in Gran Turismo Sport Using Deep Reinforcement Learning. *arXiv:2008.07971 [cs]*, August 2020. arXiv: 2008.07971.

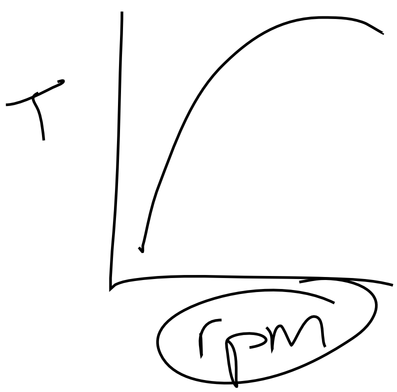
[3] Dean A. Pomerleau. ALVINN: An Autonomous Land Vehicle in a Neural Network. In D. S. Touretzky, editor, *Advances in Neural Information Processing Systems 1*, pages 305–313. Morgan-Kaufmann, 1989.

$\delta t \to T \sim \ell$

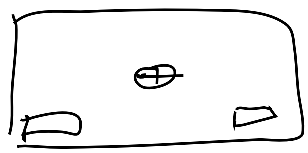
100 Hz

- path curvature, bank
- ΔS, ΔZ
- ~~slip angle & ratio~~
- ~~pitch & roll~~
- track width





I_{zz})



lap count

14

~~15~~

16

1 car multiple tracks

- train on 70% group of tracks
- test on 30% of that group of tracks
- validate on new tracks

- hyperparameter ℓ

---

collect features → catherine