CS 159
Predictive Control & Neural Network Theory

Yisong Yue
Ugo Rosolia
Jeremy Bernstein

Spring 2021

Instructors & TAs

Yisong Yue

Ugo Rosolia

Jeremy Bernstein

Natalie Bernat

Joe Marino

Alex Farhang

Course Details

- Website: https://1five9.github.io/
- Office Hours and Q&A via Discord
  - (ask instructors for invite)
- Assignments via Gradescope

- 4 Homework Assignments (20% total grade)
- Final Research Project (80% total grade)
- Work in teams of 2-3 for everything

Structure of the Course

- Two Topics:
  - Machine Learning for Predictive Control (Ugo Rosolia)
  - Neural Network Theory (Jeremy Bernstein)
  - TBA: guest lectures related to these two topics

- 6 Lectures (3 weeks) per topic
- 2 Assignments per topic
- Final project on topic of your choice
  - Instructors and TAs can help you choose a good project direction

Recap of CS 155

Supervised Learning

Linear Models

Overfitting

Loss Functions

Non-Linear Models

Learning Algorithms & Optimization

Probabilistic Modeling

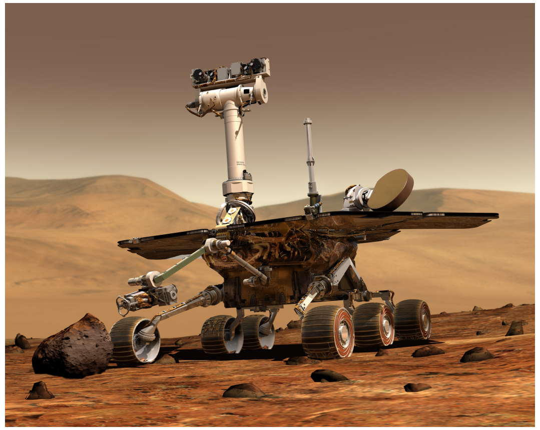
Unsupervised Learning

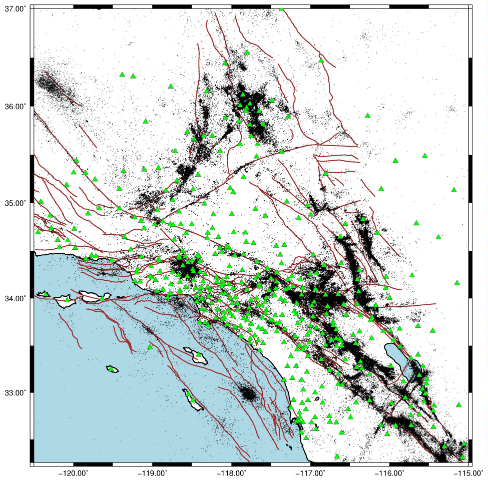
Main Lessons from CS 155

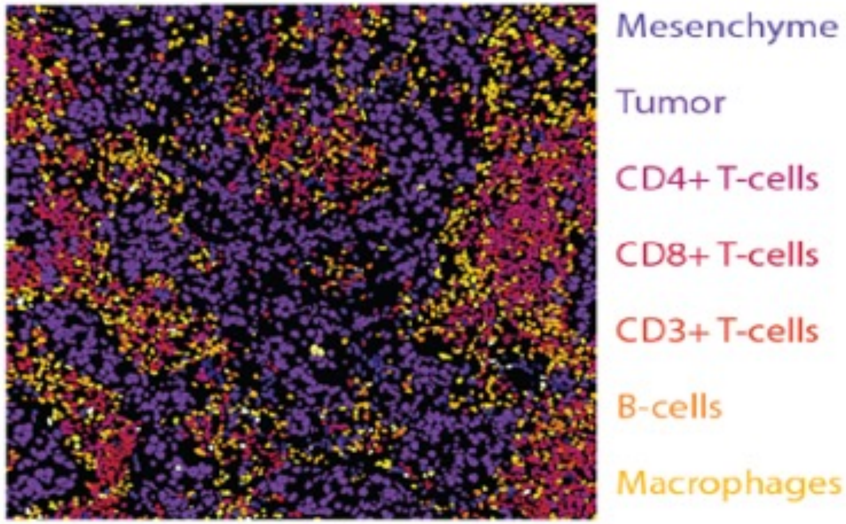
- Requirements for reliable learning:
  - Large dataset
  - Supervised learning signal

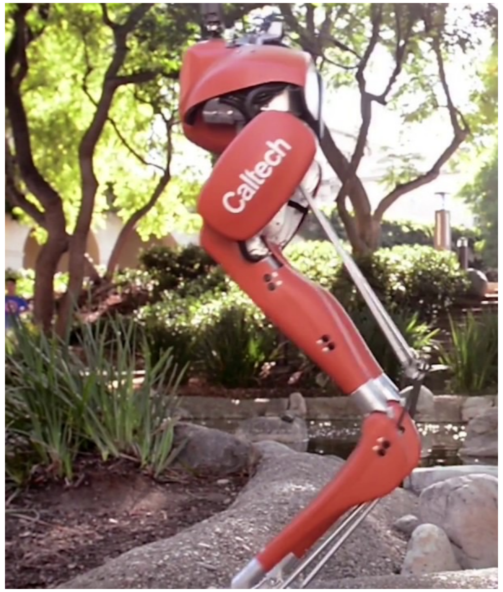
- Outcome:
  - Reasonably good performance on validation/test set

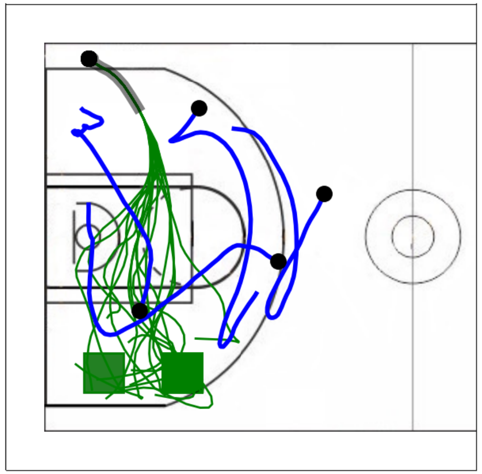
Machine Learning for the Real World





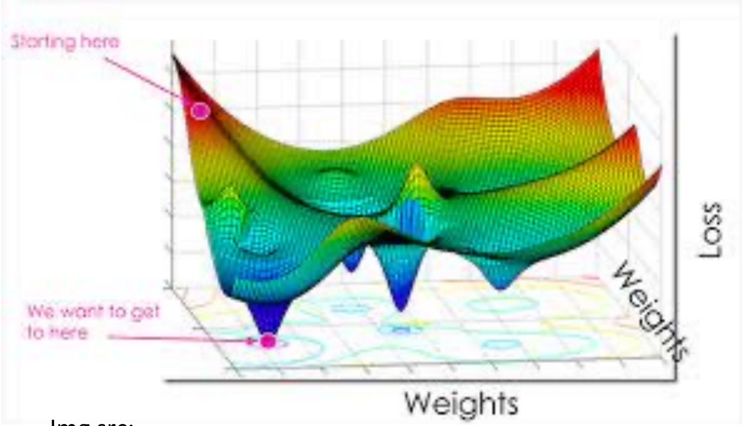






Course Topics

- Predictive Control
  - Sequential decision making
  - Design controllers using learning
    - Safety considerations
    - Distribution shift
- Neural Network Theory
  - Designing reliable learning algorithms
    - Minimize hyperparameter tuning
  - Thinking about generalization
    - Deep learning can memorize training set
    - What about overfitting?



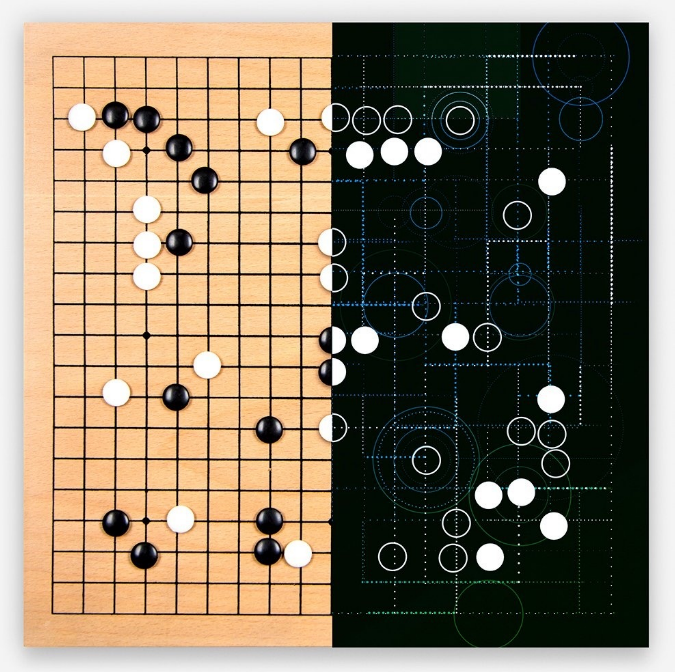
Img src: https://www.pyimagesearch.com/2019/10/14/why-is-my-validation-loss-lower-than-my-training-loss/

Questions?

First Challenge: Dynamics & Control

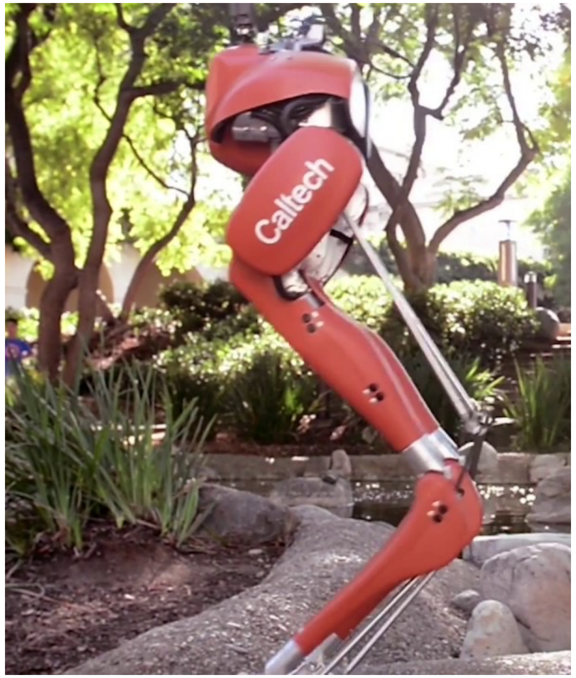
- Sequential decision making
- Your “state” evolves based on actions
- Training data may not cover test instances
- Constraints such as safety

First Topic: Predictive control & model-based RL



Discrete State Space

Continuous State Space

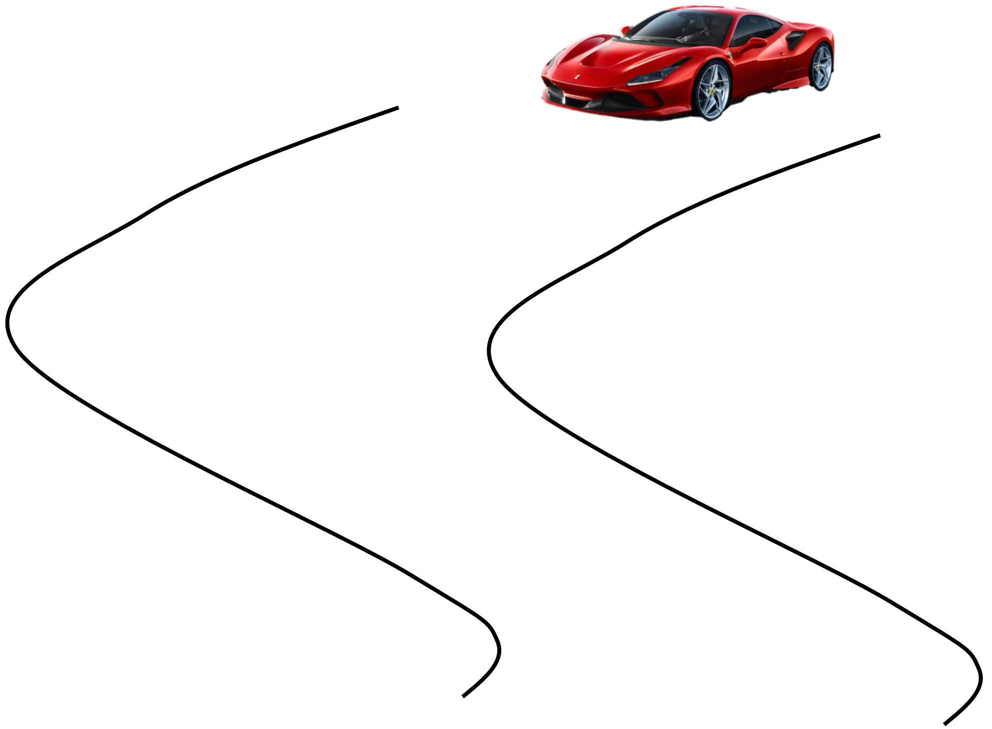


Hybrid State Space

Main Focus: Continuous State Spaces

Why?

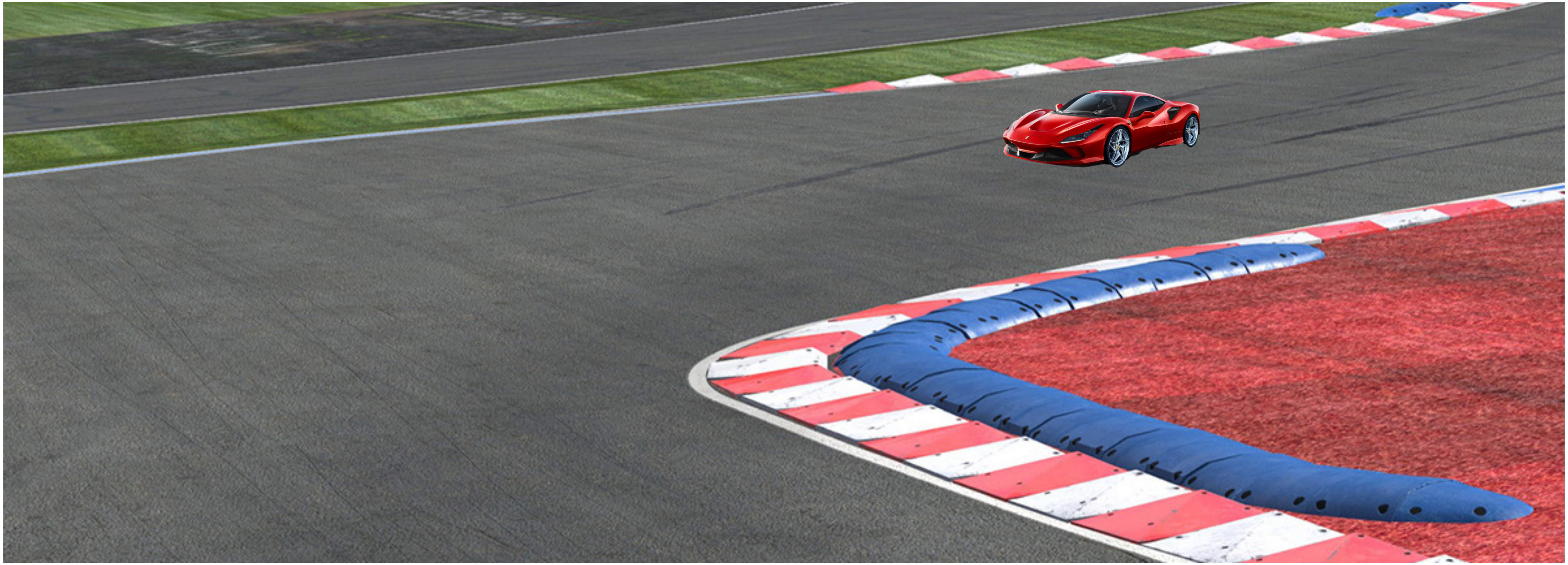
- Easy to express constraints.
  - e.g., stay within the lane boundaries.
- Low-level control actions are continuous.
  - e.g. torque send to the engine.
- Performance metrics are easy to specify.
  - e.g., minimize the derivative of the acceleration.



Main Focus: Continuous State Spaces

Tools:

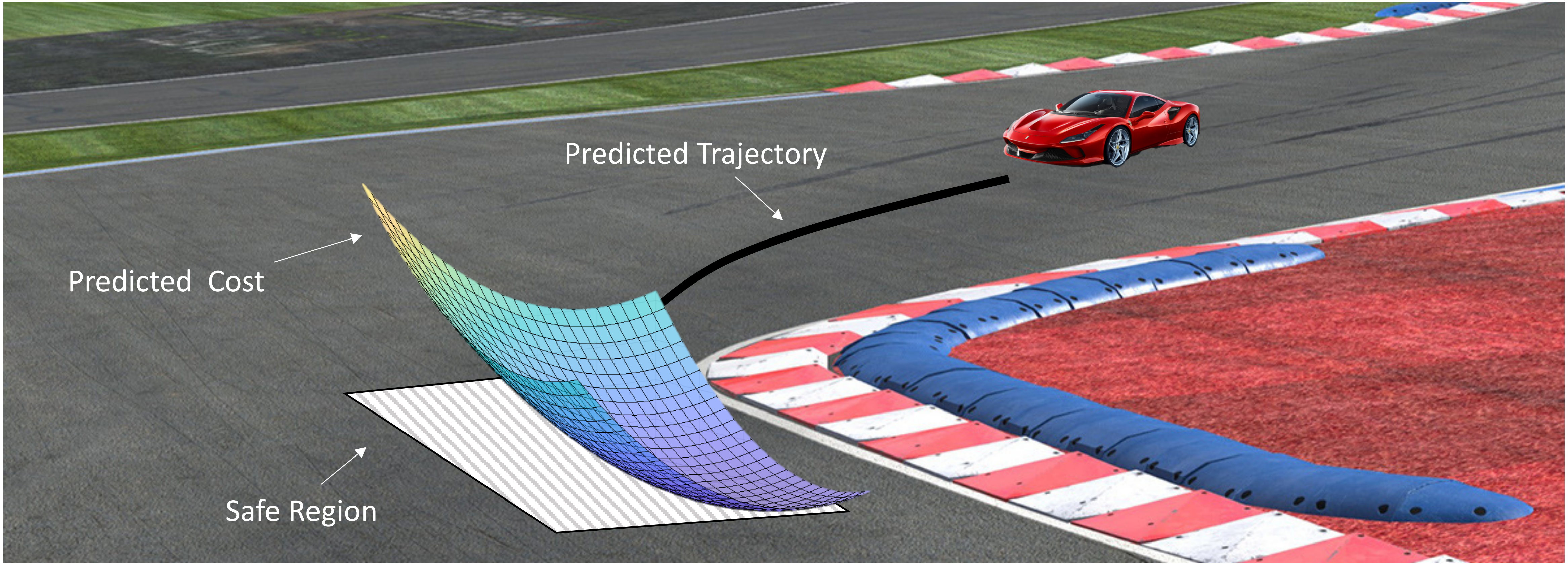
- Model Predictive Control
- System Identification (aka "supervised" model learning)
- Model-based Approximate Dynamic Programming (aka Model-based RL)



Main Focus: Continuous State Spaces

Tools:

- Model Predictive Control
- System Identification (aka "supervised" model learning)
- Model-based Approximate Dynamic Programming (aka Model-based RL)



https://www.youtube.com/watch?v=LNdH9YFzTV4

Learning Model Predictive Controller full-size vehicle experiments

Credits: Siddharth Nair, Nitin Kapania and Ugo Rosolia

Outline

Week #1:

- Discrete MDPs
- Optimal Control

Week #2:

- Model Predictive Control
- Model Predictive Control: Feasibility and Stability

Week #3:

- Learning Model Predictive Control (LMPC)
- Uncertain LMPC

Quick Recap:
1. Dynamic Programming
2. Optimal Control

MPC fundamentals

Learning + MPC

Spoiler Alert: In the continuous settings, we will use ideas inspired by the discrete case!

Homework

Homework:

- Mostly coding given a python template.
- Goal: get familiar with optimization-based controllers.

```python
def buildIneqConstr(self):
    # Hint 1: consider building submatrices and then stack them together
    # Hint 2: most likely you will need to use auxiliary variables
    ...
    G_in = ...

    ...
    E_in = ...

    ...
    w_in = ...

    if self.printLevel >= 2:
        print("G_in: ")
        print(G_in)
        print("E_in: ")
        print(E_in)
        print("w_in: ", w_in)

    self.G_in = sparse.csc_matrix(G_in)
    self.E_in = E_in
    self.w_in = w_in.T
```

Office hours:

- Monday and Wednesday @ 3PM
- **For email CS159 in the subject!**

Slides:

- More details than what we will cover in class.
- Use them as lectures notes.
- HW will be only on topics covered in class.

Final Project:

- Based on research interests.
- Encouraged to combine **Part I** and **Part II** of this class.
- Review project is also an option (Review and code existing RL algorithms. Reach out to instructors for details).

Questions?