ESE 6180-001: Learning for Dynamics and Control

Instructor: Nikolai Matni (nmatni@engineering.upenn.edu), Associate Professor, Dept. of ESE,
Teaching Assistant: Eliot Shekhtman (shekhe@engineering.upenn.edu)
Course website: https://nikolaimatni.github.io/courses/ese6180-fall2026/
Lectures: Tu/Th 1:45pm-3:14pm in DRLB 3C2
Office hours: Tu 3:30pm-4:30pm in AGH 520A (see Canvas for door code)

Course description

This course will provide students an introduction to the emerging area at the intersection of machine learning, dynamics, and control. We will investigate machine learning and data-driven algorithms that interact with the physical world, with an emphasis on a holistic understanding of the interplay between concepts from control theory (e.g., feedback, stability, robustness) and machine learning (e.g., generalization, sample-complexity). This semester will explore these concepts mainly using *imitation learning* as a central case-study, although we will touch on other concepts as well.

Prerequisites

This is an advanced theory-intensive course. A solid foundation in linear systems (at the level of ESE 5000), probability theory (at the level of ESE 5300), and optimization (at the level of ESE 6050), as well as mathematical maturity (comfort with reading and writing proofs) is required. Familiarity with Python is helpful, but not required. Undergraduates need permission.

Intended audience

This course is ideal for advanced graduate students who are interested in applying novel research concepts to their own work. By the end of this course, students will be ready to start doing research in the Learning for Dynamics and Control (L4DC) space.

Tentative List of Topics

Part 1: Basics

- **Introduction and Course Overview**: What this course is (and isn't); the interplay of learned components, feedback, dynamics, and data
- **Foundations of Control**: Open- vs. closed-loop systems, feedback interconnections, stability (local/global, open-/closed-loop), Lyapunov theory, robust stability
- **Foundations of Statistical Learning**: Empirical risk minimization, train/validation/test distributions, generalization bounds, and learning a dynamical system from data

Part 2: Imitation Learning as a Case Study

- **Introduction to Imitation Learning**: Problem setup and standard algorithms; how imitation learning differs from classical control and classical statistical learning
- **Incremental Input-to-State Stability (δ-ISS)**: Motivation, definitions, properties, certificates, and examples
- **Stability-Constrained Imitation Learning**: Learning within a stable policy class; practical approaches (model-based linear IL, open-loop stability with action chunking)
- **Taylor Series Imitation Learning (TaSIL)**: The TaSIL lemma and its application to linear and nonlinear systems
- **Noise Injection**: Persistence of excitation and related considerations

Part 3: Fundamental Limits

- **Minimax Lower Bounds and Le Cam's Method**: Two-point testing arguments, KL divergence tools, and worked examples (Gaussian mean estimation, linear regression)
- **Hardness of System Identification**: Classical methods, consistency, and minimax rates
- **Hardness of Learning to Stabilize**: The role of co-stabilizability in statistical hardness
- **Hardness of Imitation Learning, Revisited**: Connecting control-theoretic and learning-theoretic hardness
- **Hardness of Learning the Linear Quadratic Regulator (LQR)**: The coarse-ID/certainty-equivalence pipeline, minimax risk, and excess cost bounds

Part 4: Optimization Algorithms for Models Deployed in Closed Loop

- **Multi-Task Learning for Dynamics and Control**: Statistical results for multi-task learning; optimization algorithms for anisotropic, task-specific covariates
- **Test-Time Feedback and Training Data Imbalance**: How closed-loop feedback shapes learning; connections to feature learning and preconditioning methods

Grading

- **Homework (40%)**: there will four homework assignments, each worth 10%. You will be given ***5 free late days*** which you may use as you please throughout the semester, after which no late assignments will be accepted. **No exceptions beyond these 5 free late days will be made**. You are allowed, even encouraged, to work on homework in small groups, but you must write up your own homework solutions and code to hand in -- please indicate who you collaborated with on your assignments.
- **Final project (60%):** students will be expected to work on a **theory-focused** project (in groups of up to 2 students). See course website for more details.

Note that these weights are approximate, and we reserve the right to change them later.

Code of Academic Integrity: All students are expected to adhere to the University's Code of Academic Integrity.

AI policy: there are no restrictions on the use of AI in this course. However, you should indicate how AI was used. You are also responsible for ensuring that the output of AI-tools is accurate: **every identified hallucination will result in your grade on that assignment being divided by half.** For example, if you submit a project report with one hallucinated reference, the highest grade you can get on that report is 50%, with two hallucinated references, the highest grade you can get on that report is 25%, etc.