Statistical Techniques in Robotics (16-831) Fall 2018 (Rev09)

Days MW
Room NSH 1305
Time 10:30 - 11:50 AM
Lecturer Kris Kitani
TAs Adam Harley, Navyata Sangvhi
Class Discussion and Slides `https://piazza.com/cmu/fall2018/16831/home`

Description
Probabilistic and learning techniques are now an essential part of building robots (or embedded systems) designed to operate in the real world. These systems must deal with uncertainty and adapt to changes in the environment by learning from experience. Uncertainty arises from many sources: the inherent limitations in our ability to model the world, noise and perceptual limitations in sensor measurements, and the approximate nature of algorithmic solutions. Building intelligent machines also requires that they adapt to their environment. Few things are more frustrating than machines that repeat the same mistake over and over again. We will explore (1) modern learning techniques that are effective at learning online: i.e. throughout the robots operation, (2) reinforcement learning based techniques built to learn from expert demonstrations and (3) probabilistic inference algorithms for maintaining an estimate of itself despite uncertainty.

Prerequisites
Linear Algebra, Multivariate Calculus, Probability theory

Grading
(1) Programming and theory assignments: 70% (4 total), (2) Tutorial presentation:30% (one 30 minute group presentation). Grades determined on an absolute scale. 90% and above is A, 80% - 89% is B, 70% - 79% is C, 60% - 69%, 59% - 50% is D, 50% or below is R. There will be extra credit opportunities.

Late Submissions
3 late days total for the semester. You can use up to 2 late days on a single assignments. Submissions beyond the allowed late days will be penalized by a deduction in points by $33.\bar{3}\%$ per day.

Educational Outcomes
(1) Able to design, implement and prove the regret bounds of a Multiplicative Weights Algorithm.
(2) Apply concepts of online learning and understand the theoretical implication to incremental supervised learning algorithms.
(3) Implement and analyze the difference between linear programming, matrix games, quadratic programming and entropy maximization formulations of inverse reinforcement learning.
(4) Implement and analyze the optimality of multi-armed bandit problems given various environments.
(5) Design tutorial presentation to convey complex technical topics is easy to digest parts. Design theoretical and programming problems to reinforce ideas conveyed through the presentation. Improve verbal and visual presentation skills.

Academic Integrity
All encouraged to work together BUT you must do your own work (code and write up). If you work with someone, please include their name in your write up and inside any code that has been discussed. If we find highly identical write-ups or code without proper accreditation of collaborators, we will take action according to university policies. For tutorial presentations, you are encouraged to find pre-existing presentation material to aid your preparation. If you uses someone's material, please given them credit by including a citation where necessary.

Take Care of Yourself
Do your best to maintain a healthy lifestyle this semester by eating well, exercising, avoiding drugs and alcohol, getting enough sleep and taking some time to relax. This will help you achieve your goals and cope with stress. All of us benefit from support during times of struggle. You are not alone. There are many helpful resources available on campus and an important part of the college experience is learning how to ask for help. Asking for support sooner rather than later is often helpful. If you or anyone you know experiences any academic stress, difficult life events, or feelings like anxiety or depression, we strongly encourage you to seek support. Counseling and Psychological Services (CaPS) is here to help: call 412-268-2922 and visit their website at `http://www.cmu.edu/counseling/`. Consider reaching out to a friend, faculty or family member you trust for help getting connected to the support that can help.

Class Schedule

Date	Topic	Date	Topic
08-27	Overview, Learning overview	08-29	PWEA, Regret
09-03	NO CLASS (Labor Day)	09-05	PWEA, Online Perceptron
09-10	NO CLASS (ECCV)	09-12	NO CLASS (ECCV)
09-17	Winnow Algorithm	09-19	Optimization Basics
09-24	Online Convex Opt, FTL (Sun)	09-26	FTRL, Online SVM
10-01	Adaboost, Multi-Armed Bandit (Sun)	10-03	Contextual Bandit^N
10-08	RL-GPI (Held)	10-10	RL (Policy Gradients)^N
10-15	MAB-UCB (Sun), MAB-EXP3, MAB-EXP4	10-17	Game-IRL, LP-IRL
10-22	QP-IRL, MaxEnt-IRL	10-24	Guest lecture (Bagnell)
10-29	GP, Bayesian Optimization (Kroemer)	10-31	Gaussian Processes^A
11-05	Temporal Inference, HMM, KF	11-07	POMDP^A
11-12	CRF^N	11-14	RNN^A
11-19	NO CLASS (Thanksgiving)	11-21	NO CLASS (Thanksgiving)
11-26	Generative Adversarial Networks^N	11-28	Variational Autoencoder^A
12-03	Special Lecture	12-05	Special Lecture

* Topics with superscripts are student led tutorials. Superscript denotes the TA (A:Adam, N:Navyata) that will support that student team. See Piazza for student assignments.

Assignment Schedule

HW	Topic	Release Date	Submission Date	Editor
1	Weighted Majority Algorithm	09-05	09-20	Navyata
2	Online Supervised Learning	09-26	10-11	Adam
3	Multi-Armed Bandit	10-10	10-25	Adam
4	Inverse Reinforcement Learning	10-24	11-08	Navyata

Tutorial Guidelines

Each tutorial should be 30 minutes and each student in the team should present for roughly 10 minutes each. Each team will consist of 3 students. For each tutorial topic, there will be two teams presenting on the same topic and therefore there will be a repetition of ideas to reinforce learning objectives. Teams will be scored by Instructors and students.

Process:

1. Select desired tutorial topic using form posted on Piazza (8-31 to 9-03), wait for team assignments (9-05)
2. Meet with your team for coffee and discuss roles
3. Begin collecting resources and reading papers/books (ensure enough time to digest material)
4. Obtain Google doc tutorial template file, fill with tutorial content
5. Submit tutorial slides to Instructors (Lecturer and TA) at least 2 weeks before presentation. Be sure to write your script in the presenter notes.
6. Improve slides based on comments
7. Present tutorial on assigned date (come early to make sure your computer connects)
8. Tutorial presentation (PDF) material made available to class on Piazza

Grade Breakdown (Total 30 pts)

- Submit tutorial slides to Instructors (Lecturer and TA) at least 2 weeks before presentation date. Get feedback and improve. (1 pt)
- Give oral presentation using slides (3 pts)
- Develop one theory problem and solution manual, submit to instructors before presentation (3 pts)

- Develop one programming problem and solution code, submit to instructors before presentation (3 pts)
- Audience score (10 pts). Weighted average score of the audience.
- Participation (10 pts). A half point for filling out a scoring form for each tutorial.