Project Instructions
CS 6501: Learning and Game Theory (Fall'19)

General Instructions

The project will count for 40% of your total grade.

The goal of this project is to apply the basic concepts and theory you learned from this course to your own research projects or to *thoroughly* survey a research field relevant to these course topics. A successful project could be, e.g., using the concepts or theory you learned to formulate a research question in your own research field and then try to provide an (even partial) solution. Please notice that though the course is theory-oriented, your course project does *not* have to be — i.e., applied project is equally welcome! However, please make sure that you invest most of your time to the "research" side of the project while not the programming side.

You are encouraged to form a team of 2-4 members to complete the project together. You may use Piazza to find team members. If you really want to do a project alone, please come to talk with us first since we would like to make sure the project is doable for one person and also has sufficient content — remember that it counts for a big portion of of your grade.

Please feel free to discuss with me regarding, e.g., identifying a topic, references, ideas for the project, though these are not mandatory.

Notice: your project does not have to show prefect results — in fact, you could even have a successful project by showing all the thing you have tried and *why they failed*. The primary goal is for you to exercise the concepts and techniques you learned in an interesting research domain. As long as we see that you have tried and spent efforts (which is usually easy to see :-0), there should not be much to worry.

Timeline and Presentation Formats

Project Proposal — an (at most) *one-page* high-level description about what you would like to do and some initial thoughts/survey. This is due by November 7'th (11/07) 5 pm ET and counts for 5 points. You can submit it to Collab.

Project Presentation — each project will be given about 15 mins to be presented during the last week of this course (i.e., 12/03 and 12/05). This counts for 10 points.

Project Report — majority of the score (25 points) is assigned to the final report in PDF format, which should be at most 8 pages with single-column, 11 pt fonts and 1 inch margin. This is due by December 14'th (12/14) 5 pm ET to Collab as well.

Some Suggested Topics

We strongly encourage you to explore your own topics for the project. In the following we provide some suggested topics just for your information.

1. Design mechanisms for selling data/information.

   We studied in class about mechanisms for selling physical goods, like houses, ad opportunities or painting. Recent years have seen significant interests in *data markets*, partially driven by the ongoing AI revolution powered by data and information (trade of data is already happening frequently in our daily life). Data have intrinsically different nature than physical goods. For example, a buyer may even do not know the value of data without running it on his ML models. On the other hand, the buyer can use the same data for multiple purposes or tasks. This raises significant challenges about designing good data markets and data selling mechanisms and has been a very active research topic, drawing attentions from computer science, economics, operation research and business. For an introductory reading, see the paper *Markets for Information: An Introduction* by Bergemann and Bonatti. For a CS-perspective study, see a recent paper *Selling Information through Consulting*.

2. Use multiplicative weight to solve *security games*. The *security game* is a fundamental resource allocation game with significant real-world impact. See, e.g., a paper titled "*The Mysteries of Security Games: Equilibrium Computation Becomes Combinatorial Algorithm Design*" for an introductory reading. Previous research mostly used OR techniques to solve security games. You may think about using the multiplicative weight update algorithm to solve the game, and see whether it is faster than classical OR-based algorithms.

3. We looked at convergence of no-regret learning algorithm in zero-sum games, and some of you figured out that it also provably converges in 2×2 matrix games. Is there any other structured settings (e.g., a game arised in your research domain) that no-regret learning algorithm also provably or empirically converges to equilibrium? If the algorithm does not convergence, what is the patten of the its trajectory? See, e.g., this paper: *From Nash Equilibria to Chain Recurrent Sets: Solution Concepts and Topology*.

4. The above project is about no-regret learning, which is a special class of reinforcement learning (RL) algorithms. You can ask the same question for other RL algorithms, e.g., Q-learning or deep Q-learning. See, e.g., this paper: *Achieving Cooperation Through Deep Multiagent Reinforcement Learning in Sequential Prisoner's Dilemmas.*

5. In the first lecture (and later lectures), we studied deceptive behaviors or *strategic attacks* to ML algorithms used in economic decision making. If you work in ML, particularly in E-commerce domains, you may think about tackling situations with strategic manipulations.

6. Incentive issues in Bitcoin mining. One recent hot trend is to understand gaming phenomenon in bitcoin mining. See, e.g., this paper: *Matthew Effect in Computation Contests: High Difficulty Lead to 51% Dominance*. You can do a survey or study new questions on this frontier.