EECS 598-006 - Final Project Guidelines and Ideas

Jacob Abernethy Due: 12/16

Rough Guidlines

Below I will propose a number of ideas for projects that you can consider working on. This is not meant to be an exhaustive list of all possible topics that you could explore, I strongly welcome you to consider your own ideas or to find nice applications of the theory of the course to your favorite problem. By November 8 I expect you all to have thought about

You may work alone or in groups of two, but in the latter case I will have higher expectations of the output of the project. If there is some strong reason for a group of three then feel free to approach me about this.

The project writeup will be due on December 16. You must **also** create a explanatory poster for a poster session which we will do on the last day of class, Wednesday December 11.

Learning in Games There has been a lot of work on the topic of learning in games, and the literature is too great to list here. Once could do a nice review of this work, and perhaps find some key areas where the use of regret minimization could be of use. A good place to start is the book aptly titled "The Theory of Learning in Games" by Fudenberg and Levine. Here is an interesting research question that I'd love to get even a partial answer to: While regret minimization works great for computing equilibria in zero-sum games, it does not seem to do so well for general bimatrix games, and there are known bad cases. Is it possible to find a characterization for when regret minimization does and does not work?

Universal Portfolios The Universal CRP result that I presented in class, with Cover's algorithma and the $n \log T$ regret bound, is just the tip of the ice berg. It is a great algorithm but unfortunately it's not efficient. There have been a number of attempts to make this efficient with, for example, sampling, and others have proposed alternative algorithms although I don't believe there are any such algorithms with optimal regret bounds. There have also been a number of experimental papers that have played with these techniques. I know of some generalizations that allow for more general portfolios (e.g. long and short, include borrowing costs, etc.). Possible project ideas:

- Do an experimental review on stock data over a range of algorithms and over a range of time periods. I know how to get access to stock price data so please ask.
- Review the literature on existing methods and their advantages/drawbacks. Explain why different algorithms have suboptimal regret.
- Prove a new result. Find a new algorithm with a better bound. Show a new lower bound. (I can make some suggestions along this line).

Understand Boosting The original Boosting paper of Freund and Schapire used the potential function analysis of EWA in a very strong way, although they don't use the regret bound explicitly. There has been a lot of followup work on boosting, with a number of different algorithms proposed. I don't know to what extent the newest results consider the game theory viewpoint, but it would be good to understand how fundamental the "learning to shrink the duality gap" idea is.

Understand the Adversarial vs Stochastic Settings I have presented lots of results on learning in an adversarial (worst case) setting. But there is quite a bit more work on learning under stochastic assumptions and trying to prove bounds that hold in expectation or with high probability. These are often called "generalization bounds" and the field is generally referred to as "statistical learning theory". In some cases the results in the adversarial setting match up but in other cases they do not. It would be nice to understand the differences more precisely.

Convex Optimization Many techniques in regret minimization relate to methods in convex analysis and convex optimization. For example, a tool known as the *Bregman divergence* is used in learning theory bounds as well as optimization. Also the algorithm known as *Mirror Descent* is often used in convex optimization, but it is nearly equivalent to Follow the Regularized Leader. The big players here are Nesterov and Nemirovski, and you can find out about algorithms like "optimal gradient descent" and "excessive gap technique" that resemble online learning. Are there more connections to explore here?

Multi-armed Bandit Setting One of the big applications of online learning has been for the so-called "bandit setting" where the learner only receives feedback about the action chosen on that round. There are lots of real-world scenarios where this is very relevant; the problem of choosing ads to display for a web search query is a great example. There has been lots of work in the past 5 years on theoretical and applied work for bandit problems, and there are lots of avenues to explore here.

- Find a nice experimental setting where different algorithms can be compared (EXP3 for example). I already know of one paper that does an empirical performance comparison. One idea I'd love to see explored: using bandit algorithms to design a No-Limit Texas Holdem Poker bot that performs well against existing bots.
- Review the literature on adversarial bandit problems and compare this to what is understand for stochastic bandit problems.