CS639: Algorithmic Game Theory & Learning

University of Wisconsin–Madison, Spring 2026 *Instructor: Kirthevasan Kandasamy*

Homework 3. Due 03/13/2026, 11.59 pm

Instructions:

1. Homework is due on Canvas by 11:59 pm on the due date. Please plan to submit well before the deadline. Refer to the course website for policies on late submission.
2. Homework must be typeset using appropriate software, such as LaTeX. Handwritten and scanned submissions will **not** be accepted.
3. Your solutions will be evaluated on correctness, clarity, and conciseness.
4. Unless otherwise specified, you may use any result we have already discussed in class. Clearly state which result you are using.
5. If you use any external references, please cite them in your submission.
6. **Collaboration:** You may collaborate in groups of size up to 3. If you collaborate, please indicate your collaborators at the beginning of the homework. Even if you collaborate, *you must write the solution in your own words*.

1 Game theory in the real world

1. Game theory is the study of interactions between self-interested agents. In class, we have seen several stylized games and studied their equilibria. Below are some examples of such games.

   (a) Prisoner's dilemma

   (b) Stag hunt

   (c) Coordination game (a.k.a battle of the sexes)

   (d) Rock-paper-scissors

   (e) Congestion game (provide an example other than driving in a road network)

   [2×2 pts] Pick any two games from the above list. For each game you pick, describe a real-world example where the behavior of agents can be modeled by the game. The real-world example could be from politics, business, technology, or day to day life. In your solution, you should specify the real-world example, and describe which action in the game corresponds to which action in the real world. Justify your answer. Please be as clear and concise as possible. See an example below (if you pick the prisoner's dilemma, you will need to come up with a different example).

   Example: *The nuclear strategy of countries is comparable to the prisoner's dilemma. Collectively, it is better for each country to not develop any nuclear weapons (remain silent) as it will reduce the possibility of nuclear war. However, individually, developing nuclear weapons (betray) is a dominant strategy since doing so will give a country a strong advantage if others do not develop their own, or be an effective deterrent if the others do.*

2. *(Braess' paradox)* Recall Braess' paradox for non-atomic selfish routing (you may refer to example 8.1.1 and figure 8.2 in KP). Suppose you observed something similar in Madison: the presence of some high speed links in the city's road network has increased average travel times for the drivers.

   (a) **[2 pts]** Say you are in charge of the Wisconsin Department of Transportation. Describe any measures you could introduce to discourage drivers from using these high speed links (without actually closing the roads, which may lead to public backlash).

   (b) **[2 pts]** Say you are in charge of a GPS navigation app (e.g Google maps). Describe any measures you could take to reduce travel times caused due to these links.

3. *(Perverse incentives)* There is a popular anecdote about cobras in India. Worried about an increase in the number of venomous cobras, the government began rewarding individuals who killed and reported cobras. While this was successful at first, over time, individuals started breeding their own cobras, killing them, and then reporting them. Upon learning this, the government terminated the reward program. Breeders then released their cobras into the wild, and it led to a further rise in the cobra population than before the reward scheme was implemented.

   This is an example of a *perverse incentive*, where a well–intentioned, but poorly designed incentive scheme can be self-defeating in the presence of strategic agents. You can find many real-world instances of such perverse incentives in Wikipedia: wikipedia.org/wiki/Perverse_incentive.

   [4 pts] Choose one incident described in the Wikepedia article. Briefly summarize the incentive scheme [1 pt], describe why it was susceptible to manipulation by strategic agents [1 pt], and describe an alternative incentive scheme that mitigates or eliminates such strategic considerations [2 pts].

   See below for an example solution. You will need to choose a different example for your solution.

   - ***Gun buyback programs.*** *When governments have attempted to buyback guns to reduce gun violence, some individuals have 3D printed crude gun parts that qualified as a gun in order to obtain the cash payout.*
   - *As governments provided a fixed payout, this was susceptible to manipulation since the cost for 3D printing was (presumably) lower than the cash payout.*
   - *The government could offer the cash payout only if the individual provides proof of purchase or a gun registration when returning the gun, in order to avoid 3D printed returns. One shortcoming of this proposal*

is that it might dissuade agents from returning guns that are not registered. The government could then offer a smaller payout that is lower than the cost to print a gun to disincentivize strategic behavior.

2 CE and CCE in the Coordination Game

Consider the following modified version of the coordination game. Two friends Alice and Bob are planning to attend an event together. Alice wishes to attend a film festival (F) over a ballet (B), whereas Bob wishes to attend the ballet. However, they would also like to spend time with each other. As they cannot communicate with each other, they have to make this decision independently. The utility function for this game is given by the following matrix.

		Bob	
	Actions	Ballet (B)	Film (F)
Alice	Ballet (B)	(2, 3)	(0, 0)
	Film (F)	(1, 1)	(3, 2)

1. **[5 pts]** *(Correlated equilibrium.)* Consider the following distribution s. Show that it is a CE if $\alpha \leq 3/5$.

		Bob	
	Actions	Ballet (B)	Film (F)
Alice	Ballet (B)	$(1-\alpha)/2$	0
	Film (F)	α	$(1-\alpha)/2$

2. **[5 pts]** *(Coarse-correlated equilibria)* Consider the distribution given in part 1. Find all values of α for which the distribution is a coarse-correlated equilibrium.

3 Solving two player games via linear programs

In this question you will derive and implement linear programs (LP) to compute safe strategies, correlated equilibria, and coarse-correlated equilibria in two player games. In this problem, we will assume that player 1 has m actions and player 2 has n actions. We will let $u_i(j,k)$ denote the utility of player $i \in \{1,2\}$ when player 1 chooses action $j \in \mathcal{A}_1 = [m]$ and player 2 chooses action $k \in \mathcal{A}_2 = [n]$.

1. **[2 pts]** *(Safe strategies)* Write two LPs to compute the *(i)* safe strategy of player 1 and *(ii)* safe strategy of player 2 in a two player game.

2. **[4 pts]** *(Maximizing social welfare)* The social welfare $W^{\text{soc}}(s)$ of a joint distribution $s \in \Delta([m] \times [n])$ is the expected sum of agent utilities, i.e., $W^{\text{soc}}(s) = \mathbb{E}_{a\sim s}[u_1(a_1,a_2) + u_2(a_1,a_2)]$. Write two LPs to compute *(i)* the correlated equilibrium (CE) which maximizes the social welfare, and *(ii)* the coarse-correlated equilibrium (CCE) which maximizes the social welfare.

3. **[6 pts]** *(Maximizing egalitarian welfare)* The egalitarian welfare $W^{\text{egal}}(s)$ of a joint distribution $s \in \Delta(\mathcal{A}_1 \times \mathcal{A}_2)$ is the minimum expected utility[1] of either agent, i.e., $W^{\text{egal}}(s) = \min_{i\in\{1,2\}} \mathbb{E}_{a\sim s}[u_i(a_1,a_2)]$. Write three LPs to compute *(i)* the joint distribution s which maximizes the egalitarian welfare, *(ii)* the CE which maximizes the egalitarian welfare, and *(iii)* the CCE which maximizes the egalitarian welfare.

4. **[18 pts]** *(Implementation)* See the given Python starter code given with this homework. Implement the following functions in the file `solve_via_lps.py`. In all cases, the arguments to the function are two $m \times n$ matrices `Q1`, `Q2`, which represent player utilities, where the $(i,j)^{\text{th}}$ element of `Q1` and `Q2` denote $u_1(i,j)$ and $u_2(i,j)$ respectively.

[1] Note that this is different from the expected minimum utility $\mathbb{E}_{a\sim s}\left[\min_{i\in\{1,2\}} u_i(a_1,a_2)\right]$.

(a) `compute_two_player_safe_strategies_via_lp`. Using your solution to part 1, write two LPs to compute the safe strategies of both players. Return both safe strategies.

(b) `compute_optimal_social_welfare`. Return the optimal social welfare, given the utility matrices.

(c) `compute_optimal_egal_welfare`. Using your solution to part 3, write an LP to compute the optimal egalitarian welfare.

(d) `compute_two_player_social_welfare_maximzing_ce_via_lp`. Using your solution to part 2, write an LP to compute the CE with the maximum social welfare. Return both the optimal welfare value and the optimal CE.

(e) `compute_two_player_social_welfare_maximzing_cce_via_lp`. Using your solution to part 2, write an LP to compute the CCE with the maximum social welfare. Return both the optimal welfare value and the optimal CCE.

(f) `compute_two_player_egal_welfare_maximzing_ce_via_lp`. Using your solution to part 3, write an LP to compute the CE with the maximum egalitarian welfare. Return both the optimal welfare value and the optimal CE.

(g) `compute_two_player_egal_welfare_maximzing_cce_via_lp`. Using your solution to part 3, write an LP to compute the CCE with the maximum egalitarian welfare. Return both the optimal welfare value and the optimal CCE.

Follow these instructions carefully:

- Execute `python main.py` to execute the code and generate the required outputs. The starter code is set up to produce these outputs automatically. **Please print the outputs** generated by the starter code **for all four games provided**. Your grade will be based on the outputs your program produces.
- You may use any LP solver of your choice (our reference implementation uses SciPy's `optimize.linprog` module).
- If you choose to use a programming language other than Python, you are responsible for generating all instances exactly as in the starter code, and writing any necessary boilerplate to reproduce the required outputs.
- Below, we display the output for the first game. If your output does not match ours, first double-check your implementation with your peers. You may post your output for the first game on Piazza if it differs from the reference output.

```
Game 1: random_game_4_5 (m=4, n=5)
  * Computing safe strategies
      - Player 1 safe strategis: [0.         0.         0.4056338 0.5943662]
      - Player 2 safe strategis: [0.         0.06732673 0.
                                  0.93267327 0.        ]
  * Computing optimal social and egalitarian welfare
      - Optimal social welfare : 1.666
      - Optimal egalitarian welfare : 0.7707699680511182
  * Computing a correlated equilibrium to maximize social welfare
      - Optimal social welfare at a CE: 1.637335416308038
      - PoS_CE_SW = 1.0175068488755936
      - CE: [ [0.13967252379903478, 0.0, ...],  [0.0, 0.0, ...],  ...]
  * Computing a coarse correlated equilibrium to maximize social welfare
      - Optimal social welfare at a CCE: 1.657128801431127
      - PoS_CCE_SW = 1.0053533548878106
      - CCE: [ [0.0, 0.0, ...],  [0.0, 0.0, ...]  ...]
  * Computing a correlated equilibrium to maximize egalitarian welfare
      - Optimal egal welfare at a CE: 0.7165222739281247
      - PoS_CE_EW: 1.0757097107750695
      - CE: [ [0.02294275295260642, 0.09009420879998888, ...],
              [0.0, -0.0, ...]  ...]
```

```
* Computing a coarse correlated equilibrium to maximize egalitarian welfare
   - Optimal egal welfare at a CCE: 0.7608240275362749
   - PoS_CCE_EW: 1.0130725899220752
   - CCE: [ [0.0, 0.0, ...],  [0.0, 0.10799249382026084, ...]  ...]
```

5. *(Price of stability)* Let us define the price of stability for the social welfare with respect to CE and CCE as follows:

$$\mathrm{PoS}^{\mathrm{soc}}_{\mathrm{CE}} = \frac{\max_s W^{\mathrm{soc}}(s)}{\max_{s \text{ is a CE}} W^{\mathrm{soc}}(s)}, \qquad \mathrm{PoS}^{\mathrm{soc}}_{\mathrm{CCE}} = \frac{\max_s W^{\mathrm{soc}}(s)}{\max_{s \text{ is a CCE}} W^{\mathrm{soc}}(s)}$$

Above s refers to any *joint distribution* in $\Delta([m] \times [n])$. We can similarly define the price of stability for the egalitarian welfare with respect to CE and CCE as follows:

$$\mathrm{PoS}^{\mathrm{egal}}_{\mathrm{CE}} = \frac{\max_s W^{\mathrm{egal}}(s)}{\max_{s \text{ is a CE}} W^{\mathrm{egal}}(s)} \qquad \mathrm{PoS}^{\mathrm{egal}}_{\mathrm{CCE}} = \frac{\max_s W^{\mathrm{egal}}(s)}{\max_{s \text{ is a CCE}} W^{\mathrm{egal}}(s)}$$

(a) **[1 pts]** Briefly, explain why the following inequalities must be true.

$$\mathrm{PoS}^{\mathrm{soc}}_{\mathrm{CE}} \geq \mathrm{PoS}^{\mathrm{soc}}_{\mathrm{CCE}}, \qquad \mathrm{PoS}^{\mathrm{egal}}_{\mathrm{CE}} \geq \mathrm{PoS}^{\mathrm{egal}}_{\mathrm{CCE}}.$$

(b) **[1 pts]** Are the values you obtained in the games above consistent with this observation? See the outputs generated for the `PoS_XX_YY` quantities in the starter code.

4 Deterministic policies in the experts problem

Recall the setting for the experts problem. There are a set of K actions. On round t, a *learner* chooses a probability distribution $p_t = (p_t(1), \ldots, p_t(K)) \in \Delta_K$. An action A_t is sampled from p_t. The environment simultaneously (without knowledge of A_t) chooses a loss vector $\ell_t = (\ell_t(1), \ldots, \ell_t(K)) \in [0,1]^K$, where $\ell_t(i)$ is the loss for action i. The learner observes the entire loss vector ℓ_t, i.e the losses for *all* actions. The learner's expected loss on round t is $\mathbb{E}[\ell_t(A_t)] = \sum_{i=1}^K p_t(i)\ell_t(i) = p_t^\top \ell_t$.

A learner's policy π maps the previous losses to a distribution for round t, i.e., $p_t = \pi(\{\ell_s\}_{s=1}^{t-1})$. The regret $R_T(\pi, \ell)$ of a policy π for a given sequence of losses $\ell = \{\ell_1, \ldots, \ell_T\}$ is the difference between her cumulative expected loss and the best fixed action in hindsight:

$$R_T(\pi, \ell) = \sum_{t=1}^T p_t^\top \ell_t \;-\; \min_{i \in [K]} \sum_{t=1}^T \ell_t(i) = \sum_{t=1}^T p_t^\top \ell_t \;-\; \min_{p \in \Delta_K} \sum_{t=1}^T p^\top \ell_t. \tag{1}$$

Deterministic policy. We say that a policy is deterministic if, on each round, it chooses p_t to be a unit vector e_{A_t} for some $A_t \in [K]$. That is, the action A_t is a function of the losses $\{\ell_s\}_{s=1}^{t-1}$ in the previous $t-1$ rounds.

[5 pts] Show that any deterministic policy will have linear regret. That is, for any deterministic policy π and any T, there exists a loss sequence $\ell = \{\ell_1, \ldots \ell_T\}$ such $R_T(\pi, \ell) \in \Omega(T)$.

5 Other policies for the experts problem

In class, we studied the Hedge algorithm for the experts problem and showed that its regret could be bounded by $R_T \leq 2\sqrt{T \log(K)}$. In this question we will study other policies for the experts problem.

1. *(Polynomial weights)* Our first policy, shown below, takes a learning rate parameter $\eta \in (0, 1/2)$ as input. We initialize a weight vector to $\mathbf{1}_K$. On each round, we set p_t so that that the probability of selecting action i is proportional to $w_t(i)$, i.e $p_t(i) \propto w_t(i)$. At the end of the round, we update the weights as follows: $w_t(i) \leftarrow w_{t-1}(i)(1 - \eta\ell_t(i))$. Observe that if we replace the update with $w_t(i) \leftarrow w_{t-1}(i)e^{-\eta\ell_t(i))}$ instead, we obtain the Hedge algorithm.

Algorithm 1 π^{PW}

Given: Learning rate $\eta \in (0, 1/2)$.
Initialize weights: $w_0 \leftarrow [1, \ldots, 1] = \mathbf{1}_K$.
for $t = 1, \ldots, T$ **do**
  Set $p_t(i) \leftarrow \frac{w_{t-1}(i)}{\sum_{j=1}^{K} w_{t-1}(j)}$.
  Play p_t and observe ℓ_t.
  Update weights: $w_t(i) \leftarrow w_{t-1}(i)(1 - \eta \ell_t(i))$ for all $i \in [K]$.
end for

We will bound the regret of this policy π^{PW}.

(a) **[5 pts]** Let $\ell = \{\ell_1, \ldots, \ell_T\}$ be a sequence of losses. Let $\Phi_t = \frac{1}{\eta} \log\left(\sum_{j=1}^{K} w_t(j)\right)$. First show that $\Phi_t - \Phi_{t-1} \leq -p_t^\top \ell_t$.

Hint: You may use the inequality $\log(1 - x) \leq -x$ for all $x < 1$.

(b) **[6 pts]** Using the result from part 1a, show that

$$R_T(\pi^{\text{PW}}, \ell) \leq \frac{1}{\eta} \log(K) + \eta T.$$

Hint: You may use the inequality $\log(1 - x) \geq -x - x^2$ for $x \in (0, 1/2)$.

(c) **[2 pts]** Choose an optimal value for η and obtain the tightest possible upper bound on the regret.

2. *(Gradient descent)* We will now present a policy based on gradient descent. We will first introduce some notation. For vectors $x, y \in \mathbb{R}^n$, let $x^\top y = \sum_{i=1}^{n} x_i y_i$ denote the inner product, let $\|x\|_2$ denote the ℓ_2 distance of x where $\|x\|_2^2 = x^\top x$.

Consider the following policy π^{GD}. First, choose some arbitrary $p_1 \in \Delta_n$ for round 1. Then, iteratively update it based on the observed losses as follows:

$$w_{t+1} \leftarrow p_t - \eta \ell_t, \qquad p_{t+1} \leftarrow \text{proj}_{\Delta_K}(w_{t+1}).$$

Here, $\eta > 0$ is the learning rate, which you will choose below. The projection $\text{proj}_S(w)$ of any $w \in \mathbb{R}^n$ to some $S \subset \mathbb{R}^n$ is the closest point in S to w, and is defined as $\text{proj}_S(w) = \text{argmin}_{w' \in S} \|w - w'\|_2$. You can think of the above update as a gradient descent scheme since w_{t+1} takes a small step in the direction of the loss from the current p_t. However, since w_{t+1} is not guaranteed to be a probability distribution, we project it back to the Δ_K simplex to compute p_t, i.e we find the point in Δ_K that is closest to w_{t+1}.

(a) **[5 pts]** *(Bound on the instantaneous regret)* Consider any $p \in \Delta_n$. Show that on round t,

$$\ell_t^\top (p_t - p) \leq \frac{1}{2\eta} \left(\|p - p_t\|_2^2 - \|p - p_{t+1}\|_2^2\right) + \frac{\eta K}{2}.$$

Hint: You may start by trying to upper bound $\|p - p_{t+1}\|_2^2$. The following facts may be useful (you are encouraged to verify/prove that their true):

- For any $p \in \Delta_K$, $w \in \mathbb{R}^K$, and $S \subset \mathbb{R}^K$, we have $\|p - \text{proj}_S(w)\|_2^2 \leq \|p - w\|_2^2$.
- As $\ell_t \in [0, 1]^K$, we have $\|\ell_t\|_2^2 \leq K$ for all rounds t.
- For any $a, b \in \mathbb{R}^n$, we have $\|a + b\|_2^2 = \|a\|_2^2 + \|b\|_2^2 + 2a^\top b$.
- For any $a \in \mathbb{R}^n$ and $c \in \mathbb{R}$, we have $\|ca\|_2^2 = c^2 \|a\|_2^2$.

(b) **[6 pts]** *(Bound on the cumulative regret)* Using the result in part 2a, show that, for an appropriate choice of η, we have $R_T(\pi^{\text{GD}}, \ell) \in \mathcal{O}(\sqrt{KT})$ for any loss sequence ℓ.

Hint: Recall, from (1) there are two equivalent ways to define the regret. To apply the result in part 2a, choose an appropriate $p \in \Delta_K$ and relate it to the regret incurred by the policy on round t.