Homework #3
DATA 37200: Learning, Decisions, and Limits (Winter 2025)

Due end of day (midnight) **March 14**, submit on gradescope.

Each subproblem is worth 2 points. We give you 6 bonus points as well, as long as you turn in the hw, so you can potentially get a full score even if you do not do all parts (given you make no mistakes). Nevertheless, you are encouraged to do all of the parts, in part because the problems are intended to be helpful for learning the material.

Problem 1: Exponential Discounting

For (non-stationary) MDPs with infinite time horizons, it is common to use a time-discounted reward structure where future rewards are valued less than present rewards. We can do this by picking a parameter $\gamma < 1$ and defining the expected total reward of a policy π to be equal to

$$\mathbb{E}_\pi \sum_{i=1}^{\infty} \gamma^i r_i.$$

In this problem we consider some basic properties of such a reward structure. In all parts we assume rewards r_i are valued in $[0, 1]$.

(a) Approximation via truncation: show that for any $\epsilon > 0$ there exists a time horizon H depending on γ, ϵ such that

$$\mathbb{E}_\pi \sum_{i=H+1}^{\infty} \gamma^i r_i < \epsilon.$$

(b) Approximation on short time scales when γ is close to 1: let $H \geq 1$ be arbitrary and suppose that $\gamma = (1 - \delta/H)$ for some $\delta \in (0, 1/4)$. Show that

$$e^{-2\delta} \mathbb{E}_\pi \sum_{i=1}^{H} r_i \leq \mathbb{E}_\pi \sum_{i=1}^{H} \gamma^i r_i \leq \mathbb{E}_\pi \sum_{i=1}^{H} r_i.$$

(c) Write down a version of the Bellman equations which shows how to compute the Q function and value function at time t given the Q function and value function at time $t + 1$.

Problem 2: Harmonic Oscillator

Let $k > 0$ and consider the following system with continuous time:

$$dx/dt = v(t), \qquad dv/dt = -kx(t) + u(x(t), v(t)). \tag{1}$$

We are going to consider how the system behaves for different choices of $u : \mathbb{R}^2 \to \mathbb{R}$.

(a) Solve the differential equations when $u = 0$ regardless of its input, with initial conditions $x(0) = 1, v(0) = 0$.

Let $\lambda > 0$. For any u, $x(0)$, and $v(0)$

$$F(u, x(0), v(0)) = \int_0^\infty x(t)^2 dt + \lambda \int_0^\infty u(x(t), v(t))^2 dt$$

where $x(t)$ is the solution of the differential equation given this particular choice of u and initial condition $x(0)$ and $v(0)$. (F is analogous to the value function and u is analogous to the policy in a discrete time MDP.)

For any $\alpha < k$ and $\beta \in \mathbb{R}$, define $u_{\alpha,\beta}(x(t), v(t)) = \alpha x(t) + \beta v(t)$.

(b) Solve the differential equation (1) when $u = u_{\alpha,\beta}$.

Finally our goal is, in terms of k and λ, to compute α and β to minimize

$$F(u_{\alpha,\beta}, 1, 0).$$

This represents, given α and β, the total cost incurred starting from $x(0) = 1, v(0) = 0$. It is possible, but fairly time consuming, to solve this problem using the formula from part (b).

Instead, there is a more conceptual way to solve this problem using the "Bellman equations" for this problem. We will take some facts as given to simplify the argument.

Take it as a given that:

- Letting $V^*(x, v) = \min_u F(u, x, v)$ where u ranges over all differentiable functions such that the solution of (1) exists for all time, then there exists a symmetric and positive-definite[1] matrix P such that

$$V^*(x, v) = \begin{bmatrix} x & v \end{bmatrix} P \begin{bmatrix} x \\ v \end{bmatrix}.$$

  Also, there exists a unique minimizer u_* such that $V^*(x, v) = F(u_*, x, v)$.

- V^* satisfies the following continuous-time version of the Bellman equation: for any x, v

$$0 = \min_{u \in \mathbb{R}} \left[\langle \nabla V^*(x, v), (v, -kx + u) \rangle + x^2 + \lambda u^2 \right]$$

  and the minimum is attained at $u = u_*(x, v)$. (Informally, this is because $V^*(x, v) = F(u_*, x, v)$ and for any possible u, we have by the optimality of u_* that

$$V^*(x, v) = F(u_*, x, v) \leq V^*(x + hv, v + h \cdot (-kx + u)) + x^2 h + \lambda u^2 h + o(h)$$

  for small $h > 0$, with equality for $u = u_*$, and we can expand $F(u_*, \cdot, \cdot)$ to first order in h. It is related to what is called the "calculus of variations".)

[1] i.e. a matrix with only positive eigenvalues

(c) Using the above facts, show that u_* is a linear function, i.e. $u_* = u_{\alpha^*,\beta^*}$ for some α^*, β^* with $\alpha^* < k, \beta^* \in \mathbb{R}$.

(d) Using part (c), derive an equation for the entries of P. (In more general systems, this step yields what is called an "algebraic Riccati equation", in case you want to do further reading related to this.)

(e) Compute α^*, β^*, and P in terms of λ and k. (Note: this solves the problem of optimizing $F(u_{\alpha,\beta}, 1, 0)$ as a special case.)

Problem 3: Zero-Sum Games for Robust Classification

Zero-sum games turn out to be a very useful tool for robust machine learning (sometimes also called adversarial ML). In this question, you will exercise to formulate and solve robust ML as a zero-sum game for a few toy problems.

Suppose you are looking to design robust linear classifier to classify points in 2-D. Specifically, there are two samples, with 2-D feature vector $x_1 = (1, 1)$ and label $y_1 = 1$ for sample 1, and $x_2 = (-1, -1)$ and $y_2 = 0$ for sample 2. A linear classifier, with parameters $a \in R^2, b \in R$, predicts the label of any feature x as $\mathbb{I}(a \cdot x + b \geq 0)$ where $\mathbb{I}$ is the indicator function. For normalization reason, we always assume the l_2 norm $||a||_2 = 1$ (since re-scaling the parameters do not affect the classification outcomes at all). To characterize how good a classifier is, we need a loss function. Here, we consider the following loss of any sample (x_i, y_i)

$$l(x_i, y_i|a, b) = \begin{cases} 0 & if \quad \mathbb{I}(a \cdot x + b \geq 0) = y_i \\ ||a \cdot x + b||_2^2 & otherwise \end{cases}$$

That is, if the classification label is correct, the loss is 0; if classification label is incorrect, the loss is the *squared* l_2 norm of the distance from the classifier. The classifier is found by minimizing the total loss over samples, i.e., solving the following

$$(a^*, b^*) = \arg \min_{a,b:||a||_2=1} \sum_{i=1}^{2} l(x_i, y_i|a, b)$$

Finding a linear classifier to classify the above two samples is easy in standard classification. However, here, we study the situation with adversarial pertubation. Specifically, we assume *each* sample has the power to adversarially perturbate its feature x_i to some x'_i so long as $||x'_i - x_i||_2 \leq c$ where c captures the "manipulation power" of the adversary. For this question, we work with $c = \sqrt{2}$. In presence of such adverarial perturbation, the classifier can only observe the manipulated feature x'_i, hence classifying based on x'_i. The goal of adversarial classification is to find classifier (a^*, b^*) that minimizes the worst-case loss under any feasible adversarial perterbations.

(a) Formulate the above adversarial classification problem as a zero-sum game. Who is the max player and who is the min player? What values they are maximizing or minimizing?

(b) Find the optimal robust linear classifier for the above problem.

(c) Find the optimal robust linear classifier for a slightly harder problem with 4 data points: (1) $x_1 = (1, 1), y_1 = 1$, (2) $x_2 = (-1, -1), y_2 = 0$, (3) $x_3 = (1, -1), y_3 = 1$, (4) $x_4 = (-1, 1), y_4 = 0$,