MIT 6.S890 Topics in Multiagent Learning Fall 2024

Homework 3

Released on: Nov. 12, 2024 **Due on: Nov. 24, 2024, 11:59pm ET**

Instructions This homework contains three problems (each split into smaller tasks), for a total of 60 points.

Collaboration policy You are welcome to discuss the problems with other students. However you should write up the solutions by yourself.

1 The 0/1-Polyhedral Kernel (20 points)

As we discussed in class, the 0/1-polyhedral kernel introduces a computationally tractable way to handle the exponentially large set $\mathcal{V} \subseteq \{0,1\}^d$. Specifically, we define the 0/1-polyhedral feature map $\phi_{\mathcal{V}} : \mathbb{R}^d \to \mathbb{R}^{\mathcal{V}}$ as follows:

$$\phi_{\mathcal{V}}(x)[v] := \prod_{k:v[k]=1} x[k], \quad \forall v \in \mathcal{V}.$$

The 0/1-polyhedral kernel $K_{\mathcal{V}} : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ is defined as the inner product of the feature map:

$$K_{\mathcal{V}}(x,y) := \langle \phi_{\mathcal{V}}(x), \phi_{\mathcal{V}}(y) \rangle_{\mathcal{V}} = \sum_{v \in \mathcal{V}} \prod_{k:v[k]=1} x[k]y[k].$$

The kernel method enables handling the exponentially large space $\mathbb{R}^{\mathcal{V}}$ while requiring access only to the relatively small space $\mathbb{R}^d$. In this question, you will derive some properties of the 0/1-polyhedral kernel.

Problem 1.1 (2 points). Someone gives you a kernel $K_{\mathcal{V}}$ but doesn't tell you what $\mathcal{V}$ is. How do you find out the cardinality of $\mathcal{V}$?

Problem 1.2 (7 points). Someone gives you a kernel $K_{\mathcal{V}}$ but doesn't tell you what $\mathcal{V}$ is. How do you find out how many elements $v \in \mathcal{V}$ have the first bit hot, that is, $v[1] = 1$?

Problem 1.3 (8 points). Someone gives you a kernel $K_{\mathcal{V}}$ but doesn't tell you what $\mathcal{V}$ is. How do you find out how many elements $v \in \mathcal{V}$ have the first two bits hot, that is, $v[1] = v[2] = 1$?

Problem 1.4 (3 points). Show that, given oracle access to the kernel function $K_{\mathcal{V}}(x, 1)$ for any $x \in \mathbb{R}^d$, one can compute $K_{\mathcal{V}}(u, v)$ for any $u, v \in \mathbb{R}^d$ in polynomial time.

2 Kernel on Combinatorial Games (30 points)

In this problem, you will compute the 0/1 polyhedral kernel for certain combinatorial game instances. As some games are large, you may want to compute the kernel operations programmatically.

2.1 m-set (10 points)

An m-set refers to the decision problem of choosing m distinct actions from d available actions simultaneously. The pure strategy of the decision problem can be represented as a vector of length d, where the entries corresponding to the m chosen actions are set to 1. In this way, the vertices of an m-set are given by $\mathcal{V} := \{v \in \{0,1\}^d : \|v\|_1 = m\}$.

Problem 2.1 (5 points). For an m-set instance, explain how one can evaluate the kernel function in time polynomial in m and d.

Problem 2.2 (2 points). For an m-set instance with $d = 5$ and $m = 2$, compute the kernel function $K_{\mathcal{V}}(x, 1)$ for the vector $x[i] = i/d$ for $i \in [\![d]\!]$.

Problem 2.3 (3 points). For an m-set instance with $d = 40$ and $m = 10$, compute the kernel function $K_{\mathcal{V}}(x, 1)$ for the vector $x[i] = i/d$ for $i \in [\![d]\!]$.

2.2 Colonel Blotto (10 points)

The Colonel Blotto game is a type of allocation game in which two players each write down m *ordered positive* integers that sum to a pre-specified number S. Subsequently, the two players reveal their choices and compare the corresponding numbers. The player who has more numbers higher than the corresponding ones of the opponent wins the game.

The pure strategy of each player's decision problem can be represented as a vector of length $m \times S$, where the entry (i, j) (using a two-dimensional index for clarity) is set to 1 when the i-th number is j. Of course, only bit strings that denote valid allocations are considered.

Problem 2.4 (4 points). For an Colonel Blotto instance with $S = 6$ and $m = 3$, compute the kernel function $K_{\mathcal{V}}(x, 1)$ for the vector $x[i, j] = i/m + j/S$ for $i \in [\![m]\!]$ and $j \in [\![S]\!]$. Explain how your algorithm works.

Problem 2.5 (6 points). For an Colonel Blotto instance with $S = 40$ and $m = 10$, compute the kernel function $K_{\mathcal{V}}(x, 1)$ for the vector $x[i, j] = i/m + j/S$ for $i \in [\![m]\!]$ and $j \in [\![S]\!]$. Explain how your algorithm works. Does it always run in time polynomial in m and S?

2.3 Online Shortest Path (10 points)

Online shortest path refers to the decision problem where an agent chooses a path from the source to the sink in a directed graph $G = (V, E)$. The adversary generates a weight assignment for each of the edges simultaneously, and the agent receives a loss equal to the summation of the weights of the chosen edges.

The pure strategy of the decision problem can be encoded using a vector of length equal to the number of edges, $|E|$, where the entries corresponding to the chosen edges are set to 1. In this way, the feasible solutions of the online shortest path problem are given by

$$\mathcal{V} := \{v \in \{0,1\}^{|E|} : \text{edges } e \text{ with } v[e] = 1 \text{ form a path from the source to the sink}\}.$$

Problem 2.6 (3 points). For the online shortest path instance in Figure 1, compute the kernel function $K_{\mathcal{V}}(x, 1)$ for the vector $x[(u, v)] = (v - u)/2$ for each directed edge $(u, v) \in E$. The source vertex is 1, and the sink is 4.

Explain how your algorithm works.

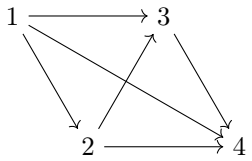

Figure 1: Online shortest path instance with source vertex 1 and sink vertex 4.

Problem 2.7 (7 points). For the online shortest path instance in Figure 2, compute the kernel function $K_{\mathcal{V}}(x, 1)$ for the vector $x[(u, v)] = (v - u)/2$ for each directed edge $(u, v) \in E$. The source vertex is 1, and the sink is 40.

Explain how your algorithm works. If the graph pattern were to be repeated, would your algorithm run in polynomial time in the number of vertices?

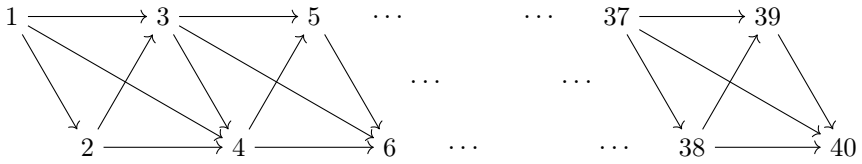

Figure 2: Graph with 40 vertices, showing two rows of vertices with directed edges. The first row consists of vertices 1, 3, 5, ..., 39, and the second row consists of vertices 2, 4, 6, ..., 40.

3 Project Progress Report (10 points)

Problem 3.1 (10 points). Give a progress report on your course project. Briefly describe the progress you have made so far. Additionally, outline any issues you have encountered (if any), and explain how you addressed or plan to address them.

6.S890 Topics in Multiagent Learning

Fall 2024