1 Monotone Submodular Maximization [Recommended]

Consider a set \( U \) of \( n \) elements that we can buy and a function \( f : 2^U \rightarrow \mathbb{R}^+ \), where for each subset \( S \subseteq U \), the value \( f(S) \) determines our profit if we buy exactly the elements of set \( S \).

We assume two properties about this profit function: (A) Function \( f \) is monotone in the sense that \( f(S) \leq f(T) \) for any two sets \( S, T \) such that \( S \subseteq T \), and (B) Function \( f \) is submodular in the sense that \( f(S \cup i) - f(S) \geq f(T \cup i) - f(T) \) for any \( i \in U \) and any two sets \( S, T \) such that \( S \subseteq T \). In simple words, the submodularity means that the marginal gain that we have by adding \( i \) to our purchase set diminishes as we move from one purchase set \( S \) to a superset of it \( T \). That is, roughly speaking, the more that we already have in the purchase set, the less extra gain by adding an element to it.

Devise an algorithm that purchases a set \( S \) of (approximately) maximum profit, subject to the constraint that \( |S| \leq k \), for some given value \( k \in \{1, 2, 3, \ldots, n\} \). What approximation factor do you get?

2 2-Approximation for Knapsack (Vazirani 8.2)

In the knapsack problem (discussed in the class), discard all elements that are larger than the budget \( B \), and then sort the remaining elements by decreasing ratio of profit to size, let this order be \( a_1, a_2, \ldots, a_n \). Let \( k \) be the smallest number such that the total size of the first \( k \) elements \( a_1, a_2, \ldots, a_k \) exceeds the budget \( B \). Pick the more profitable of the following two options: \( \{a_1, a_2, \ldots, a_{k-1}\} \) and \( \{a_k\} \). Prove that this gives a 2-approximation for the most profitable set that fits in the knapsack.

3 Connected Dominating Set

Given an \( n \)-node graph \( G = (V, E) \), a set \( S \subseteq V \) of vertices is called a Connected Dominating Set (CDS) if the following two properties are satisfied: (1) each node \( v \in V \) is either in \( S \) or has a neighbor in \( S \), (2) the subgraph \( G[S] \) induced by \( S \)—i.e., the one made of \( S \)-vertices and all edges whose both endpoints are in \( S \)—is connected. Devise an approximation algorithm for finding a minimum-cardinality CDS.
Optional/Additional Problems

4 Max-weight Matroid Base (Williamson-Shmoys 2.12)

A matroid \((\mathcal{E}, \mathcal{I})\) is defined by a ground set \(\mathcal{E}\) of elements and a collection \(\mathcal{I} = \{S_1, S_2, \ldots, S_\ell\}\) of independent subsets \(S_i \subseteq \mathcal{E}\), subject to the following conditions:

1. For any two subsets \(S\) and \(S'\) such that \(S \subseteq S'\), if \(S'\) is independent, then so is \(S\), i.e., \((S' \in \mathcal{I}) \implies (S \in \mathcal{I})\).

2. For any two independent sets \(S\) and \(T\) such that \(|S| < |T|\), there exists an element \(e \in T \setminus S\) such that \((S \cup \{e\}) \in \mathcal{I}\).

We call an independent set \(S\) a base if there is no \(T \in \mathcal{I}\) such that \(S \subseteq T\). That is, \(S\) is in some sense maximal with regard to independence.

Suppose that each element \(e \in \mathcal{E}\) has a weight \(w_e \geq 0\). Devise an algorithm that finds a maximum-weight base of the matroid.

5 \(k\)-Center

Consider a set of \(n\) points \(P = \{p_1, p_2, \ldots, p_n\}\) and a distance metric \(d : P \times P \rightarrow [0, \infty)\) where \(d(p_i, p_j)\) indicates the distance between the two points \(p_i, p_j \in P\). For a set \(S \subseteq P\), and an arbitrary point \(p' \in P\), the distance of \(p'\) to \(S\) is equal to \(\min_{p'' \in S} d(p', p'')\). The objective is to find a set \(S \subseteq P\) of \(k\) points, called centers, such that the maximum distance of any point \(p' \in P\) to the set \(S\) is minimized. Devise a 2-approximation algorithm for this problem.

6 Walking on the Hypercube

Consider the \(d\)-dimensional hypercube, which has vertex set \(\{0, 1\}^d\) and where every two vertices that differ in exactly one coordinate are connected by an edge. Suppose that each edge has a random length drawn from an exponential distribution with mean 1, and the lengths of different edges are independent. Devise an algorithm that finds a walk from the vertex \((0, 0, \ldots, 0)\) to the vertex \((1, 1, \ldots, 1)\) with expected length \(O(\log d)\).