CSE 102 Spring 2021
Advanced Homework Assignment 4

Jaden Liu
University of California at Santa Cruz
Santa Cruz, CA 95064 USA

May 5, 2021

1 AdvHW4

Greedy Approximation to 0-1 Knapsack Problem: (up to 10 points: simple proof without external resources) Consider the 0-1 knapsack problem: Given n items with weights w_1, ..., w_n and value v_1, ...v_n and a total weight of W, where each w_i, v_i and B are positive integers, find a subset S of items that a thief would like to steal so that the total weight is smaller than W and the total value is maximum.

Greedy strategy:

1. Calculate the value of each item's unit weight vi/wi
2. Pack items with the highest unit weight into the backpack.
3. If total weight of the items in the backpack does not exceed give weight maximum W, then select items with the second highest value per unit weight.
4. Loop process 2 and 3 to pack as many backpacks as possible until the backpack is full.
5. After the looping, we need to compare v_{k+1} and $v_1+v_2+v_3+\cdots+v_k$ unless we could get wrong other. Our final answer would be $\max(v_{k+1}, v_1+v_2+v_3+\cdots+v_k)$ for $1 \leq k \leq n$

Proof. First the we sort the knapsack's items in decreasing order of the value density v_i/w_i, then item n_1 has biggest value density with value v_1 and w_1. Assume we have the optimal total value v_{max}, then obviously $v_1+v_2+v_3+\cdots+v_k \leq v_{max}$. Also we can have $v_{max} \leq v_1+v_2+v_3+\cdots+v_k+v_{k+1}$. This situation is like partial Knapsack Problem. When we add partial v_{k+1} to fill the pack completely. The pack has already been at the maximum value density, hence adding partial v_{k+1} is equal to v_{max}. Not to mention adding whole v_{k+1} is larger or equal to v_{max}. Then we have to two inequalities:

$$\begin{cases} v_{max} \leq v_1+v_2+v_3+\cdots+v_k+v_{k+1} \\ v_1+v_2+v_3+\cdots+v_k+v_{k+1} \leq 2\max(v_{k+1}, v_1+v_2+v_3+\cdots+v_k) \end{cases}$$

It's easy to prove the second inequality if we consider $v_{k+1}=a$, $v_1+v_2+v_3+\cdots+v_k=b$, then $\max(a,b) \geq a$ and $\max(a,b) \geq b$. Thus $2\max(a,b) \geq a+b$.
Combine these two inequalities, we can get $v_{max} \leq 2\max(v_{k+1}, v_1+v_2+v_3+\cdots+v_k)$. Thus the

solution of our greedy algorithm is at least $\frac{V_{max}}{2}$.

For further convenience, I want to change the name "v_{max}" to "$v_{optimal}$" or "v_{opt}", and name the solution of our greedy algorithm "max($v_k + 1, v_1 + v_2 + ..v_k$)" as "$v_{max}$".

1. It's hard to compare v_{max} and v_{opt}. Therefore, we need to come up with a "bridge" to connect these two variables. It's natural to think of max$(a,b) \geq a$ or b. In order to get the coefficient "2" in the final result, we can easily come up with inequality that $2 * max(a,b) \geq a$+b. Then we find a+b, which is "$v_1 + v_2 + .. + v_{k+1}$" is always larger than v_{opt}.

2. 0-1 Knapsack problem is different from general Knapsack problem mainly because of its element divisibility. In general Knapsack problem, for example, we have fulfilled the pack with v_1 to v_k with only w_{rem}. However, we have w_{k+1} for the last element which cannot be packed in the bag. Thus, we choose w_{rem}/w_{k+1} of the last element, so the total value is $v_1 + ... + v_{k+1} * \frac{w_{rem}}{w_{k+1}}$. It's easy to prove $v_{max} = v_{opt}$ in this situation since we have chosen from most valuable to least valuable to let the final pack density reaching maximum. However, in the 0-1 Knapsack problem, the elements are not divisible. Hence, $v_1 + .. + v_{k+1}$ is always larger to v_{opt}. $\square$