Prepared for submission to JCAP

Greedy Method

Mohammed Rizin [1][1]

[1]Umemployed, Navi Mumbai, Maharashtra

E-mail: mrizin2013@gmail.com

Abstract. Greedy Method is a well know algorithm for optimization problems

ArXiv ePrint: 3423.2342

1 Introduction

Greedy Method is a well known algorithm used in variety of application of problem from self driving car to simple optimization problems. The optimization problem could be minimizing or maximing. Its just we greedily select the most optimized problem. In this paper, we will analyze the Greedy Method Algorithm and its various usage in different applications.
Suppose we have to go from city A to city B and you got 3 options:

1. By Bus
2. By car
3. Walk

In this optimization problem, you can consider the cost of travelling, the time to travel. Based on our criteria, we greedily select the best option. For e.g., You are in short of money and short of time as well, then the best option would be to choose to go by Bus. If you have lots of money , You could choose car then and so and on...

1.1 Constraint

As in above example, we didn't choose to take bus just like that. There would be questions:

Why not taking car instead of bus? Because we don't have much money.

Why dont you walk? Because we don't have much time as well.
In this example, money and time are the constraints. Based on these constraints, we choose the best route to our destination. This decision-making process ensures that the solution is feasible and satisfies the given constraints.

[1] He's Amazing

1.2 Feasibility

For example, in the above scenario, choosing to walk would not be feasible if time is a critical constraint. Similarly, choosing to travel by car would not be feasible if money is a critical constraint. A solution is considered feasible if the solution to the problem satisfies all the constraints of the problem. If any one of constraint is not satisfied then it is not feasible solution. There could be multiple feasible solution and also no feasible solution for some constraint.

1.3 Optimality

The Greedy Method ensures that the solution is not only feasible but also optimal for the given problem. Optimality means that the solution is the best among all feasible solutions. For instance, if both time and money are limited, the optimal solution would be to take the bus, as it balances both constraints effectively. There can be only one optimal solutions. Problems that require either minimum or maximum results is called **Optimzation Problems**

Other Optimzation Strategies:

1. Greedy Method
2. Dynamic Programming
3. Branch and Bound

1.4 Applications

The Greedy Method is widely used in various fields, such as:

- **Graph Algorithms:** Algorithms like Dijkstra's shortest path and Prim's minimum spanning tree use the Greedy Method.
- **Scheduling Problems:** Tasks like job scheduling and resource allocation often rely on greedy strategies.
- **Optimization Problems:** Problems like the Knapsack problem and Huffman encoding are solved using greedy approaches.

2 Methodology

Greedy Method says that a Problem should be solved in stages. Basically. It checks each and every input one after another and check if the input is feasible and add to the list.
Suppose we have array of inputs:

1	35	52	52	2
0	1	2	3	4

Figure 1: Sample Array

Here's the Algorithm:

Algorithm 2.1 Greedy Method

```
procedure GREEDY($a, n$)
    for $i \leftarrow 1$ to $n$ do
        $x \leftarrow Select(a)$
        if $Feasible(x)$ then
            $Solutions \leftarrow Solutions + x$
        end if
    end for
end procedure
```

$a \leftarrow$ Problem we are solving.
$a_i \leftarrow$ is input for that problem.
Suppose you are buying a car. Then you started looking at all the cars in the market and you finally select. But is this reliable. **No**. This method is very very time-consuming But yeah this gives the best car.
But what if you have selection procedure like You want only Toyota or Cadillac. Based on this contraint, You select some cars. This is stage 1. Then you would say, I want only sedan type cars. This will take (feasible) the sedan cars from Toyota and Cadillac only list. This is stage 2. This will go on and on... We find our best car. But is that the best car? According to you, yeah. According to your selection procedure (eg choosing only toyota and Cadillac cars), this is best (optimal car). This would reduce the hurdle of checking every single car in the market by filtering out stage by stage.
Lets look at some Problems in the following sections.

3 Knapsack Problem

Suppose you have n of objects to choose from to fill into your bag to sell it at the next town. The profit of each object and weight of each object is also given. You have to choose the right objects into the bag so that it doesn't exceed the given weight of the bag and also maximizes the profit. For our case, we have 7 elements and the limit of our sack is $\boldsymbol{15Kg}$. If the total weight of all objects is greater than the capacity of the bag. Then this is Knapsack Problem.

Objects	Profits	Weights
1	10	2
2	5	3
3	15	5
4	7	7
5	6	1
6	18	4
7	3	1

These objects can be divided, we can take fraction of the objects, then only we can use Knapsack Method.
To solve this problem using the Greedy Method, we calculate the profit-to-weight ratio for each object and sort the objects in descending order of this ratio. Then, we select objects greedily until the bag's weight capacity is reached. So the selection procedure we follow is

to take the maximum profit-to-weight ratio objects. That is our selection strategy for this problem.

Why profit-to-weight ratio? The reason we choose profit-to-weight. Since the constraint of this problem is the total weight of objects or partial objects should be less than or equal to the weight of the bag. If we know the profit of an object for 1 kg, and we greedily fill based on that value, we get the maximum profit because we are choosing the maximum profit-to-weight ratio in every single step

Objects	**Profits**	**Weights**	**Profit/Weight Ratio**
1	10	2	5
2	5	3	1.67
3	15	5	3
4	7	7	1
5	6	1	6
6	18	4	4.5
7	3	1	3

Table 1: Object with its Profit/Weight Ratio

Steps to Solve:

1. Sort the objects by their profit-to-weight ratio in descending order.
2. Start adding objects to the bag until the total weight exceeds the bag's capacity.
3. If adding an object exceeds the capacity, add a fraction of the object to fill the bag.

Solution:

- Add object 5 (Profit = 6, Weight = 1).
- Add object 1 (Profit = 10, Weight = 2).
- Add object 6 (Profit = 18, Weight = 4).
- Add object 3 (Profit = 15, Weight = 5).
- Add object 7 (Profit = 3, Weight = 1)
- Add $\boldsymbol{\frac{2}{3}}$ of object 2 (Profit = $\boldsymbol{\frac{10}{3}}$, Weight = $\mathbf{2}$).

Total Profit: $\boldsymbol{\sum x_i \cdot p_i = 6 + 10 + 18 + 15 + 3 + 3.333 = 55.333}$
Total Weight: $\boldsymbol{\sum x_i \cdot w_i = 1 + 2 + 4 + 5 + 1 + 2 = 15}$
There is also Zero-One Knapsack problem where you are not allowed to take fraction of an object. Either you can or you can't include that element.