

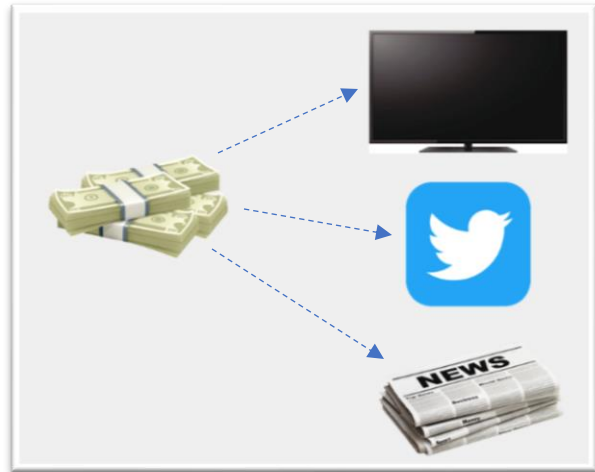
Assignment 2 – Marketing Budget Allocation Problem

About the problem:

Marketing can help scale and sustain a business, but it comes with a cost! It is expensive to run paid campaigns, so careful planning is necessary. Given a marketing budget and a set of marketing channels, we would like to allocate the budget to the different marketing channels such that the total profit is maximized.

What you are required to do:

Write a genetic algorithm to solve the marketing budget allocation problem.



What the input looks like:

You'll be given the marketing budget, the set of marketing channels, the return-on-investment (ROI) of each channel and a set of constraints (on investing in each channel). For simplicity, we will assume we have only 2 types of investment constraints:

- An investment in marketing channel (i) **can't be less than** some value (in thousands)
- And/or an investment in marketing channel (i) **can't exceed** some percent of the total budget.

These constraints form the lower and upper bounds of investment in each channel.

Example:

```
Enter the marketing budget (in thousands):
100

Enter the number of marketing channels:
4

Enter the name and ROI (in %) of each channel separated by space:
TV advertising 8
Google 12
Twitter 7
Facebook 11

Enter the lower (k) and upper bounds (%) of investment in each channel:
(enter x if there is no bound)
2.7 58
20.5 x
x 18
10 x

Please wait while running the GA...

The final marketing budget allocation is:
TV advertising -> 2.7k
Google -> 76.2k
Twitter -> 5.6k
Facebook -> 15.5k

The total profit is 11.457k
```

Notes:

- The ROI is given as a percentage.
- In this example, the investment in TV advertising can't be less than 2.7k and can't be more than 58% of the total budget. The investment in marketing through Google can't be less than 20.5k and there is no restriction on the upper bound. The investment in marketing through Twitter can't exceed 18% of the total budget and there is no restriction on the lower bound. The investment in marketing through Facebook can't be less than 10k and there is no restriction on the upper bound.
- Assume a single currency is used in the entire problem.
- Your final solution doesn't have to match the one in this example, but it should produce high profit. You could also display the output in a different format.

Important remarks:

1. The first thing you need to do is to think about how you will encode the solution in your chromosome. Formulate the problem with the objective function and constraints to help you figure out the next steps.
2. You should use a **floating-point representation** of the chromosome.
3. The population size and initialization method you use are up to you. You can actually try different population sizes to see how this will affect your results. The maximum number of generations is also up to you.
4. Think about how you will handle **infeasible solutions**. Infeasible solutions are solutions that violate the constraints of the problem; therefore, they are not allowed.
5. Implement **tournament selection, 2-point crossover** and **elitist replacement**. Implement both **uniform and non-uniform mutation**.
6. Run the algorithm **20 times** with the same input, **write the output to a file** and **print the best solution** obtained from these experiments.

Assignment submission notes:

- The **maximum** number of students in a team is **3** and the **minimum** is **2**.
- The **deadline** is 20/11/2020 and **no late submission** is allowed.
- Please submit **one compressed folder**. The folder name should follow this structure: ID1_ID2_ID3_MAJOR_MINOR_GROUP
- **Cheating** students will take **negative grades** and no excuses will be accepted. If you have any problems during the submission, contact your TA but don't, under any circumstances, give your code to or take the code from your friends.

Grading Criteria: (6 marks)

| | |
|---------------------------------------|-----|
| Representation and initialization | 0.5 |
| Fitness function | 0.5 |
| Handling infeasible solutions | 1 |
| Selection | 1 |
| Crossover | 0.5 |
| Mutation | 1 |
| Replacement | 0.5 |
| Output (file + best solution printed) | 1 |

Good luck