OHM Term Project - Spring 2021

Optimizing Bank Lending Decisions Using Metaheuristics

Name: Darshan Santosh Gangurde
RN: 18IM10010

Python Notebook:
https://colab.research.google.com/drive/1457IbZw29qq1cHEbCTvhwCC1m-hSWER6?usp=sharing

Introduction:

- The problem of bank lending decision in a credit crunch environment- where all applicable customers are eligible to get the desired loan is an NP-hard optimization problem that can be solved using meta-heuristic algorithms such as evolutionary algorithms (Ex.Genetic Algorithm)

- Genetic Algorithm (GA) can be used to organize bank lending decision in a highly competing environment with credit crunch constraint.

- The main focus of the GA model is two-fold:
1. To stabilize banks systemically while achieving maximum profit, and
2. To establish the capital base so that banks would increase lending efficiently

- This paper proposes an efficient, GA-based model is developed to maximize bank profit in lending decision. The lending decision is dynamically decided based on customer's loan characteristics

A) Using Genetic Algorithm

The GA's fitness function (Fx) simply consists loan revenue (θ), loans cost (μ), total transaction cost (π), and cost of demand deposit (β). The main objective is to maximize F(x)

Loan revenue:

$$\vartheta = \sum_{i=0}^{n} (r_L L - \lambda)$$

Loans cost (μ):

$$\mu = \sum_{i=0}^{n} L\delta$$

Total transaction cost (π):

$$\varpi = \sum_{i=0}^{n} r_L T \qquad\qquad T = (1-K)D - L$$

cost of demand deposit (β):

$$\beta = r_D D$$

The Fitness Function(Fx):

$$F_x = \vartheta + \varpi - \beta - \sum_{i=0}^{n} \lambda$$

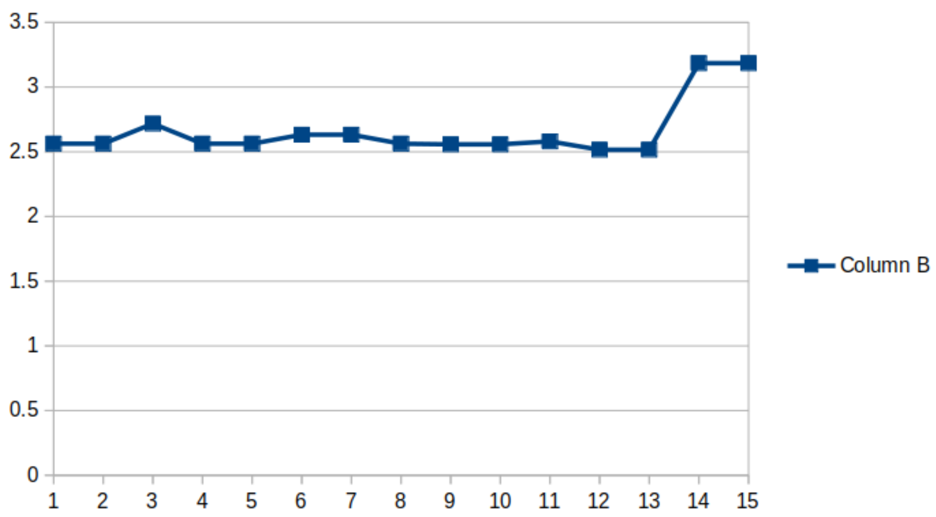
Steps Followed in GA :

1. Generate Initial Random Solution-Chromosomes are generated according to given constraint
2. Calculate the fitness of each chromosome
3. Roulette Wheel Selection for creating the parent pool
4. Do crossover and Mutation using givren probabilities
5. Add favourable children to the population
6. Find the best solution occured during interations

Observations:

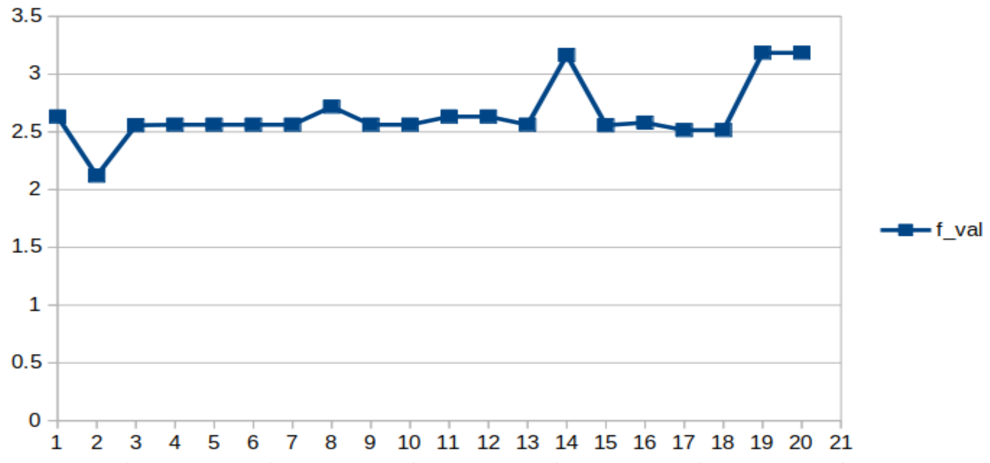
Observations for 15 iterations

# iterations	F_max
1	2.562
2	2.562
3	2.7175999999
4	2.562
5	2.562
6	2.6324
7	2.6324
8	2.562
9	2.5574
10	2.5574
11	2.58
12	2.5154
13	2.5154
14	3.185
15	3.185



Observations for 30 Iterations:

# Iterations	F_max
1	2.6324
2	2.12059999
3	2.5574
4	2.562
5	2.562
6	2.562
7	2.562
8	2.71759999 99
9	2.562
10	2.562
11	2.6324
12	2.6324
13	2.562
14	3.1654
15	2.5574
16	2.58
17	2.5154
18	2.5154
19	3.185
20	3.185



Conclusion:

1. Roulette wheel selection helps in passing better solutions for reproduction
2. The crossover part gives the exploitation part(local Search) of search.
3. The mutation helps in escaping local optima and exploring the global best.
4. Final solutions also depends on number of iterations and the crossover and mutation probabiities
5. With given approach and parameters, GA struggles to escape the local optima in many cases.

B) Using Amalgamation of GA and Simulated Annealing

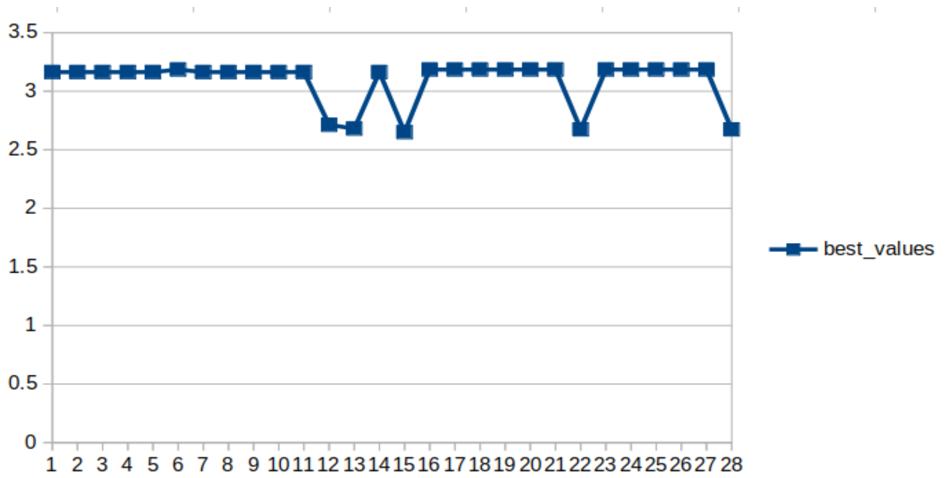
- The population pool in GA keeps on changing irrespective to the betterment of solution.
- To avoid this GA and Simulated annealing can be used together.
- The new population is accepted in this case only if the solution is satisfiying the acceptance criteria of SA
- This helps the algorithm to accept better solutions.

Steps Followed:

1. Define the parameters used in Simulated Annealing.
2. Create a GA solution for each iteration at perticular temperature
3. Check whether the given solution satisfies the acceptance criteria of SA
4. Update the population accordingly.

Observations:

# Steps	F_max
1	2.6594
2	3.1636
3	3.1636
4	3.1636
5	3.1636
6	3.1636
7	3.1636
8	3.1854
9	3.1636
10	3.1636
11	3.1636
12	3.1636
13	3.1636
14	2.713
15	2.681
16	3.1636
17	2.6508
18	3.1854
19	3.1854
20	3.1854
21	3.1854
22	3.1854
23	3.1854
24	2.6744
25	3.1854
26	3.1854
27	3.1854
28	3.1854
29	3.1854
30	2.6744



Conclusion:
1. From observations it can be observed the steps in simulated annealing gives multiple solutions obtained from GA
2. With addition of SA , GA can escape the local optima and search for global optima
3. The acceptance criteria of SA avoids the selection of worse solutions.
4. With multiple populations being reproduced , the probability of reaching at global optima is high.
5. This gives better solution than Genetic Algorithm alone.