

Assignment 1 – Report

IT3708 – Bio-Inspired Artificial Intelligence

Fredrik Robertsen

February 9, 2026

Source code available at:

<https://git.pvv.ntnu.no/frero-uni/IT3708>

Introduction

The binary knapsack problem and its derivations like feature selection are NP-hard, meaning no polynomial-time solution exists. Genetic algorithms can approximate solutions by tweaking hyperparameters. This report details solving the knapsack problem and feature selection.

Background & Setup

The binary knapsack problem finds the optimal combination of items to maximize value without exceeding weight capacity. Without the capacity limit, this becomes feature selection, relevant to regression and machine learning.

Genetic algorithms approximate solutions by finding global optima in a search space. Items or features are encoded as bits in a bitstring representing individuals/chromosomes. Starting with a population, we repeat until satisfied:

- 1) select parents from population;
- 2) perform crossover with pairs of parents;
- 3) mutate offspring.

I implemented a modular genetic algorithm in over a thousand lines of [Odin](#) code with memoized fitness values for performance. [Uiua](#) handles plotting.

Results & Reflection

Running the algorithm

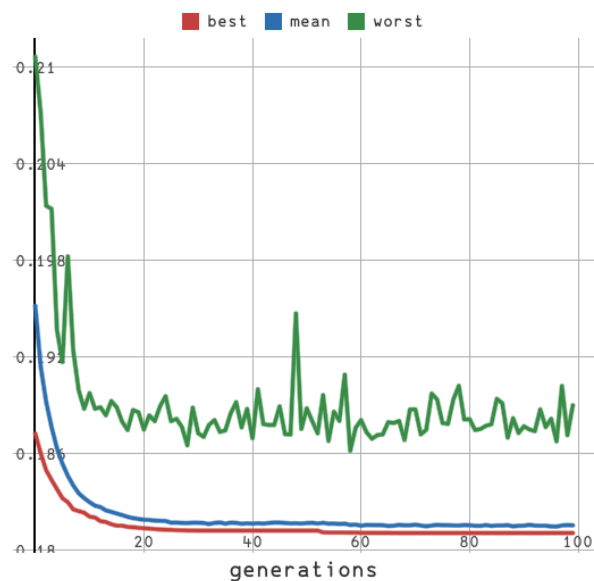
Initial output with random parent selection, 70% single-point crossover, 1% bit-flip mutations and full generational replacement:

1	Baseline RMSE: 0.1952
2	Gen 0: Best=0.1885 Mean=0.1957
	Worst=0.2062 Entropy=49.5620
3	...
4	Gen 99: Best=0.1920 Mean=0.1974
	Worst=0.2088 Entropy=40.3545

Results worsened because random parent selection doesn't approach optima [1].

Best and Worst RMSE

Baseline RMSE (all features selected) is 0.195. Using tournament selection with $k = 10$, $\mu = 1000$, $P_C = 0.7$ and $P_M = 0.01$:

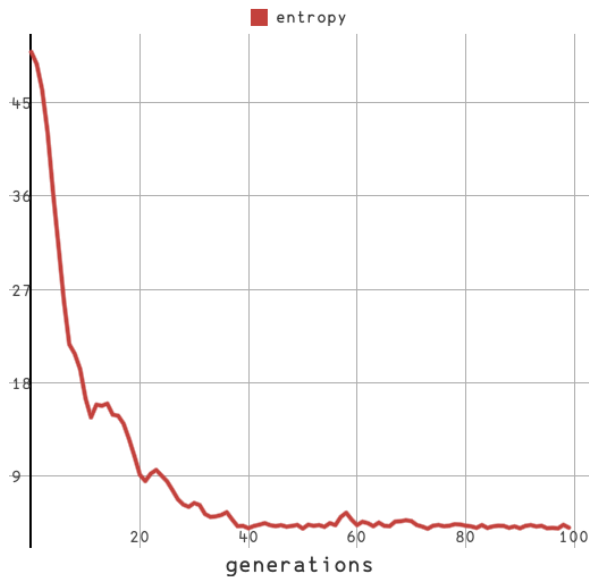


Roulette selection yields identical results. All hyperparameter configurations I've tested converge near 0.1811.

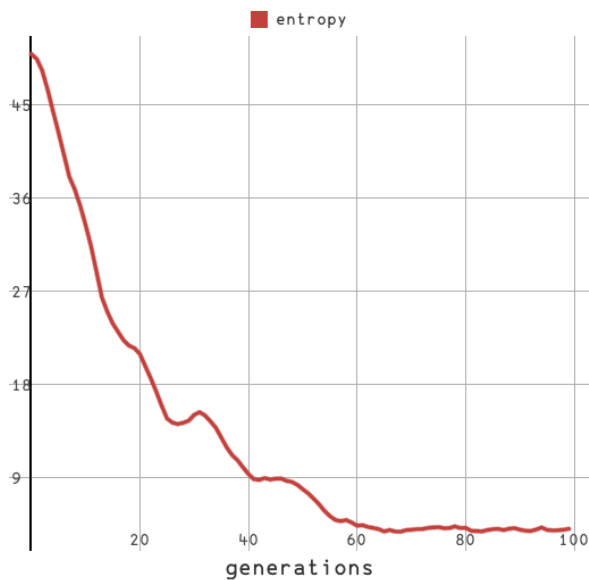
Crowding

Crowding maintains diversity through niching. Despite testing deterministic and probabilistic crowding, no configuration improved beyond 0.1811. The seed (42) may be stuck at a local optimum.

Generational replacement entropy:



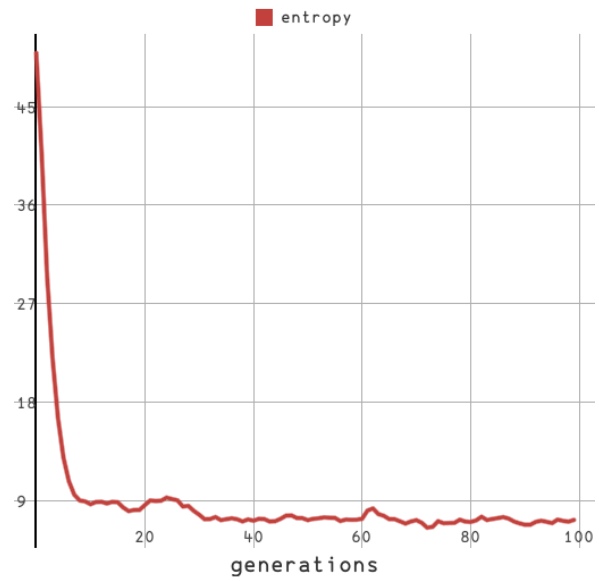
Probabilistic crowding entropy:



Probabilistic crowding maintains entropy longer despite high selection pressure.

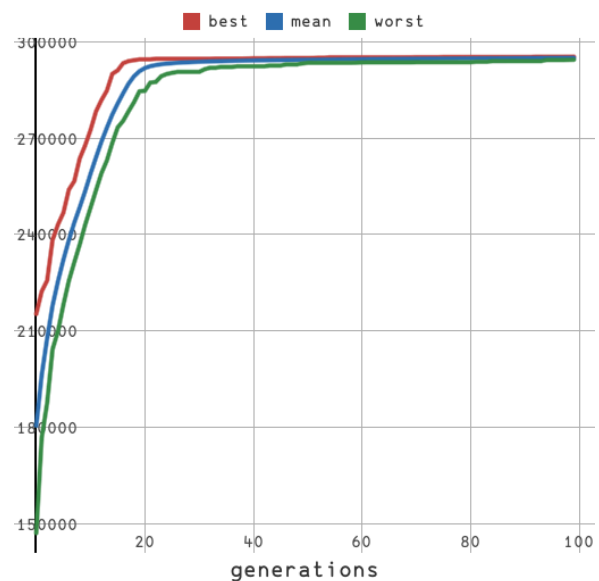
Elitism

With probabilistic crowding, 10 elites, roulette selection, $P_M = 0.02$ and $P_C = 0.8$:

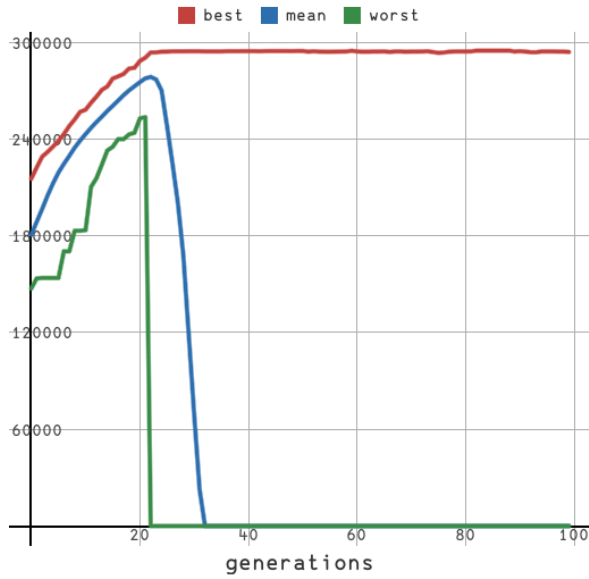


Comparing with the Knapsack Problem

Deterministic crowding approaches optimum (295246):



Probabilistic crowding has a fitness penalty bug causing negative values:



Conclusion

Balancing selection pressure and diversity requires careful hyperparameter tuning.

Further work

Potential improvements include fixing implementation flaws, comparing multiple configurations in single plots, automating parameter tuning, and further modularizing the code.

References

- [1] T. Pavlic, “Guide to Tuning the Many Hyperparameters of a Genetic Algorithm.” Accessed: Feb. 08, 2026. [Online]. Available: <https://www.youtube.com/watch?v=TwZxTuU8LUI>