

Assessing lineage information in genetic programs

Gary D. Boetticher and Jason Rudisill

ABSTRACT—Although much research has been performed with genetic programs (GPs) over the last decade, very little research was been applied to the use of lineage within a GP. Most likely, researchers are using genetic programs that only apply elitism (they only carry over one chromosome with the highest fitness value to the next generation) or that have an arbitrarily hard-coded percentage of lineage applied. This research investigates several different lineage percentages to determine which is most beneficial to a genetic program. Six experiments are performed, and a lineage percentage is statistically superior in 60 percent of the cases.

INTRODUCTION

Traditionally, genetic programs (GPs) solve problems by generating a set of mathematical equations, or chromosomes, that represent a mapping between two sets of variables. Collectively, these chromosomes form a population. GPs repetitively breed new generations of chromosomes seeking to find an optimal, or at least satisfactory, solution. Normally, a GP experiment runs until a satisfactory solution is found, until the GP runs for n generations, or until the user terminates the inquiry.

GPs are frequently deployed on large, complex, noisy datasets in which the search space is extremely large. Finding a solution within the search space is an extremely difficult challenge. For example, theoretically there may be billions of equation permutations. If a GP consists of 1,000 chromosomes and runs for 1,000 generations, then the GP generates at most one million possible equations (1,000 chromosomes * 1,000 generations). Such an experiment would cover less than one-tenth of a percent of the total search space!

Boetticher¹ demonstrated the value of using chromosome lineage information for performing better breeding within a population of equations. In this context, the top 20 percent of the population produced statistically superior results across five different experiments.

The next step, and theme of this research, explores various lineage scenarios for optimizing the breeding process. This extends the proof-of-concept from the previous paper by examining different lineage ranges from 2 percent to 50 percent.

The key question emerges, “What is an optimal, or at least preferred, amount of lineage to apply to a genetic program?” Our hypothesis is that the small (2 percent and 5 percent) and large (33 percent and 50 percent) lineage percentages will not significantly improve the r^2 or fitness values. Instead, the mid-



Gary D. Boetticher

dle percentages (10 percent through 20 percent) will cause an improvement in these values. The next section describes how the experiments are set up and conducted to test this hypothesis.

Six different synthetic datasets are created for conducting experiments using six lineage percentages ranging from 2 percent up to 50 percent.

The findings show that there is a statistically significant difference among the lineage percentages. (See Figure 1.)

LINEAGE IN GENETIC PROGRAMS

Once a new generation is created in the GP modeling process, all legacy information about the previous generation is discarded. Perhaps this discarded ancestral fitness information could offer valuable clues on how to make better propagation decisions. A chromosome’s lineage refers to its fitness values across multiple generations.

Lineage differs from elitism in that elitism carries chromosomes into the next generation without modification. Lineage identifies a subset of the population with very high fitness values and breeds this subset until a full population is attained. It greatly rewards chromosomes that best fit the dataset. Figure 2 provides a lineage example in which the top 10 percent of the population breeds the next generation.

Previous research performed by Boetticher and Kaminsky shows that “lineage-based GP modeling produces better results faster.”² Their experiments were performed on five datasets, and they show that repeatedly breeding the top 20 percent of chromosomes produces better average fitness and r^2 results than cross-breeding the middle or bottom 20 percent of chromosomes.

However, lineage can be viewed as a burden to a genetic program because it restricts the ability of the GP to create new chromosomes. When lineage is applied, chromosomes are copied from the previous generation; as such, the available

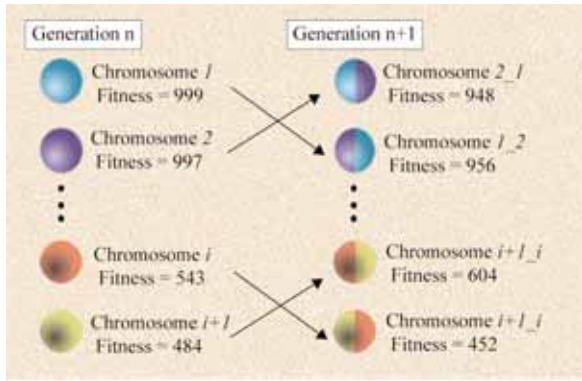


Figure 1. Tracking fitness values for two generations

chromosome space is restricted for newly created, crossed-over, or mutated chromosomes. This prevents “new life” from being breathed into the chromosome space.

This raises the question of how to find the optimal amount of lineage to apply. Applying too much lineage can result in a thrashing situation. Applying too little lineage results in a vanilla-based GP. The next section explains a series of experiments to find an optimal, or at least preferred, lineage setting.

GP LINEAGE EXPERIMENTS

The goal of the lineage experiments is to determine if there is a significant difference among different lineage settings. If there is a statistical difference among the populations, the experiments should show the best lineage percentage to use in a genetic program.

Each dataset will be executed 20 times for each of the following lineage percentages: 2, 5, 10, 20, 33, and 50. These percentages were selected to cover a wide range of potential “best values.” In total, 720 executions are performed: 6 datasets * 6 lineage percentages * 20 repetitions.

General Experimental Settings

To perform the genetic program lineage experiments, six different data sets were created. Each data set included 200 rows of data, and the data for each attribute was randomly generated. The output data for each dataset was slightly perturbed to prevent the GP from prematurely converging on the solution. What follows are the equations used in the six experiments.

- $Z = A * B + C * 5$ (1)
- $Z = C - (D * B) + (C + D)/A$ (2)
- $Z = A * B * C + (D + D)/B$ (3)
- $Z = A * A + B + B - C - C - (C - C)/B$ (4)
- $Z = \text{SIN}(A - C) * B + C * C$ (5)
- $Z = A/C + B * B - A$ (6)

All lineage experiments run for a maximum of 100 generations, each generation contains 1,000 chromosomes, and the chromosomes are constrained to a maximum length of 250.

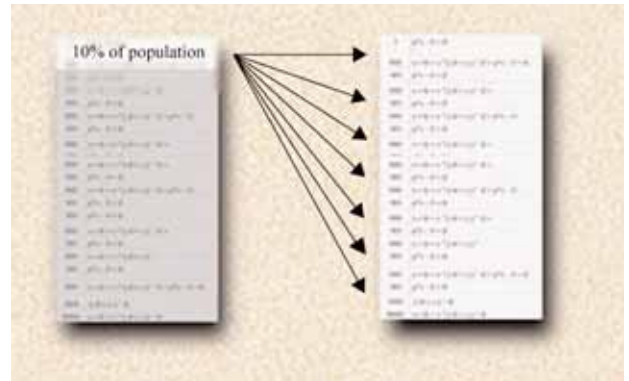


Figure 2. Lineage example, top 10 percent breed next generation

Table 1. Dataset 1 results

Lineage	Avg. Fitness	Avg. r ²	Avg. SE
2%	447.4	0.8516	8.2590
5%	371.8	0.8438	9.3571
10%	518.6	0.8734	7.3350
20%	558.8	0.8912	7.0415
33%	475.8	0.8730	8.1388
50%	467.9	0.8669	8.2571

Table 2. Dataset 2 results

Lineage	Avg. Fitness	Avg. r ²	Avg. SE
2%	587.6	0.9146	12.1223
5%	447.5	0.8530	13.4907
10%	603.3	0.9161	11.9657
20%	619.1	0.9264	11.3906
33%	501.7	0.8589	13.5644
50%	387.6	0.6944	19.5798

Table 3. Dataset 3 results

Lineage	Avg. Fitness	Avg. r ²	Avg. SE
2%	153.8	0.6383	59.7340
5%	134.1	0.5887	62.3474
10%	109.0	0.5860	64.3333
20%	118.8	0.6125	57.4806
33%	104.3	0.6304	49.8423
50%	180.0	0.6300	58.8662

Additionally, the experiments are repeated 20 times for each of the data sets, and each dataset is tested against six lineage options: 2%, 5%, 10%, 20%, 33%, and 50% lineage carryover. Thus, 720 experiments (6 data sets * 6 lineage options * 20 experiments) are performed.

Experiment Results

Tables 1-6 show the average results for the six equations and six lineage percentages. The fitness, r², and standard error

Table 4. Dataset 4 results

Lineage	Avg. Fitness	Avg. r ²	Avg. SE
2%	677.6	0.9337	6.9769
5%	698.8	0.9424	6.7688
10%	732.3	0.9456	5.8188
20%	703.7	0.9433	6.5176
33%	755.3	0.9526	5.5396
50%	612.4	0.9251	8.0236

Table 5. Dataset 5 results

Lineage	Avg. Fitness	Avg. r ²	Avg. SE
2%	455.7	0.8805	10.6294
5%	493.0	0.8947	10.1035
10%	493.1	0.8940	10.1736
20%	534.9	0.9053	9.5290
33%	491.5	0.8953	10.3288
50%	465.1	0.8848	10.5193

Table 6. Dataset 6 results

Lineage	Avg. Fitness	Avg. r ²	Avg. SE
2%	409.3	0.8712	48.6775
5%	436.7	0.8810	45.2063
10%	390.2	0.8646	48.6575
20%	410.5	0.8718	48.9126
33%	418.5	0.8749	48.2298
50%	398.6	0.8673	50.6929

Table 7. Average dataset results

Lineage	Avg. Fitness	Avg. r ²	Avg. SE
20%	490.9	0.8584	23.4786
2%	455.3	0.8483	24.3998
33%	457.8	0.8475	22.6073
10%	474.4	0.8466	24.7140
5%	430.3	0.8339	24.5456
50%	418.6	0.8114	25.9898

Table 8. T-test of average values for various lineage settings

	Lineage Setting = 20%
Lineage Setting = 2%	0.0374
Lineage Setting = 5%	0.0031
Lineage Setting = 10%	0.2401
Lineage Setting = 33%	0.0995
Lineage Setting = 50%	0.0006

(SE) values are displayed. (Each dataset and lineage percentage was evaluated 20 times, and the average value is displayed.) The best experimental results for each dataset are boldfaced.

Finally, the average of all the results from the six experiments is presented. The results are sorted by r².

DISCUSSION

With the exception of Dataset 3, all experiments have an evident winner in terms of best lineage value. The 20 percent setting produces the best results in three of the six experiments.

The results for Dataset 3 appear to be suspect. It had the lowest fitness and correlation values. Results from the data set appear less reliable in assessing the benefits of using specific lineage settings. Thus, Dataset 3 is removed from the final analysis.

Table 8 shows various t-tests between the fitness values of the overall best lineage setting, 20 percent, versus the other respective values.

The boldfaced values show that a lineage setting of 20 percent is statistically superior (alpha = 0.05) in three out of the five cases. If alpha is set to 0.1, then a 20 percent setting is statistically superior in four out of five, or 80 percent of the cases.

CONCLUSIONS

Based on the experiments, a 20 percent lineage is the preferred setting. It is statistically superior in 60 percent of the experiments. This information is extremely helpful for those seeking to build better models using genetic programs.

REFERENCES

1. Boetticher, G. and Kaminsky, K. Building a genetically engineerable evolvable program (GEEP) using breadth-based explicit knowledge for predicting software defects. *Proceedings of the IEEE Annual Meeting of the North American Fuzzy Information Processing Society*. Banff, Alberta, Canada, **1**, 10-15 (June 27-30, 2004).
2. Boetticher, G. and Kaminsky, K. The assessment and application of lineage information in genetic programs for producing better models. *IEEE Information Reuse and Integration Conference*, Waikoloa, HI, Sept. 16-18 (2006).