BITI 3143 EVOLUTIONARY COMPUTING
SEM 2 19/20 GROUP'S PROJECT

TITLE : FEATURE SELECTION BY USING GA IN NN PREDICTION MODEL

TANG LI HO	**B031810122**
MUHAMMAD FARID BIN ED NOOR	**B031810455**
AFIFI BIN KHALILI	**B031810123**
AGILAN A/L MANIVASAGAM	**B031810281**

ABSTRACT : This study is to improve the performance of prediction on a dataset by using genetic algorithm based feature selection. The performance is based on the accuracy and the number of selected attributes. It is also a testing on the efficiency of genetic algorithm in feature selection.

1. INTRODUCTION

Genetic Algorithm (GA) is a search-based optimization technique based on the principles of Genetics and Natural Selection. GA is a type of artificial intelligence algorithm to find out optimal or near-optimal solutions of a problem. In this study, we are using GA in feature selection to improve the performance of prediction on a dataset.

In this era of information-driven world, we need not worry on not enough information, instead of it our problem has been changed to be how to extract out the useful information. Even though there are various type of machine learning techniques or algorithm in appling prediction, it is still hard to remove the noise from the dataset. In order to solve with this problem, a new concept called feature selection has been invented, which can be used by combining with various machine learning techniques and genetic algorithm.

2.IMPROVED PREDICTION USING FEATURE SELECTION

2.1 PROBLEM :

With the data Connectionist Bench (Sonar, Mines vs. Rocks) Data Set that we get from UCI Machine Learning Repository which contains 61 variables include the label and totally 208 instances. However, the performance of using neural network prediction in this dataset is not good enough, it may contains lot of useless attributes which has became noise and affect the performance of training In order to get the better analysis result, we have designed a feature selection project by using genetic algorithm to filter out the unimportant attributes. It is an optimization problem on genetic algorithm.

2.2 OBJECTIVE :

To improve the accuracy of NN model by genetic based feature selection on the dataset.

3.FEATURE SELECTION BASED ON GENETIC ALGORITHM

- **Basic setting for GA training**

In this project, the chromosome will has 60 bits as there has 60 non-labeled attributes inside the dataset, the gene is in binary form, 1 means using the attribute at that position while 0 means not to use. The population size has been set as 40 because too large in population size will cause longer time to train up a model and get the fitness in a generation but too small will cause crossover's feature extraction become meaningless.

- **Initialization**

Because of we do not know which attributes will give positive impact on the performance of prediction, so the initial chromosomes is random generated at all. The only thing that we have some limitation on the chromosome which has all genes as 0 (no attributes be selected), if it appears then its fitness will directly become 0(worst),

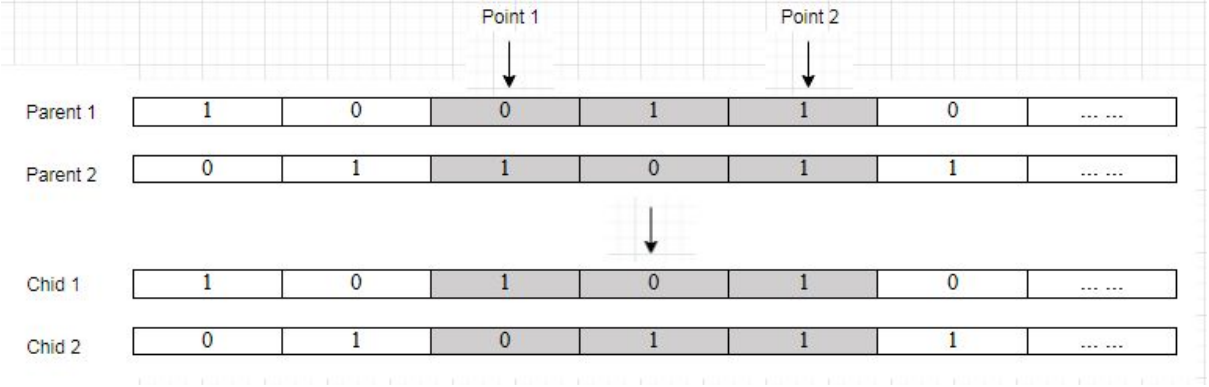
- **Parent Selection**

Tournament Selection is used in the parent selection process. 2 random individuals will be selected from the population, the one with higher fitness will be kept to be a parent. This process will be repeated to select the second parent and it will stop if the two selected

parents are not from the same position (it is possible that a chromosome appears multiple times in the same generation).

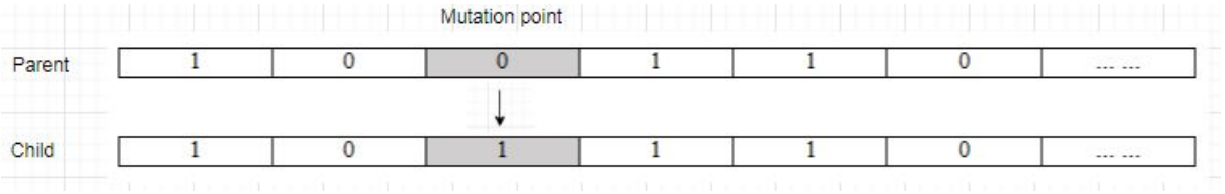
- **Crossover**

Crossover function is working as a convergence operation to pull the population towards a better performance by extracting the useful genes. The way of getting the new chromosome is by recombination where we use 2-point crossover. 2 random points will be selected and cut to do crossing between 2 parents to form 2 new children.



- **Mutation**

The mutation function is working as a divergence operation to explore more possible solutions. As the gene is in binary form, Bit Flip Mutation is selected to be used in GA training. After the mutation,the genes will be inverted from before(0 to 1 or 1 to 0).



- **Survival Selection**

The selected survival method is Age-based method, all the chromosomes in previous generation will be replaced by their children. After the competition in parent selection and updated by crossover & mutation, these 2 children will replace their parents to be part of the new generation. The processes from parent selection until survival selection will repeat 5 times to generate 10 new children ready for next generation. As the probability of crossover and mutation is not high, some of the children may be same as their parents.

- **Fitness function**

The fitness value is calculated by the validation accuracy from a single hidden layer neural network and the number of selected attributes. In order to ensure that the result will not drift unexpectedly, the random seed for the model is fixed and redefined for every training process and the same training set (128 rows) will be used for every turn. The reason of using only 128 rows of data is to prevent overfitting in training the model. The accuracy of the model will be tested by the whole dataset (all rows) so that it will be fair for testing in every chromosome.

The fitness for the GA training is based on both accuracy of the model and the number of selected attributes but with different weights. Because of it is far more important in accuracy than number of attributes, we set the weight of 0.99 for accuracy and 0.01 for number of attributes. The fitness function is :

fitness = 0.99(A) + 0.01(60 - N) / 59

where A = validation accuracy from the model

N = number of selected attributes

60 is the total number of attributes except the label

the minimum number of attribute should be 1

For this fitness function, the higher fitness means better performance, the highest score will be 1.0. The reason for taking the number of attributes as part of the fitness calculation is to solve the condition that 2 different chromosomes with same accuracy but different number of attributes, it is common to see that less number of attributes will be better.

- **Termination strategy**

The GA training will be terminated under 2 conditions:

- Completed 30 generations
- The fitness reaches 1.0 (100% accuracy with only 1 selected attributes)

In most of the conditions, the fitness is impossible to reach 1.0, hence it can be considered that the training will only be terminated after 30 generations.

4.EXPERIMENTAL RESULT

4.1 EXPERIMENT TESTING

For testing the efficiency of different parameter setting and their stability, 3 different set of parameter settings are prepared and each of them will be run for 3 times.

Blue line : Best fitness

Orange line : Average fitness

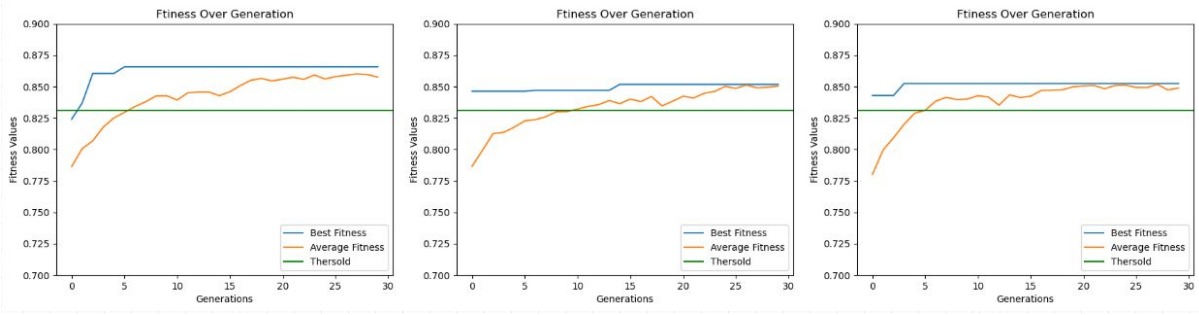
Green line : Threshold (the fitness of using all 60 attributes in training the model)

Graph y-axis range : 0.70 - 0.90

Graph x-axis range : 0 - 30 (30 generations)

Parameter setting 1:

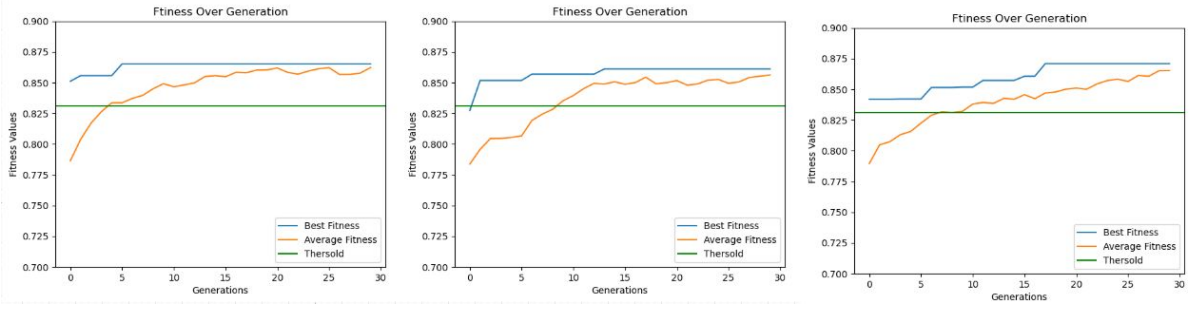
population size = 40, crossover rate = 0.4 , mutation rate = 0.1



From the 3 graphs above, we can see that evaluations occur only few times, the average line is so close to the best fitness line which show that the evaluation is nearby reaching its limit and a slow evaluation speed. The no enough rate of crossover and mutation cause it to hard to improve itself, it just show a small improvement at last.

Parameter setting 2:

population size = 40, crossover rate = 0.4 , mutation rate = 0.2

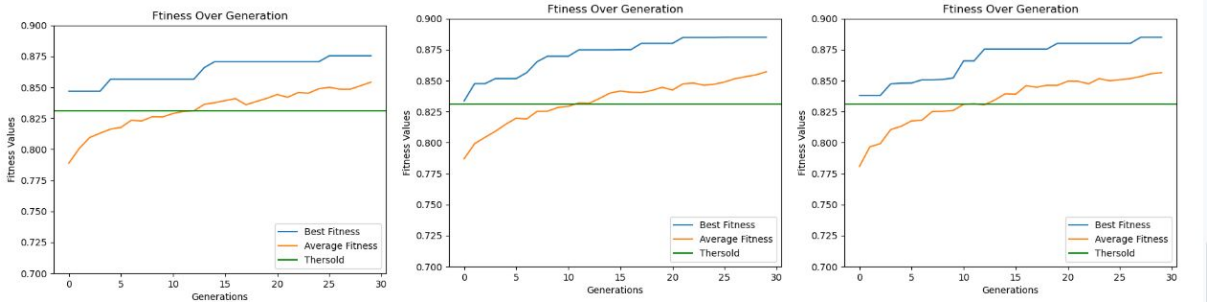


The graphs show a better look compared with previous parameter setting, more evaluations occur in the process and finally show a better average performance than before.

The only weakness is that it has no breakthrough at the later period because of inefficient mutation rate to make divergence.

Parameter setting 3:

population size = 40, crossover rate = 0.8 , mutation rate = 0.2



The algorithm is working well as it is still able to improve itself gradually at 30th generations. Even though it has slowed down its evaluation speed gradually, but overall it show a greater potential than the 2 parameter settings before. The average line and the best line are becoming closer after time which show that the overall chromosomes has been improved in stable and converged well.

4.2 FURTHER TESTING ON THE EXPERIMENT RESULT

Based on the experiment above, we can get the conclusion that the third parameter setting(population size = 40, crossover rate = 0.8 , mutation rate = 0.2) has the best performance. For further testing on its stability, we use that setting to run 20 times and record the result of every runs. Afterwards we calculate the standard deviation by using formula :

$$\sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - \mu)^2}$$

Results of 20 runs :

Turn	BestFitness
1	0.8803
2	0.8750
3	0.8752
4	0.8706
5	0.8752

6	0.8747
7	0.8745
8	0.8845
9	0.8801
10	0.8759
11	0.8757
12	0.8896
13	0.8705
14	0.8754
15	0.8752
16	0.8749
17	0.8757
18	0.8846
19	0.8849
20	0.8801

Highest fitness = 0.8896, lowest fitness = 0.8706

Sum of all = 17.5526

Average fitness over 20 runs = 0.8776

Standard deviation = 0.0048

As the Standard deviation is merely 0.0049, it means the results are maximumly float 0.49% in the fitness, which is small and can be considered that the algorithm is stable.

After the training of 200 generations, we get the following result:

	Before GA training	After GA training (take the average fitness)
Fitness	0.8234	0.8776

From the table above, we can see that the accuracy is increasing by 5.42% when useless attributes are filtered out, the noise of the data is decreased and only the important attributes left.

5.CONCLUSION

Comparing with no feature selection involved one, the performance of the model has been improved largely. It proves that the usage of feature selection successfully remove the non useful attributes from the dataset and the involvement of genetic algorithm inside feature selection has optimize the solution in fact. The result shows that it is working well in combination of feature selection and genetic algorithm. Other than the increasing in accuracy, the required number of attributes has been reduced to around half of original one, which will causing the largely simplify the further testing on real world and able to prevent wasting of time in prediction or retraining the model. The only weakness of this project is that the time required to reach a requested fitness is unexpected, it will largely be affected by luck on generating the initial stage and evolution process. Lastly, as this project is not be designed specific for one dataset only, the same theory and method are able to solve similar problems too.

6.REFERENCES

https://www.tutorialspoint.com/genetic_algorithms/

Genetic Algorithm-based Feature Selection for Depression Scale Prediction by Seung-Ju Lee, Hyun-Ji Moon, Da-Jung Kim, Yourim Yoon from Gachon University Gyeonggi-do, Korea at 2019

https://www.researchgate.net/publication/334381474_Genetic_algorithm-based_feature_selection_for_depression_scale_prediction