NOVA IMS
Information Management School

MAA

Master Degree Program in Data Science and Advanced Analytics

COMPUTATIONAL INTELLIGENCE FOR OPTIMIZATION PROJECT

MNIST PREDICTION

GROUP H:

MOHAMED SHAMSUDEEN (m20210707)
DAVID SANTOS (r20181082)
FOAZUL ISLAM (m20200750)
ARMANN KUKREJA (m20210643)

NOVA Information Management School
Instituto Superior de Estatística e Gestão de Informação
Universidade Nova de Lisboa

GitHub Link: https://github.com/foazul/CIFO_PROJECT_GROUP_H_CNN

1. Introduction

A convolutional neural network is a deep learning neural network designed for processing structured arrays of data. These networks are widely used in computer vision and have become essential for many visual applications, such as image classification, on which we will be focusing for this project.
Genetic algorithms are adaptive heuristic search algorithms that simulate the process of natural selection, which means those species that can adapt to changes in their environment are able to survive, reproduce, and pass on to the next generation. They simulate and demonstrate the expression "survival of the fittest" among individuals of consecutive generations to solve a problem. Each generation consists of a population of individuals, and each individual represents a point in search space and a possible solution. They are commonly used to generate high-quality solutions for optimization problems and search problems.
Our aim with this project is to trick CNN into misclassifying images from the dataset MNIST, which contains numbers from 0 to 9, using a genetic algorithm to disrupt the images.

2. Operators and Fitness function:

2.1. GA Operators

A GA operator is employed to help the algorithm find a solution to a problem. Three major operators such as selection, crossover, and mutation were executed in this project on the GA. Fitness proportion selection (FPS), and tournament selection were the methods used in the selection process. In a crossover operator, more than one parent is chosen, and one or more offspring are produced using the parent's genetic material. Single point crossover, arithmetic crossover,Partially Mapped Crossover (PMX) and cycle crossover were utilized. A crossover is generally used in a high-probability GA. Finally, binary mutation, inversion mutation and swap mutations were applied for mutation.

Combining the above-mentioned operators yielded the adversarial images, which evolved into 4 new populations and were analyzed for every population to see the impact of the GA operators. They used a single well-trained CNN model to classify the data. After analyzing all of the data from the four new populations, the one that was most influenced by GA operators was chosen.

2.2. Fitness function

The most important stage in carrying out everything discussed above is establishing the fitness function, it allows to compute the fitness and driving force for GA which is dependent on the problem.

The MNIST test dataset comprises 10,000 images. Using a CPU to evolve through 100 generations would take a long time. Because of these considerations which led to computational limitations as a result, GA was implemented using the constants listed below:

- N = 1000; On the test dataset, 1000 were selected randomly to be evolved.
- gens = 5; Number of generations.
- cross_prob = 0.6; Probability of crossover.
- mutation_prob = 0.5; Probability of mutation.
- tournement_size = 20; Randomly selected number of individuals on the tournament size.

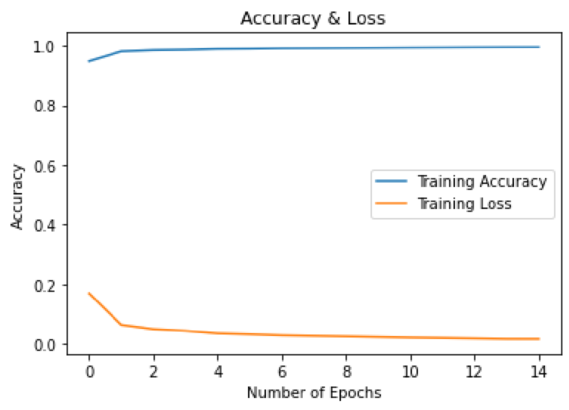
3. Results

The initial step we took was to train the model to measure the accuracy of the model after loading the dataset and preprocessing. As aforementioned, the adversarial images produced by each combination of the operator were classified using this well-trained model. Subsequently, we have found adversarial images for 4 new populations, and the performance of the model is analyzed for every generation using the classification report and plotting the graph of accuracy, f1 score, recall, and precision.

The reason why we did the comparison and the graph plotting for each population is to understand the impact of GA operators over others.

3.1. Trained model

The performance of the trained model shows the accuracy and loss that is clear in the line graph (figure 1). The model is well-trained in general since it can properly predict all of the labels (0-9). Our model achieved the best accuracy of 0.99 in precision, recall, and f1-score succeeding this the adversarial images are analyzed with the model created.

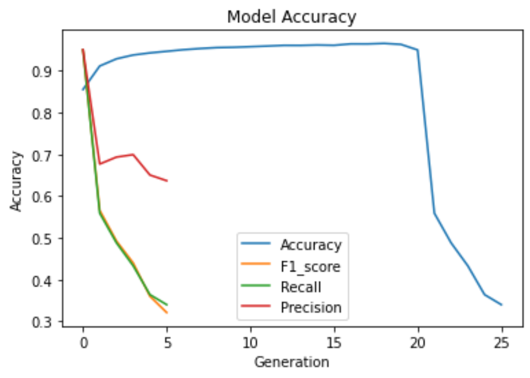


```
              precision    recall  f1-score   support

           0       0.99      1.00      0.99       980
           1       1.00      1.00      1.00      1135
           2       0.99      1.00      0.99      1032
           3       1.00      0.99      0.99      1010
           4       0.99      1.00      0.99       982
           5       0.99      0.99      0.99       892
           6       1.00      0.99      0.99       958
           7       0.99      0.99      0.99      1028
           8       0.99      0.99      0.99       974
           9       0.99      0.99      0.99      1009

   micro avg       0.99      0.99      0.99     10000
   macro avg       0.99      0.99      0.99     10000
weighted avg       0.99      0.99      0.99     10000
 samples avg       0.99      0.99      0.99     10000
```

Figure 1: Performance of the trained model

3.2. Adversarial images from FPS_SPC_SM

The result has been found for these adversarial images with the GA operators combination such as: fitness proportion selection, single point crossover and swap mutation (FPS_SPC_SM). Figure 2 represents a report of classification which has outcomes of a well-trained model ‘s performance after exhibiting the images with 10 generations. We use plot graphs to show the result of the accuracy f1_score, recall and precision for each of the generations. Before exhibiting, it is possible to classify correctly for most of the images although it breaks the images classification for all the labels whenever starting the exhibition with 10th generations the model and process finished with having a micro average score of 0.34 for precision as well as for recall and f1 score.

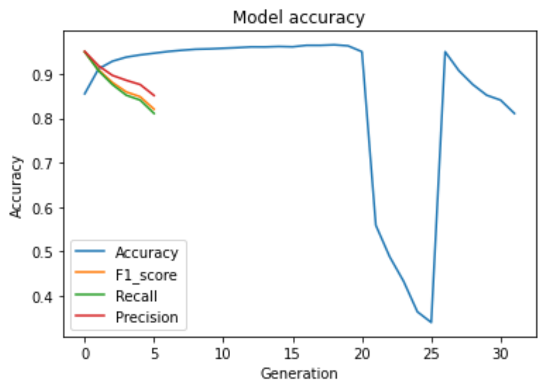


```
              precision    recall  f1-score   support

           0       0.60      0.03      0.06        95
           1       1.00      1.00      1.00         3
           2       0.60      0.29      0.39       120
           3       0.36      0.09      0.15       170
           4       0.00      0.00      0.00         1
           5       0.98      0.58      0.73        98
           6       0.98      0.21      0.35       260
           7       0.69      0.12      0.20        76
           8       0.21      0.95      0.34       158
           9       0.38      0.63      0.47        19

   micro avg       0.34      0.34      0.34      1000
   macro avg       0.58      0.39      0.37      1000
weighted avg       0.64      0.34      0.32      1000
 samples avg       0.34      0.34      0.34      1000
```

Figure 2: Performance of the trained model on adversarial images using FPS_SPC_SM.

3.3. Adversarial images from FPS_AC_IM

The result has been found for these adversarial images with the GA operators combination such as: fitness proportion selection, single point crossover and inversion mutation (FPS_SPC_IM). Figure 3 represents a report of classification which has outcomes of a well-trained model ‘s performance after exhibiting the images with 10 generations. We use plot graphs to show the result of the accuracy f1_score, recall and precision for each of the generations. Before exhibiting, it is possible to classify correctly for most of the images although it breaks the images classification for all the labels whenever starting the exhibition with 10th generations the model and process finished with having a micro average score of 0.81 for precision as well as for recall and f1 score.

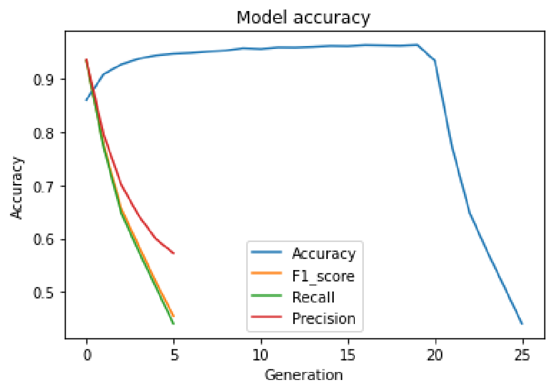


```
              precision    recall  f1-score   support

           0       0.97      0.75      0.85        84
           1       1.00      0.94      0.97       146
           2       0.88      0.76      0.81       133
           3       0.69      0.97      0.80        93
           4       0.94      0.78      0.85        80
           5       0.90      0.72      0.80       111
           6       0.85      0.85      0.85        88
           7       0.90      0.81      0.85        98
           8       0.41      0.82      0.55        65
           9       0.77      0.70      0.73       102

   micro avg       0.81      0.81      0.81      1000
   macro avg       0.83      0.81      0.81      1000
weighted avg       0.85      0.81      0.82      1000
 samples avg       0.81      0.81      0.81      1000
```

Figure 3: Performance of the trained model on adversarial images using FPS_AC_IM.

3.4. Adversarial images from TS_SPC_IM

Tournament selection, single point crossover, and inversion mutation (TS SPC IM) were used to create these hostile images. Different types of behavior, as stated above, can be observed in general. TS SPC IM, unlike the other operators, generates a large number of adversarial pictures capable of deceiving classifications and causing the model to misclassify. After all was said and done, the model received a micro average score of 0.44 for precision, recall, and f1 score, respectively.



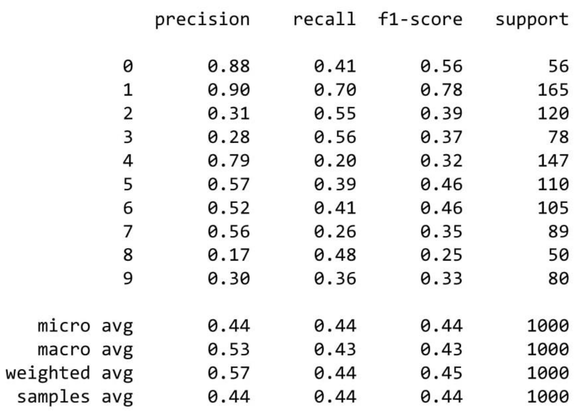
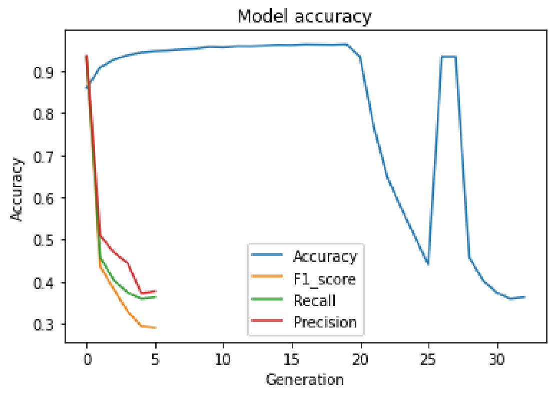

```
              precision    recall  f1-score   support

           0       0.88      0.41      0.56        56
           1       0.90      0.70      0.78       165
           2       0.31      0.55      0.39       120
           3       0.28      0.56      0.37        78
           4       0.79      0.20      0.32       147
           5       0.57      0.39      0.46       110
           6       0.52      0.41      0.46       105
           7       0.56      0.26      0.35        89
           8       0.17      0.48      0.25        50
           9       0.30      0.36      0.33        80

   micro avg       0.44      0.44      0.44      1000
   macro avg       0.53      0.43      0.43      1000
weighted avg       0.57      0.44      0.45      1000
 samples avg       0.44      0.44      0.44      1000
```

Figure 4: Performance of the trained model on adversarial images using TS_SPC_IM.

3.5. Adversarial images from TS_AC_SM

The following GA operators were used to create these hostile images: tournament selection, arithmetic crossover, and swap mutation (TS AC SM). In general, the TS SPC IM does not generate many hostile images that could cause the model to misclassify. The model's performance does not converge to zero quickly over the course of ten generations, and the model ended up with a micro average score of 0.36 for precision, recall, and f1 score, respectively.



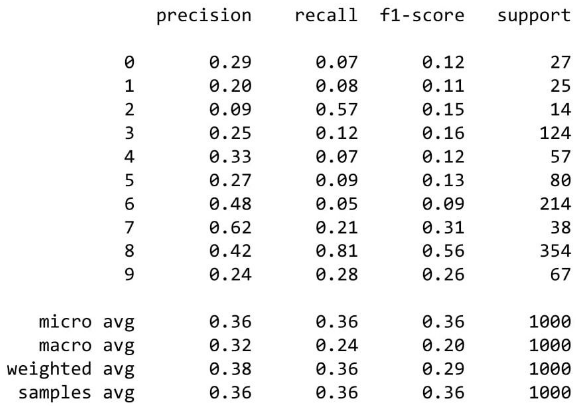

```
              precision    recall  f1-score   support

           0       0.29      0.07      0.12        27
           1       0.20      0.08      0.11        25
           2       0.09      0.57      0.15        14
           3       0.25      0.12      0.16       124
           4       0.33      0.07      0.12        57
           5       0.27      0.09      0.13        80
           6       0.48      0.05      0.09       214
           7       0.62      0.21      0.31        38
           8       0.42      0.81      0.56       354
           9       0.24      0.28      0.26        67

   micro avg       0.36      0.36      0.36      1000
   macro avg       0.32      0.24      0.20      1000
weighted avg       0.38      0.36      0.29      1000
 samples avg       0.36      0.36      0.36      1000
```

Figure 5: Performance of the trained model on adversarial images using TS_AC_SM.

3.6. Fitness on the populations

The performance of the model was analyzed on every population change by the GA's operators in order to compare the result throughout every 5 generations. It shows the distribution of the fitness on each of 4 new populations through 5 generations with boxplots for each population. The resultant plot illustrates the population in all 5 generations with the greater values of fitness are the ones with additional adversarial images. These images tend to continue the model with misclassification. Thus, the operators prosper to increase the fitness values throughout all 5 generations. In a different way, the operators did not create several adversarial images for misclassification. Misclassify model clearly shows more generation is needed to succeed the operations.

To add the different combinations, the GA operators such as fitness proportion selection, arithmetic crossover and swap mutation produce many adversarial images which might misclassify the images.

4. Conclusion

The model shows the GA's operators are able to create adversarial images able to make a well-trained model misclassify. The MNIST images versus adversarial images from GA operators shows the implementation of the GA's operators on MNIST was attained. The resultant of these outcomes are

- ➔ The constant FPS got many images, which led to misclassifications of the model.
- ➔ Operators like TS_SPC_SM, RS_AC_SM need additional generations to create many images which aid the model to misclassifications.
- ➔ The number of the generations in the evolving process only maintained 10 of the finest individuals out of a total of 1000.

5. Future work

Owing to the computational limitations in this project, we have only selected 5 numbers of generations, the probability of crossover is 0.6 and the probability of mutation is 0.5. As a result, the probability value may need to be decreased in future work to increase the number of generations and discover the results. Additionally, we have done adversarial images for only four new populations due to the computation limitation. In order to do so for all the populations, try to apply some extra methods to reduce the code running time.

References

https://deepai.org/machine-learning-glossary-and-terms/convolutional-neural-network
https://www.geeksforgeeks.org/genetic-algorithms/