Evolutionary Machine Learning – HW01
Mandar Angchekar (386916341)

Based on the SUID 386916341, implemented **(4+d2) ES** as 1 % 4 = 1.
The conclusions were derived from 10 trials across 256 generations, with data recorded for more generations than given in HW description for better understanding [1, 2, 4, 8, 16, 22, 50, 100, 128, 150, 156, 200, 256].

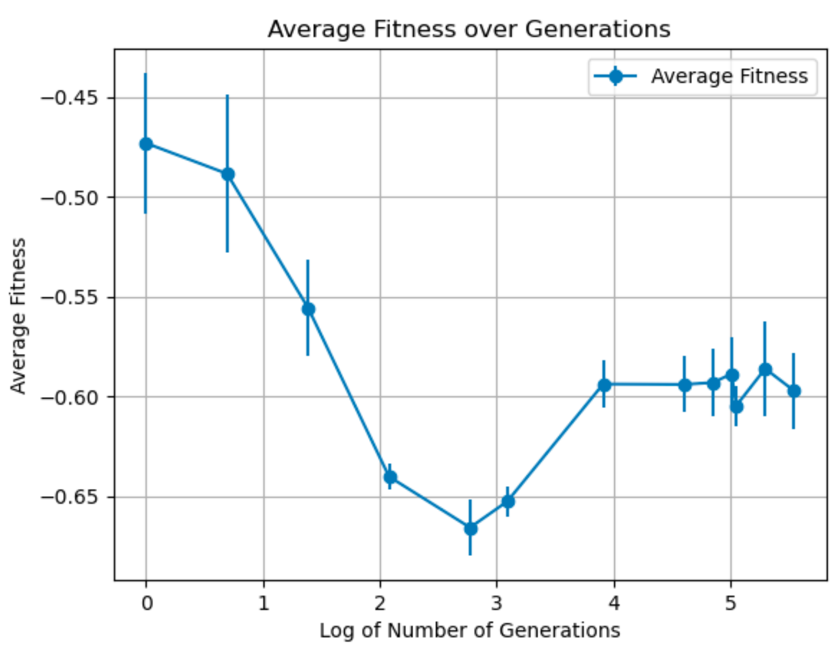
Mutation:

The initial mutation rate is set to be **0.001.**
The code adjusts the mutation rate based on the success of recent mutations.
If the success_ratio, which is the proportion of mutations that lead to an improvement in fitness, is **greater than 0.2 (20%),** the algorithm interprets this as a sign that mutations are not disruptive enough. Thus, it increases the mutation_rate to encourage more exploration of the solution space. The new mutation rate is determined by **dividing the current mutation_rate by 0.85**, effectively increasing it. However, it also ensures that the mutation rate does not exceed a certain limit **D = 0.1**.

Conversely, if **the success_ratio is less than 0.2(20%)**, it decreases the **mutation_rate by multiplying it by 0.85** to reduce the extent of exploration and encourage more exploitation of the current solution space. The min function is again used to ensure the mutation rate does not go below a threshold defined by **D = 0.1.**

Conclusion:

In the Evolutionary Strategy trials, results varied due to the stochastic nature of the process, with a general trend of improvement in fitness scores across generations. Convergence appeared to occur when the average fitness values began to plateau, roughly between 50 and 256 generations. The confusion matrices indicated that Backpropagation had fewer false positives and negatives, showing possibly better performance and generalization than the ES. Finally, the ES took significantly longer, between approximately 5000 to 18600 seconds per trial, compared to Backpropagation, which was likely much faster due to the fixed number of epochs and more efficient weight updates.



Evolutionary Machine Learning – HW01
Mandar Angchekar (386916341)

1. **How much did results vary between the different trials?**

   The Evolutionary Strategy (ES) results showed variability between trials, which is a characteristic of evolutionary algorithms due to their stochastic nature. The fitness values at each generation, as given below and the plot of average fitness over generations given above, showed fluctuations. The standard deviations in fitness scores at different generations give a quantitative measure of this variability. Notably, the average fitness did not steadily improve but had peaks and troughs, which is typical in ES due to the exploration of the search space.

   Generation 1: Avg fitness = -0.4731003612279892, Std fitness = 0.03521721633487554

   Generation 2: Avg fitness = -0.488438805937767, Std fitness = 0.03978685506343584

   Generation 4: Avg fitness = -0.5556656420230865, Std fitness = 0.023843571884629024

   Generation 8: Avg fitness = -0.6402679562568665, Std fitness = 0.0062901011093288274

   Generation 16: Avg fitness = -0.6657958924770355, Std fitness = 0.013956089879791795

   Generation 22: Avg fitness = -0.6527134954929352, Std fitness = 0.0076153263906415325

   Generation 50: Avg fitness = -0.593863719701767, Std fitness = 0.012148776186813988

   Generation 100: Avg fitness = -0.5940596640110016, Std fitness = 0.014005615598503421

   Generation 128: Avg fitness = -0.5931085050106049, Std fitness = 0.017011050363156595

   Generation 150: Avg fitness = -0.5887706398963928, Std fitness = 0.018425140153275947

   Generation 156: Avg fitness = -0.6047889411449432, Std fitness = 0.01025583521236226

   Generation 200: Avg fitness = -0.5861323714256287, Std fitness = 0.023645744672107994

   Generation 256: Avg fitness = -0.5972532331943512, Std fitness = 0.01902797366930064

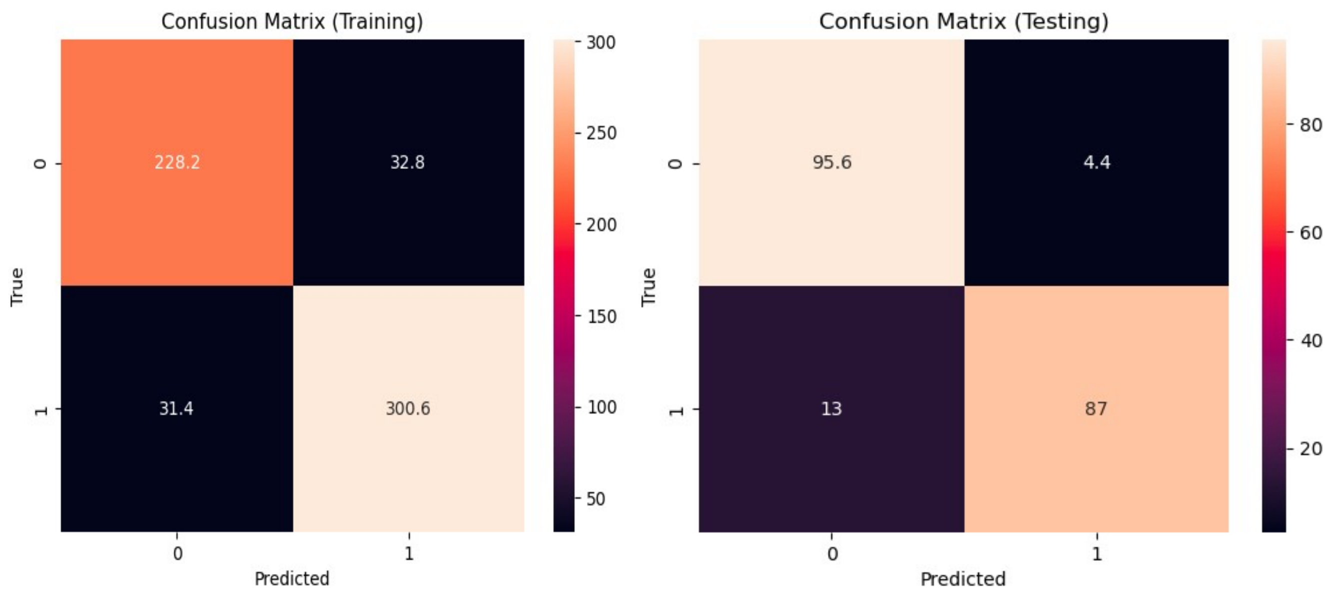
2. **After how many generations did the results appear to "converge" with no more progress in *q* values?**

   Convergence in ES can be identified when there is little to no improvement in the fitness values over successive generations. The provided text output does not conclusively specify when convergence occurred, as the highest fitness values continued to fluctuate across the generations. However, from the plot of average fitness over generations, it appears that the average fitness started to plateau after a certain point i.e **50 generations**, indicating a form of convergence.

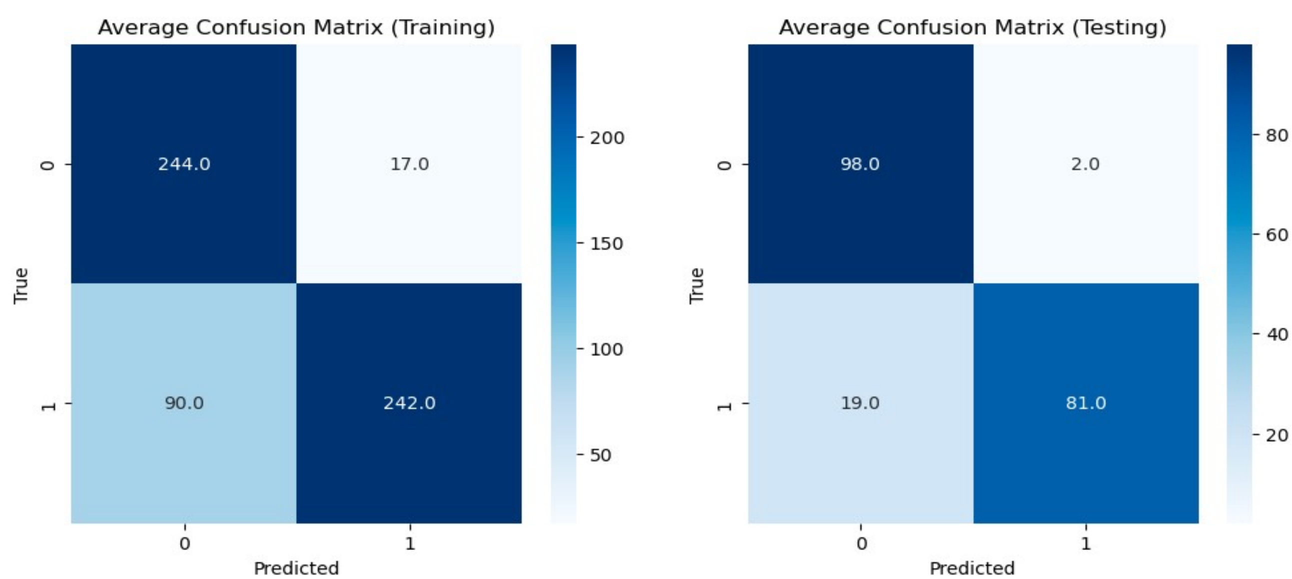
3. **How did the final confusion matrices compare, between ES and Backpropagation?**

   The final confusion matrices for both the Evolutionary Strategy and Backpropagation algorithms, indicates that both methods performed well, with a significant number of true positives and true negatives. In both training and testing, the Neural Network trained with Backpropagation seemed to have fewer false positives and negatives compared to the ES, suggesting a better performance. The confusion matrix of the Backpropagation method also appeared more consistent between training and testing, indicating better generalization.

Confusion Matrix for Backpropagation:



Confusion Matrix for Evolution Strategy



4. **How long did the ES take, compared to Backpropagation, to achieve similar results?**

The time taken for each trial of the ES varied between approximately 5000 to 18600 seconds. The large variability in time was due to the stochastic nature of ES and the changing mutation rates. In contrast, the time taken for the Backpropagation trials with the number of epochs fixed to 100, was significantly less. Backpropagation is generally faster as it uses gradient information to update weights efficiently. The ES, on the other hand, requires evaluating the fitness of multiple individuals across generations, which is more computationally expensive. Therefore, Backpropagation can achieve similar or better results in a much shorter time frame compared to ES.