



Code 39 Issues 28 Pull requests 28 Actions Projects 2 Wiki Security Insights

Notebook Akshat Karwa

[Edit](#)[New page](#)[Jump to bottom](#)

Karwa, Akshat edited this page now · 115 revisions

Name: Akshat Karwa

Email: akshatkarwa@gatech.edu

Cell Phone: +1 (404)-490-9911

Interests: Machine Learning, Blockchain Technology, Cybersecurity, Problem Solving, Adventure Sports

Table of Contents:

[Syllabus Fall 2023](#)

[Island Migration Team](#)

[Steps to Connect to PACE](#)

[SubTeam Weekly Report Spring 2024](#)

[April 22, 2024](#)

[April 15, 2024](#)

[April 8, 2024](#)

[April 1, 2024](#)

[March 25, 2024](#)

[March 11, 2024](#)

[March 4, 2024](#)

[February 26, 2024](#)

[February 19, 2024](#)

[February 12, 2024](#)

[February 5, 2024](#)

[January 29, 2024](#)

[January 22, 2024](#)

[January 8, 2024](#)

[SubTeam Weekly Report Fall 2023](#)

Table of Contents:[December 4, 2023](#)[November 27, 2023](#)[November 20, 2023](#)[November 13, 2023](#)[November 6, 2023](#)[October 30, 2023](#)[October 23, 2023](#)[October 16, 2023](#)[October 9, 2023](#)[October 2, 2023](#)[September 25, 2023](#)[September 18, 2023](#)[September 11, 2023](#)[September 4, 2023](#)[August 28, 2023](#)

[SubTeam Weekly Report Spring 2023](#)[April 24, 2023](#)[April 17, 2023](#)[April 10, 2023](#)[April 3, 2023](#)[BootCamp Team 2](#)[March 13, 2023](#)[March 8, 2023](#)[March 1, 2023](#)[February 22, 2023](#)[February 15, 2023](#)[February 8, 2023](#)[February 1, 2023](#)[January 25, 2023](#)[January 18, 2023](#)[January 11, 2023](#)[EMADE](#)

Island Migration Team :

1. Daniel Euno Park - dpark371@gatech.edu
2. Dongbin Seo - dseo33@gatech.edu
3. Aadarsh Vavilikolanu - avavilikanu3@gatech.edu
4. Jacob Grinberg - jgrinberg3@gatech.edu
5. Vudit Pokharna - vudit@gatech.edu
6. Keigo Hayashi - khayashi31@gatech.edu
7. Guozhen Miao - gmiao8@gatech.edu
8. Harish Tirumalai - htirumalai3@gatech.edu
9. Mattias Anderson - manderson343@gatech.edu
10. Atharva Gorantiwar - agorantiwar6@gatech.edu
11. Saahil Khanna - skhanna71@gatech.edu

Subteam Weekly Report

Steps to Connect to PACE -

- | ssh gt-username@login-ice.pace.gatech.edu
- | salloc -N1 --ntasks-per-node=1 -t8:00:00
- | mysqld_safe
- Update xml file with node name and open new terminal window.
- | ssh gt-username@login-ice.pace.gatech.edu
- | module load anaconda3
- | conda activate env_name
- | cd scratch/emade_island_migration

Week of Apr 22, 2024 -

Meeting Notes:

- [Final Presentation](#)
- Discussed how our presentation would be structured.
- Discussed how we want to record everything in our notebooks very clearly so that future students in this subteam could build on our work.
- Guozhen is trying to get a successful CIFAR-10 run. He was able to get a valid individual.
- Everyone discussed their run results and we all had very random results but with similar attributes. Initially, the first valid individuals dominates the process and we don't see another individual for a while. Then, after 300 generations, we start seeing some individuals that are different.
- Everyone agreed that we need more computational power and more time than 8 hours to get conclusive results. Seeded runs could definitely be a solution.
- [Island Migration Report](#)

Individual Work:

- Ran test runs on MNIST (4 islands, 512 individuals). The objectives were F1 score for class '0', F1 score for class '1', F1 score for class '2', and the number of elements. Migration start: 50 generations, Migration Interval: 5 generations. :

- In one run, I had each of the islands caring about three of the four objectives.
 - Island 1 doesn't care about F1 score for class '0'
 - Island 2 doesn't care about F1 score for class '1'
 - Island 3 doesn't care about F1 score for class '2'
 - Island 4 doesn't care about the number of elements

```
<islandObjectives>0111</islandObjectives>
<islandObjectives>1011</islandObjectives>
<islandObjectives>1101</islandObjectives>
<islandObjectives>1110</islandObjectives>
```

- In this run, I had three islands caring about one objective each and one island caring about all the objectives.
 - Island 1 cares about F1 score for class '0'
 - Island 2 cares about F1 score for class '1'
 - Island 3 cares about F1 score for class '2'
 - Island 4 cares about all the objectives

```
<islandObjectives>1000</islandObjectives>
<islandObjectives>0100</islandObjectives>
<islandObjectives>0010</islandObjectives>
<islandObjectives>1111</islandObjectives>
```

- In this run, I had three islands caring about three of the four objectives and one island caring about all the objectives.
 - Island 1 doesn't care about F1 score for class '0'
 - Island 2 doesn't care about F1 score for class '1'
 - Island 3 doesn't care about F1 score for class '2'
 - Island 4 cares about all the objectives

```
<islandObjectives>0111</islandObjectives>
<islandObjectives>1011</islandObjectives>
<islandObjectives>1101</islandObjectives>
<islandObjectives>1111</islandObjectives>
```

- I couldn't get conclusive results as PACE was too slow due to finals week. I only got some valid individuals but I think for testing we either need to have seeded runs or go upto at least 500-700 generations for MNIST to conclude something.
- I would recommend that future students play around more with the migration start and interval. I had it set as I previously had for Titanic, but I believe we need to tune more for every dataset.
 - I think starting migration at 500 generations once we get a few valid individuals would be good for MNIST and then migrating every 30 generations should be good for a starting point. Then, we can fine tune.
 - For CIFAR-10, I feel a migration start of 1000 generations or more and a migration interval of 70 generations.
 - Dynamic migration topologies may also be a great way to migrate properly. An approach where we look at island diversity and migrate if the island is saturated with similar individuals would be excellent.
- Finished my slides on:
 - The phenotypic diversity implementation

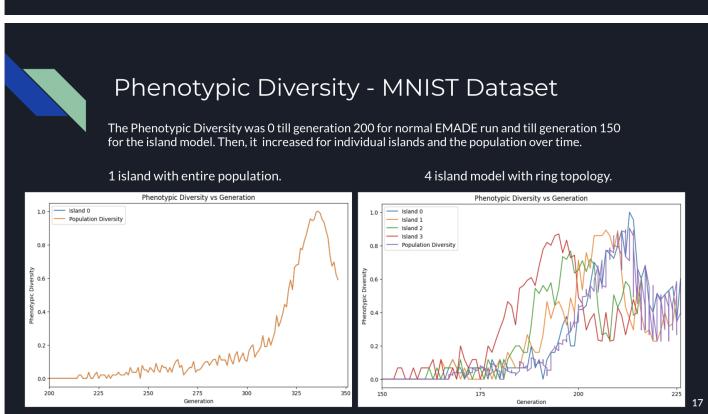
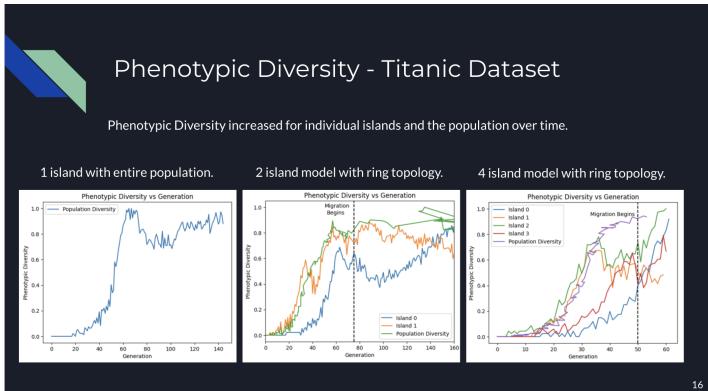
Phenotypic Diversity

- Objectives are set within certain limits and goals.
- Then, the objective space is divided into sections, or bins, to help measure diversity.
- To ensure consistency and simplicity, individual fitness values are normalized and reduced in complexity.
- The diversity within the population is then calculated using the Shannon diversity index, which adjusts for any invalid entries.
- The final outcome is a quantifiable measure of population diversity, which helps in exploring a wider range of solutions in evolutionary algorithms.

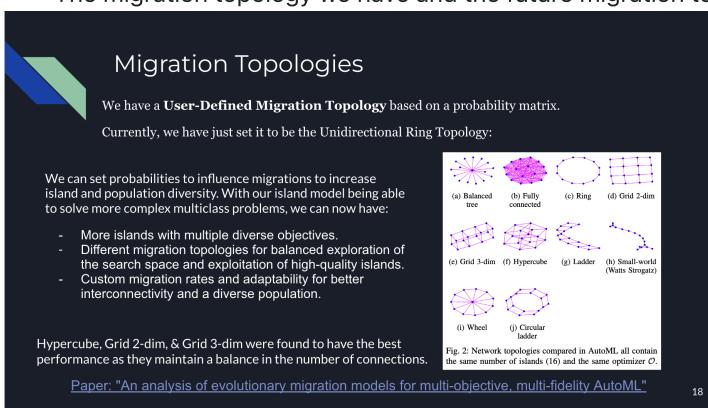
$$E(P) = - \sum_k p_k \cdot \log p_k$$

15

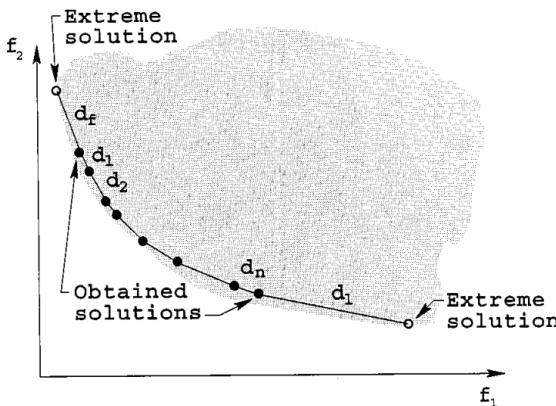
- The test run results showing diversity increase for Titanic and MNIST



- The migration topology we have and the future migration topologies we want to have.



- This diversity implementation can be integrated to our island migration model in future - [DEAP Diversity Implementation](#)
- This function from DEAP return a metric of the diversity of the front as explained in the original NSGA-II article by K. Deb. - [A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II](#)



Task	Current Status	Date Assigned	Suspense Date	Date Resolved
Analyse and discuss the test run results with everyone and add to the final presentation	Completed	Apr 22, 2024	Apr 26, 2024	Apr 25, 2024
Add paper links, recommendations for future work to notebook (My final semester)	Completed	Apr 22, 2024	Apr 26, 2024	Apr 25, 2024
Complete the final presentation slides	Completed	Apr 22, 2024	Apr 26, 2024	Apr 25, 2024
Complete notebook for final submission	Completed	Apr 22, 2024	Apr 28, 2024	Apr 28, 2024

Week of Apr 15, 2024 -

Meeting Notes:

- Everyone is focussing on finishing up everything and conducting rigorous testing.
- New members were having issues with database connection. We then helped them solve the issues. Two new members now are able to successfully run our island model on MNIST. Rest are still solving the errors.
- Everyone started creating the final presentation slides.
 - Dongbin is working on the first half of the genotypic diversity slides (his implementation).
 - Guozhen is working on the second half of the genotypic diversity slides (his implementation) and CIFAR-10 run.
 - Keigo is working on the dashboard part.
 - Aadarsh is working on the individualized islands part and what does this subteam's future look slide.
 - Vudit is working on combined diversity metric part and presenting the research papers he went through.
 - Jacob is working on our test run results and their interpretation slides.
 - I am working on slides for phenotypic diversity, diversity visualizations of test runs, migration topologies and future works in these areas.
- [Island Migration Report](#)

Individual Work:

- Our model is now ready for more complex multiclass and multiobjective problems. I believe that if we have enough computational power, then we could really benefit from seeing good results. I did some more research on migration topologies that I think future students could benefit from when they build on our model.
- Paper: [An analysis of evolutionary migration models for multi-objective, multi-fidelity AutoML](#)

- This paper conducts a comprehensive analysis of 10 network topologies distributed over a generalized island model for AutoML. This model supports dynamic migration to reduce task complexity. The model is also compared against state-of-the-art AutoML frameworks.
- These are images showing visually these network topologies.

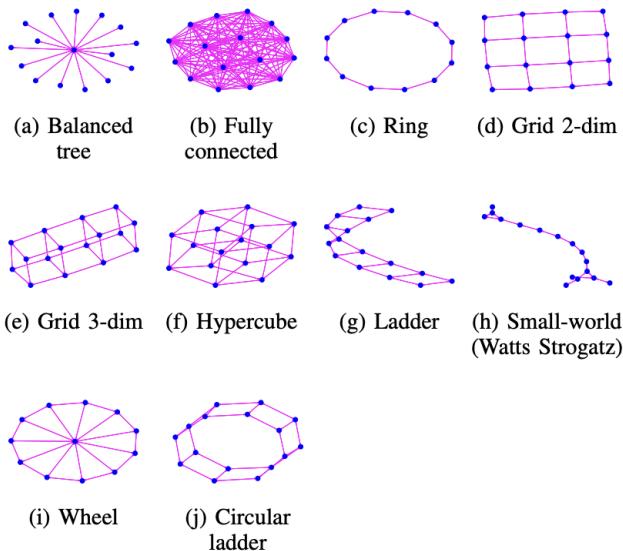
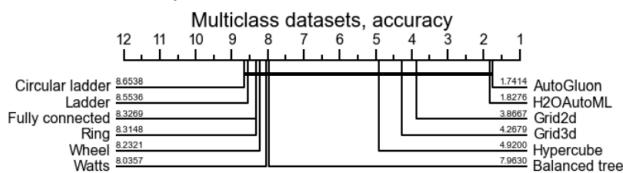


Fig. 2: Network topologies compared in AutoML all contain the same number of islands (16) and the same optimizer \mathcal{O} .

- This is the accuracy they achieved. The study found that Hypercube, Grid 2-dim, and Grid 3-dim topologies performed best as they maintained a balance in the number of connections.



- I also found this Journal on Evolutionary Computation which I believe has several important papers: [Evolutionary Computation](#)

- Had test runs on MNIST with normal EMADE, 2 islands, and 4 islands; one run with 256 individuals and one with 512 individuals.

Task	Current Status	Date Assigned	Suspense Date	Date Resolved
Get more test runs on MNIST, CIFAR-10 with different migration intervals and migration starts	Completed	Apr 15, 2024	Apr 26, 2024	Apr 22, 2024
Add paper links, recommendations for future work to notebook (My final semester)	Completed	Apr 15, 2024	Apr 26, 2024	Apr 21, 2024
Work on the final presentation slides.	Completed	Apr 15, 2024	Apr 26, 2024	Apr 24, 2024

Week of Apr 8, 2024 -

Meeting Notes:

- Everyone observed very different results for the time to get a valid individual with MNIST. I got a valid individual at generation 40 and then didn't get another and the run timed out. Jacob got one at generation 50 and then at 350. Aadarsh got the first

valid individual directly at generation 350. The elapsed time was high for everyone for the valid individuals. We all got very few valid individuals as compared to titanic which makes sense as the dataset is more complex.

- Aaron explained some reason why our runs are so different and gave suggestions for improvement.
- Decided that we have to conduct more rigorous testing on MNIST and if possible get some runs with CIFAR-10 with only a subset of classes.
- Jacob added 3 more objective functions that I plan to use for my future runs on MNIST.
- I checked on how new members were doing with PACE. Everyone had got PACE setup and now are working on getting successful runs. Told them to try to get a run on MNIST using the latest branch mnist_objectives.
- Dongbin is finished with his genotypic diversity implementation.
- Keigo is constantly adding new features every week to the Dashboard.
- Vudit created a script to remove unnecessary files to free up space (deletion of logs, temporary data).
- Guozhen has also got MNIST working. He is setting up the CIFAR-10 dataset for EMADE.
- [Island Migration Report](#)

Individual Work:

- I created a colab notebook to plot diversity and shared with Keigo to integrate in the dashboard. This is the link - [Island_Diversity.ipynb](#)
- I have converted the diversity table from our database to a pandas dataframe named diversity.

	id	island	generation	pheno_diversity	geno_diversity	geno_ratio
0	1	100	0	1.000000e-15	119.773	0.374292
1	2	0	0	1.000000e-15	119.773	0.374292
2	3	100	1	1.000000e-15	173.010	0.337909
3	4	0	1	1.000000e-15	173.010	0.337909
4	5	100	2	1.000000e-15	173.033	0.337954

- This is the code to create a plot for phenotypic diversity vs generations.

```
gen = diversity['generation']
num_gen = len(gen)

# # Get every 5 generations (Based on migration interval)
# diversity = diversity[gen % 5 == 0].copy()

scaler = MinMaxScaler()
# Fit and transform the 'generation' column
diversity.loc[:, 'phenotypic_diversity_normalized'] = scaler.fit_transform(diversity[['pheno_diversity']])

# Group by 'island' column
grouped = diversity.groupby('island')
plt.figure(figsize=(10, 6))
plt.xlim(200, 351)

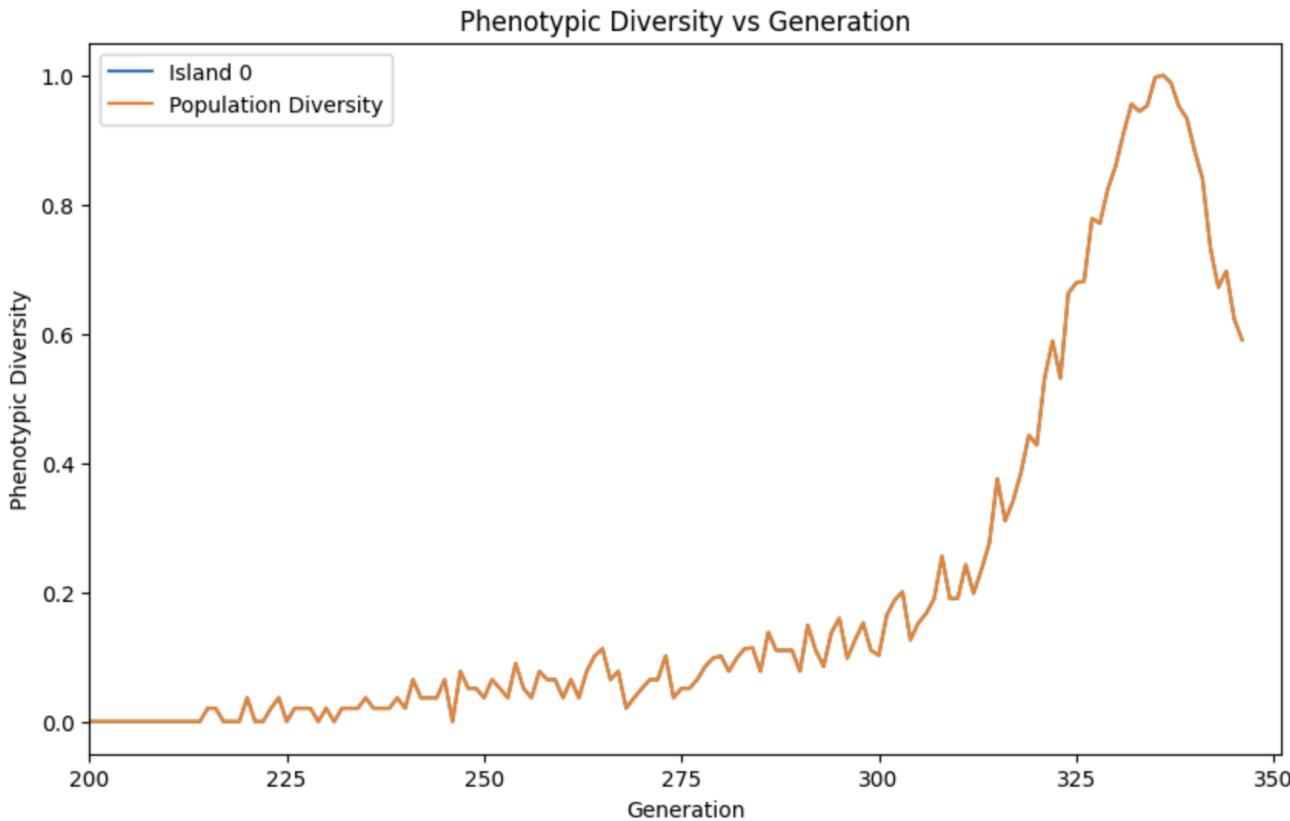
# Plotting each group separately
for island, group in grouped:
    label = 'Population Diversity' if island == 100 else f'Island {island}'
    plt.plot(group['generation'], group['phenotypic_diversity_normalized'], label=label)

plt.xlabel('Generation')
plt.ylabel('Phenotypic Diversity')
plt.title('Phenotypic Diversity vs Generation')

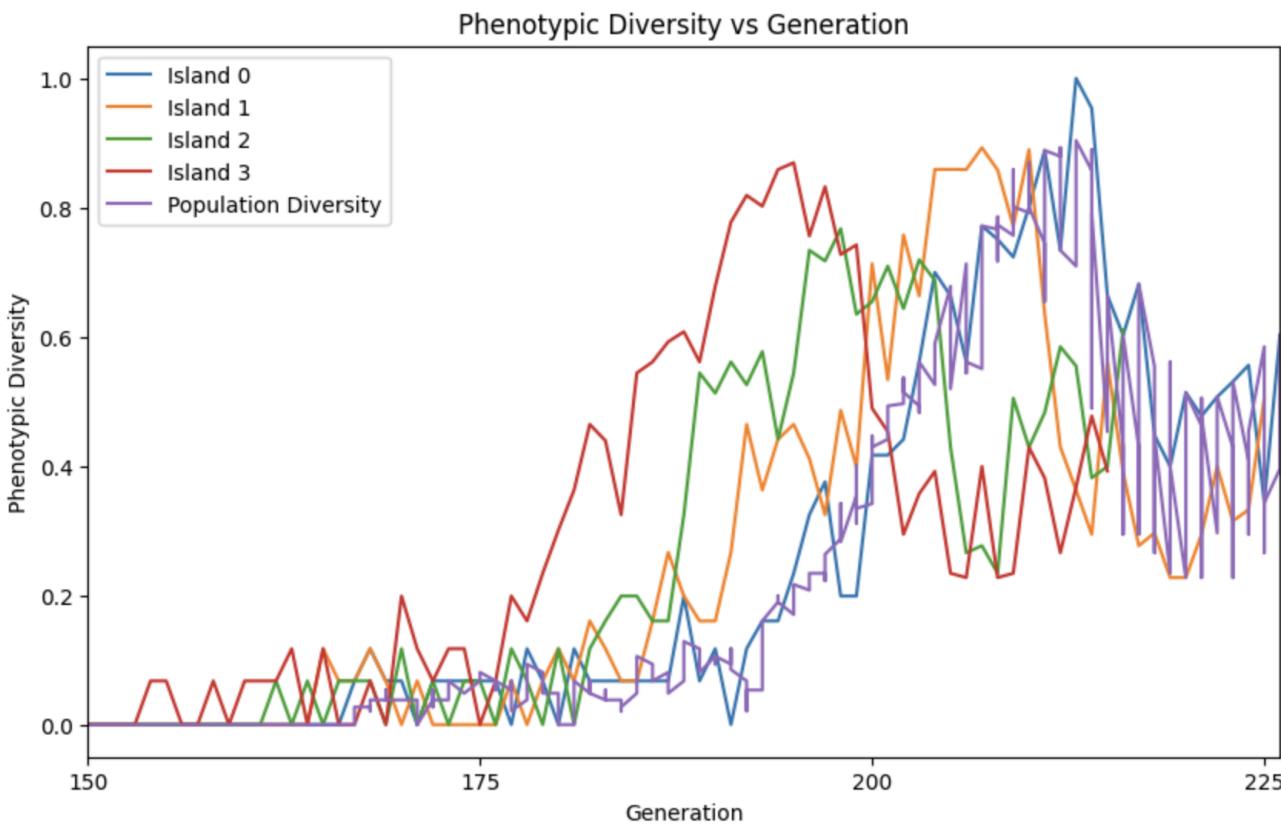
plt.xticks(list(range(200, 351, 25)))
# # Code to add a line to indicate migration start
# plt.axvline(x=50, color='black', linestyle='--')
# plt.text(102, plt.ylim()[1]*0.95, 'Migration', color='black')
# plt.text(104, plt.ylim()[1]*0.90, 'Begins', color='black')

plt.legend()
plt.show()
```

- Here is the plot for an MNIST run with normal EMADE. The plot is only after 200 generations because the population diversity before that was 0 as there was no valid individual.



- Here is the plot for an MNIST run with 4 islands, 512 individuals. The plot is only after 150 generations because the population diversity before that was 0 as there was no valid individual.



- Similarly, this is the code to create a plot for genotypic diversity vs generations.

```
# # Get every 5 generations (Based on migration interval)
# diversity = diversity[gen % 5 == 0].copy()

scaler = MinMaxScaler()
# Fit and transform the 'generation' column
diversity.loc[:, 'genotypic_diversity_normalized'] = scaler.fit_transform(diversity[['geno_ratio']])

# Group by 'island' column
grouped = diversity.groupby('island')
plt.figure(figsize=(16, 6))

# Plotting each group separately
for island, group in grouped:
    label = 'Population Diversity' if island == 100 else f'Island {island}'
    plt.plot(group['generation'], group['genotypic_diversity_normalized'], label=label)

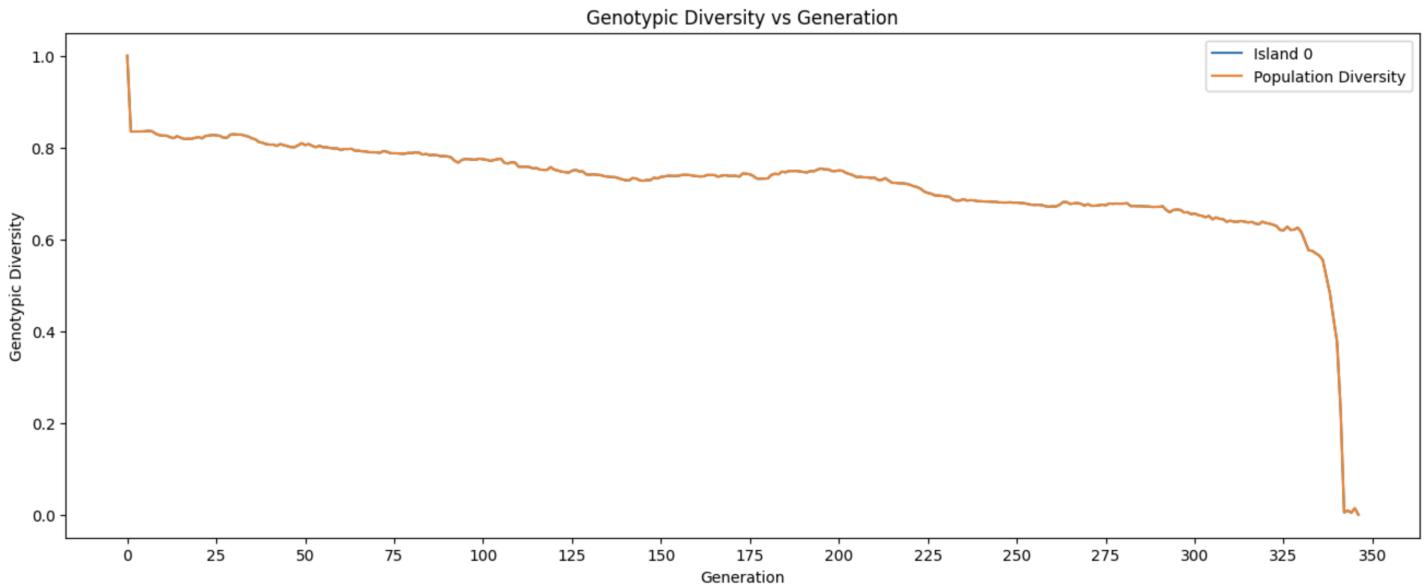
plt.xlabel('Generation')
plt.ylabel('Genotypic Diversity')
plt.title('Genotypic Diversity vs Generation')

plt.xticks(range(0, 226, 25))

# plt.axvline(x=75, color='black', linestyle='--')
# plt.text(50, plt.ylim()[1]*0.95, 'Migration', color='black')
# plt.text(52, plt.ylim()[1]*0.90, 'Begins', color='black')

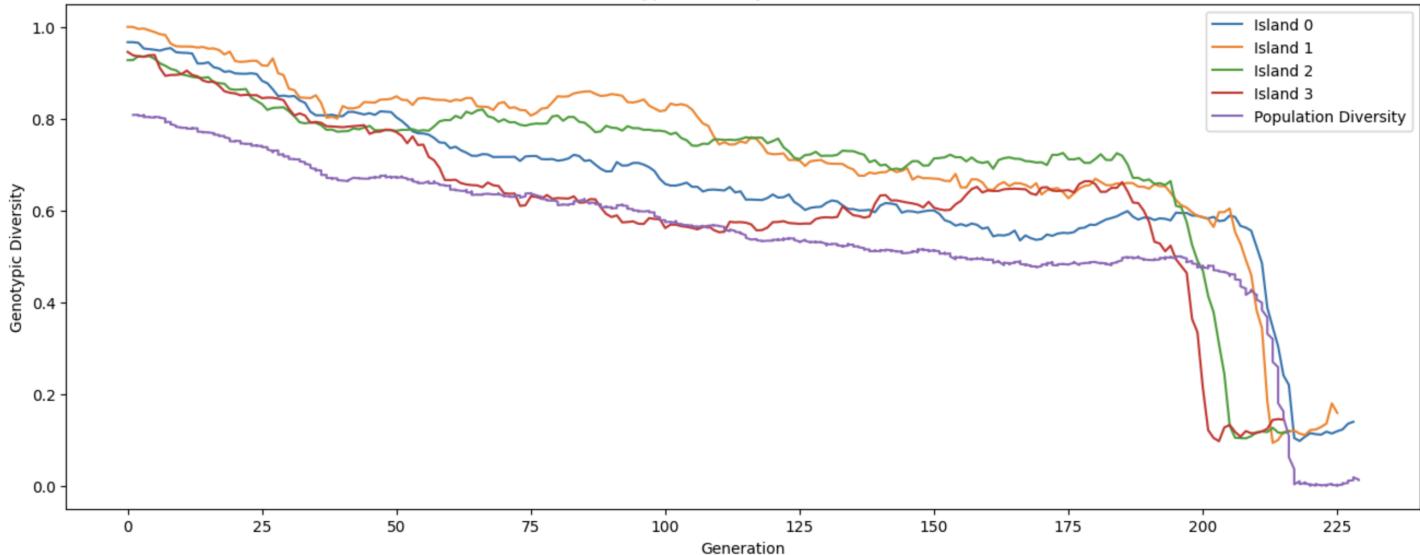
plt.legend()
plt.show()
```

- Here is the plot for an MNIST run with normal EMADE.



- Here is the plot for an MNIST run with 4 islands, 512 individuals.

Genotypic Diversity vs Generation



Task	Current Status	Date Assigned	Suspense Date	Date Resolved
Read research papers on experiments with migration topologies	Completed	Apr 1, 2024	Apr 15, 2024	Apr 12, 2024
Work with Keigo to integrate the code for these plots to the Dashboard	Completed	Apr 8, 2024	Apr 15, 2024	Apr 10, 2024
Work on creating the combined diversity metric, read papers on this	Completed	Apr 8, 2024	Apr 15, 2024	Apr 14, 2024
Start working on the final presentation slides.	Completed	Apr 8, 2024	Apr 15, 2024	Apr 10, 2024

Week of Apr 1, 2024 -**Meeting Notes:**

- Helped new members setup PACE, setup the database, and clone the island migration branch.
- Discussed that we should work with a few labels from the MNIST dataset, like 0, 1, 2.
- Decided on testing MNIST with normal EMADE, 2 islands, and 4 islands, with 512 individuals. Each island with 2 out of 3 objectives and one island with all three objectives.
- Discussed about how to get CIFAR-10 setup.
- Aadarsh was able to setup MNIST and got a valid individual after around 350 generations. The elapsed time was a lot for the valid individual.
- We can implement seeded runs to optimize on the limited computational resources we have. To truly understand island migration, we need a lot of valid individuals that we can then migrate and see how it enhances our evolutionary algorithm. With seeded runs, we would benefit a lot and would be able to actually realise our model's effectiveness.
- Keigo is working to integrate heat map visualizations to the dashboard.
- [Island Migration Report](#)

Individual Work:

- Worked with Vedit to understand Guozhen's genotypic diversity implementation in order to utilize it to enhance our migration process. The goal is to promote randomness in the evolutionary process but we feel that utilizing the diversity of individuals to guide migration would help us better explore the search space and if we have islands that start getting saturated with similar individuals, then we would be able to understand that and migrate accordingly to enhance diversity.

- Guozhen implementation of the diversity measure: The ratio between the cardinality of the set of all subtrees/the sum of the cardinality of individual sets of subtrees.

This reveals the similarity of d with the Tanimoto distance d_t , which corresponds to

$$d_t(X, Y) = \frac{|X \cup Y| - |X \cap Y|}{|X \cup Y|} \quad (13)$$

Contrary to the case of d , however, d_t satisfies the triangle inequality and is therefore a metric distance (Levandowsky and Winter, 1971; Lipkus, 1999).

In an analogous way, we can interpret Equation 1 as a particular case of the following measure of diversity for a collection (or multiset (Monro, 1987)) $\{X_1, X_2, \dots, X_n\}$ of finite and not all empty sets X_j

$$D(\{X_1, X_2, \dots, X_n\}) = n \frac{|\bigcup_{j=1}^n X_j|}{\sum_{j=1}^n |X_j|} \quad (14)$$

- Researched on how we could make our model benefit from population diversity.
- Link to Paper: [The Benefits of Population Diversity in Evolutionary Algorithms: A Survey of Rigorous Runtime Analysis](#)

8 The Benefits of Population Diversity in Evolutionary Algorithms

375

Algorithm 8.8: Scheme of an island model with migration interval τ

```

1 Initialize a population made up of subpopulations or islands,
   $P^{(0)} = \{P_1^{(0)}, \dots, P_m^{(0)}\}$ .
2  $t \leftarrow 1$ .
3 while termination criterion not met do
4   for each island  $i$  do in parallel
5     if  $t \bmod \tau = 0$  then
6       Send selected individuals from island  $P_i^{(t)}$  to selected neighboring
         islands.
7       Receive immigrants  $I_i^{(t)}$  from islands for which island  $P_i^{(t)}$  is a
         neighbor.
8       Replace  $P_i^{(t)}$  by a subpopulation resulting from a selection among
          $P_i^{(t)}$  and  $I_i^{(t)}$ .
9       Produce  $P_i^{(t+1)}$  by applying reproduction operators and selection to  $P_i^{(t)}$ .
10     $t \leftarrow t + 1$ 

```

- The paper conducts rigorous runtime analyses of evolutionary algorithms with explicit diversity mechanisms, ranging from avoiding genotype or fitness duplicates, deterministic crowding, fitness sharing, and clearing island models. They also explore how diversity can also emerge naturally, without any explicit mechanisms, through independent mutations, phases of independent evolution in the context of island models, or through the interplay of different operators such as crossover followed by mutation and selection. Diversity would be beneficial to enhance:
 - Global Exploration Capabilities
 - Effective Crossover
 - Performance and Robustness for Dynamic Optimization
 - Speeding up Optimization Time
- The most significant and interesting thing I learned was that diversity mechanisms that are effective for one problem may be ineffective for a different problem. Therefore different types of diversity measures are needed for different problems or a different interpretation of the same measures may be required when dealing with different problems.
- All this was important to understand how we want to utilize the diversity metrics to influence migration for a better exploration of the search space and to enhance diversity in future without letting certain individuals take over the entire evolutionary loop.
- I am also working on finalizing the migration logic. I feel that with MNIST, we get valid individuals very late and the elapsed time is too much so within the 8 hours we get from PACE, a complex migration topology would not be beneficial, especially because we have less islands and less objectives. Therefore, my focus is more on reading research papers in the area, and understanding the tests and experiments other researchers have run. I would provide links to them in my notebook and short summarizes so that in future, the team could then implement these topologies with more complex multiclass problems.

Task	Current Status	Date Assigned	Suspense Date	Date Resolved
Complete the setup for MNIST and get test runs with diverse parameters	Completed	Apr 1, 2024	Apr 8, 2024	Apr 3, 2024

Task	Current Status	Date Assigned	Suspense Date	Date Resolved
Read research papers on experiments with migration topologies	Completed	Apr 1, 2024	Apr 15, 2024	Apr 12, 2024
Create a colab notebook for graphing diversity results and send to Keigo to integrate in Dashboard	Completed	Apr 1, 2024	Apr 8, 2024	Apr 7, 2024
Graph results of phenotypic and genotypic diversity from test runs on MNIST	Completed	Apr 1, 2024	Apr 8, 2024	Apr 7, 2024

Week of Mar 25, 2024 -

Notes:

- Everyone worked on discussing goals for the final presentation and getting tasked. This is the week after spring break, and after the midterm presentations.
- Onboarded new member onto the team.
- Discussed with Aadarsh and Jacob our future goals by the end of the semester:
 - Test our island migration model on a more complex multi-class problem (CIFAR-10).
 - Get different experiment results with diverse parameters.
 - Finish code for diversity calculations and get graphical results showing diversity measures with increasing generations.
 - Get the dashboard completed to allow for rigorous testing with real-time visual data analysis.
 - Test individualized islands with different objectives.
- Helped vidit with his combined diversity metric implementation. Working on adding it to the database.

Task	Current Status	Date Assigned	Suspense Date	Date Resolved
Make a list of all the experiments to run	Completed	Mar 25, 2024	Apr 1, 2024	Mar 30, 2024
Read research papers on diversity implementations and testing	Completed	Mar 25, 2024	Apr 1, 2024	Mar 31, 2024
Onboard new members, help them setup PACE, and explain them our past work	Completed	Mar 25, 2024	Apr 1, 2024	Mar 27, 2024
Start setting up MNIST for test runs	Completed	Mar 25, 2024	Apr 8, 2024	Apr 3, 2024

Week of Mar 11, 2024 -

Notes:

- [Island Migration Midterm Presentation for Spring 2024](#)
- Ran some more tests and added graphs to the midterm presentation.
- Semester Goals:
 - Combine the genotypic diversity and phenotypic diversity measures to create a metric based on which we can migrate to increase island and population diversity. The idea is to:
 - Tailor migrations to provide best results and adapt to any problem
 - Get the right balance between exploration of the search space and exploitation of high-quality islands

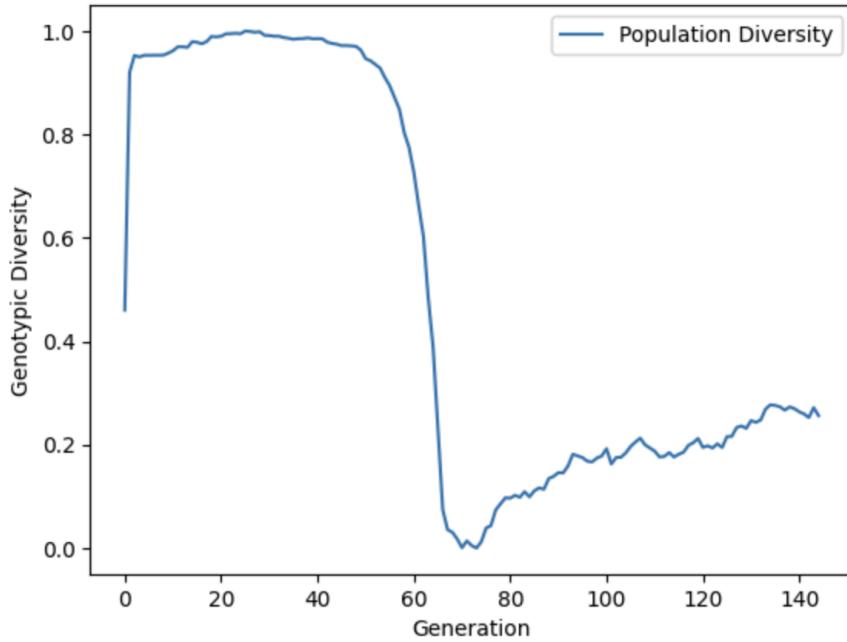
- Reduce the risk of island isolation -> get better interconnectivity and a more diverse population
- Continue testing by experimenting with the following:
 - Different number of islands with different objectives (two or more objectives preferably)
 - Migration Intervals - currently 5 generations; have to find the right number
 - Migration Cold Start - have tried 50 and 75 generations; titanic has not provided any conclusive result, have to experiment with a different problem
 - Migration Topology - currently using the unidirectional ring topology; will test different topologies once we have a good island model where diversity metrics and individualised islands work
- Will implement a combined diversity metric that can be used to decide how to migrate with increasing generations.
- Will have more comparisons for runs with and without migration (Graphs as well)
- Combine everyone's code in a consolidated branch so that other sub-teams can built on and gain from our work.
- Get island migration model working with the CIFAR-10 dataset. (multi-class problem with more objectives)

Week of Mar 4, 2024 -

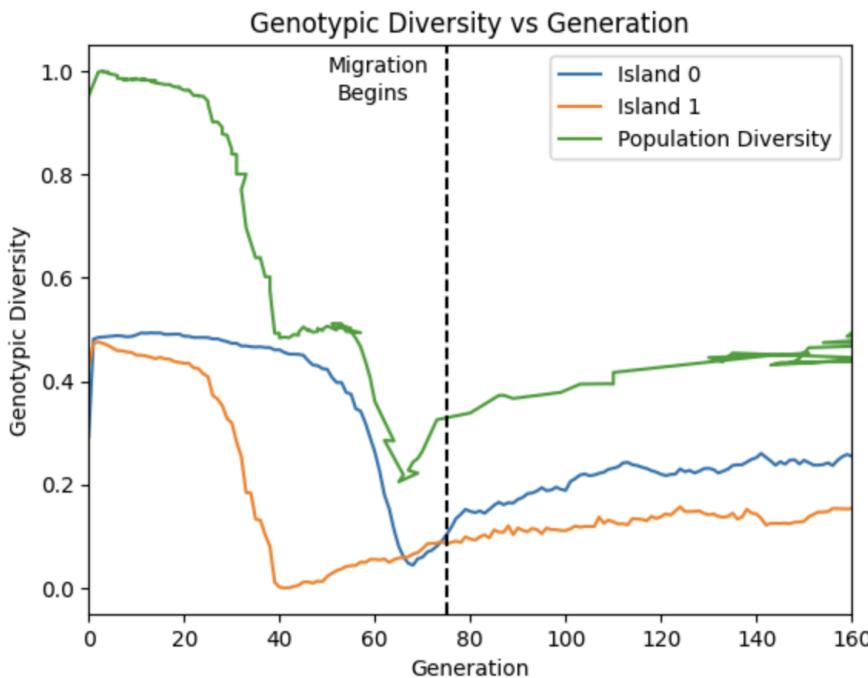
Notes:

- Got the csv output files from Jordan which have diversity implementation. Created graphs to understand genotypic diversity. The plot below shows the genotypic diversity of individuals over time with respect to generation. This was the result of a normal EMADE run on the titanic dataset with our island model with 1 island and population of 512 individuals.
- The plot shows how initially because of unevaluated random individuals the genotypic diversity is high. Then the diversity drops when we start getting evaluated individuals. Then gradually, we see an increase in genotypic diversity with more diverse evaluated individuals.

Genotypic Diversity vs Generation



- The plot below shows the results of a run with population of 512 individuals and 2 islands. The genotypic diversity with respect to increasing generations is shown for the titanic dataset. Migration starts at generation 75 and the migration interval is 5 generations.
- The plot initially shows high genotypic diversity because individuals are random and unevaluated. Then, when migration starts, we have evaluated and similar individuals. The genotypic diversity then increases with increasing number of generations demonstrating how migration increases diversity.



- Starting point for combined metric (weights = (0.6, 0.2, 0.2)) -> migration_metric = weight_fitness * normalized_fitness + weight_pheno_div * normalized_phenotypic + weight_geno_div * normalized_genotypic

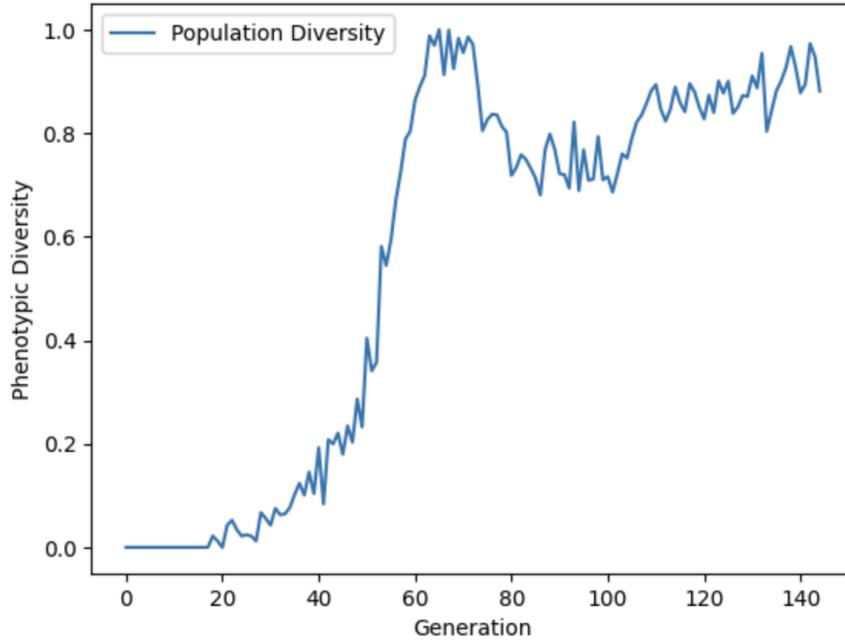
Task	Current Status	Date Assigned	Suspense Date	Date Resolved
Add slides to midterm presentation	Completed	Mar 4, 2024	Mar 11, 2024	Mar 10, 2024
Run more tests showing whether or not migration performs better than EMADE	Completed	Mar 4, 2024	Mar 11, 2024	Mar 8, 2024
Visualise genotypic diversity and phenotypic diversity implementation results from test runs for Final Presentation	Completed	Mar 4, 2024	Mar 11, 2024	Mar 8, 2024

Week of Feb 26, 2024 -

Notes:

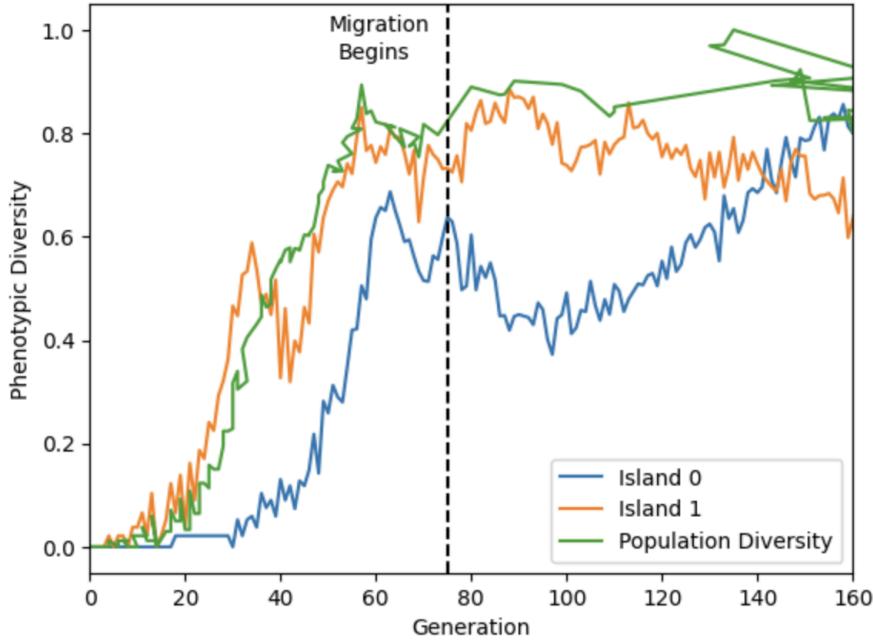
- Using csv outputs from a test run, created graphs to understand phenotypic diversity. The plot below shows the phenotypic diversity of individuals over time with respect to increasing generations. Utilised results from a normal EMADE run on the titanic dataset with our island model with 1 island and population of 512 individuals.
- The plot below shows how overall population diversity increases over time with increasing number of generations.

Phenotypic Diversity vs Generation



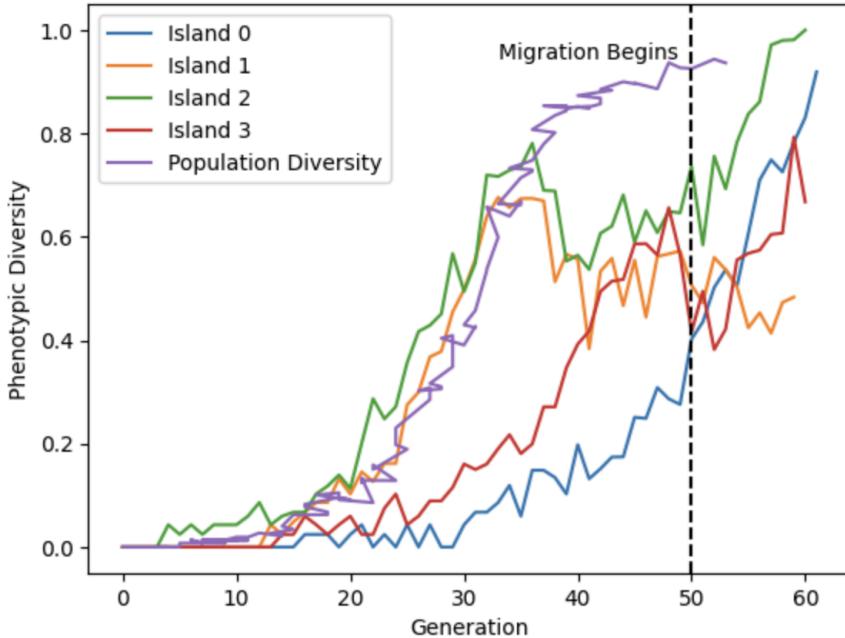
- The plot below shows the results of a run with population of 512 individuals and 2 islands having different objectives. The phenotypic diversity with respect to increasing generations is shown for the titanic dataset. Migration starts at generation 75 and the migration interval is 5 generations.
- The plot below shows how diversity for individual islands as well as the entire population increases over time with increasing number of generations.

Phenotypic Diversity vs Generation



- The plot below shows the results of a run with population of 256 individuals and 4 islands having different objectives (FPR, FNR, Num Elements). The phenotypic diversity with respect to increasing generations is shown for the titanic dataset. Migration starts at generation 50 and the migration interval is 5 generations.
- The plot below shows how diversity for individual islands as well as the entire population increases over time with increasing number of generations.

Phenotypic Diversity vs Generation



-

Task	Current Status	Date Assigned	Suspense Date	Date Resolved
Run more tests on genotypic diversity implementation with and without migrations	Completed	Feb 26, 2024	Mar 4, 2024	Mar 2, 2024
Visualise genotypic diversity results from test runs	Completed	Feb 26, 2024	Mar 4, 2024	Mar 2, 2024
Work with Vudit on the combined diversity metric implementation	Completed	Feb 26, 2024	Mar 4, 2024	Mar 1, 2024

Week of Feb 19, 2024 -

Notes:

- Implemented the user-defined diversity metric. Using probability matrix for 4 islands right now. Have not pushed the changes I have made to XML file for now, so that it is easier to work with different topologies through changing the variables in the master algorithm loop.

```
# Ring Topology
prob_matrix = np.array([[0.0, 1.0, 0.0, 0.0], [0.0, 0.0, 1.0, 0.0], [0.0, 0.0, 0.0, 1.0], [1.0, 0.0, 0.0, 0.0]])

for (i, j), prob in np.ndenumerate(prob_matrix):
    if i == j or prob == 0.0:
        continue
    else:
        if len(parents[i]) > 0:
            if ((gens_elapsed[i] > cold_start) and (((gens_elapsed[i] - cold_start) % frequency) == 0)):
                # Migrate individuals from one island to another.
```

- We currently implement a unidirectional ring topology, migrating in a ring.

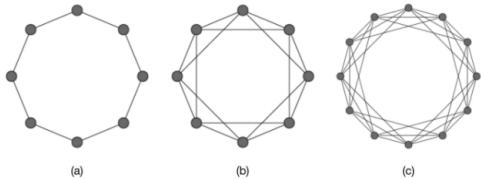


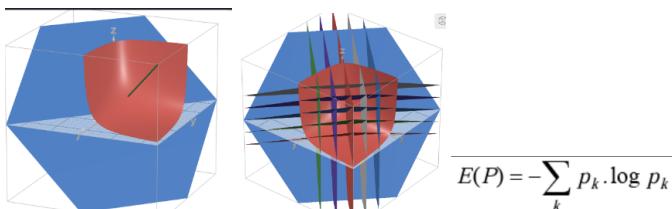
Figure 2: Ring (a), Ring+1+2 (b) and Ring+1+2+3 (c) topologies.

Task	Current Status	Date Assigned	Suspense Date	Date Resolved
Run more tests on phenotypic diversity implementation with and without migrations	Completed	Feb 19, 2024	Feb 26, 2024	Feb 23, 2024
Visualise phenotypic diversity results from test runs	Completed	Feb 19, 2024	Feb 26, 2024	Feb 25, 2024

Week of Feb 12, 2024 -

Notes:

- Understanding how phenotypic diversity implementation works so that we can utilise it in the combined diversity metric. Most work for the phenotypic diversity is done and no members are currently working on it so I and vidit have decided to test it and see if the implementation needs any changes for better results.
- The code right now compares fitnesses of individuals to calculate population diversity. The population is partitioned based on a fitness value, and p_k is the proportion of the population with that fitness value in the partition. All fitnesses are normalised; then each fitness tuple is projected onto a plane orthogonal to the vector connecting the best possible and worst possible individuals. Then this plane is partitioned uniformly along both axes.



Task	Current Status	Date Assigned	Suspense Date	Date Resolved
Implement the user-defined topology and do test runs	Completed	Feb 12, 2024	Feb 19, 2024	Feb 17, 2024
Research regarding how to utilise the current phenotypic diversity implementation to create the combined metric	Completed	Feb 12, 2024	Feb 19, 2024	Feb 17, 2024

Week of Feb 5, 2024 -

Notes:

- Got the islands_fixed branch to work which has the latest island model with phenotypic diversity implemented and individualised islands implemented.
- Have started testing how the model works and how to change migration parameters to make it work well.
- Have to conduct rigorous testing, and design a combined diversity metric.

Task	Current Status	Date Assigned	Suspense Date	Date Resolved
Run tests with different hyper-parameters for islands_fixed branch with latest model	Completed	February 5, 2024	February 12, 2024	February 10, 2024
Create combined diversity metric	Not started	February 5, 2024	February 12, 2024	February 10, 2024

Week of Jan 29, 2024 -

Notes:

- Tried fixing the errors in island_vikit branch but got new errors as it merged with my code locally.
- Jacob has fixed the errors and created an islands_fixed branch which has the phenotypic diversity implementation merged with the individualised islands implementation. I will clone from this branch and then manually add my changes of migration topologies.
- Looked at vidit's phenotypic diversity implementation from last semester. Will integrate that with migrations.
- Have to test the most recent model, and create new branch with individualised islands, diversity implementation and migration topologies.

Task	Current Status	Date Assigned	Suspense Date	Date Resolved
Clone islands_fixed branch and run final model from this branch	Completed	January 29, 2024	February 5, 2024	February 3, 2024
Integrate the phenotypic diversity implementation from last semester	Completed	January 29, 2024	February 5, 2024	February 3, 2024

Week of Jan 22, 2024 -

Notes:

- Ran EMADE on PACE with the updated code (aadarsh and vidit's merged code) from new branch but I was getting errors.
- Read Paper - [Benefits of Population Diversity in EAs](#). The paper explores the significance of population diversity, and emphasizes its role in enabling global exploration and preventing premature convergence. I learnt from the paper how rigorous runtime analyses can be conducted to show the various benefits of diversity, such as enhancing global exploration, improving crossover effectiveness over mutation, helping in dynamic optimization scenarios, and facilitating the search for the entire Pareto front in multiobjective optimization.

Task	Current Status	Date Assigned	Suspense Date	Date Resolved
Test our latest island model from last semester	Completed	January 22, 2024	January 28, 2024	January 26, 2024
Modify my code for migration topologies to incorporate diversity metrics	Completed	January 22, 2024	January 28, 2024	January 26, 2024
Research on how to test diversity metrics	Completed	January 22, 2024	January 28, 2024	January 27, 2024

Week of Jan 8, 2024 -

Notes:

Goals for this semester:

- We decided to combine the code changes of everyone and to create a final island model with individualised islands and a migration topology that migrates using a combined metric of genotypic and phenotypic diversity.
- Jacob will continue to implement the dashboard. He is planning to add ways to create more graphs with migration results and diversity measures.
- I will work on creating a combined metric that uses the genotypic diversity measures, phenotypic diversity measures and other parameters. This metric would then be used to migrate individuals using different migration topologies.

Task	Current Status	Date Assigned	Suspense Date	Date Resolved
Set Goals for the semester and divide tasks	Completed	January 8, 2024	January 22, 2024	January 15, 2024
Test running EMADE on PACE	Completed	January 8, 2024	January 22, 2024	January 15, 2024

Week of Dec 4, 2023 -

Team Meeting Notes:

- Working Session:
 - Created my slides for the final presentation. Added images of code to the presentation.
 - Divided and discussed parts with all the team members.
 - Me & Vudit: Migration Topologies and combined Metric for island-quality assessment. Dynamic migrations and researching the possibility of self-learning migrations.
 - Jacob & Dennis: Visualisation Dashboard for graphs of pareto front and analysis of individuals.
 - Vudit & Diya: Phenotypic Diversity Metric and graphs of diversity measurement.
 - Aadarsh & Sneha: Individualised islands implementation and graphs of results. (FP, FN and both objectives islands)
 - Dongbin & Guozhen: Genotypic Diversity Metric and graphs of diversity measurement.
 - [Final Presentation](#)

Task	Current Status	Date Assigned	Suspense Date	Date Resolved
Final Presentations	Completed	December 4, 2023	December 8, 2023	December 8, 2023
Finalise notebook for final collection	Completed	December 4, 2023	December 10, 2023	December 9, 2023

Week of Nov 27, 2023 -

Team Meeting Notes:

- Running the migration topologies locally, did not commit my code changes because Aadarsh and Vudit added theirs and we had some errors. I would have to manually change the code instead of merging. Focussing on code for phenotypic and genotypic diversity measures and individualised islands, as migration topologies are not the priority for now.
- Read paper on Q-Learning based migration topology that regularly computes the fitness of individuals and migrates accordingly. [Learning & Migration Processes for Island Models](#)

- Hackathon Notes (December 2, 2023):

- Discussed with Aadarsh on how to change matrix probabilities based on hyper-volume calculations or diversity measures.
- Vedit is looking at creating a metric based on a combination of the individual's fitness, phenotypic diversity and genotypic diversity that can be used to figure the probabilities of migrating individuals.

Task	Current Status	Date Assigned	Suspense Date	Date Resolved
Work on Final Presentations	In Progress	November 27, 2023	December 7, 2023	In Progress
Research possibility of self-learning migrations	In Progress	November 27, 2023	December 5, 2023	In Progress

Week of Nov 20, 2023 -

Team Meeting Notes:

- Looking at how to update probability matrix, considering island quality improvement obtained by migrations.
- Read a paper on dynamic regulation of migrations, that takes in account the pertinence of recent migrations distributing individuals to the most promising islands, based on efficient operators in each stage of the search. [Dynamic Island Model with Reinforcement Learning Based Migration](#)

Task	Current Status	Date Assigned	Suspense Date	Date Resolved
Make code changes to allow for probability matrix input in XML file	Completed	November 20, 2023	November 27, 2023	November 26, 2023
Research island quality based migration	In Progress	November 20, 2023	December 4, 2023	In Progress

Week of Nov 13, 2023 -

Team Meeting Notes:

- Successfully implemented a user-defined migration topology, where migration occurs if randomly generated number is less than the given probability.

```
probability_matrix = np.array(prob_matrix)
# probability_matrix = np.array([[0.0, 0.2, 0.5, 0.3], [0.6, 0.0, 0.1, 0.3], [0.4, 0.3, 0.0, 0.3], [0.2, 0.4, 0.4, 0.0]])
for (sending_island, receiving_island), probability in np.ndenumerate(probability_matrix):
    if len(parents[sending_island]) > 0:
        if ((gens_elapsed[sending_island] > cold_start) and (((gens_elapsed[sending_island] - cold_start) % frequency) == 0)):
            # Do not send to itself
            if sending_island != receiving_island and random.random() >= probability:
                print("No Migration!")
            else:
                print('Migrating individuals on generation ' + str(gens_elapsed[sending_island]))
```

Task	Current Status	Date Assigned	Suspense Date	Date Resolved
Implement a user-defined topology and analyse results	Completed	November 13, 2023	November 20, 2023	November 15, 2023
Research how to incorporate island quality while migrating	Completed	November 13, 2023	November 27, 2023	November 20, 2023

Week of Nov 6, 2023 -

Team Meeting Notes:

- I implemented a complete graph topology with 4 islands, where the top 5% individuals were migrating from each island to all the other islands every 5 generations.

```

if ((gens_elapsed[i] > cold_start) and (((gens_elapsed[i] - cold_start) % frequency) == 0)):

    islands_to_send_to = np.arange(num_islands)
    islands_to_send_to = np.delete(islands_to_send_to, i)

    print('Sending individuals to opposite island on generation ' + str(gens_elapsed[i]))

    for ind in parents[i]:
        data = database.select(ind.hash_val)
        print('This is individual in parents ', my_str(ind),
              'with fitness', str(ind.fitness.values), 'hash', ind.hash_val, 'island id', data.island_id, 'and age',
              str(ind.age))

    for ind in elitePool[i]:
        data = database.select(ind.hash_val)
        print('This is individual in elite pool ', my_str(ind),
              'with fitness', str(ind.fitness.values), 'hash', ind.hash_val, 'island id', data.island_id, 'and age',
              str(ind.age))

    inds_to_send = elitePool[i][0:int(0.05 * NPOP)]
    for island in islands_to_send_to:
        print('Sending ' + str(len(inds_to_send)) + ' individuals to island ' + str(island))

        for ind in inds_to_send:
            data = database.select(ind.hash_val)
            new_id = data.island_id
            # new_id = new_id - (2**i)    # if we ever want to remove the individual from the sending island

            # Checking whether this individual is already in the island we want to add to. If not, we add it.
            new_id = new_id | (1 << island)

            if not data:
                print("This should not be happening, as this is an individual that has already been evaluated!")
            else:
                database.update(row=data, individual=cp.deepcopy(ind),
                                age=ind.age, island_id=new_id,
                                evaluation_status=IndividualStatus.EVALUATED)

            evaluated_offspring[island].append(ind)

```

- Discussed how we could implement a dynamic topology with Aadarsh, where we would use an adjacency list or adjacency matrix to represent islands and migration pathways and migrate accordingly based on edges.
- I am currently researching how to implement such a dynamic topology based on the user defined graph.

Task	Current Status	Date Assigned	Suspense Date	Date Resolved
Help remaining new members set up PACE	Completed	November 6, 2023	November 13, 2023	November 13, 2023
Research dynamic topologies and their implementation	Completed	November 6, 2023	November 13, 2023	November 10, 2023

Week of Oct 30, 2023 -

Team Meeting Notes:

- New members from Bootcamp joined our team: Vudit Pokharna, Sneha Pal, Diya Jain, Dennis Nguyen
- Helped new members set up PACE.
- Discussed our goals for the Final Presentation.
- Instead of using new datasets, thinking of providing our island model to other sub-teams to test on their datasets.
- I started looking at the topologies that I want to implement and test before the final presentation.

Task	Current Status	Date Assigned	Suspense Date	Date Resolved
Help new members set up PACE	Completed	October 30, 2023	November 6, 2023	November 6, 2023
Implement different migration topologies (Implemented ring & complete graph)	Completed	October 30, 2023	November 6, 2023	November 5, 2023

Week of Oct 23, 2023 -

Team Meeting Notes:

[Midterm Presentation](#)

Task	Current Status	Date Assigned	Suspense Date	Date Resolved
Finished the midterm presentations	Completed	October 16, 2023	October 23, 2023	October 23, 2023

Week of Oct 16, 2023 -

Team Meeting Notes:

- Discussed each team member's progress and divided parts of the midterm presentation.
- Vudit and Dongbin will work on implementations of phenotypic and genotypic diversity measures.
- Jacob got successful runs for 1 island, 2 islands, and 4 islands. He will analyse and present the results for the wine dataset.
- I will present the migration topologies that we are planning to implement.

Task	Current Status	Date Assigned	Suspense Date	Date Resolved
Looked at datasets to run our island model on	Completed	October 16, 2023	October 23, 2023	October 20, 2023
Added migration topology research to the presentation	Completed	October 16, 2023	October 23, 2023	October 18, 2023

Week of Oct 9, 2023 -

Team Meeting Notes:

- Discussed each team member's progress.
- Vidit and Dongbin shared how they are trying to implement phenotypic and genotypic diversity measures.
 - Vidit is planning to containerise individuals based on output.
 - Dongbin wants to compare tree structure
- Jacob shared his wine test run results. He got an improved RMSE of 0.21.
- I shared my research on migration topologies; ring, modified-ring, and star topologies.

Task	Current Status	Date Assigned	Suspense Date	Date Resolved
Start creating midterm presentations	Completed	October 9, 2023	October 16, 2023	October 11, 2023
Further research migration topologies	Completed	October 9, 2023	October 16, 2023	October 14, 2023

Week of Oct 2, 2023 -

Team Meeting Notes:

- Discussed problems and progress team member's made.
- Vidit explained how the best way to get outputs of individuals would be to compare phenotype diversity and we discussed how we want to store individual's information.
- Dongbin explained genotypic diversity and how we can compare nodes that are present in the tree.
- I started analysing how individuals were migrating.

Task	Current Status	Date Assigned	Suspense Date	Date Resolved
Started researching migration topologies	Completed	October 2, 2023	October 9, 2023	October 4, 2023
Ran EMADE on wine again and analysed results	Completed	October 2, 2023	October 9, 2023	October 4, 2023

Week of September 25, 2023 -

Team Meeting Notes:

- Everyone ran EMADE on PACE on the wine quality dataset.
- Read research papers: [Migration and Importance of Diversity](#), [Genetic Algorithms and Poker Rule Induction](#)
- We assigned tasks to all team members:
 - Dongbin - Researching genotypic diversity; testing model on wine dataset.
 - Vudit - Researching phenotypic diversity.
 - Aadarsh - Individualising islands; testing model on wine dataset.
 - Jacob - Data pipeline for faster graphs; testing model on wine dataset.
 - Akshat - Researching datasets and problems that island models could optimally solve; testing model on wine dataset.
 - Daniel - Researching datasets.
- Trying to figure out how history of individuals could be maintained.

Task	Current Status	Date Assigned	Suspense Date	Date Resolved
Research datasets and problems	Completed	September 25, 2023	October 2, 2023	September 27, 2023
Analyse model results with different island/individual combinations	Completed	September 25, 2023	October 2, 2023	September 27, 2023

Week of September 18, 2023 -

Meeting Notes:

- Got everyone to run EMADE on PACE and everyone was able to setup MySQL.
- Looked at goals for midterm presentation:
 - To have unique mating algorithms and different mutations.
 - To have different objective functions.
 - To increase diversity of populations and individualise islands.
- Discussed the possibilities of having a master of master vs a modified master loop.
- Aadarsh suggested that we could use island migration model for function value approximation.
- We could look at something like Stirling's Approximation but a simpler approximation for now.
- Researched problems that can be solved better using island migration models than existing solutions.

Task	Current Status	Date Assigned	Suspense Date	Date Resolved
Analysed the results from the wine quality dataset	Completed	September 18, 2023	September 25, 2023	September 20, 2023
Research potential datasets we could use for island model	In progress	September 18, 2023	September 25, 2023	September 23, 2023

Week of September 11, 2023 -

Meeting Notes:

- Discussed problems that team members were having related to MySQL.
- Considered potential datasets to run island migration models on.
- Ran EMADE on Wine Quality Dataset.

Task	Current Status	Date Assigned	Suspense Date	Date Resolved
Setup MySQL Workbench and run EMADE on PACE	Completed	September 11, 2023	September 18, 2023	September 16, 2023
Run EMADE on the wine quality dataset	Completed	September 11, 2023	September 18, 2023	September 13, 2023

- Steps to setup MySQL database:
 - Create a db folder within the scratch directory in PACE.
 - Create a .my.cnf config file in the home directory and paste in it the following lines:


```
[mysqld]
datadir=/storage/ice1/{a/b/gt-username}/db
socket=/storage/ice1/{a/b/gt-username}/db/mysql.sock
user={gt-username}
symbolic-links=0
port=3040
innodb_force_recovery=4

[mysqld_safe]
log-error=/storage/ice1/{a/b/gt-username}/db/mysql.log
pid-file=/storage/ice1/{a/b/gt-username}/db/mysql.pid

[mysql]
socket=/storage/ice1/{a/b/gt-username}/db/mysql.sock

[mysqladmin]
socket=/storage/ice1/{a/b/gt-username}/db/mysql.sock
```
 - Run:
 - pace-quota
 - Get your "a/b/{gt-username}" and change it everywhere in the lines above. Also change the port to any number close to but not equal to 3040 like 3042 or 3038. Ensure that it is not same as someone in your sub-team, otherwise there would be server connection problem.
 - In the home-directory, run:
 - mysql_install_db --datadir=/storage/ice1/{a/b/gt-username}/db
 - salloc -N1 --ntasks-per-node=1 -t1:00:00
 - mysqld_safe
 - This would initialise and setup your database, running an interactive session.
 - In a new terminal window, ssh into PACE and ssh into the node that you can see on your previous terminal window, for example:
 - ssh {gt-username}@atl1-1-01-005-2-1

- Connect to MySQL:
- mysql -u root
- Enter your password and run these commands:
- DELETE FROM mysql.user WHERE user='';
- Change username and password in the next line:
- GRANT ALL PRIVILEGES ON *.* TO '{gt-username}'@'%' IDENTIFIED BY '{password}' WITH GRANT OPTION;
FLUSH PRIVILEGES;
- CREATE SCHEMA emade;
- Update the xml file for the dataset you will use for example the input_titanic.xml file. Change USER to your gt-username, password to your password, ENVIRONMENT_NAME to your env name.
- Finally change server to your node name and port number, like {atl1-1-02-010-7-2:3044}.
- <server> {node-name:port-num} </server>
- Now to run EMADE, create a file named launchEMADE.sbatch and paste this into it:
- ```
#!/bin/bash
#SBATCH -JIslandMigrationTest # Job name
#SBATCH -N1 --ntasks-per-node=4 # Number of nodes and cores per node required
#SBATCH --mem-per-cpu=1G # Memory per core
#SBATCH --time=08:00:00 # Duration of the job (Ex: 15 mins)
#SBATCH -oReport-%j.out # Combined output and error messages file

cd $SLURM_SUBMIT_DIR
echo "Started on /bin/hostname" # prints the name of the node job started on

module load jdk/1.8.0-hdolxd
module load openmpi/4.1.4
module load gcc/12.1.0
module load anaconda3/2022.05.0.1

export CC=gcc
conda activate {your-env-here} # TODO: update with your conda environment

echo "STARTING EMADE"
python src/GPFramework/launchEMADE.py templates/input_titanic.xml # use your XML file
```
- You can make changes to the lines above as per your requirements. Before running this file, enter your conda env and change the xml file name to correspond to your dataset.
- Run this file using the command:
- sbatch launchEMADE.sbatch
- This would submit a job request and run EMADE on PACE!

## Week of September 4, 2023 -

---

### Meeting Notes:

---

- Ran EMADE on titanic getting it to work on PACE. Successfully connected to MySQL server.
- Aadarsh created a word document on how to get EMADE working on PACE. I have outlined the steps from my experience and the word document below.

| Task                        | Current Status | Date Assigned     | Suspense Date      | Date Resolved     |
|-----------------------------|----------------|-------------------|--------------------|-------------------|
| Setup EMADE in PACE         | Completed      | September 4, 2023 | September 11, 2023 | September 8, 2023 |
| Run island model on titanic | Completed      | September 4, 2023 | September 11, 2023 | September 9, 2023 |

- Steps to get EMADE working in PACE:
  - ssh {gt-username}.login-ice.pace.gatech.edu
  - module load anaconda3
  - conda create --name {your-env-name} python==3.9
  - conda activate {your-env-name}
  - cd to the scratch folder on PACE and run:
  - git lfs install
  - To reduce username, password prompts run:
  - git config --global credential.helper cache
  - git clone {most updated remote repository}
  - git checkout island\_migration
  - I opened PACE in Visual Studio Code by connecting to host, ssh into PACE to view files in PACE.
  - In requirements.txt, set mysqlclient==2.1.1 and tensorflow==2.12
  - conda install opencv
  - pip install -r requirements.txt
  - We have successfully installed EMADE in PACE.
  - To correctly install GPFramework run:
  - pip install .

## Week of August 28, 2023 -

---

### Meeting Notes:

- Potential sub-teams and goals for the semester were discussed.
- New teams like Data Pipeline Overhaul, Engagement Detection, Neural Architecture Search, LLM Integration, etc were proposed.
- Goals:
  - To work on the implementation of different evolutionary algorithms in different islands.
  - To get our island migration model working on PACE.
  - To represent the history of migration of individuals; analyse and utilise it to enhance diversity.
  - To run our island migration model on a more complicated dataset and get insights from the results.
  - To find a way to specify migration topologies.
- Started setting up PACE and getting our remote repository cloned in PACE.

| Task                                      | Current Status | Date Assigned   | Suspense Date     | Date Resolved   |
|-------------------------------------------|----------------|-----------------|-------------------|-----------------|
| Start setting up PACE                     | In Progress    | August 28, 2023 | September 4, 2023 | August 28, 2023 |
| Clone directory and install EMADE on PACE | Completed      | August 28, 2023 | September 4, 2023 | August 30, 2023 |

## Week of April 24, 2023 -

### Team Meeting Notes:

1. Final Presentation
2. Assigned tasks, divided parts of the presentation that everyone would individually work on.
3. Deciding on running local tests to find any errors and get good final results.
4. Dongbin ran over 170 generations of single island on titanic.
5. Aadarsh ran our model on PACE with 1024 individuals and 2 islands.
6. Started with a cold start of 30 generations and migration interval of 5 generations.
7. Completed my slides on notable problems and challenges we faced and goals for semester.

| Task                                                                    | Current Status | Date Assigned  | Suspense Date  | Date Resolved  |
|-------------------------------------------------------------------------|----------------|----------------|----------------|----------------|
| Complete Final Presentation                                             | Completed      | April 24, 2023 | April 28, 2023 | April 25, 2023 |
| Test island model, migration and create AUC graphs for different models | Completed      | April 24, 2023 | April 28, 2023 | April 26, 2023 |

## Week of April 17, 2023 -

### Team Meeting Notes:

1. Played around with code to try to get a working island model.
2. Added an n-bit island\_id to every individual where the n-th bit represented the presence of that individual in island n.
3. We made changes to EMADE.py. We now check each island and determine if the island should produce the next generation now or not.
4. Need to get PACE working on one of the computers. Aadarsh is going to try to get it running on his machine.
5. Decided to experiment with different island sizes and multiple islands.
6. Tested migration by tracking our pareto optimal individual. It was not working well as we kept getting the same individual (No diversity).

| Task                                                | Current Status | Date Assigned  | Suspense Date  | Date Resolved  |
|-----------------------------------------------------|----------------|----------------|----------------|----------------|
| Add island_id to each individual                    | Completed      | April 17, 2023 | April 24, 2023 | April 20, 2023 |
| Make code changes to implement working island model | Completed      | April 17, 2023 | April 24, 2023 | April 20, 2023 |

| Task                               | Current Status | Date Assigned  | Suspense Date  | Date Resolved  |
|------------------------------------|----------------|----------------|----------------|----------------|
| Implement circular migration model | Completed      | April 17, 2023 | April 24, 2023 | April 20, 2023 |

## Week of April 10, 2023 -

### Team Meeting Notes:

- Discussed what we learnt from reading research papers.
- We decided to add a parameter for islandID rather than create separate tables for each island.
- Deciding on whether we should remove the worst 5% individuals or not.
- Trying to figure out how to make sure each island is at the same generation when we migrate.
- Would migrating independently and then adding to the individuals for selection would be a good idea?
- Focus is on implementing whatever is most convenient for now; do not want many changes.
- Understand the current evolutionary loop and decide on what should be changed to get multiple iterations working.
- Islands should be arranged in a ring and migration would occur in some circular pattern.
- Deciding on what changes to make to the input file, and how to adjust migration parameters.
- Have to figure how the main loop would work with different populations.

| Task                                           | Current Status | Date Assigned  | Suspense Date  | Date Resolved  |
|------------------------------------------------|----------------|----------------|----------------|----------------|
| Get cache-v2 working for everyone in the team  | Completed      | April 10, 2023 | April 17, 2023 | April 12, 2023 |
| Read papers to decide migration implementation | Completed      | April 10, 2023 | April 17, 2023 | April 16, 2023 |
| Change input file                              | Completed      | April 10, 2023 | April 17, 2023 | April 12, 2023 |

## Week of April 3, 2023 -

### Team Meeting Notes:

- Joined the Island Migration Subteam.
- Decided that we would create a two island migration model by the end of this semester.
- Set meeting time to 5 - 6pm every Wednesday.
- Using tables in the database to store island populations would be a good idea.
- Read research papers: [Island Model GA](#), [Dual Migration](#), [Distributed Island-Model GA using heterogeneous parameter settings](#)

| Task                                     | Current Status | Date Assigned | Suspense Date  | Date Resolved  |
|------------------------------------------|----------------|---------------|----------------|----------------|
| Create team Island Migration slack       | Completed      | April 3, 2023 | April 10, 2023 | April 3, 2023  |
| Create team wiki                         | Completed      | April 3, 2023 | April 10, 2023 | April 3, 2023  |
| Discuss goals for the remaining semester | Completed      | April 3, 2023 | April 10, 2023 | April 3, 2023  |
| Read research papers on Island Migration | Completed      | April 3, 2023 | April 10, 2023 | April 10, 2023 |

### Bootcamp Team 2 :

1. Edison Liao - [wliao75@gatech.edu](mailto:wliao75@gatech.edu)
2. Eric Li - [eli301@gatech.edu](mailto:eli301@gatech.edu)
3. Tariq Almawash - [talmawash3@gatech.edu](mailto:talmawash3@gatech.edu)
4. Adil Shaik - [ashaik39@gatech.edu](mailto:ashaik39@gatech.edu)
5. Aditya Yadavalli - [ayadavalli6@gatech.edu](mailto:ayadavalli6@gatech.edu)

|                                   |
|-----------------------------------|
| <b>Bootcamp:</b>                  |
| <a href="#">Bootcamp Lectures</a> |
| <a href="#">Bootcamp Labs</a>     |

► **Self-Graded Rubric -**

## Week of March 13, 2023 -

### Team Meeting Notes:

- Presentation Day
- [Midterm Presentation](#)

### Action Items:

| Task                                           | Current Status | Date Assigned  | Suspense Date  | Date Resolved  |
|------------------------------------------------|----------------|----------------|----------------|----------------|
| Review sub-team reports and presentations      | Completed      | March 13, 2023 | March 25, 2023 | March 13, 2023 |
| Decide on sub-team: Stocks or Island Migration | Completed      | March 13, 2023 | March 25, 2023 | March 24, 2023 |

## Week of March 8, 2023 -

### Team Meeting Notes:

- Completed EMADE setup and tried running EMADE on everyone's machines.
- Everyone tried connecting to Tariq's computer (master). There were errors due to some computer (worker).
- I downgraded my tensorflow packages from 2.11.0 to 2.10.0, tensorflow-metal from 0.7.1 to 0.6.0.
- Removed tensorflow from setup.py and reinstalled because of alternate builds for M1.
- data.py had error because of importing from keras.processing.image which was changed to tensorflow.keras.utils

### Action Items:

| Task                                      | Current Status | Date Assigned | Suspense Date  | Date Resolved |
|-------------------------------------------|----------------|---------------|----------------|---------------|
| Help team members with EMADE installation | Completed      | March 8, 2023 | March 13, 2023 | March 8, 2023 |

| Task                                                | Current Status | Date Assigned | Suspense Date  | Date Resolved  |
|-----------------------------------------------------|----------------|---------------|----------------|----------------|
| Organise team meeting and complete EMADE assignment | Completed      | March 8, 2023 | March 13, 2023 | March 11, 2023 |
| Prepare presentation                                | Completed      | March 8, 2023 | March 13, 2023 | March 12, 2023 |

## Week of March 1, 2023 -

### Team Meeting Notes:

#### Installing EMADE:

Steps for Mac-Apple Silicon

1. Install git.
2. Run this Homebrew command in terminal to install `git lfs`: `brew install git-lfs`
3. Run in terminal to reduce username and password prompts: `git config --global credential.helper cache`
4. Clone git repository: `git clone https://github.gatech.edu/emade/emade`
5. Install [Anaconda](#).
6. Go to [gtMOEP](#) in your local repository, and change lines:

```
1771: class MyPool(multiprocess.pool.Pool):
1772: Process = NoDaemonProcess
```

to

```
class MyPool(multiprocess.pool.Pool):
 if sys.version[:3] == '3.8':
 if sys.version_info.minor >= 8:
 @staticmethod
 def Process(ctx, *args, **kwds):
 return NoDaemonProcess(*args, **kwds)
 else:
 Process = NoDaemonProcess
```

from [Python Support](#).

7. Next, run `cd emade` and `conda install opencv`. (this will take time)
8. Install tensorflow in M1 mac separately using [M1 Installation Guide](#). Follow the steps, using homebrew to install mini forge.
9. Install required packages using `conda install numpy pandas keras scipy psutil lxml matplotlib PyWavelets sqlalchemy networkx cython scikit-image mysqlclient pymysql scikit-learn`. If not M1, add tensorflow to list of packages.
10. Install: `pip install xgboost lmfit multiprocess hmmlearn deap opencv-python`.
11. Run `bash reinstall.sh` to build all the required files.
12. Finally, install MySQL and connect with server. Configure: localhost for server, shareable password, reuse can be set to 0 or 1 (1 -> resumes next time),

13. Provide required parameters in the .XML file ([templates/input\_titanic.xml])  
([https://github.gatech.edu/emade/emade/blob/master/templates/input\\_titanic.xml](https://github.gatech.edu/emade/emade/blob/master/templates/input_titanic.xml))
14. Workers read input from master and write results through database, while master handles things like the selection process.
15. Input to EMADE is in form of zipped datasets.
16. [evalFunctions.py](#) contains the evaluation functions. Memory Limit (memory used until fatal error): 20% is a good value for our device.
  - Assignment: Run EMADE as a group:
    - 1 master database, rest everyone connects to their server.
    - 1 runs the master process while rest of the processes (worker) are run by everyone else.
    - Play around with SQL (try queries after running EMADE)
    - Try running for a good number of generations
    - Compare with ML & GP
    - Presentation on March 13th

## Action Items:

| Task                                                        | Current Status | Date Assigned | Suspense Date | Date Resolved |
|-------------------------------------------------------------|----------------|---------------|---------------|---------------|
| Run EMADE                                                   | Completed      | March 1, 2023 | March 8, 2023 | March 4, 2023 |
| Organise team meetings to ensure everyone has EMADE running | Completed      | March 1, 2023 | March 8, 2023 | March 5, 2023 |
| Understand EMADE outputs and try different SQL queries      | Completed      | March 1, 2023 | March 8, 2023 | March 5, 2023 |

## Week of February 22, 2023 -

### Team Meeting Notes:

#### Lecture on EMADE (Evolutionary MultiObjective Algorithm Design) -

1. Multi-objective evolutionary search with high-level primitives, to automate the process of designing ML algorithms.
2. Configure mysql server, download & install git-lfs, clone EMADE repository, and run the setup module to install the package.
3. Module to run EMADE using input file: [GP Framework, GT MOEP](#) Titanic Problem: [Titanic Input](#)
4. The input xml file helps us configure all moving parts of EMADE. Configure python, and database (mysql server:localhost).
5. MySQL config in xml file: Reuse is a boolean - 0 wipes database and starts again next time we run the program, and 1 starts from the previous Pareto frontier, if we want to continue from where we left (generates some random and fresh individuals).
6. EMADE can run across multiple datasets. Data is preprocessed and cross folded (Monte Carlo trials).
7. Clean data and create test train split, describe objectives, Evaluation function: [Methods](#), achievable & goal variables steer optimization, lower & upper are for bounding.
8. Individual using excessive memory is fatal!; define memory limit in percent; workersPerHost: number of evaluations to run in parallel, and evolutionary parameters.
9. Run MySQL queries to understand the EMADE output (May take time).
10. GPFramework: main body of code, gtMOEP.py: main EMADE engine (has evolutionary loop), gp\_framework\_helper.py: location of primitives, templates has input files, datasets has datasets.

11. Presentation due on March 13, 2023.

## Action Items:

| Task                                                                                   | Current Status | Date Assigned     | Suspense Date | Date Resolved     |
|----------------------------------------------------------------------------------------|----------------|-------------------|---------------|-------------------|
| Setup EMADE based on lecture slides                                                    | Completed      | February 22, 2023 | March 1, 2023 | February 26, 2023 |
| Organise team meetings to ensure everyone has EMADE setup                              | Completed      | February 22, 2023 | March 1, 2023 | February 27, 2023 |
| Understand the <a href="#">gtMOEP.py file</a> and <a href="#">evaluation functions</a> | Completed      | February 22, 2023 | March 1, 2023 | February 27, 2023 |
| Play around with SQL to understand EMADE outputs                                       | Completed      | February 22, 2023 | March 1, 2023 | February 28, 2023 |

## Week of February 15, 2023 -

Lecture: [Presentation Day](#)

| Task                   | Current Status | Date Assigned     | Suspense Date     | Date Resolved     |
|------------------------|----------------|-------------------|-------------------|-------------------|
| Start setting up EMADE | Completed      | February 15, 2023 | February 22, 2023 | February 21, 2023 |

## Week of February 8, 2023 -

### Lecture on Presentation Techniques -

1. Presentation should be clear, concise, informative.
2. Understand graphing techniques for Pareto Front.

### Notes -

1. We used multiple primitives and mapped them to specific features.
2. Next, we set up our toolbox, evaluation function, registered functions, and set max tree height.
3. We ran the evolutionary loop, and then graphed our results.
4. We then compared our ML results with MOGA results, graphing the comparisons.
5. [Google Colab Code](#) for Multi-objective Genetic Algorithm on Titanic Dataset

### Team Meeting Notes:

1. We decided to use strongly type genetic programming.
2. We decided to experiment individually using different primitives and approaches.
3. We figured that we were accidentally considering the index column while training data.
4. We analysed and agreed on features and the test-train split.
5. We divided different parts of the presentation among us, and decided to prepare our parts.

#### ► ML Minimisation Result

► DEAP Minimisation Result

## Action Items:

| Task                                                     | Current Status | Date Assigned    | Suspense Date     | Date Resolved     |
|----------------------------------------------------------|----------------|------------------|-------------------|-------------------|
| Create MOGA                                              | Completed      | February 8, 2023 | February 15, 2023 | February 11, 2023 |
| Organise & Work in Team Meetings                         | Completed      | February 8, 2023 | February 15, 2023 | February 11, 2023 |
| Create Comparison Graphs (MOGA vs ML)                    | Completed      | February 8, 2023 | February 15, 2023 | February 13, 2023 |
| Create <a href="#">Presentation</a> on Titanic Challenge | Completed      | February 8, 2023 | February 15, 2023 | February 13, 2023 |

## Week of February 1, 2023 -

### Team Meeting Notes:

#### Lecture on Titanic - ML from Disaster (Kaggle), Scikit Learn, Classification -

1. Clean the train\_data and test\_data datasets downloaded from Kaggle, use feature imputation. Then train classification model of choice on the train\_data dataset and predict binary values (Survived) for the test\_data.
2. Use confusion matrix to see accuracy of model and graph classification model scores of teammates. (create a Pareto Frontier)

#### Notes -

1. Read the documentation (sklearn) and understand different classification algorithms - [DecisionTreeClassifier](#), [KNeighboursClassifier](#), [MLPClassifier](#) (neural networks), [SupportVector Classifier](#)
2. Types of Feature Imputation (removing NaN values from data) -
  - removing entire columns
  - replacing with mean, mode or median
  - KNearestNeighbours - replacing with the mean of the closest k samples from dataset to the sample with NaN Value.
  - Multiple Imputation by Chained Equations (MICE) - Uses Linear Regression (on present values) to predict missing values and then imputes NaN values with those values.
  - KNN & MICE use entire dataset while mean & median use only the column of the missing value.

### Personal Work -

1. Preprocessed Name column based on prefix values.
- **Code for Name column Preprocessing**
2. Preprocessed Age column using [multivariate iterative feature imputation](#).
- **Code for Age column Preprocessing**
3. Trained Neural Network MLP classifier. 4. Experimented with different parameters and random seed values.

## Action Items:

| Task                                                                                             | Current Status | Date Assigned    | Suspense Date    | Date Resolved    |
|--------------------------------------------------------------------------------------------------|----------------|------------------|------------------|------------------|
| Review ML Lecture Notes                                                                          | Completed      | February 1, 2023 | February 8, 2023 | February 6, 2023 |
| Self-Grade (Notebook) Rubric                                                                     | Completed      | February 1, 2023 | February 8, 2023 | February 6, 2023 |
| Organise & Work in Team Meetings                                                                 | Completed      | February 1, 2023 | February 8, 2023 | February 6, 2023 |
| Clean Downloaded Data from Titanic Dataset (Kaggle) (Feature Imputation on train_data/test_data) | Completed      | February 1, 2023 | February 8, 2023 | February 6, 2023 |
| Train Classification Model (NeuralNetwork MLP Classifier) & Make Predictions on test_data        | Completed      | February 1, 2023 | February 8, 2023 | February 7, 2023 |
| Create Pareto Frontier based on scores of classification models of teammates                     | Completed      | February 1, 2023 | February 8, 2023 | February 7, 2023 |

## Week of January 25, 2023 -

### Team Meeting Notes:

#### Lecture on Multiple Objective Optimisation, Classification, Pareto Dominance -

- Multiple objective optimisation is powerful in terms of supplying us with a population of solutions.
- Fitness Computation: Translation of the vector of scores (from evaluation) to a fitness value.
- Using classification measures - confusion matrix (TP, FN, FP, TN)
- MAXIMISE -> Sensitivity:  $TPR = TP/(TP + FN)$ , Specificity:  $TNR = TN/(TN + FP)$ , Precision (Positive Predictive Value):  $PPV = TP/(TP + FP)$ , Negative Predicted Value:  $NPV = TN/(TN + FN)$
- MINIMISE ->  $FNR = FN/P = 1 - TPR$ , Fallout:  $FPR = FP/N = 1 - TNR$ , False Discovery Rate:  $FDR = FP/(TP + FP)$

| MAXIMISE                           | MINIMISE             |
|------------------------------------|----------------------|
| TPR, TNR                           | FNR, FPR             |
| Positive,Negative Predictive Value | False Discovery Rate |

#### Understanding Confusion Matrix

- Phenotype -> Objectives scores give each individual a point in the objective space, based on the evaluation using objective functions.
- Pareto Optimality: Individual that outperforms every other individual on all objectives.
- Pareto Frontier: set of all Pareto individuals. (Represent unique contributions)
- Favour Pareto individuals, but consider diversity.
- Non-dominated Sorting Genetic Algorithm (NSGA) II - Binary Tournament to select individuals, Lower Pareto Ranks are better, Ties on same front are broken using crowding distance. (Sum of normalised Euclidian distances, highest crowding wins)

11. Strength Pareto Evolutionary Algorithm 2 (SPEA2) - Uses Strength & Rank, distance to k-nearest neighbours is calculated and fitness is obtained.

## Action Items:

| Task                                | Current Status | Date Assigned    | Suspense Date    | Date Resolved    |
|-------------------------------------|----------------|------------------|------------------|------------------|
| Review Lecture 3 Notes              | Completed      | January 25, 2023 | February 1, 2023 | January 31, 2023 |
| Complete Lab 2 (Multiple Objective) | Completed      | January 25, 2023 | February 1, 2023 | January 31, 2023 |

## Week of January 18, 2023 -

### Team Meeting Notes:

#### Lecture on Genetic Programming

- Concept: Instead of having an individual and a function evaluator -> the individual itself is the function.
- Some Important Terms:
  1. Tree Representation: Program -> Tree Structure -> Nodes are Primitives (functions), Leaves are Terminals (parameters) -> Output at root
  2. Stored in the form of Lisp Preordered Parse Tree (Preordered Traversal)
  3. Crossover -> exchanging subtrees to produce children
  4. Mutation -> inserting/deleting/changing node/subtree
  5. We looked at Symbolic Regression, Third Order Expressions, feeding input points to get outputs, and utilising primitives to make evolution easier.

[Genetic Programming DEAP Documentation](#)

### Lab Notes:

Evolutionary approach to generating computer programs -

- Code Overview for single objective genetic programming (symbolic regression):
  1. We import gp from DEAP and inherit from gp.PrimitiveTree
  2. Initialise primitive set and add primitives.
  3. Define toolbox, individual, population, and compiler.
  4. Define evaluation function and optimise the function generated by minimising mean squared error.
  5. Register genetic operators
  6. Tried Node Replacement Mutation method which replaced a randomly chosen primitive from individual by a randomly chosen primitive with the same number of arguments from pset of individual.
  7. Add tree height constraints to crossover and mutation functions (optional).
  8. Program the main evolutionary algorithm. (eval population -> begin evolution -> select next gen individuals -> clone individuals -> apply crossover & mutation -> eval individuals with invalid fitness -> replace population)
  9. Best individual is outputted in Tree format.

## Action Items:

| Task                              | Current Status | Date Assigned    | Suspense Date    | Date Resolved    |
|-----------------------------------|----------------|------------------|------------------|------------------|
| Review Lecture 2 Notes            | Completed      | January 18, 2023 | January 25, 2023 | January 23, 2023 |
| Complete Lab 2 (Single Objective) | Completed      | January 18, 2023 | January 25, 2023 | January 24, 2023 |

## Week of January 11, 2023 -

### Team Meeting Notes:

#### Lecture on Genetic Algorithms

- Concept: creation through mutation and constant evaluation of fitness.
- Some Important Terms:
  1. Fitness proportionate: Higher probability of getting selected due to greater fitness value.
  2. Tournament: Tournament winners selected for mating.
  3. Mate/Crossover: single or double point to represent mating between individuals.
  4. Mutate: Random Modifications to maintain diversity.
- Steps:
  1. Randomly Initialise Population
  2. Determine Fitness
  3. Keep repeating: select parents, perform crossover, mutation, determine fitness again. Stop when best individual is found!!
- One Max Problem: Individuals have a list of 100 values (0 or 1) each. Eventually they have 100 1s.

#### Discussion on Expectations

- Notebook maintenance
- Team meeting notes and work during the week.

### Lab Notes:

- We solved the One Max Problem utilising the functionality of the DEAP Python Library.
- Code Overview:
  1. import base, creator, tools
  2. define the fitness objective and individual classes
  3. define functions using DEAP's Toolbox, create random boolean (0/1) generator
  4. define four functions for our genetic algorithm: evaluate, mate (2-point crossover function), mutate (bit-toggle, independent probability of 5%), select (tournament selection, 3 individuals)
  5. Evaluation, Evolution (40 generations), Tournament Selection (Cloning), Crossover (Mates two with 50% probability) & Mutation (20% probability - Mate & Mutate function), Deleting statements to remove previously stored data
  6. Keep replacing old population with new offspring
  7. Output achieved for Generation 39 -->

- Min 89.0
- Max 100.0
- Avg 97.94
- Std 2.4061864710227803

- We then solved the N Queens Problem (Goal is to minimise the number of conflicts between two queens.)
- Code Overview:
  1. Create fitness objective with different weight.
  2. Chess Board: Column -> Index of number, Line is stored
  3. Use partially matched crossover (Better than a two-point crossover)
  4. Use shuffle indexes mutation function or any mutation that works within set bounds without duplicates.
  5. Register functions into toolbox, along with tournament selection method.
  6. Define and run main evolutionary loop.
  7. [Solving the N-Queen's Problem: GA](#)

## Action Items:

| Task                                              | Current Status | Date Assigned    | Suspense Date    | Date Resolved    |
|---------------------------------------------------|----------------|------------------|------------------|------------------|
| Join Slack Channel                                | Completed      | January 11, 2023 | January 18, 2023 | January 15, 2023 |
| Create Notebook                                   | Completed      | January 11, 2023 | January 18, 2023 | January 15, 2023 |
| Review Lecture 1 Notes                            | Completed      | January 11, 2023 | January 18, 2023 | January 15, 2023 |
| Complete Lab 1 (OneMaxProblem, TheNQueensProblem) | Completed      | January 11, 2023 | January 18, 2023 | January 16, 2023 |

+ Add a custom footer

▶ Pages 569

### Navigation

[AAD Home](#)

[Syllabus](#)

[Calendar](#)

[Help Desk](#)

[Notebooks](#)

[Sub-team Weekly Reports](#)

**Clone this wiki locally**<https://github.gatech.edu/emade/emade.wiki.git>