Helwan University
Faculty of Computing & Artificial Intelligence – Mainstream Programme
Artificial Intelligence Department – Module: AI420 Evolutionary Algorithms – Spring “Semester 2” 2024-2025

[5] Feature Selection in Medical Image Analysis Using Genetic Algorithms [Preferably in combination with other EAs/SI approaches of your choice]:

o Identify a publicly available medical imaging dataset (e.g., for diagnosing diseases).
o Define an objective function for feature selection that maximizes classification accuracy while reducing feature count.
o Implement a Genetic Algorithm (GA) where each individual represents a feature subset.
o Run experiments comparing different selection and crossover strategies on the dataset.
o Summarize findings with charts, accuracy metrics, and a final report.
o You may employ other EAs/SI approaches instead of GAs.

Detailed Description:
Context and Problem Statement: In medical image analysis, reducing the number of features (or image characteristics) can simplify models and improve performance. Selecting the best features is critical for accurate diagnosis.

Key Terms and Concepts:
o Feature Selection: The process of choosing a subset of relevant features (variables) for model training.

o Genetic Algorithm (GA): An algorithm that evolves candidate solutions using techniques inspired by biological evolution.
o Objective Function: In this case, it balances high accuracy and lower complexity.

Requirements/Deliverables:
o A GA implementation for selecting a meaningful subset of image features.
o A detailed comparison of different GA operators (selection, crossover, mutation).
o Performance metrics (accuracy, reduction percentage) across multiple runs.
o A comprehensive report with visual aids (graphs and tables) showing algorithm performance.

Faculty of Computer Science and artificial intelligence – Helwan University – AI

Project Documentation

Project Number: 5

Project Name: Feature Selection in Medical Image Analysis Using Genetic Algorithms

Team #: 10

Discussion Time: 9:00 AM

Team Members:

Member #	Member ID	Member Name	Member Department
1	20210656	فادي هاني عبد الملاك مسيحه	Dept. AI
2	20220521	ميرنا ناجح بطرس	Dept. AI
3	20210850	محمد ياسر فؤاد احمد	Dept. AI
4	20220558	يوساب شنوده شفيق	Dept. AI
5	20220368	مايكل سمير عبدالنور زكي	Dept. AI

Introduction and Overview:

- Project idea and overview.

Feature selection in medical image analysis involves identifying the most relevant characteristics or patterns within medical images that contribute to accurate diagnosis, classification, or pathology detection. Genetic Algorithms (GAs) offer an evolutionary strategy for optimizing this selection process by emulating the principles of natural selection.

Core Concept:

1. Extract a broad set of potential features from medical images (e.g., texture, shape, intensity, statistical measures).
2. Apply genetic algorithms to intelligently search through combinations of these features.
3. Identify optimal subsets that maximize diagnostic performance while minimizing redundancy and computational cost.

Why this is valuable in medical imaging:

- The feature space is typically high-dimensional.
- Many features may be irrelevant or even detrimental to model performance.
- Efficient computation is essential in time-sensitive clinical settings.
- Interpretability is crucial for supporting medical decisions and gaining clinician trust.

- Applications (desktop, web, or mobile) similar

Similar applications exist in:

1 AIMe (AI Model Explorer)

- A web-based interface for exploring machine learning models and datasets, including feature selection and performance metrics.
- Could be adapted for medical image-based datasets.

2 SegNet Medical

- An online platform for image segmentation in medical data.
- Could be extended to support feature selection pipelines for diagnostic purposes.

- A Literature Review of Academic publications (papers/books/articles) relevant to the problem

1. Feature Selection and Extraction in Medical Imaging

◆ *"A Survey on Feature Extraction in Medical Image Processing"*

Authors: U. Rajendra Acharya et al.
Published in: *Computer Methods and Programs in Biomedicine, 2009*
Summary:

- Reviews techniques for feature extraction from medical images such as CT, MRI, and ultrasound.
- Highlights statistical, texture, and shape features.
- Establishes the need for dimensionality reduction in high-dimensional feature spaces.

---

2. Genetic Algorithms for Feature Selection

◆ *"Genetic algorithms for feature selection in medical diagnosis: A review"*

Authors: L. P. Vidal, M. A. Silva, M. P. Souza
Published in: *Artificial Intelligence in Medicine, 2018*
Summary:

- Explores GA-based feature selection for improving diagnostic accuracy.
- Compares performance with other heuristic algorithms like PSO and ACO.
- Emphasizes the trade-off between model complexity and performance.

---

3. Application in Disease Diagnosis

◆ *"Genetic algorithm-based feature selection approach for detection of Alzheimer's disease using brain MR images"*

Authors: A. Mwangi, T. Tian, J. Soares
Published in: *Neurocomputing, 2014*
Summary:

- Applies GA to select features from MRI data for early Alzheimer's diagnosis.
- Demonstrates improved classification accuracy using SVM after GA-based selection.

Proposed Solution & Dataset: -

This project focuses on using Genetic Algorithms (GAs) for feature selection in classifying chest X-ray images into two categories:

- **Normal**
- **Pneumonia**

The aim is to develop an efficient diagnostic system by:

1. **Extracting features (e.g., texture, intensity patterns) from X-ray images.**
2. **Using Genetic Algorithms to evolve and optimize feature subsets.**
3. **Selecting the best-performing feature combinations to improve classification accuracy and reduce computational cost.**

This technique addresses the following challenges:

- **High dimensionality of image data**
- **Redundant or irrelevant features affecting model performance**
- **Need for interpretable and clinically reliable results**

Dataset Description

Dataset Name: Chest X-Ray Images (Pneumonia)

Source: Guangzhou Women and Children's Medical Center, Guangzhou, China
Patient Group: Pediatric patients aged 1–5 years
Imaging Type: Anterior–Posterior (AP) chest X-rays
Total Images: 5,863 JPEG images
Classes:

- **NORMAL**
- **PNEUMONIA**

Dataset Structure:

- **Divided into 3 main folders:**
  - **train/**
  - **val/**
  - **test/**
- **Each folder contains subfolders:**
  - **NORMAL/**

- ○ **PNEUMONIA/Faculty of Computer Science and artificial intelligence – Helwan University – AI**

Applied Algorithms:

This project applies a Genetic Algorithm (GA) for feature selection in the classification of chest X-ray images as either Normal or Pneumonia. The selected features are passed to a KNN classifier to improve predictive accuracy and reduce feature usage complexity.

---

1. Genetic Algorithm for Feature Selection

The GA is customized with various evolutionary strategies and designed to explore a large search space of possible feature subsets. Here's how it operates:

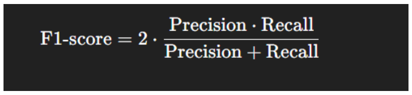
Chromosome Representation:

- **Each chromosome is a binary vector representing a subset of features.**
- **A 1 means the feature is selected; a 0 means it is excluded.**

Fitness Function:

- **Each individual in your genetic algorithm represents a subset of features. The fitness of that individual is evaluated using the performance of a K-Nearest Neighbors (KNN) classifier trained on the selected features.**
- **The fitness score is a composite based on:**
- **1. F1-Score:**

The F1-score is the harmonic mean of precision and recall. It balances the trade-off between false positives and false negatives.

- **Formula:** 

$$\text{F1-score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

Where:

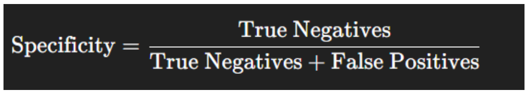
- **Precision = True Positives / (True Positives + False Positives)**

Recall = True Positives / (True Positives + False Negatives)

It's especially useful in imbalanced datasets, like medical diagnostics (e.g., pneumonia detection), where both false positives and false negatives are costly.

- ---
- **2. Specificity *(a.k.a. True Negative Rate):***

Specificity measures the ability of the classifier to correctly identify negative cases (i.e., "Normal" X-rays in your context).

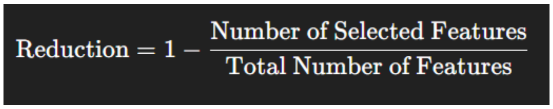
- **Formula:** 

$$\text{Specificity} = \frac{\text{True Negatives}}{\text{True Negatives} + \text{False Positives}}$$

- **Ensures the model isn't over-predicting pneumonia.**
- **High specificity means fewer false alarms—important in clinical settings.**

- 3. Reduction (Feature Reduction Ratio)

This metric represents how much the algorithm has reduced the number of selected features compared to the total.

- 📊 Formula: 

$$\text{Reduction} = 1 - \frac{\text{Number of Selected Features}}{\text{Total Number of Features}}$$

- Encourages simpler models that generalize better and train faster.
- Useful for dimensionality reduction without compromising much on accuracy.

- A typical fitness score might look like this:
- $\text{Fitness}= \alpha \cdot \text{F1} \; + \; \beta \cdot \text{Specificity} + \; \gamma \cdot \text{Reduction}$
- Where:

α, β, γ are weights (e.g., 5000, 2500, 2500) that determine how much each metric contributes to the overall fitness.

- Fitness Sharing is applied to promote diversity and prevent premature convergence.

Selection Strategy:

- A configurable selection mechanism (roulette wheel, tournament , rank selection) chooses parents based on fitness.
- Selection probabilities are adjusted using fitness sharing to encourage population diversity.

Crossover Strategy:

- A crossover rate of 0.8 determines the probability of crossover between parents.
- Crossover generates new offspring by combining selected features from both parents using methods (one-point , two-point , multiple-point ,uniform crossover)

Mutation Strategy

- Mutation is applied with an adaptive mutation rate that decays over generations (simulating simulated annealing).
- If no improvement is observed for several (5) generations, the mutation rate is temporarily boosted to escape local optima.

Elitism

- The top-performing individuals (top 7%) are carried forward to the next generation to preserve high-quality solutions.

Early Stopping

- **Evolution stops if no improvement in best fitness is observed over 15 generations (early stopping threshold).**

---

2. GA Configuration

Parameter	Value
Population Size	150
Number of Generations	125
Mutation Rate	0.08 (adaptive decay)
Crossover Rate	0.8
Elitism Rate	Top 7%
Early Stopping Rounds	15

---

3. Performance Tracking

- **Tracks:**
  - **Fitness history**
  - **Average Hamming distance (to measure population diversity)**
  - **Feature selection frequency**
  - **Mutation effectiveness (number of times it improved/worsened fitness)**

---

4. Classification Model

After optimal features are selected:

- **KNN is evaluated on the reduced feature set.**
- **This leads to:**
  - **Higher accuracy**
  - **Reduced overfitting**
  - **Faster training and inference**

Benefits of GA-based Feature Selection

- **Reduces dimensionality without losing key discriminatory features**
- **Boosts classification performance**
- **Encourages global search over greedy or local optimization methods**
- **Adds adaptivity through simulated annealing-like mutation dynamics and diversity preservation**

Experiments & Results:

1. Population Size:

- **Exploration: The population size was tested through trial and error with values of 250, 200, and 150.**
- **Best Result: The optimal population size was 150, which produced the best results after multiple runs. Larger populations (250, 200) were found to introduce unnecessary computational complexity without significant improvement in performance.**

2. Number of Generations:

- **Exploration: Different numbers of generations were tested: 200, 150, and 125.**
- **Best Result: 125 generations proved to be the most efficient. No substantial improvements were observed after this number of generations, leading to the conclusion that further generations would not yield better results. This number strikes a balance between performance and computational cost.**

3. Mutation Rate:

- **Exploration: The mutation rate was initially set to 0.08, but it was adapted using a simulated annealing-like decay for early exploration.**
- **Best Result: The mutation rate began at 0.08 and was then decayed over generations. This dynamic mutation rate helped maintain diversity in the population during the initial generations, while reducing mutation frequency as the algorithm approached convergence.**

4. Early Stopping Rounds:

- **Exploration: Different values for early stopping rounds were experimented with, specifically 40 and 15.**
- **Best Result: 15 rounds was found to be optimal. This setting allows for enough exploration without unnecessarily delaying the termination of the algorithm due to stagnation.**

5. Elitism:

- **Exploration: The percentage of elitism was tested with values of 15%, 10%, and 7%. Additionally, generational replacement was also tested but was found less effective in this context.**
- **Best Result: 7% elitism proved to be the best configuration. With elitism at this level, the algorithm maintained a balance between preserving the best individuals and promoting diversity in the population. A higher elitism rate (15%, 10%) resulted in too much bias toward the best individuals, which hindered exploration.**

6. Crossover Rate:

- **Exploration: An adaptive crossover rate was initially explored but did not produce satisfactory results.**
- **Best Result: A fixed crossover rate of 0.8 was found to be optimal. This allowed sufficient genetic exchange among individuals while avoiding the potential complications that arose with an adaptive crossover approach.**

7. Fitness Function Weights (α, β, γ):

- **Exploration: Various values for the fitness function weights were tested, particularly focusing on the trade-off between F1-score, specificity, and reduction.**
- **Best Result: The optimal values for the weights were found to be:**
    - **α = 2500 (weight for F1-score)**
    - **β = 2500 (weight for specificity)**
    - **γ = 5000 (weight for reduction)**

These values balanced the need for high classification accuracy while also encouraging feature reduction for model simplicity.

Analysis, Discussion, and Future Work:

1. Analysis of Results:

The Genetic Algorithm (GA) was applied to optimize feature selection for a K-Nearest Neighbors (KNN) classifier, focusing on improving the F1-score, specificity, and reduction. After multiple trials, the following conclusions were drawn from the experimental results:

Fitness Evaluation:

The fitness of each individual (feature subset) was evaluated using a combination of F1-score, specificity, and reduction, ensuring that the algorithm selected feature subsets that maximized classification performance while minimizing the feature set size. The best result was achieved using 130 features, with a fitness value of 9240.5851. This indicates that the selected feature subset provided a balanced trade-off between classification performance and feature reduction, ensuring both high predictive power and simplicity in the model.

Optimal Parameters:

Based on the experimental setup, the following optimal parameters were determined:

- **Population size: The best population size was 150, balancing diversity and computational efficiency.**
- **Number of generations: 125 generations were sufficient to reach convergence, as no further improvement was observed beyond this point.**
- **Mutation rate: The adaptive mutation rate, decaying over time similar to simulated annealing, contributed to a gradual refinement of the solution space.**
- **Elitism: A 7% elitism rate helped preserve the best solutions while allowing exploration of new solutions through genetic operations.**
- **Crossover rate: A fixed crossover rate of 0.8 enabled effective genetic recombination without causing instability.**

These parameters collectively facilitated the evolution of the population toward an optimal feature subset with high performance.

2. Discussion:

Impact of Mutation Rate:

The adaptive mutation rate played a critical role in achieving the optimal solution. Initially, a higher mutation rate allowed the algorithm to explore the search space broadly, while the decay function ensured that the search became more refined as the algorithm progressed. This approach mimicked the principles of Simulated Annealing, promoting exploration in early generations and refinement as the solution neared optimality.

Elitism and Crossover:

The elitism rate of 7% was crucial in maintaining the quality of the population. By retaining the best individuals, elitism ensured that high-performing solutions were carried forward, preventing them from being lost due to random genetic operations. Meanwhile, a fixed crossover rate of 0.8 allowed for sufficient genetic mixing to maintain diversity without destabilizing the algorithm.

Performance Metrics:

- F1-score: The primary performance metric, the F1-score, was maximized through the feature selection process, reflecting a balance between precision and recall in the KNN classifier.
- Specificity: The incorporation of specificity into the fitness function helped the GA focus on reducing false positives, which is particularly important in applications involving imbalanced datasets.
- Feature Reduction: The feature set was reduced to 130 features, significantly simplifying the model without compromising performance. This reduction aligns with the goal of building efficient models while retaining predictive power.

The best fitness score of 9240.5851 indicates that the selected feature subset offered a high level of classification performance with a minimal number of features, validating the effectiveness of the GA-based feature selection approach.

3. Limitations:

While the algorithm produced promising results, there are several limitations that could be addressed in future work:

- Computation Time: The GA's computational expense increases with larger datasets and higher-dimensional feature spaces. While the selected parameters worked well for this case, scalability to larger problems may require optimization strategies such as parallelization or dimensionality reduction techniques.
- Local Optima: GAs are susceptible to local optima, particularly in complex solution spaces. Although the adaptive mutation rate helps mitigate this issue, further diversity-preserving techniques, such as crowding or niching, could be explored to prevent premature convergence.
- Parameter Sensitivity: The performance of the GA is sensitive to the choice of parameters. While trial and error provided good results in this case, systematic optimization of parameters through methods like Grid Search or Bayesian Optimization could lead to more robust solutions.

4. Future Work:

Several areas for improvement and future exploration exist:

1. Hyperparameter Optimization:

To further enhance the GA's performance, hyperparameter optimization techniques, such as Grid Search, Random Search, or Bayesian Optimization, could be employed to automate the process of parameter tuning. This would reduce the reliance on trial and error and potentially lead to more efficient algorithms.

2. Enhanced Fitness Function:

Future work could explore expanding the fitness function to include additional performance metrics, such as accuracy, AUC (Area Under the Curve), or Matthews correlation coefficient (MCC). These metrics could provide deeper insights into the performance of the selected feature subsets, especially in cases where the dataset is highly imbalanced.

3. Parallelization:

Given the computational expense of GAs, particularly when working with larger datasets, parallelization could be applied to speed up the evolutionary process. This could involve evaluating fitness in parallel or distributing genetic operations (selection, crossover, mutation) across multiple processors or machines.

4. Diversity Preservation:

Maintaining diversity within the population is critical for avoiding premature convergence. Exploring diversity-preserving techniques such as fitness sharing or crowding could help ensure that the GA explores a broader space and avoids getting trapped in local optima.

5. Hybrid Approaches:

Combining GA with other optimization or machine learning techniques could yield even better results. For example, a hybrid approach that uses GA for feature selection followed by an ensemble model (e.g., Random Forest or Gradient Boosting) could provide superior classification performance. Alternatively, GA could be combined with deep learning models for more complex feature selection tasks.

Conclusion:

The GA-based feature selection process, optimized using 130 features and achieving a fitness score of 9240.5851, demonstrated the ability to effectively balance classification performance and feature reduction. The identified optimal parameters contributed to robust feature selection, resulting in an efficient and high-performing KNN model. While the algorithm performed well, future work could focus on improving scalability, enhancing diversity preservation, and automating hyperparameter optimization to further refine the feature selection process.

Strategies combination

| 3 | Uniform | Random resetting | Tournament :

Run: 1

seed = 42

Generation 1: Best Fitness = 7289.5845, Mutation Rate = 0.240000, Features Selected: 462 out of 1000

Generation 2: Best Fitness = 7362.7895, Mutation Rate = 0.232907, Features Selected: 457 out of 1000

Generation 125: Best Fitness = 8888.1129, Mutation Rate = 0.200000, Features Selected: 190 out of 1000

Run: 2

seed = 43

Generation 1: Best Fitness = 7162.8065, Mutation Rate = 0.240000, Features Selected: 476 out of 1000

Generation 2: Best Fitness = 7314.5845, Mutation Rate = 0.232907, Features Selected: 457 out of 1000

Generation 125: Best Fitness = 9240.5851, Mutation Rate = 0.240000, Features Selected: 133 out of 1000

Run: 3

seed = 44

Generation 1: Best Fitness = 7225.7196, Mutation Rate = 0.240000, Features Selected: 470 out of 1000

Generation 2: Best Fitness = 7265.7697, Mutation Rate = 0.232907, Features Selected: 470 out of 1000

Generation 125: Best Fitness = 9255.5851, Mutation Rate = 0.080000, Features Selected: 130 out of 1000

Run: 4

seed = 45

Generation 1: Best Fitness = 7260.7697, Mutation Rate = 0.240000, Features Selected: 471 out of 1000

Generation 2: Best Fitness = 7260.7697, Mutation Rate = 0.232907, Features Selected: 471 out of 1000

Generation 125: Best Fitness = 9164.2159, Mutation Rate = 0.080000, Features Selected: 138 out of 1000

Run: 5

seed = 46

Generation 1: Best Fitness = 7243.1593, Mutation Rate = 0.240000, Features Selected: 465 out of 1000

Generation 2: Best Fitness = 7243.1593, Mutation Rate = 0.232907, Features Selected: 465 out of 1000

Generation 125: Best Fitness = 9058.6968, Mutation Rate = 0.120000, Features Selected: 161 out of 1000

Run: 6

Random seed: 47

Generation 1: Best Fitness = 7245.7196, Mutation Rate = 0.240000, Features Selected: 466 out of 1000

```
Generation 2: Best Fitness = 7259.8086, Mutation Rate = 0.232907, Features Selected: 476 out of 1000

Generation 125: Best Fitness = 9075.5851, Mutation Rate = 0.140000, Features Selected: 166 out of 1000

--------------------------------------------------------------------------------------------------------------------------------------------

Run: 7

Random seed: 48

Generation 1: Best Fitness = 7140.0424, Mutation Rate = 0.240000, Features Selected: 473 out of 1000

Generation 2: Best Fitness = 7173.7432, Mutation Rate = 0.232907, Features Selected: 463 out of 1000

Generation 125: Best Fitness = 9240.4323, Mutation Rate = 0.200000, Features Selected: 135 out of 1000

--------------------------------------------------------------------------------------------------------------------------------------------

Run: 8

Random seed: 49

Generation 1: Best Fitness = 7283.4513, Mutation Rate = 0.240000, Features Selected: 449 out of 1000

Generation 2: Best Fitness = 7283.4513, Mutation Rate = 0.232907, Features Selected: 449 out of 1000

Generation 125: Best Fitness = 9104.8071, Mutation Rate = 0.100000, Features Selected: 155 out of 1000

--------------------------------------------------------------------------------------------------------------------------------------------

Run: 9

Random seed: 50

Generation 1: Best Fitness = 7199.3604, Mutation Rate = 0.240000, Features Selected: 472 out of 1000

Generation 2: Best Fitness = 7231.8875, Mutation Rate = 0.232907, Features Selected: 472 out of 1000

Generation 125: Best Fitness = 8973.1129, Mutation Rate = 0.140000, Features Selected: 173 out of 1000

--------------------------------------------------------------------------------------------------------------------------------------------

Run: 10

Random seed: 51

Generation 1: Best Fitness = 7148.1593, Mutation Rate = 0.240000, Features Selected: 484 out of 1000

Generation 2: Best Fitness = 7214.5825, Mutation Rate = 0.232907, Features Selected: 466 out of 1000

Generation 125: Best Fitness = 9169.4688, Mutation Rate = 0.080000, Features Selected: 144 out of 1000
```

| 6 | Single-point | Bit-flip | Roulette wheel:

```
Run: 1

Random seed: 42

Generation 1: Best Fitness = 7289.5845, Mutation Rate = 0.240000, Features Selected: 462 out of 1000

Generation 2: Best Fitness = 7289.5845, Mutation Rate = 0.232907, Features Selected: 462 out of 1000

Generation 125: Best Fitness = 7753.9464, Mutation Rate = 0.080000, Features Selected: 395 out of 1000
```

```
-----------------------------------------------------------------------------------------------------------------------------

Run:  2

Random seed: 43

Generation 1: Best Fitness = 7162.8065, Mutation Rate = 0.240000, Features Selected: 476 out of 1000

Generation 2: Best Fitness = 7187.0095, Mutation Rate = 0.232907, Features Selected: 473 out of 1000

Generation 125: Best Fitness = 7665.0327, Mutation Rate = 0.120000, Features Selected: 403 out of 1000

-----------------------------------------------------------------------------------------------------------------------------

Run:  3

Random seed: 44

Generation 1: Best Fitness = 7225.7196, Mutation Rate = 0.240000, Features Selected: 470 out of 1000

Generation 2: Best Fitness = 7225.7196, Mutation Rate = 0.232907, Features Selected: 470 out of 1000

Generation 125: Best Fitness = 7717.4546, Mutation Rate = 0.220000, Features Selected: 409 out of 1000

-----------------------------------------------------------------------------------------------------------------------------

Run:  4

Random seed: 45

Generation 1: Best Fitness = 7260.7697, Mutation Rate = 0.240000, Features Selected: 471 out of 1000

Generation 2: Best Fitness = 7260.7697, Mutation Rate = 0.232907, Features Selected: 471 out of 1000

Generation 125: Best Fitness = 7647.5887, Mutation Rate = 0.100000, Features Selected: 418 out of 1000

-----------------------------------------------------------------------------------------------------------------------------

Run:  5

Random seed: 46

Generation 1: Best Fitness = 7243.1593, Mutation Rate = 0.240000, Features Selected: 465 out of 1000

Generation 2: Best Fitness = 7243.1593, Mutation Rate = 0.232907, Features Selected: 465 out of 1000

Generation 125: Best Fitness = 7692.7134, Mutation Rate = 0.140000, Features Selected: 424 out of 1000

-----------------------------------------------------------------------------------------------------------------------------

Run:  6

Random seed: 47

Generation 1: Best Fitness = 7245.7196, Mutation Rate = 0.240000, Features Selected: 466 out of 1000

Generation 2: Best Fitness = 7245.7196, Mutation Rate = 0.232907, Features Selected: 466 out of 1000

Generation 125: Best Fitness = 7602.8941, Mutation Rate = 0.120000, Features Selected: 422 out of 1000

Run:  7

Random seed: 48

Generation 1: Best Fitness = 7140.0424, Mutation Rate = 0.240000, Features Selected: 473 out of 1000

Generation 2: Best Fitness = 7203.6069, Mutation Rate = 0.232907, Features Selected: 484 out of 1000

Generation 125: Best Fitness = 7706.6473, Mutation Rate = 0.140000, Features Selected: 398 out of 1000

-----------------------------------------------------------------------------------------------------------------------------
```

```
Run:  8

Random seed: 49

Generation 1: Best Fitness = 7283.4513, Mutation Rate = 0.240000, Features Selected: 449 out of 1000

Generation 2: Best Fitness = 7283.4513, Mutation Rate = 0.232907, Features Selected: 449 out of 1000

Generation 125: Best Fitness = 7674.0071, Mutation Rate = 0.120000, Features Selected: 398 out of 1000

------------------------------------------------------------------------------------------------------------------------------------------

Run:  9

Random seed: 50

Generation 1: Best Fitness = 7199.3604, Mutation Rate = 0.240000, Features Selected: 472 out of 1000

Generation 2: Best Fitness = 7199.3604, Mutation Rate = 0.232907, Features Selected: 472 out of 1000

Generation 125: Best Fitness = 7668.3679, Mutation Rate = 0.080000, Features Selected: 404 out of 1000

------------------------------------------------------------------------------------------------------------------------------------------

Run:  10

Random seed: 51

Generation 1: Best Fitness = 7148.1593, Mutation Rate = 0.240000, Features Selected: 484 out of 1000

Generation 2: Best Fitness = 7223.7432, Mutation Rate = 0.232907, Features Selected: 453 out of 1000
```

Most selected features:

```
[961, 659, 31, 15, 696, 732, 43, 989, 650, 165, 277, 779, 929, 455, 577, 632, 340, 646, 731, 757, 842, 964, 0, 19, 343, 8, 197, 294, 324, 474, 573, 607, 690, 741,
836, 869, 90, 114, 123, 128, 141, 196, 20, 254, 305, 331, 388, 458, 464, 470, 486, 511, 545, 566, 568, 583, 619, 628, 658, 671, 675, 692, 773, 847, 894, 920,
926, 951, 955, 967, 996, 997, 7, 99, 105, 130, 135, 138, 144, 148, 178, 223, 227, 228, 230, 236, 241, 242, 263, 265, 266, 270, 284, 287, 297, 311, 317, 332,
333, 336, 342, 344, 348, 350, 351, 359, 376, 377, 378, 379, 380, 391, 397, 400, 405, 406, 408, 410, 421, 422, 423, 444, 446, 453, 46, 50, 56, 62, 79, 96, 103,
187, 189, 202, 211, 213, 220, 221, 232, 234, 237, 251, 255, 256, 257, 291, 301, 312, 315, 326, 335, 354, 356, 362, 365, 386, 399, 411, 449, 452, 467, 472, 485,
487, 509, 526, 551, 559, 571, 591, 595, 600, 603, 614, 616, 623, 644, 645, 648, 661, 670, 674, 676, 677, 679, 684, 687, 697, 698, 700, 705, 709, 713, 716, 738,
744, 753, 755, 767, 769, 771, 785, 790, 792, 798, 806, 815, 817, 819, 843, 853, 856, 860, 870, 875, 877, 882, 884, 891, 897, 898, 910, 922, 923, 925, 933, 937,
941, 953, 958, 960, 963, 965, 966, 971, 974, 976, 979, 980, 982, 984, 999, 1, 3, 5, 6, 10, 24, 30, 49, 53, 61, 78, 86, 91, 101, 116, 120, 146, 156, 160, 163, 166,
182, 190, 207, 210, 212, 214, 247, 252, 261, 304, 316, 327, 338, 341, 366, 371, 381, 392, 419, 421, 442, 447, 450, 457, 463, 465, 473, 496, 506, 508, 518, 527,
531, 532, 541, 557, 558, 572, 588, 602, 612, 629, 631, 649, 652, 667, 668, 673, 70, 725, 729, 75, 764, 766, 777, 801, 81, 848, 859, 861, 865, 873, 874, 888,
889, 893, 948, 949, 950, 956, 966, 983, 985, 14, 16, 29, 35, 60, 85, 87, 106, 115, 122, 125, 131, 132, 151, 154, 162, 169, 172, 179, 193, 201, 224, 226, 233,
239, 240, 243, 249, 260, 288, 321, 339, 355, 357, 358, 364, 377, 379, 382, 390, 394, 398, 401, 416, 418, 430, 435, 436, 440, 451, 466, 482, 490, 493, 499, 501,
517, 537, 538, 543, 560, 576, 581, 582, 589, 590, 592, 595, 610, 617, 618, 638, 641, 654, 669, 680, 686, 688, 703, 717, 719, 736, 745, 772, 798, 800, 809, 810,
822, 830, 834, 849, 854, 856, 858, 876, 901, 902, 924, 935, 941, 950, 963, 969, 970, 980, 995, 4, 13, 23, 34, 40, 64, 65, 77, 80, 84, 88, 108, 118, 124, 142, 16,
177, 204, 217, 218, 224, 239, 244, 250, 275, 298, 320, 329, 346, 357, 370, 384, 395, 407, 414, 432, 438, 443, 462, 471, 494, 498, 501, 513, 515, 519, 525, 526,
539, 562, 563, 574, 579, 580, 592, 601, 625, 640, 664, 67, 686, 712, 720, 763, 765, 768, 774, 776, 778, 781, 808, 811, 812, 816, 826, 827, 841, 860, 876, 881,
894, 902, 907, 926, 939, 942, 945, 950, 961, 963, 987, 995, 999, 11, 21, 25, 26, 27, 36, 41, 44, 50, 52, 55, 58, 69, 71, 72, 73, 76, 82, 83, 89, 93, 97, 102, 107,
111, 116, 121, 126, 127, 139, 145, 147, 149, 150, 152, 155, 164, 167, 170, 183, 185, 191, 192, 194]
```