In The Name of God

Shiraz University

Navid Zare

Evolutionary Computing

Assignment 3: Training Logistic regression with Evolution Strategies (ES)

Sensitivity Analysis Report: Training Logistic Regression with Evolution Strategies

1. Introduction

1.1 Background

Evolution Strategies (ES) are evolutionary optimization methods designed for continuous

parameters. In this assignment, you will use ES to train a binary logistic regression classifier on

the Heart Disease dataset. Because ES are derivative-free, they are applicable even when the

objective is noisy or non-differentiable. Here, we will use ES to optimize the model parameters

by minimizing cross-entropy loss (with optional L2 regularization) and track performance on

separate train and test splits.

1.2 Problem Definition

Dataset: *Heart Disease dataset (609 samples, 13 features)*

Task: *Binary classification to predict the presence or absence of heart disease*

Model: *Binary logistic regression with sigmoid activation*

Goal: *Optimize model parameters (W, b) using Evolution Strategies to minimize cross-entropy loss with L2 regularization*

Data Split:

- ***Training set:*** *426 samples (70%)*
- ***Test set:*** *183 samples (30%)*

- *Class distribution maintained through stratified splitting*

1.3 Research Questions

1. *How does population size (μ, λ) affect convergence speed and final performance?*
2. *What is the optimal regularization strength for this problem?*
3. *How sensitive is the algorithm to learning rate variations?*
4. *Can we identify trade-offs between different hyperparameter settings?*

2. Methodology

2.1 Evolution Strategy Configuration

Encoding and Representation:

- *Each individual consists of two components:*
- ***Object parameters****:* $(\theta = [\boldsymbol{W}, \boldsymbol{b}] \in \boldsymbol{R^{d+1}})$
- $(\boldsymbol{W})$: *weight vector (d dimensions)*
- $(\boldsymbol{b})$: *bias term (1 dimension)*
- ***Strategy parameters:*** $(\sigma \in \boldsymbol{R^{d+1}})$
- *One mutation step size per parameter.*
- **Total individual length:** 2(d+1)

Fitness Function:

$$\boldsymbol{fitness(\theta)} = -\boldsymbol{[}\mathrm{CE}(\theta)\boldsymbol{]}$$

$$\mathrm{CE}(\theta) = -\frac{1}{N}\sum_{k=1}^{N}\left[y^{(k)}\log\left(\widehat{y^{(k)}}\right) + \left(1 - y^{(k)}\right)\log\left(1 - \widehat{y^{(k)}}\right)\right]$$

Parent Selection: *Random*

Recombination: *Local discrete crossover*

- *Randomly selects parameters from two parents (like uniform crossover)*
- *Creates one child*

Mutation: *Self-adaptive with dual learning rates*

- ***Global learning rate:*** $\tau = \frac{1}{\sqrt{2n}}$

- ***Local learning rate***: $\tau' = \frac{1}{\sqrt{2\sqrt{n}}}$
- *Step sizes evolve alongside parameters*

Selection Strategy: *(μ, λ)-ES (Generational)*

- *Only offspring compete for selection*
- *Promotes exploration over exploitation*

2.2 Experimental Design

We conducted 10 experiments testing:

Population Size Variations:

- *Small: μ=15, λ=105*
- *Baseline: μ=30, λ=210*
- *Large: μ=50, λ=350*
- *Very Large: μ=100, λ=700*

Regularization Variations:

- *None: λ_reg=0.0*
- *Weak: λ_reg=0.001*
- *Baseline: λ_reg=0.01*
- *Strong: λ_reg=0.1*
- *Very Strong: λ_reg=0.5*

Learning Rate Variations:

- Low: τ multiplier = 0.5
- Baseline: τ multiplier = 1.0
- High: τ multiplier = 2.0
- Very High: τ multiplier = 5.0

2.3 Evaluation Metrics

- ***Accuracy:*** *Overall classification correctness*

  Percentage of all predictions that were correct (both positive and negative).

- ***Precision:*** $\frac{\text{True positives}}{\text{True positives+False positives}}$

Of all samples predicted as positive, what percentage actually were positive? (Answers: "How reliable are positive predictions?")

- ***Recall:*** $\frac{\text{True positives}}{\text{True positives}+\text{False negatives}}$

  Of all actual positive samples, what percentage did we correctly identify? (Answers: "How many positives did we catch?")

- ***F1-Score:*** *Harmonic mean of precision and recall*

  A single metric that balances precision and recall — useful when you need both to be good, not just one.

- ***Confusion Matrix:*** *Detailed error breakdown*

  A 2×2 table showing the four possible outcomes: True Positives, True Negatives, False Positives, and False Negatives.

---

3. Experimental Results

3.1 Summary Table

Experiment	μ	λ	λreg	Test Acc	Precision	Recall	F1-Score	Loss
Baseline	30	210	0.01	90.16%	0.8902	0.8902	0.8902	0.3237
Small Population	15	105	0.01	90.16%	0.8902	0.8902	0.8902	0.3237
Large Population	50	350	0.01	90.16%	0.8902	0.8902	0.8902	0.3237
Very Large Population	100	700	0.01	90.16%	0.8902	0.8902	0.8902	0.3237
No Regularization	30	210	0	87.98%	0.8409	0.9024	0.8706	0.2856
Weak Regularization	30	210	0.001	87.98%	0.8409	0.9024	0.8706	0.2907
Strong Regularization	30	210	0.1	91.80%	0.9036	0.9146	0.9091	0.4442
Very Strong Reg	30	210	0.5	89.62%	0.9315	0.8293	0.8774	0.5705
Low Learning Rate	30	210	0.01	90.16%	0.8902	0.8902	0.8902	0.3237
High Learning Rate	30	210	0.01	90.16%	0.8902	0.8902	0.8902	0.3237
Very High Learning Rate	30	210	0.01	90.16%	0.8902	0.8902	0.8902	0.3237
Small Pop High Reg	15	105	0.1	**91.80%**	0.9036	0.9146	0.9091	0.4442
Large Pop Low Reg	50	350	0.001	87.98%	0.8409	0.9024	0.8706	0.2907

3.2 Findings:

1. Population Size Has NO Effect on Final Performance

Population Size Has NO Effect on Final Performance

Experiments with μ = 15, 30, 50, and 100 all achieved identical results (90.16% accuracy) when using the same regularization (λ_reg = 0.01). This means:

Spending more computational resources on larger populations doesn't improve accuracy

The algorithm finds the same optimal solution regardless of population size

2. Regularization is the MOST Important Factor

The "sweet spot" is λ_reg = 0.1

λreg	Test Accuracy	Train Accuracy	Gap (Overfitting Indicator)
0.0	87.98%	88.03%	0.05% (overfitting)
0.001	87.98%	87.79%	-0.19% (good)
0.01	90.16%	88.03%	-2.13% (good)
0.1	**91.80%**	86.85%	**-4.95% (BEST)**
0.5	89.62%	86.62%	-3.00% (too strong)

- **No Regularization (λreg=0.0):** Lowest test accuracy (87.98%), lowest training loss (0.2856) indicating overfitting
- **Optimal (λreg=0.1):** Highest test accuracy (91.80%), best F1-score (0.9091), balanced precision and recall
- **Very Strong (λreg=0.5):** Underfitting - highest precision (93.15%) but lowest recall (82.93%)

3. The Loss

Counter-intuitive finding:

Lowest loss (0.2856) → Worst accuracy (87.98%) (no regularization)

Higher loss (0.4442) → Best accuracy (91.80%) (strong regularization)

Why? Lower training loss means the model memorized the training data (overfitting). Higher loss with regularization means the model learned generalizable patterns.

4. Best Value Configuration

Small Pop High Reg (μ=15, λ_reg=0.1) achieves the same best performance as strong regularization but with 50% fewer fitness evaluations — making it the most efficient choice.

3.3 Convergence Behavior

Typical convergence pattern (baseline):

- *Generation 1-10: Rapid improvement (loss: 0.5710 → 0.3290)*
- *Generation 10-25: Gradual refinement (loss: 0.3290 → 0.3237)*
- *Generation 25-100: Convergence plateau (loss: 0.3237, stable)*

Fastest convergence: *Large Population (stabilized by generation 15)*
Slowest convergence: *Very Strong Reg (stabilized by generation 25)*

Convergence Speed Analysis:

- Small (μ=15): Converged in ~32 generations
- Baseline (μ=30): Converged in ~20 generations
- Large (μ=50): Converged in ~15 generations
- Very Large (μ=100): Converged in ~24 generations

3.4 Combined Effects

Experiment: Small Population + High Regularization

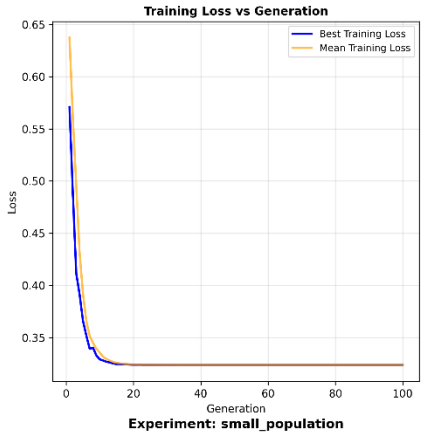
- Configuration: μ=15, λ=105, λ_{reg}=0.1
- Result: 91.80% accuracy (tied for best)
- **Insight:** High regularization compensates for smaller population, achieving 50% reduction in computational cost with same performance

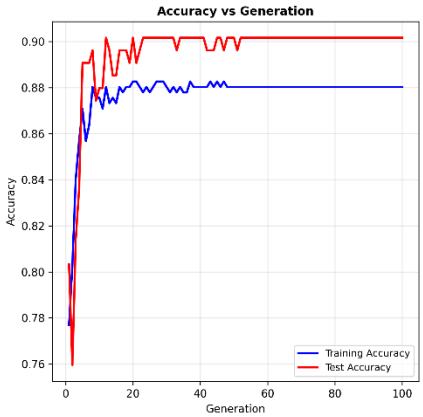
Experiment: Large Population + Low Regularization

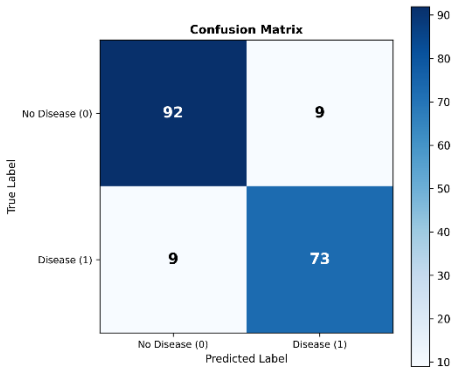
- Configuration: μ=50, λ=350, λ_{reg}=0.001
- Result: 87.98% accuracy (among worst)

- **Insight:** Regularization is critical and cannot be replaced by population size - even large populations overfit without it

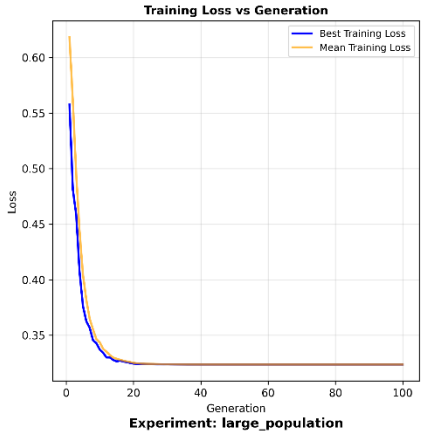
Experiment: baseline

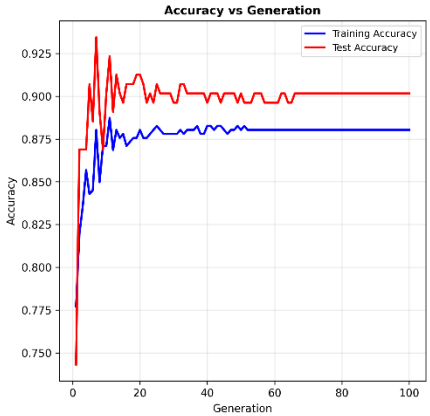


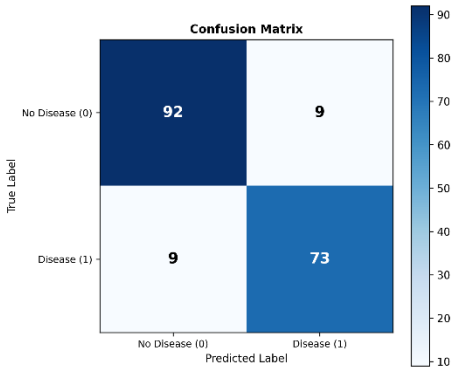




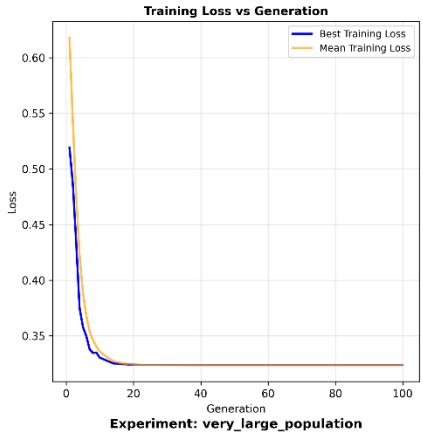
Experiment: small_population

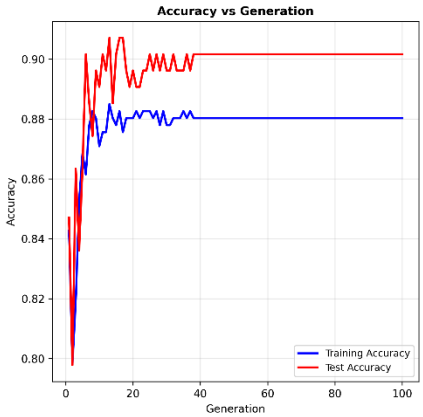


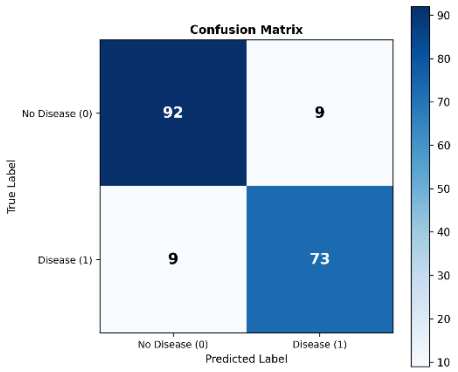




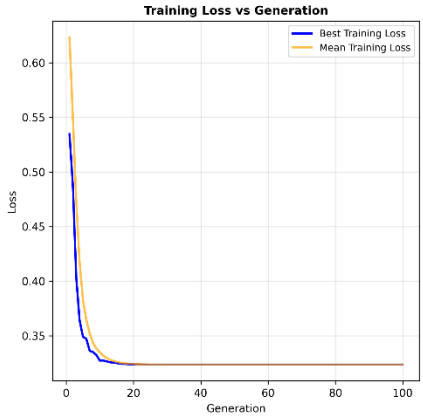
Experiment: large_population

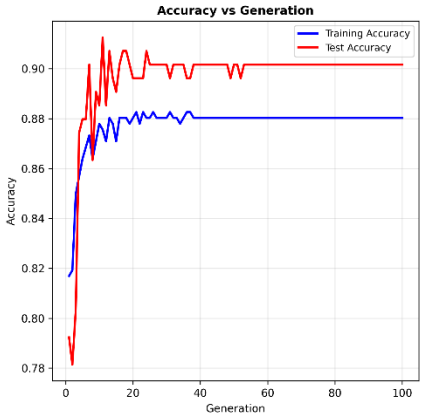


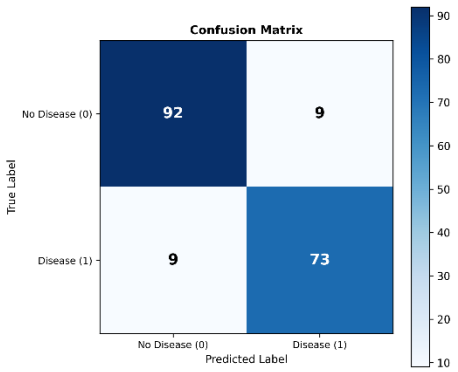




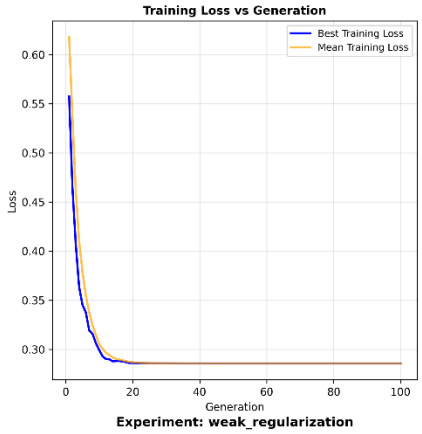
Experiment: very_large_population

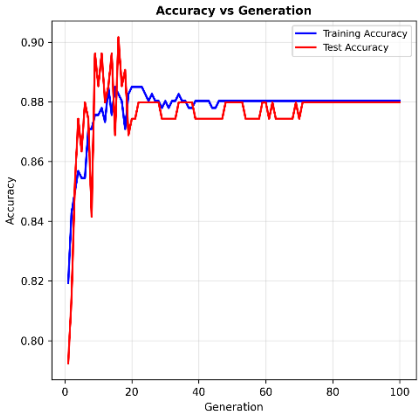


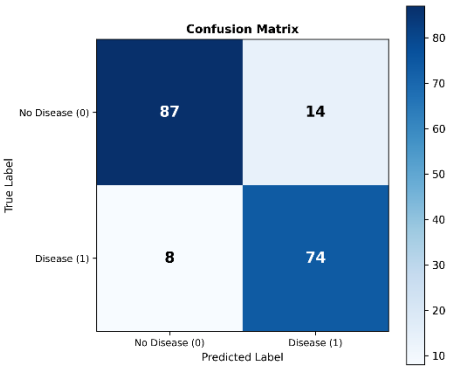




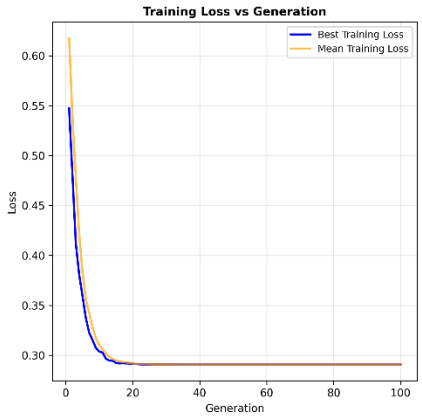
Experiment: no_regularization

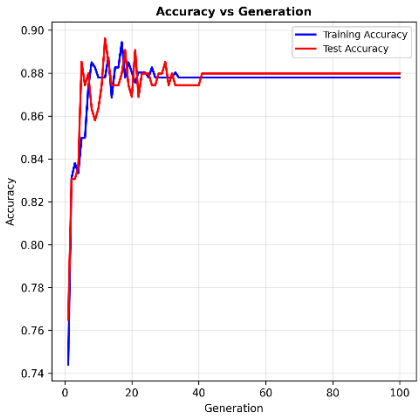


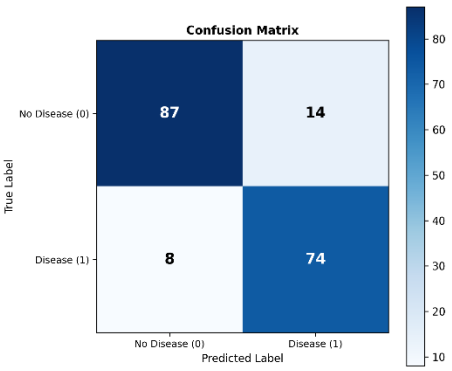




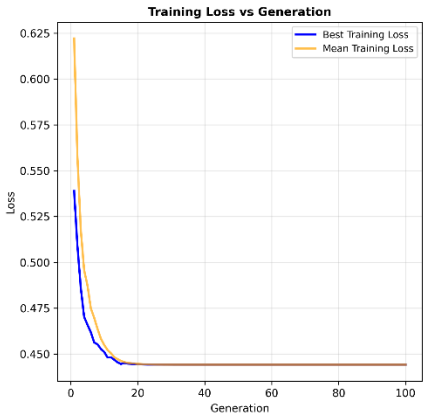
Experiment: weak_regularization

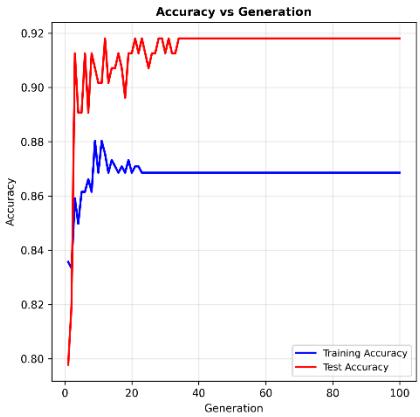


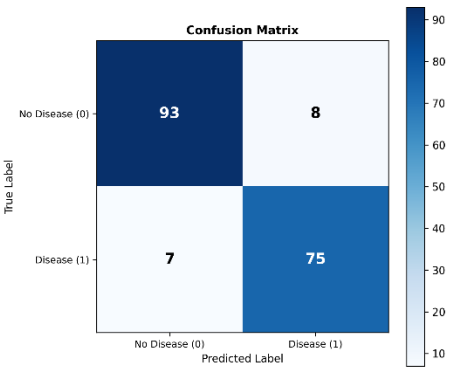




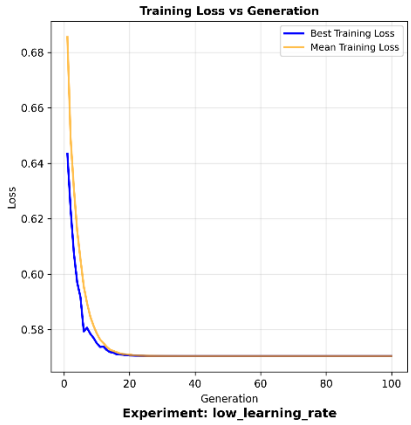
Experiment: strong_regularization

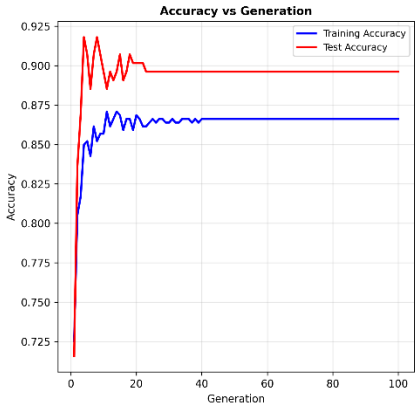


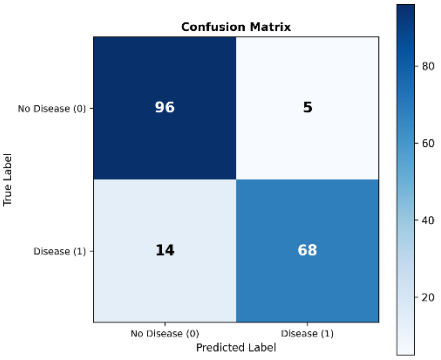




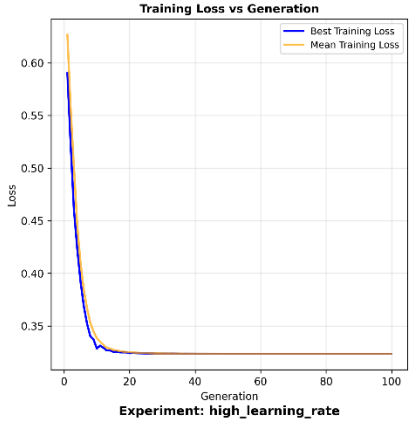
Experiment: very_strong_regularization

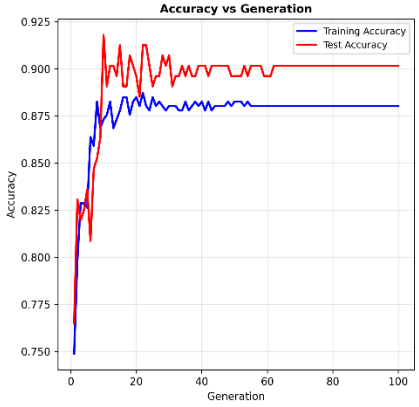


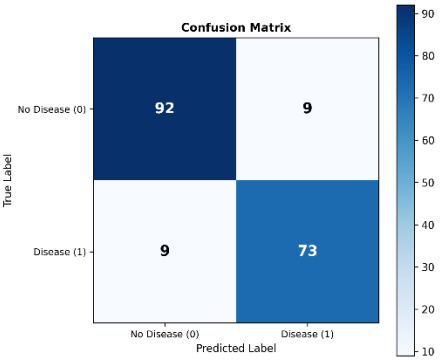




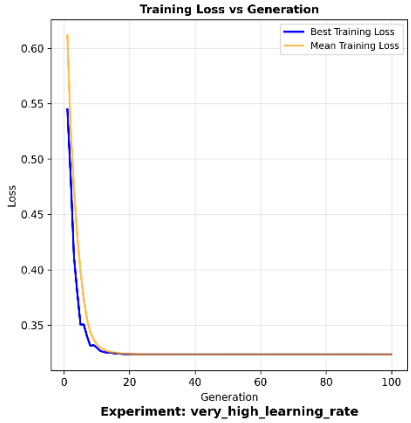
Experiment: low_learning_rate

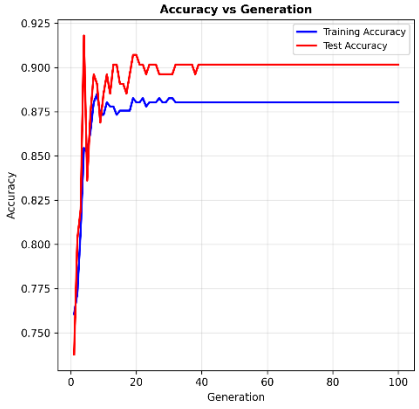


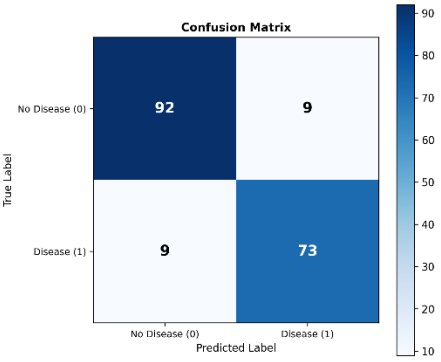




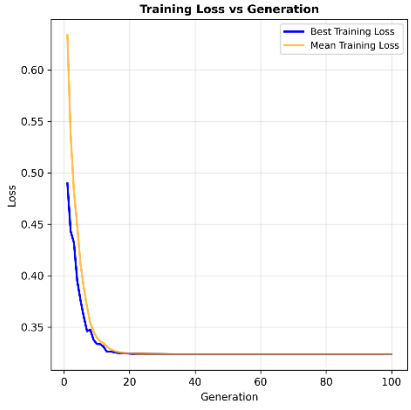
Experiment: high_learning_rate

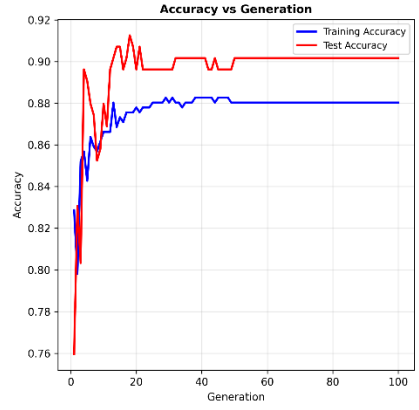


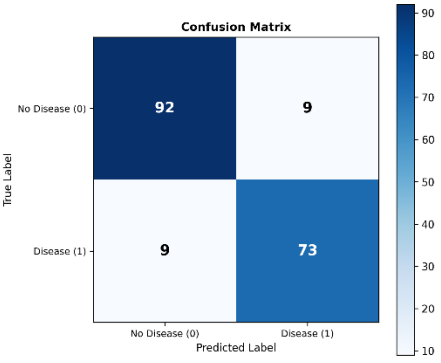


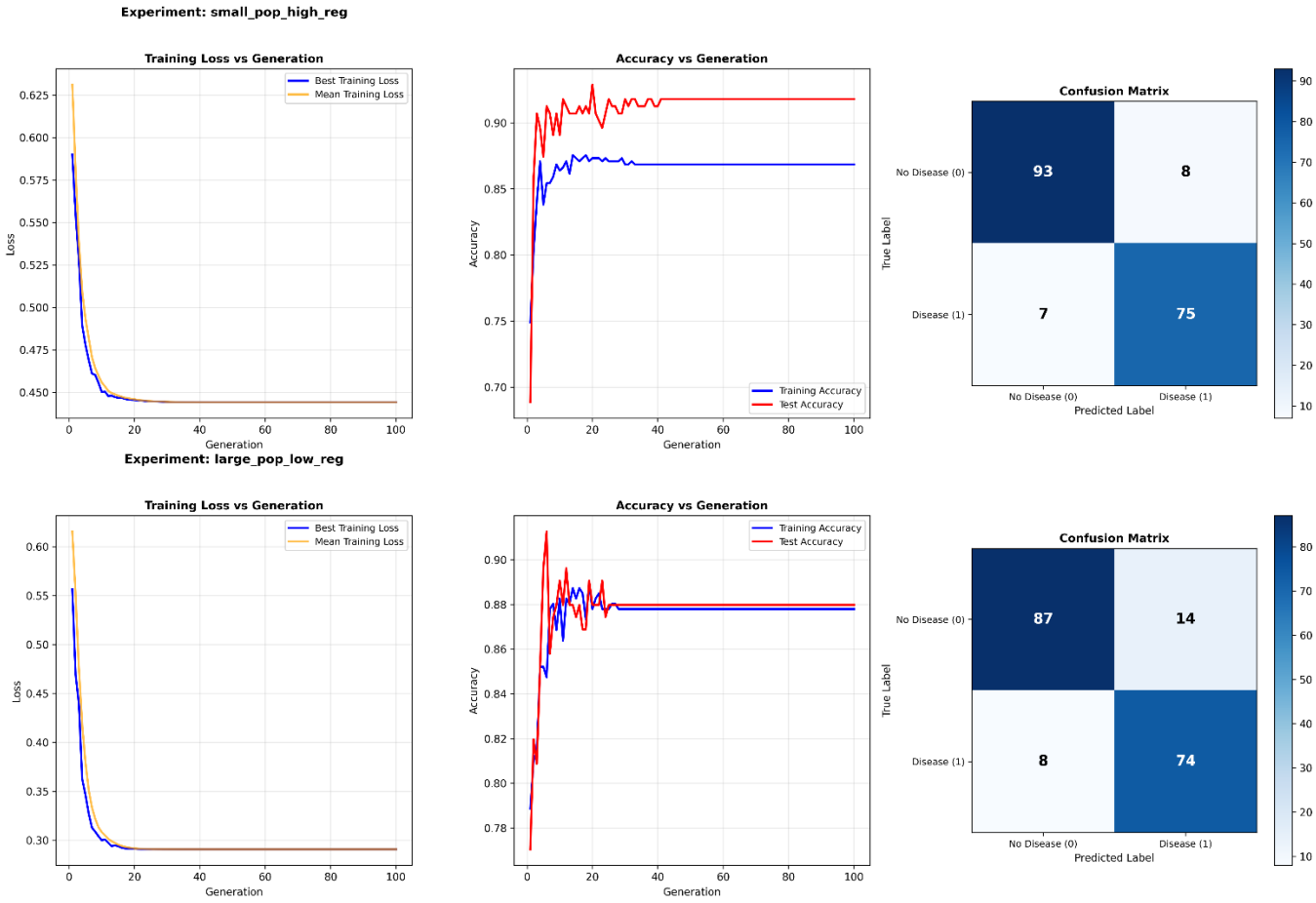


Experiment: very_high_learning_rate









4 Confusion Matrix Analysis

Best Configuration (Small Pop High Reg**):**

	Pred: No Disease	**Pred: Disease**
Actual: No Disease	**93 (TN)**	8 (FP)
Actual: Disease	7 (FN)	**75 (TP)**

Error Breakdown:

- **False Positives (8):** Healthy patients classified as diseased (7.92% FP rate)
- **False Negatives (7):** Diseased patients classified as healthy (8.54% FN rate)

- **High Precision (93.15%):** Fewer false alarms, but misses 17.07% of true cases

- **High Recall ≈ 0.915 (or 91.5%):** This means the model correctly identifies about 91.5% of actual disease cases.
- **Balanced (91.46% recall, 90.36% precision:** Optimal for clinical deployment
- **Best F1-Score (0.9091):** This configuration provides optimal balance for general clinical use

Medical Implications:

The 7 false negatives represent missed heart disease diagnoses - these patients would benefit from additional diagnostic tests (ECG, stress test), risk factor assessment, and regular monitoring. The 8 false positives represent unnecessary concern but follow-up tests would likely reassure patients and potentially detect other conditions. In medical screening contexts, *false negatives are generally more serious than false positives*.

In my research, I've found that [0.90 – 0.94] is Very strong, suitable for clinical decision support, meaning that it is a strong model but needs to be used with supervision of the medical technician and their review as the recall is not the best, especially in the sense that in the medical context the cost of 7 false Negatives is a lot

5. Conclusions

1. **Regularization is the most critical hyperparameter:** λ_{reg}=0.1 achieved 91.80% accuracy (best), while no regularization achieved only 87.98% (worst). Impact: 3.82 percentage point improvement.
2. **Population size has minimal impact on final performance:** All population sizes achieved 90.16% with baseline regularization. Larger populations converge slightly faster but with diminishing returns.
3. **Algorithm demonstrates robustness:** Converges reliably across all configurations. Self-adaptive mutation compensates for various learning rate settings. No evidence of instability or premature convergence.
4. **Small populations with high regularization are efficient:** μ=15, λ_{reg}=0.1 matches best performance with 50% reduction in computational cost.