Genetic Algorithm for Diagnosing Diabetes

Project Overview

This project focuses on optimizing a Support Vector Machine (SVM) classifier using a Genetic Algorithm (GA) to improve the accuracy of diabetes diagnosis. The Pima Indians Diabetes dataset is utilized for this purpose. The core idea is to leverage the power of GA to automatically tune the SVM's hyperparameters (specifically 'C' and 'kernel'), thereby overcoming the limitations of traditional manual or trial-and-error tuning methods that often lead to suboptimal accuracy and misclassifications.

Problem Statement

Many traditional diabetes prediction models use fixed parameters or trial-and-error tuning, which often fail to deliver the best accuracy. This results in misclassification, delaying diagnosis or falsely alarming healthy individuals. Diabetes affects millions worldwide and early diagnosis is crucial. Inaccurate models can lead to missed diagnoses, wrong treatments, and increased healthcare burden. Every patient’s data is different, but using one-size-fits-all parameters fails to capture these differences.

Our Innovation

We used a Genetic Algorithm to automatically optimize the SVM model's hyperparameters, leading to a more accurate, customized, and efficient diabetes prediction system.

Objective

The primary objective is to optimize the hyperparameters of a Support Vector Machine (SVM) classifier using a Genetic Algorithm (GA) in order to maximize the classification accuracy for diagnosing diabetes using the Pima Indians Diabetes dataset.

Dataset

The project utilizes the Pima Indians Diabetes Dataset, obtained from Kaggle / UCI Repository, in CSV format. It contains 768 real-world patient details.

Features in the dataset include:

- Pregnancies
- Glucose
- Blood Pressure
- SkinThickness
- Insulin
- BMI
- DiabetesPedigreeFunction
- Age

These features are used to train and evaluate an SVM classifier that predicts whether a patient is diabetic.

Libraries & Tools Used

The following libraries and tools were used in this project:

Data Handling & Manipulation

- pandas: For reading and processing CSV files.
- numpy: For numerical operations & arrays.

Preprocessing & Dimensionality Reduction

- sklearn.preprocessing.StandardScaler: For feature normalization (scaling).

- sklearn.decomposition.PCA: To reduce features to 2D for visualization.

Model Building and Optimization

- sklearn.svm.SVC: The SVM model used for classification.
- deap: The Genetic Algorithm framework for optimization.

Visualization

- random: Used to generate initial fireflies.
- matplotlib.pyplot: For plotting clusters, visuals.
- seaborn: For heatmaps, feature visuals.

Genetic Algorithm (GA) - Nature-Inspired Optimization

The Genetic Algorithm is a bio-inspired evolutionary optimization technique based on natural selection and genetics. GA mimics the process of evaluation using selection, crossover, and mutation to evolve better solutions. The fitness is the objective function, which in our case is model accuracy or cross-validation score. Each individual (chromosome) represents a solution. Our objective is to use GA to find the best combination of C, gamma, and kernel. The goal is to maximize the accuracy of the SVM classifier on diabetes prediction.

Key GA Parameters needed:

- Population size
- cxpb [Crossover Probability]
- Mutpb [Mutation probability]

Architecture

The overall architecture of the system involves the following steps:

1. **Load Dataset:** The Pima Indians Diabetes dataset is loaded.
2. **Preprocess Data:** The loaded data undergoes preprocessing.

3. **Initialize Genetic Algorithm:** The GA is initialized. This involves defining an individual as a set of SVM hyperparameters (C and kernel).
4. **Fitness Function (Accuracy):** The accuracy of the SVM model is evaluated for each individual, serving as its fitness.
5. **Optimize SVM Hyperparameters:** The GA iteratively optimizes the SVM hyperparameters. This involves:
    - **Selection:** Tournament Selection was used to pick the fittest individuals.
    - **Crossover:** Blend crossover helped combine features from parent solutions.
    - **Mutation**: Gaussian mutation introduced slight variations to explore better parameters.
6. **Train SVM on Best Params:** Once the GA identifies the best parameters, the SVM is trained using these optimal settings.
7. **Predict + Evaluate Accuracy:** The trained SVM model is used for prediction, and its accuracy is evaluated.
8. **End:** The process concludes.

Optimization Outcome (Results)

The Genetic Algorithm successfully optimized the SVM hyperparameters, yielding the following results:

Parameter	Value
Best C	**52133**
Best kernel	**RBF**
Best Accuracy	**0.7786**

Conclusion & Key Takeaways

- **Dataset Insight:** The Pima Indians dataset contains key medical predictors such as glucose level, BMI, age, and more, which are crucial for diabetes diagnosis.
- **Model Used:** An SVM (Support Vector Machine) classifier was used due to its effectiveness in handling classification problems with high-dimensional data.
- **GA Optimization:**
    - **Selection:** Tournament Selection was used to pick the fittest individuals.
    - **Crossover:** Blend crossover helped combine features from parent solutions.
    - **Mutation:** Gaussian mutation introduced slight variations to explore better parameters.
- **Performance Improvement:** The SVM with GA-optimized hyperparameters outperformed the default model, validating the importance of hyperparameter tuning.
- **Automation Advantage:** GA removes the need for manual grid search, making hyperparameter tuning more efficient and adaptive.
- **Scalability:** The same GA-based approach can be applied to other datasets or models, making it a reusable and adaptable tool in ML pipelines.