# **Gene Expression Classification**

## Introduction

This document provides an overview of the gene expression classification project, including the methodologies, data preprocessing, model training, and results. The project explores the use of various machine learning models to classify gene expression data.

## Prerequisites

To work with the code and data in this project, ensure you have the following Python libraries installed:

- pandas

- numpy

- matplotlib

- seaborn

- scikit-learn

- xgboost

- keras

## Dataset

The project uses gene expression data provided in CSV format. The dataset consists of training and test sets, along with actual labels for classification.

## Data Preprocessing

Data preprocessing includes several key steps:

1. Handling missing values

2. Label encoding

3. Feature scaling

4. Principal Component Analysis (PCA) for dimensionality reduction

## Model Training

The project includes the training of various machine learning models, each with its specific focus and methodology.

### Support Vector Machine (SVM)

- Hyperparameter optimization

- Visualization of decision boundaries

- Classification report and confusion matrix

### Random Forest

- Hyperparameter tuning using GridSearchCV

- Feature importance analysis

- Learning curve and ROC curve

### Neural Network

- Hyperparameter search

- Model training with early stopping

- Confusion matrix

- Validation loss and accuracy analysis

## Result

The models' accuracy is compared and displayed in a bar plot. The maximum accuracy achieved is highlighted.

For in-depth details and code, please refer to the provided Jupyter Notebook and code files in this project.

Gene Expression Classification