

Linear Classifiers: Perceptron, SVM, Softmax, Logistic Regression

This project focuses on implementing and experimenting with linear classifiers using Python and scientific computing tools. The goal is to understand the fundamentals of core classification models and apply them to real datasets while exploring hyperparameter tuning and model evaluation practices.

Datasets Used

- Rice Dataset: A small binary classification dataset with labels 0 and 1.
- Fashion-MNIST: A multi-class image classification dataset consisting of 28×28 grayscale images across 10 categories.

Models Implemented

1. Perceptron (`models/perceptron.py`)

- Binary and multi-class classification using a simple update rule.
- Hyperparameters explored: learning rate, number of epochs.

2. Support Vector Machine (SVM) (`models/svm.py`)

- Implemented soft-margin SVM with hinge loss optimized via Stochastic Gradient Descent (SGD).
- Hyperparameters explored: learning rate, epochs, regularization constant.

3. Softmax Classifier (`models/softmax.py`)

- Multi-class linear classifier trained using softmax activation and cross-entropy loss, optimized with SGD.
- Hyperparameters explored: learning rate, epochs, regularization constant.

4. Logistic Regression (`models/logistic.py`)

- Binary classification using sigmoid activation and binary cross-entropy loss.
- Applied only to the Rice dataset due to the binary label structure.
- Hyperparameters explored: learning rate, epochs, threshold.

Main Notebook

All implementations are brought together in the notebook:

Training_and_Evaluating_Linear_Models.ipynb

This notebook walks through data loading, training, hyperparameter tuning, and performance evaluation of all four models on both datasets.