Problem Set 2: Linear Classifiers and Gradient Descent

Ankit Jambhulkar

October 12, 2024

1 Introduction

This report describes the implementation of linear classifiers using Support Vector Machines (SVM) with hinge loss and Softmax classifiers with cross-entropy loss, We use gradient descent to optimize the classifier's parameters.

2 Methodology

2.1 Score Function

The score function for a linear classifier is computed as:

$$f(x) = W \cdot X + b$$

Where W represents the weights, X is the input data, and b is the bias term.

2.2 Hinge Loss (SVM)

The hinge loss for an SVM is given by:

$$L = \sum_i \max(0, 1 - y_i \cdot f(x_i))$$

This loss function maximizes the margin between data points and the decision boundary.

2.3 Cross-Entropy Loss (Softmax)

For Softmax classifiers, the cross-entropy loss is computed as:

$$L = -\sum_i \log \frac{e^{f(x_i)_{y_i}}}{\sum_j e^{f(x_i)_j}}$$

This measures how well the predicted probabilities match the true labels.

2.4 Regularization

L2 regularization is used to penalize large weights:

$$L_{reg} = \lambda \sum_{i,j} W_{i,j}^2$$

Where λ controls the strength of the regularization.

2.5 Gradient Descent

Gradient descent is used to minimize the loss function by iteratively updating the weights and biases:

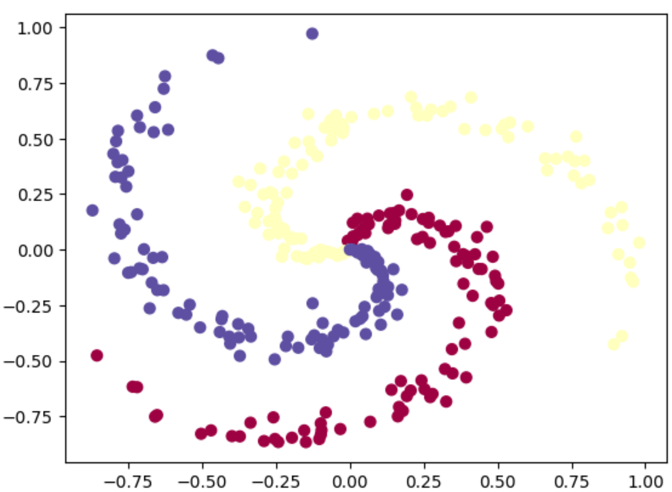
$$W = W - \alpha \nabla_W L$$

$$b = b - \alpha \nabla_b L$$

Where α is the learning rate, and $\nabla_W L$ and $\nabla_b L$ are the gradients of the loss with respect to W and b.

3 Results

Include any plots or results generated from your Python code.



4 Conclusion

This assignment helped me reinforce my understanding of linear classifiers, loss functions, regularization, and gradient descent.