Project1 Machine learning Report
Neural Networks

Abdelateef Khaled Abdelateef 202001344

Department of Engineering, University of Science and Technology at
Zewail City, Egypt

Contents

1 Problem Definition and Motivation

In this project, I aim to implement a neural network from scratch and apply it to two different datasets: IRIS and MNIST. The motivation behind this task is to gain a deeper understanding of neural network fundamentals, evaluate the model's performance on real-world datasets, and compare it with established machine learning libraries like scikit-learn and Keras.

2 Dataset

2.1 IRIS Dataset

The IRIS dataset is a well-known dataset in machine learning, consisting of features such as sepal length, sepal width, petal length, and petal width for three different species of iris flowers: setosa, versicolor, and virginica.

2.2 MNIST Dataset

The MNIST dataset is a collection of handwritten digits widely used for training various image processing systems. It consists of 28x28 pixel grayscale images of handwritten digits (0-9).

3 Approach and Methodology

My approach involves implementing a simple neural network with the option to choose between one and two hidden layers. I perform forward and backward propagation for training. The model's hyperparameters, such as the learning rate (η), maximum number of iterations, precision error enhancement, and desired accuracy, can be adjusted based on the dataset and problem.

4 Implementation

4.1 Tools and Libraries

I utilized Python for my implementation, making use of popular machine learning libraries such as scikit-learn and Keras. The code was executed in a Jupyter Notebook environment.

4.2 Data Pre-processing

For both IRIS and MNIST datasets, I applied standardization using `StandardScaler` and shuffled the data to ensure randomness. For the MNIST dataset, I also performed one-hot encoding using `LabelBinarizer`.

4.3 Model Parameters

The neural network has the flexibility to choose between one or two hidden layers, with the number of neurons determined by the number of features. I applied the sigmoid activation function for hidden layers and softmax for the output layer. The hyperparameters were fine-tuned based on iterative testing.

4.4 Model Evaluation

I evaluated the model using accuracy as the performance metric. Additionally, for MNIST, I visualized the confusion matrix using Seaborn.

5 Conclusion

My neural network achieved satisfactory results on the IRIS dataset, with a final accuracy of 96.67

During implementation, I gained insights into neural network training, hyperparameter tuning, and the importance of data pre-processing. To improve models, further exploration of hyperparameters and advanced architectures could be considered.