Kevin Lopez

CIS400 - Evolutionary Machine Learning

2/4/2024

HW0 Code Report

Purpose and Overview

The provided code implements a binary classification task on the Digits dataset using a shallow feedforward neural network (NN). It involves data preprocessing, model training, evaluation, and visualization. Specifically, it does the following:

Loads the Digits dataset, keeping only two classes for binary classification.

Splits the data into training and testing sets, with a balanced number of samples for each class.

Introduces class imbalance in the training data and random label noise.

Defines a shallow feedforward neural network model using PyTorch.

Implements functions for training the model, plotting loss curves, and calculating confusion matrices.

Trains the model for multiple epochs, repeating the experiment ten times with shuffled training data.

Visualizes confusion matrices for both training and testing data for 10 iterations

Plots average loss curves across experiments.

Libraries and Dependencies

The code relies on the following libraries:

NumPy: For numerical computing.

PyTorch: For building and training neural network models.

Matplotlib: For data visualization.

Scikit-learn: For loading the dataset, data preprocessing, and calculating confusion matrices.

Data Preprocessing

The Digits dataset is loaded and filtered to include only samples belonging to two specific classes.

100 data points from each class are set aside for testing, while the rest are used for training.

Class imbalance is introduced in the training data by duplicating samples from the minority class.

Random label noise is introduced by randomly relabeling 5% of the training samples.

Model Architecture

The shallow feedforward neural network model consists of an input layer, one hidden layer, and an output layer.

Training and Evaluation

The model is trained for a specified number of epochs, with training and testing losses recorded at each epoch.

Confusion matrices are calculated for both training and testing data to evaluate model performance.

The training process is repeated ten times with shuffled training data to assess model robustness.

Visualization

Confusion matrices for both training and testing data are visualized to understand the model's performance in classifying each class.

Average loss curves across the ten experiments are plotted to analyze the training and testing convergence.

Conclusion

The provided code demonstrates the process of building and training a shallow neural network for binary classification on the Digits dataset. Overall, the code serves as a comprehensive example of training a neural network for binary classification tasks.