CS587: Assignment 3

Build, Train & Test a Multilayer Neural Networks using PyTorch

Issued: Monday, 15/04/2024
Deadline: Tuesday, 23/04/2024, 23:59

Description

The goal of this assignment is to implement, train and test a Multi-Layer Feed Forward Neural Network using the SGD algorithm and automatic differentiation based on the PyTorch framework.

You will use the Digits dataset (10 class handwritten digits) for the task of image classification. The given source code automatically downloads the dataset using the Scikit-learn Python library (if you use the Anaconda distribution, it is already installed in your Python environment) and processes it with PyTorch.

You can use any available function of *NumPy*, *SciPy*, *PyTorch*, *TensorBoard* required for the following tasks.

1. Build and train a feed-forward neural network using the Stochastic Gradient Descent method. Use the set of hyperparameters provided in the source code.

   - Build the structure of your model connecting the input layer of your network to one hidden layer h of size $hid\_size = 15$ and then to an output layer. Create the weights W_h and the bias b_h required for the hidden layer, as well as those for the output layer W_o, b_o. Your output layer produces 10 values z^o_j, $j = \{0, .., 9\}$, which match the number of target classes in the dataset. The scores z^o of your output neurons are computed as:

   $$z^o(x) = W_o\, h(x) + b_o, \tag{1}$$

   where $h(x)$ are the activation responses of the neurons in the hidden layer.
   - Use *Sigmoid* as your activation function for the hidden layer $h(x)$.
   - Use the Softmax cross entropy error function to estimate the probability errors between the scores of the output neurons o_j in your model and the target labels t_j of your training samples for $j = \{0, .., 9\}$.
     For each sample of the training batch/set that contains N training samples:

   $$t_j = \begin{cases} 1 & \text{iff } j = \text{ correct class} \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

     Let o_j denote the resulting softmax function for each output neuron j:

   $$o_j = softmax(z^o_j) = \frac{e^{z^o_j}}{\sum_j e^{z^o_j}}, \tag{3}$$

     where the sum in the denominator is over each neuron in the output layer.
     Then the cross entropy function $CE(t; o)$ for a training sample $i \in \{0, N-1\}$ is defined as follows:

   $$CE(t, o) = -\sum_j t_j log(o_j), \tag{4}$$

   - Finally, compute the loss of your model to be minimized as the sum of CE for all samples x of your training batch/set.

   $$loss = \sum_{i=1}^{N} CE(t(i), o(i)), \tag{5}$$

2. Train your model and check the generalization on the test samples.

   - Train your model using the Stochastic Gradient Descent algorithm with mini-batches and check the accuracy in the train set.
   - Compute the predictions in the test set.
   - Compute the average accuracy of the model in the test set.
   - Visualize the graph of your model, the histories of loss and accuracy scores on train/test sets, and other quantities (i.e. weights) you may find useful using *TensorBoard*.

3. In order to maximize the performance on the given dataset try different settings for your model:

   A. Experiment with different hyperparameters
   (e.g. learning rate $= 0.001, \ldots, 0.1$, batch size $= 8, \ldots, 128$, hidden layer size $= 5, \ldots, 25$).

   B. Try different activation functions (e.g., ReLU, TanH).

   C. Try to add more hidden layers (using the same or different activation functions) to the model and increase their size.

   D. Add L2 regularization (e.g., with regularization strength 10^{-4})

   Explain your choices briefly and include the generated TensorBoard figures for the new model in your report.

   BONUS: Extra points (up to +15%) will be distributed to the top-performing models based on the accuracy on the test set.

4. Answer briefly the following theoretical questions in your report:

   (a) Describe the *sigmoid* activation function. How is it defined, when do we use it, and what are its advantages and disadvantages?

   (b) How does the *Softmax Cross Entropy* loss function work?

Important Notes

Setup your environment

You can continue using the Anaconda environment from the previous assignments for this task. However, for visualization purposes, you'll also need to install the TensorBoard package. The installation instructions can be found in the relevant Class Tutorial or within the provided assignment code.

Dataset

You do not need to manually download the Digits dataset. It will automatically be set once you run the given source code. Check the provided comments in your code for more details on the Digits dataset.

Submission info

- Create a .pdf (preferred) or .doc file to report the resulting scores, images/figures, and any other comments or description of your work you may need to submit. Do not forget to include your name and ID in the report. Save this file in your working folder.
- Do not forget to include your TensorBoard summaries in your submission.
- Use zip/rar/gz to compress your working folder and rename it to cs587_mylogin_assignment3.xxx in order to submit a single file.
- The submission of your implementation will be via the e-learn platform. Submissions via e-mail will not be accepted.
- You can upload your submission as many times as you need, keeping the same filename.

- Late submissions will be accepted **within two days** of the original deadline. However, **a penalty of 25% per day** will be applied to the final grade for each late day. Assignments submitted more than two days late will not be accepted.

Academic integrity

The assignment is individual. You may discuss with each other in general terms, but the code and the report should be written individually. If you use existing material, be sure to cite the sources in your report. The use of AI language models to produce the report or your implementation is strictly prohibited, and submissions will be verified with an AI detection tool. You may be asked to take an additional short oral examination.

Troubleshooting & Contact

If you encounter any errors or bugs in the provided code, you can report them by sending an email to kaziales@csd.uoc.gr or the course mailing list hy578-list@csd.uoc.gr. Please note that the course mailing list is the preferred channel for general questions about the assignment, as it allows everyone in the class to benefit from the answer.

For any questions of a more personal nature that are not relevant to the entire class, you can reach out to the teaching assistant directly at his email address kaziales@csd.uoc.gr.