Advanced Topics in Machine Learning
Assignment 1

University of Bern

Due date: March 24th, 2020

1 Introduction

In this assignment you need to upload a zip file to ILIAS which includes: 1) A Jupyter Notebook file Assignment1.ipynb completed with code and answers and 2) a Jupyter Notebook exported to HTML (File / Export Notebook as / HTML). The zip file name must be FirstName LastName.zip. If your implementation requires auxiliary functions, you must implement that function inside the corresponding .py file. Please state your name at the beginning of the notebook.

1.1 Notes on code and submission quality

In addition to answering the different questions, you are also expected to provide well written submissions. Here are some recommendations to take into consideration.

- Please answer the question in the same order as in the assignment and use the same question numbers;
- Don't answer the questions in the code comments. Use the text cells in your notebook;
- Remove clutter such as unused code lines instead of turning them into comments;
- Make sure the right execution order of the notebook cells is from top to bottom. A TA should be able to reproduce your results by simply clicking "Run All" without having to guess which cells should be executed first.

Poorly written submissions might result in points deduction.

2 Problem

You are given 5000 images (64x64 pixels) and their corresponding label - each image belongs to 1 of 10 classes. Your task is to train a linear classification model.

Tasks:

1. **Prepare data for training and model selection**
   - Load the data and visualize an image for each class.
   - Set aside 90% of training data for training, and the remaining 10% for validation. Make sure the label distribution is the same in training and validation set.
   - Implement a Dataset class that extends Pytorch's Dataset class. Create dataset objects and dataloaders for training and validation datasets. Make sure you normalize your data.
2. **Train a linear classification model**
   - Implement a linear classification model that will predict the label of a given image. Implement the training of the model with PyTorch's optimizer and a loss function appropriate for the task.
   - Train at least 3 models using different (reasonable) learning rates, train for at least 10 epochs. For each model evaluate on validation dataset after every epoch. Plot training and validation loss and accuracy per epoch. Select the best model based on the results on the validation set
   - Report learning rate for the selected model as well as loss and accuracy on training and validation sets.
3. **Evaluate on the test set**
   - Evaluate your selected model on the test set. How is the accuracy different from the training and validation set? What is this a sign of?
   - Visualize a few images from your test dataset. Are there any differences between train and test images?
   - Visualize the weights of your linear model. What did the model learn? Why does it fail on test images?
4. **Fix the observed problem**
   - Based on observations from the previous point, propose a way to fix your solution; you can still only use a simple linear model.
   - Repeat training and evaluation from points 2 and 3 with your improvements. Report accuracy on training, validation and test sets.
   - Show if your model learns something more meaningful than before (visualize weights of the new model). Comment on the differences.