UNIVERSITY OF CAMBRIDGE

MPhil in Data Intensive Science

Michaelmas Term 2024

M1 Machine Learning Coursework

S. Krippendorf

*Attempt **all** parts of the coursework.*

The anticipated number of marks allocated to each part of a question is indicated in the right margin, to be assessed from the both the code repository and your report.

This coursework should be submitted via a GitLab repository which will be created for you. Place all of your code and your report into this repository. The report should be in pdf format and placed in a folder named `report`*. You will be provided access to your repository until 23:59pm on Wednesday the 18th of December, after which we will assess whatever the repository contains at that time.*

You are expected to submit code and associated material that demonstrates good software development practices as covered in the C1 Research Computing module.

Your report should not exceed 3000 words (including tables, figure captions and appendices but excluding references); please indicate its word count on its front cover. You are reminded to comply with the requirements given in the Course Handbook regarding the use of, and declaration of use of, autogeneration tools.

Failure to include the wordcount and a declarations of use of autogeneration tools will result in automatic loss of marks.

Preamble

In this coursework we consider the following problem. We want to build an inference pipeline which can successfully add two handwritten MNIST digits as illustrated in Figure 1.

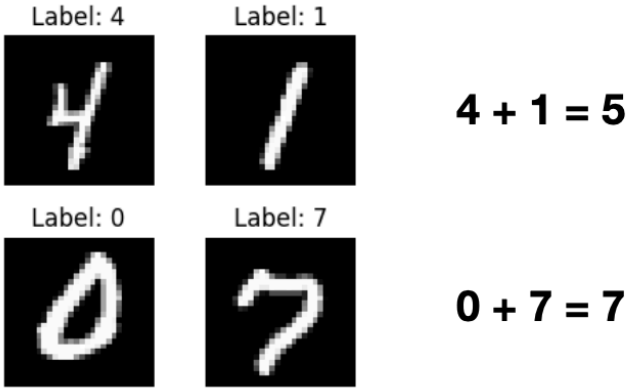

Figure 1: Two examples of the task you should perform: The first row shows two MNIST digits with labels four and one respectively. The target output of your regressor should be 5 in this case. Similarly for the second row the target output should be 7.

Please comment your code and provide code such that your results are reproducible. Your key plots need to be labelled such that they are useable for scientific publications. Your report should be concise and to the point rather than aiming at the upper word count limit.

Tasks

1 Write code that allows you to construct a dataset which combines the images and provides the appropriate labels. The input images should be of shape 56×28. Generate appropriate training, validation, and test datasets. Justify your choices in the generation of these datasets, i.e. ensure that the appropriate statistical properties are guaranteed. [10]

2 Using these datasets, develop a neural network pipeline using fully connected neural networks which allows you to address this inference/classification process. The output of your neural network can correspond to a single number, e.g. $7 + 3 = 10$. You should perform some hyper-parameter tuning to establish a good architecture (a choice of five hyper-parameters is what we look for and you should restrict the number of experiments that you can run them on a personal laptop using no more than a few hours). Your report should include a short overview of the results you have obtained in this hyper-parameter search. You should provide the weights of your best performing neural

network and the neural networks utilised in plots (i.e. the plots should be reproducible). [40]

3 Now, showcase the performance using other inference algorithms implemented in `scikit-learn` covered in the lectures such as random forest classifiers and support vector machines and compare the performance. Note that you are not required to describe the algorithms in detail. [15]

4 Using a weak linear classifier, compare the classifier probabilities for a linear classifier trained on the 56×28 dataset with the probabilities of a single linear classifier which is applied on the two images sequentially. How do the performances on the test set compare when trained with few samples (i.e. train with varying number of samples including 50,100,500, and 1000 samples respectively). [25]

5 For your best performing neural network, show the t-SNE distribution of the various classes in the embedding layer (i.e. the layer before the output). Compare this representation with the representation obtained by directly applying t-SNE on the input dataset. You should optimise the perplexity. [10]

END OF PAPER