

ECS759P Artificial Intelligence: Coursework 2

Due date: 8 December 2023 at 10.00 am

If you have any questions, you are encouraged to ask them and lead discussions via the Student Forum on the module QMPlus page. Additionally, Dr. Julia Ive's office is open for inquiries during office hours (Fridays 12pm-1pm, CS 309) and via email to j.ive@qmul.ac.uk. That said, using the forum for discussion is preferable as it can help us broadcast essential clarifications to the entire group.

1 Introduction

The coursework is composed of two questions. The first is a writing exercise about logic and reasoning. The second exercise is on classification and involves programming.

- Crystal clear (Logic problem): 50%
- Lost in the closet (Classification): 50%

2 Crystal clear (Logic problem)

Although you are looking for it everywhere, you cannot find your true love. A bit desperate, you decide to see Madame Irma, the most famous (and serious) fortune teller of the city. On the entrance, you see a sign stating: *Everything that I say must be proved to be believed*. More perplexed than ever, you still go inside. After glaring at you for some time, she looks into her crystal ball, which has a strange glow, and says in a mysterious voice:

- You have a dog.
- The person you are looking for buys carrots by the bushel.
- Anyone who owns a rabbit hates anything that chases any rabbit.
- Every dog chases some rabbit.
- Anyone who buys carrots by the bushel owns either a rabbit or a grocery store.
- Someone who hates something owned by another person will not date that person.

The sentences you just heard reminds you of a person: Robin. But before you leave, she challenges you with a conclusion:

- If the person you are looking for does not own a grocery store, she will not date you.

Remembering the sentence at the entrance, you realise that what she has told you is true only if you can prove her challenging conclusion. Since you do not want any awkward situation, you decide to provide proof of her conclusion before going to see Robin.

1. Express Madame Irma's six statements into First Order Logic (FOL). **Note:** You can use two constants: YOU and ROBIN.
This question carries 10% of the mark for this coursework.
2. Translate the obtained expressions to Conjunctive Normal Forms (CNFs, Steps 1-6 of Lecture 9: Logic). Show and explain your work.
This question carries 10% of the mark for this coursework.
3. Transform Madame Irma's conclusion into FOL, negate it and convert it to CNF (Steps 1-6 of Lecture 9: Logic). Show and explain your work.
This question carries 10% of the mark for this coursework.

4. Based on all the previously created clauses (you should have at least 7 depending on how you split them), finalise the conversion to CNF (Steps 7-8 of Lecture 9: Logic) and provide proof by resolution that Madame Irma is right that you should go to see Robin to declare your (logic) love to her. Show and explain your work, and provide unifiers.

This question carries 20% of the mark for this coursework.

Note: Make sure to follow the order of steps for the CNF conversion as given in Lecture 9, and report all the steps (state “nothing to do” for the steps where this is the case).

3 Lost in the closet (Classification)

You are an artist who secluded yourself for years to come up with the perfect design for a new brand of clothes. However, your time off from civilisation was not so beneficial since you cannot distinguish a T-shirt from a dress or a sneaker from a sandal any more. In order to address that issue, you choose to train a Convolutional Neural Network (using PyTorch) that will help you identify each cloth to match the perfect design you created. In order to train it, you decide to rely on the dataset fashion MNIST (<https://github.com/zalandoresearch/fashion-mnist>).

You can access the data using the following lines (we strongly advise you to copy this code from the `fashion_mnist.py` file attached to this coursework):

```
import torchvision
import torchvision.transforms as transforms
import torch
import torch.nn as nn
import matplotlib.pyplot as plt

train_set = torchvision.datasets.FashionMNIST(root = ".", train=True,
download=True, transform=transforms.ToTensor())
test_set = torchvision.datasets.FashionMNIST(root = ".", train=False,
download=True, transform=transforms.ToTensor())
train_loader = torch.utils.data.DataLoader(train_set, batch_size=32, shuffle=True)
test_loader = torch.utils.data.DataLoader(test_set, batch_size=32, shuffle=False)
# Fix the seed to be able to get the same randomness across runs and
# hence reproducible outcomes
torch.manual_seed(0)

# If you are using CuDNN, otherwise you can just ignore
torch.cuda.manual_seed(0)
torch.backends.cudnn.deterministic=True
torch.backends.cudnn.benchmark=False
```

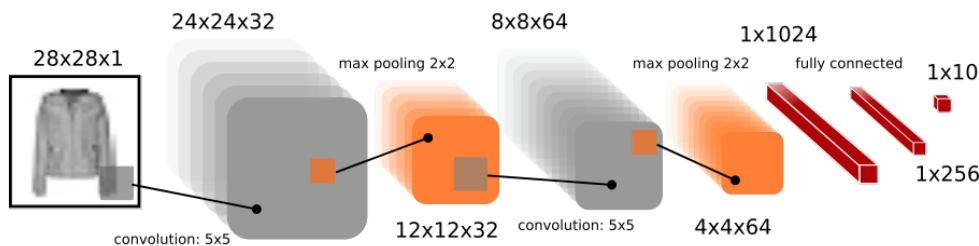
1. Given the problem, what is the most appropriate loss function to use? Provide the name of the loss, its formula and the formula interpretation in your report.

This question carries 10% of the mark for this coursework.

2. Create and train a Convolutional Neural Network corresponding to the following architecture (also see the diagram below):

1. Input image size: $28 \times 28 \times 1$ (height \times width \times number of channels).
2. First convolutional layer: Kernel size (5×5), Stride size (1×1) and 32 output channels. Activation function.
3. Max pooling layer: Kernel size (2×2) and Stride size (2×2).
4. Second convolutional layer: Kernel size (5×5), Stride size (1×1) and 64 output channels. Activation function.
5. Max pooling layer: Kernel size (2×2) and Stride size (2×2).

6. First fully-connected layer with input size being the output size of max pooling layer in 5. (flattened, i.e. 1024) and output size 1024. Activation function.
7. Second fully-connected layer with input size being the output size of fully connected layer in 6. (i.e. 1024) and output size 256. Activation function.
8. Output layer with input size being the output size of fully-connected layer in 7. (i.e. 256) and output size 10.



For training, initialise your weights using the Xavier Uniform initialisation, use ReLU as the activation function, a learning rate of 0.1 with the SGD optimiser. You will train your neural network for 30 epochs. In your report, provide the following: (a) final (train and test) accuracy obtained; (b) plot of the accuracy on the training and test sets per each epoch, comment on the speed of performance changes across epochs; (c) plot of the train loss per epoch (total sum of per batch losses for each epoch) and comment on the speed of decrease.

This question carries 15% of the mark for this coursework.

3. Run three experiments each time changing all the current activation functions to one of the following: Tanh, Sigmoid and ELU. In your report, provide only the final classification accuracy values (train and test) per activation function and comment on the result.

This question carries 6% of the mark for this coursework.

4. Keeping ReLU, use 5 different learning rates: 0.001, 0.1, 0.5, 1, 10. In your report, provide the final train loss, as well as the final accuracy values for both train and test for each learning rate and comment on the trade-offs between speed and stability of convergence. Comment on why you get the Nan loss if any.

This question carries 9% of the mark for this coursework.

5. Add a dropout of 0.3 rate on the second fully connected layer (keeping ReLU and learning rate 0.1). In your report, provide the final train and test accuracy values and explain how the dropout affects the performance.

This question carries 10% of the mark for this coursework.

4 Submission

Submit your coursework to QMPlus as a single zip file (filename ending in <yourStudentNumber>.zip) containing:

- Report (a single PDF format for both parts, no more than 5 pages of 10 point). The PDF can include scans of handwritten material as long as it is legible, but a typewritten version is encouraged.
- *Commented* source code in Python ≥ 3.7 (extension .py, Jupiter Notebook file with the extension .ipynb). Comments are important as they indicate your understanding of the code. The source code must contain a readme.txt file that includes instructions on how to run your code.

Note that a human will be reading the code, and cases of plagiarism will be reported. Strategy, algorithms, originality, code quality, and report quality will be also assessed.

IMPORTANT RE USE OF AI:

While the use of Generative AI (ChatGPT, Github Co-pilot, etc) to produce text or code for your coursework is NOT prohibited, it MUST be acknowledged. A detailed explanation of how you used those tools while preparing your coursework does not count towards the page limit. If you insert AI-generated text or code in your report/code, you must include a share link or name the AI system. Additionally, you must specify the prompt(s) used to produce the content and how you critically analysed it (including the analysis of AI-generated mistakes and your subsequent refinements to the prompts).

Failing to acknowledge and describe the use of AI-generated code and text will mean that mistakes and issues in this code and text will be considered as your own. If you fail to acknowledge the use of Generative AI and thoroughly evaluate its results, you can not contest the grade given on the grounds that the use of Generative AI is not prohibited, and its outcomes can sometimes be fallible.

Note that cases of identical (or similar) code and/or text to your peers resulting from the usage of Generative AI (without acknowledgement or critical analysis of this usage) will be considered plagiarism and will be reported.

When adding citations, note that fraudulent reporting of source material is considered academic misconduct (see the latest academic misconduct policy): for example, if you provided an inaccurate summary of an existing paper written by generative AI, that would be considered fraudulent reporting of source material, which in turn qualifies as academic misconduct.