Biologically-Inspired Computation (F20BC/F21BC) Coursework

This assessment is worth 40% of your overall course mark

Due: 3:30pm, Friday 24th November 2023 [To be confirmed]

Overview

This assessment aims to increase your understanding of both artificial neural networks (ANNs) and particle swarm optimisation (PSO), two biologically-inspired techniques which are taught in the course. It involves implementing both ANNs and PSO from scratch, and experimentally investigating how PSO can be used to optimise an ANN to carry out a specified task.

Please read through the following important points before you begin:

- In order to encourage discussion, **the assessment involves working in pairs**, i.e. 2 people. You need to choose a partner from the same level (i.e. an F20BC student cannot be paired with an F21BC student). You should report your team choice on Canvas by **Friday 29th September**:
  - For Edinburgh students, go to Edinburgh Information → Edinburgh Team Formation
  - For Dubai students, go to Dubai Information → Dubai Team Formation
- Each member of a pair should contribute equally. We will be asking you to summarise your contributions when you submit each stage of the work, and we will rebalance the marks within a pair in cases where one member contributes substantially more than the other.
- **Do not copy code from the internet or from other students**. This is very easy for us to detect, and every year we have to report students to the academic misconduct board for plagiarism. This involves a lot of work for us, and it very stressful for students to go through this process, so please *please* don't submit code that isn't your own without acknowledging its source!

This is assessed coursework. You are allowed to discuss this assignment with students outside of your pair, but you should not copy their work, and you should not share your own work with other students. We will be carrying out automated plagiarism checks on both code and text submissions.

Special note for using existing code or large language models (e.g. ChatGPT). If you are using code that you have not yourself written, then this must <u>clearly</u> be indicated, making clear which parts were not written by you and clearly stating where it came from. If your code is found elsewhere by the person marking your work, and you have not mentioned this, you may find yourself having to go before a disciplinary committee and possibly face grave consequences.

Late submission and extensions. Late submissions will be marked according to the university's late submissions policy, i.e. a 30% deduction if submitted within 5 working days of the deadline, and a mark of 0% after that. The deadline for this work is not negotiable. If you are unable to complete the assignment by the deadline due to circumstances beyond your control (e.g. illness or family bereavement), you should complete and submit a mitigating circumstances application: https://www.hw.ac.uk/students/studies/examinations/mitigating-circumstances.htm

What you are asked to do:

1. Implement a multi-layer ANN architecture
2. Implement a PSO algorithm
3. Using PSO, train the ANN to fit a specified dataset
4. Experimentally investigate the effect of ANN and PSO hyperparameters
5. Write a short report and submit both the report and your code to Canvas
6. Include the “Coursework Group Signing Sheet” and “Declaration of Authorship”

These tasks are described in more detail below. Implementation should be done using a language of your choice (e.g. Java, Python, Matlab, C, C++). The aim is for you to learn how to implement biologically-inspired approaches from scratch, so you should **not** use existing ANN or PSO libraries.

You are strongly encouraged to start work on Step 1 of the assessment once ANNs have been covered in lectures. At that point, you can also write the evaluation code for Step 3, and can test this code on ANNs with random weights. PSO will not be covered in lectures until week 7, though if you are eager to get started on this part, the introductory lecture on PSO is already available under Week 7 on Canvas.

1. Implement a multi-layer ANN architecture

You should implement a simple feedforward multilayer architecture. Both the number of neurons in each layer and the number of layers should be configurable. You do not need to implement any classical training algorithms, e.g. backpropagation, since you will be using PSO to do the training. Here is a list of activation functions that should be implemented, though you may also investigate other suitable functions:

- Logistic function: $\frac{1}{1+e^{-x}}$
- ReLU (rectified linear unit): $\max(0, x)$
- Hyperbolic tangent: $\tanh x$

2. Implement a PSO algorithm

You should implement a version of Particle Swarm Optimisation (PSO) that uses informants, i.e. each particle should be influenced by a subset of the other particles in the swarm, rather than just the swarm best.

Here are some points to bear in mind:

- How you allocate informants is up to you; for instance, they could be randomly allocated at initialisation. You might want to compare different strategies for allocating informants.
- You might find the pseudocode in the book “Essentials of Metaheuristics” useful as a starting point: https://cs.gmu.edu/~sean/book/metaheuristics/

There are lots of variants of PSO. Whilst you are not expected to be aware of all of these, you are encouraged to read about PSO and experiment with different approaches. You will get more marks if you investigate multiple approaches or go beyond a basic version of PSO.

3. Using PSO, train the ANN to fit a specified dataset

PSO is an optimisation algorithm that optimises a vector of values. In this case, this vector of values represents the set of parameters (i.e. weights and biases) of an ANN, and the goal is to find values for each of these parameters so that the ANN correctly solves a specific problem (see below for details about this). So, your task is to use the PSO code you wrote to optimise the parameters of your ANN, i.e. you'll need to figure out how to couple together your PSO and ANN code. There will be some hints on how to do this in the in person lecture in Week 7.

Here are some points to bear in mind:

- Each particle within the PSO swarm represents an ANN as a fixed-length vector of floating-point values, each of which encodes the value of a particular parameter of the ANN.
- Each time a particle is evaluated in PSO, the values in its vector should be used to set the parameters of the ANN, and the ANN should then be evaluated in order to measure its accuracy on the given problem. This accuracy value then becomes the particle's fitness.
- Although there are versions of PSO that can handle variable-length vectors, you are not expected to know about these. Consequently, the architecture of the ANN (i.e. the number of layers and neurons) should be specified at the beginning of a PSO run and remain fixed.
- One advantage of using PSO, rather than backpropagation, is that you can optimise other aspects of the ANN in addition to the weights and biases. For instance, you are encouraged to also try encoding the activation functions used by the ANN within the PSO solution vector, and extra marks will be available if you do this.

The problem domain for this work is binary classification. Your task is to build an ANN that fits the UCI banknote authentication dataset. You can download the dataset from the following link:
https://archive.ics.uci.edu/ml/machine-learning-databases/00267/data_banknote_authentication.txt

The dataset is provided in CSV format. It comprises 1372 records, each with 4 features (i.e. inputs to the ANN) and a classification (i.e. the desired output class, represented as either a 0 or 1 in this case). You can find more information here:
https://archive.ics.uci.edu/ml/datasets/banknote+authentication#

Note: *Since not all students are taking the data mining and machine learning course, we do not require you to consider overfitting and thus you don't need to split the data into training and testing sets. You can just fit the ANN with the entire dataset and record the classification accuracy. Of course, if you have come across these concepts, you are welcome to go the extra mile by splitting the data and investigating overfitting of ANNs.*

To evaluate how well an ANN fits the dataset, you should use classification accuracy; that is, the number of correctly classified data samples divided by the total number of data samples.

4. Experimentally investigate the effect of ANN and PSO hyperparameters

As discussed in the lectures, both ANNs and PSO have numerous hyperparameters. For ANNs, this includes the number of layers, the number of neurons in a layer, and the activation function used in each layer. For PSO, it includes the values of the acceleration coefficients, the number of informants, the swarm size and the number of iterations. The aim of this experimental investigation is to get some insight into how these values effect the ability of PSO to optimise an ANN than correctly solves the specified problem.

Here are some points to bear in mind:

- You can begin by informally trying out different values for the various hyperparameters and getting a feel for how much they affect the results.

- Pick some hyperparameters which you think have a significant effect and then carry out a more formal experimental investigation.
- Pick a sensible range of values for each hyperparameter that you investigate. This could be guided by values you find in lectures, books, published papers etc. You can also use a more systematic approach to exploring the hyperparameter space, such as random search or grid search.
- PSO is a stochastic algorithm. That is, each time you run it, the particles start in different positions and how they move has a random element. This means that, for the same hyperparameter values, you will likely get different results each time you run it. To address this, it is strongly recommended that you repeat each run at least 10 times and consider the average and standard deviation across runs, rather than the results of individual runs.

5. Write a short report and submit both the report and your code to Canvas

Your report should:

- Be no more than **<u>6 pages</u>** in length (max of 3000 words), excluding references and appendices. You should take this into account when planning your experiments. If you have more results than you have space for, then pick the results that you think are most insightful and briefly mention which other experiments you carried out.
- Be written in Arial, or a similar font, with a minimum **font size of 12**.

It should contain the following sections:

- **Implementation**: Briefly describe your implementations of ANN and PSO. Note any interesting aspects, and describe the motivation behind any design decisions.
- **Experimental investigation**: Describe how you approached the investigation. State which hyperparameters you chose to investigate, and why you chose to focus on these in particular. Indicate which value ranges you looked at, and why you chose these ranges. You can use references to the literature to motivate your approach.
- **Results**: Using tables and plots, show your results. When carrying out multiple runs to compensate for stochasticity (see above), tables should show averages (and ideally standard deviations) rather than the results of individual runs. Plots are useful for illustrating trends that can be hard to spot in tables alone.
- **Discussion and Conclusions**: What did you discover? How well did your ANNs solve the problem, and how did the hyperparameter settings affect this? Don't just describe what you saw; also try to explain why this is the case, e.g. why did a particular hyperparameter have a particular effect? You can use references to the literature to support your arguments.
- **References**: These should be in a standard format, e.g. Harvard style, and should be cited in the earlier sections. Don't include references that you don't use.
- **Appendices (optional)**: You can include additional results here, if you want to. However, your marker won't look at this section in any detail, so anything important and key to your discussion should appear in the earlier sections.

You should submit both your report (as a **pdf** file) and your code (as a **zip** file) to Canvas using the links provided. Grading will use the assessment criteria given in the tables in Section 7.

6. Include the "Coursework Group Signing Sheet" and "Declaration of Authorship"

The Coursework Group Signing Sheet is available on Canvas. Both members of your team should outline their contribution to the coursework, indicating which parts of the code and report they contributed to, and sign the sheet. You should also fill in the university's standard declaration of authorship form. Both should then either be embedded in your report, or included in the code submission zip file. **Note that no marks will be issued until these have been submitted.**

7. Marking Scheme

See the table below for the marking scheme. Note that the weightings are slightly different for F20BC and F21BC, and reflect a higher expectation in terms of interpretation and analysis for the latter. The grade requirements outlined in the table are indicative only, and we'll take into account your achievements across the assessment when deciding a mark. This means, for instance, if you choose to put a lot of extra effort into investigating different variants of PSO, or optimising elements of the ANN beyond just the weights, then we wouldn't expect quite as much work in the experimental part. Conversely, if your code is quite basic, then you can compensate for this by putting a lot more work into the experimental investigation and by making greater use of the literature in your discussion.

Criteria	Weight	A (70-100%)	B (60-69%)	C (50-59%)	D (40-49%)	E/F (<40%)
Implementation ***(i.e. code, comments, description and motivation in the report)***	50% for F20BC 45% for F21BC	Creative implementations of ANN and PSO that exceed the basic requirements. ANN and PSO are correctly coupled together. Correct evaluation code. Easy to read and well structured.	Correct implementations of the basic requirements. ANN and PSO are correctly coupled together. Generally good coding, structure and documentation.	Some significant issues in terms of correctness, structure, coding practice (including lack of comments) and documentation. ANN and PSO are correctly coupled together.	Major issues in terms of correctness, structure, coding practice and documentation and/or ANN and PSO are not correctly coupled together.	Critical errors: for example, the code does not compile and/or run, or inappropriate algorithms have been implemented.
Experimental study ***(i.e. choice and validity of experiments performed, presentation of results)***	20%	Hyperparameters investigated are well motivated and their values are well chosen. Suitable results have been collected and are clearly presented and meaningful.	Some minor issues in terms of the motivation or description of hyperparameters, the experiments performed, or the presentation of results.	Some significant issues in terms of the motivation or description of hyperparameters, the experiments performed, or the presentation of results.	Some major issues: experiments do not make sense, have invalid results, or the study is not adequately described.	Some critical issues: experimental study is nonsensical or missing, inappropriate experiments, or the description of the study is uninformative.
Discussion ***(i.e. results interpretation, understanding demonstrated, conclusions, use of the wider literature or internet resources)***	20% for F20BC 25% for F21BC	Clear, insightful discussion that shows a good understanding of ANNs and PSO and includes well chosen references to the wider literature or internet resources.	Generally clear and insightful, but shows some misunderstanding of ANNs or PSO. Adequate use of the wider literature or internet resources.	The discussion is limited in terms of the depth or volume of understanding it demonstrates. Little or no use of the wider literature or internet resources.	Some major issues in terms of depth or volume of understanding. No use of the wider literature or internet resources.	No real demonstration that the subject matter has been understood, or very limited in its scope.
Report ***(i.e. structure, language, referencing etc.)***	10%	Report is well structured and correctly divided into sections; readable, with good use of language; respects page limit and formatting guidelines	Report is suitably structured and divided into sections; mostly readable with good use of language; respects page limit and formatting guidelines	Report is structured but not divided into sections; language issues that affect readability; notable formatting issues	Report is poorly structured; poor readability; significant formatting issues	Report has a nonsensical structure; very hard to read; problematic formatting