CSC4025Z (2025): Artificial Intelligence

Assignment 2: Application of Neural Networks

Lecturer: Dr. Jan Buys (jbuys@cs.uct.ac.za)
TA: Claytone Sikasote (SKSCLA001@myuct.ac.za)

Deadline (report and code): Monday 27 October 2025 at 23:59. Assignment **presentations** will also take place on Monday 27 October.

You should work in groups of **2 to 3** students. For any questions related to the project please e-mail **BOTH** the lecturer and the TA. For any clarifications please contact us as early as possible.

The goal of the project is to develop a neural network-based AI system and to evaluate its performance.

The system should be implemented in a numerical computation or neural network library: PyTorch is recommended, but you may use Tensorflow, Keras or other appropriate libraries. You may use an existing codebase as starting point for your implementation (with appropriate acknowledgement), as long as you train, validate and test the model yourself. For this assignment you should **not** be using or fine-tuning pretrained neural networks.

As a platform to train and run models, including to train neural networks, Google Colab or Kaggle Notebooks is recommended. Most neural networks train more efficiently on GPUs, which can be done on Colab through cloud GPUs. Note that there are time restrictions on training time on the free version of these online services, so if you wait until too close to the deadline you may not have enough time to complete training.

The project will consist of the following steps:

1. Choose a dataset to work with. You may use any publicly available dataset that Is appropriate for supervised classification. One good source of datasets is Kaggle.
2. Formulate the machine learning problem, i.e. the prediction task. You should formulate the task as a *multi-class classification* problem that predicts some attribute in the dataset. You may aggregate a range of discrete or continuous values in the data into a single class if appropriate. There should be more than two output classes. Decide what the inputs and output of the model should be, and split the data input training, validation and test sets. Choose an appropriate evaluation metric. You should also consider if there are any *ethical issues* associated with building an AI system for this task.
3. Pick a baseline. This should be the simplest possible approach to the problem. Options for a baseline include using count-based probability estimates, Naïve Bayes, or K-nearest neighbours.
4. Develop the model: Decide which type of neural network architecture and which features or input representation to use. Implement and train the model. Tune hyperparameters and evaluate different choices of features or model architectures. Perform the final evaluation.
5. Analyse the model's performance. If the model has a low accuracy, that could be a perfectly valid outcome and you won't be penalized for that by itself. What is more important is that you show evidence that you carefully considered the various modelling choices, optimized the model on the chosen dataset, evaluated the model appropriately and analysed what its shortcomings might be.

The project has the following deliverables:

- **Project report**: The suggested length is 5 pages (this is a guideline, not a hard constraint). The report should be a comprehensive description of your work: Address everything mentioned above. You need to include a problem description, explain the baseline, model design choices, experimental setup, and results on both the validation set (for tuning) and the test set (for final evaluation), and an analysis of your model's performance.
- **Code and data**: Submit the code for the project. If you use external code as part of the project that should be indicated clearly. Submit the *test data*, as well as your model's final predictions on the test set. If the test data is >10MB you may include only a sample. Include instructions or a script for reproducing your results.
- **Final presentation**: Each group will give a 5-minute class presentation on their work. The presentation should cover the problem formulation, model architecture, and evaluation and analysis of model performance. The time limit will be enforced strictly.

The group leader should submit the assignment as a single zip file on Amathuba, including the report, code, data, and final presentation slides. Include the students numbers of all group members in the file name. The standard penalty of 10% for a submission that is 1 day late will apply.

Marking rubric

Criteria	**Mark**
Problem formulation: Formulate problem as a classification task, discuss usefulness and ethics	3
Baseline: Include appropriate baseline	3
Model design: Choose appropriate model architecture and input representations or features	3
Model validation: Evaluate different hyperparameter choices, evidence of optimizing model performance	3
Evaluation: Use appropriate evaluation metric(s) and training/validation/test split, report results	3
Analysis of model performance	3
Software: Code implements model, appropriate choice of tools	3
Reproducibility: Include test results, hyperparameters and instructions or script to reproduce results	3
Overall report quality	3
Final presentation quality	3
Total	30