Project 1

CS 4033/5033 Machine Learning Fundamentals, Spring 2026

Due: Tuesday, February 24, 11:59 pm

---

1. Motivation

Artificial neural networks (ANNs) are one of the most prominent, best understood, and most widely used types of machine learning (ML). Within ANNs, feedforward neural networks (FFNNs) trained using supervised learning via error backpropagation (backprop) are one of the most prominent, best understood, and most widely used types. Therefore, knowing how to design and train backprop nets for various problems are fundamental skills for ML practitioners and understanding the functioning of these ANNs is a fundamental skill for ML researchers. Moreover, backprop nets are quite often used for pattern classification and function approximation. In this programming assignment, as with Homework 1, you will focus on classification.

2. Goal

The goals of this assignment are:

- to give you baseline experience with implementing an ANN,
- to give you experience with backprop learning systems for FFNNs in particular,
- to familiarize you with some of the issues facing ANN implementors,
- to give you experience with the supervised learning process.

3. Assignment Overview and Data

You will design, program, and run a simple backprop net to learn several two-class classification problems consisting of simple synthetic data found in the provided comma-separated value (csv) files, as follows.

1. Gaussian

The first set of classification problems involves data sampled from Gaussian distributions in two and three dimensions. Within each dimension, there are three variants:

1. Wide separation between the classes (denoted “Wide” in the dataset title).
2. Narrow separation between the classes (denoted “Narrow” in the dataset title).
3. Overlap between the classes (denoted “Overlap” in the dataset title).

1. Two Dimensions

Each dataset consists of 100 data items for Class 0 and 100 data items for Class 1. The first two columns in each file give the first and second dimension (that is x_1 and x_2) coordinates of the 100 data items in Class 0, while the third and fourth columns in each file give the first and second dimension (that is x_1 and x_2) coordinates of the 100 data items in Class 1. The three datasets for Gaussian two-dimensional data are:

1. Gaussian 2D Wide.csv
2. Gaussian 2D Narrow.csv
3. Gaussian 2D Overlap.csv

2. Three Dimensions

Each dataset consists of 100 data items for Class 0 and 100 data items for Class 1. The first three columns in each file give the first, second, and third dimension (that is x_1, x_2 and x_3) coordinates of the 100 data items in Class 0, while the fourth, fifth, and sixth columns in each file give the first, second, and third dimension (that is x_1, x_2 and x_3) coordinates of the 100 data items in Class 1. The three datasets for Gaussian two-dimensional data are:

1. Gaussian 3D Wide.csv
2. Gaussian 3D Narrow.csv
3. Gaussian 3D Overlap.csv

2. Crescent Moons

The second set of classification problems involves data sampled from semicircular distributions that vaguely resemble crescent moons. For this set of classification problems, the data is only provided in two dimensions. As with the first set of classification problems, there are three variants:

1. Wide separation between the classes (denoted “Wide” in the dataset title).
2. Narrow separation between the classes (denoted “Narrow” in the dataset title).
3. Overlap between the classes (denoted “Overlap” in the dataset title).

Each dataset consists of 500 data items for Class 0 and 500 data items for Class 1. The first two columns in each file give the first and second dimension (that is x_1 and x_2) coordinates of the 500 data items in Class 0, while the third and fourth columns in each file give the first and second dimension (that is x_1 and x_2) coordinates of the 500 data items in Class 1. The three datasets for semicircular data are:

1. Moons 2D Wide.csv
2. Moons 2D Narrow.csv
3. Moons 2D Overlap.csv

4. Assignment Overview and Data

Carry out the following steps. Underlined steps require a written response, those in **code** require you to write software, and those in *italics* require you to collect data. Written responses, **code**, and *data* will be turned in for grading. Note that the steps below discuss an "ANN learning system" in the singular, although you may decide to have different ANNs to handle different datasets.

1. Consider the choices one needs to make regarding the design and implementation of any ANN learning system.
2. List the choices that need to be made when designing an ANN learning system that have already been made for you in this assignment. You should be able to list at least four.
3. For each of these choices, list which option I chose for you in making this assignment.
4. List the choices you need to make regarding the design of your ANN. (Note, these do not include purely implementation choices such as programming language.) You should be able to list at least four.
5. Choose an option that seems reasonable to you for each of these design choices and explain why it seems reasonable to you. If you do not have a good reason for your chosen option, say so. Note that you may want to choose different options for your networks for each of the functions you are to approximate.
6. **`Implement your ANN.`**
7. Now that you have implemented your ANN, you are likely to have recognized more choices that you needed to make along the way. List the choices you needed to make regarding the design of your ANN. Including the choices you listed previously, your list should now contain at least eight choices.
8. For each of these design choices, list the option you chose, and explain why it seems reasonable to you. If you do not have a good reason for your chosen option, say so.
9. Answer the following questions about data collection, reporting, and conclusions so that you are ensured of collecting the appropriate data. Attempt to justify your answers to these questions. As you do so, keep in mind that what you want is performance data that allows you to understand the workings of your ANN as a classification tool and you want to report the minimum amount that allows your reader to thoroughly understand what you have learned. Note that you may lack a justification for your answers to some of these questions at this time. That is acceptable since this is your first programming assignment in this course. However, you should keep all of these questions in your mind as the course progresses and be able to give good, justified answers to similar questions for later assignments.
    - How many data samples will you use to train your ANN? To validate it? To test it?

- How many times will you run (train, validate, and test) your ANN in order to be able to give an accurate measure of its performance on these functions? Once? Ten times? Twenty times? 100 times?
- If you run your ANN more than once, what will you change from run to run? What will you keep the same?
- What performance data will you collect?
- What performance data will you report?
- How will you report this data? Text? Numbers? Graphs? What form will these take?
- What conclusions will you be able to draw from your results?

10. *Run your ANN on the first dataset (Gaussian 2D Wide).*
11. Report your results and conclusions regarding the application of your ANN to this dataset.
12. *Run your ANN on the second dataset (Gaussian 2D Narrow).*
13. Report your results and conclusions regarding the application of your ANN to this dataset.
14. *Run your ANN on the third dataset (Gaussian 2D Overlap).*
15. Report your results and conclusions regarding the application of your ANN to this dataset.
16. *Run your ANN on the fourth dataset (Gaussian 3D Wide).*
17. Report your results and conclusions regarding the application of your ANN to this dataset.
18. *Run your ANN on the fifth dataset (Gaussian 3D Narrow).*
19. Report your results and conclusions regarding the application of your ANN to this dataset.
20. *Run your ANN on the sixth dataset (Gaussian 3D Overlap).*
21. Report your results and conclusions regarding the application of your ANN to this dataset.
22. *Run your ANN on the seventh dataset (Moons 2D Wide).*
23. Report your results and conclusions regarding the application of your ANN to this dataset.
24. *Run your ANN on the eighth dataset (Moons 2D Narrow).*

25. Report your results and conclusions regarding the application of your ANN to this dataset.
26. *Run your ANN on the nineth dataset (Moons 2D Overlap).*
27. Report your results and conclusions regarding the application of your ANN to this dataset.

5. What to Turn In

1. Write Up

You will turn in an electronic copy of your write up. Your write-up should be a coherent document that covers all of the underlined steps from the assignment above. Note that selected data in a digested form (such as tables or graphs) should be included in your writeup; however, your raw data should not be included here.

2. Code

You will turn in an electronic copy of your code. You will turn in the source code you have written for this ANN. Your source code should be well structured and well commented. It should conform to good coding standards (e.g., no memory leaks).

3. Data

You will turn in an electronic copy of your data. This may be in a single file or multiple files. You will also need to include a brief writeup on how the data is organized.

6. Notes on this Assignment

Be sure to use proper terminology (e.g., epoch, repetition) when describing your choices.

You may write your program from scratch or may start from programs for which the source code is freely available (such as on the web or from friends or student organizations). If you do not start from scratch, you **must** give a complete and accurate accounting of where all of your code came from and indicate which parts are original or changed, and which you got from which other source.

As an alternative to writing your own code, either from scratch or building on existing code, you may complete this assignment by using one of the many ANN packages available these days. However, it has to be something freely available and needs to allow the configuration you've used to be exported, so that our TA and I can download it and try out the configuration you've submitted. In addition, if you use a third-party package, you **must** graph the decision regions learned by your ANN.

For the written components of this assignment you may follow the format or content of other written works but you **must** give a complete and accurate accounting of who deserves credit for all parts of your documents.

Failure to give credit where credit is due is academic fraud and will be dealt with accordingly. See the University's web pages on academic integrity.