**CS2400 Lab7: Tic-Tac Neural Network v2 Spring 2020**

Overview: In this lab you will be building your own neural network to train a winning Tic-Tac-Toe board recognizer. You are not allowed to use any neural network or machine learning libraries to implement this. Instead you will start with your basic neural network implementation from last week, and you will modify it according to the instructions and goals below.

NOTE: This is a pair-programming lab.

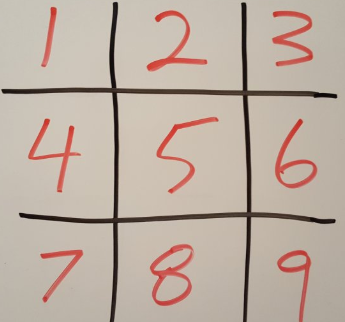
Learning Outcomes:

* Knowledge representation for capturing new network functionality
* Modification of a neural network structure
* Interpretation of outputs from a neural network
* Presentation of results

Instructions:

You are given an input file that contains training data for a new tic-tac-toe neural network (tic-tac-toeWBlanks.csv). Each row of this file contains a “winning” configuration of a tic-tac-toe board as well as which letter (x or o) won. This data is different than last week’s because it includes “blanks” (b).

Reminder: The format of the file is as follows: Columns 1-9 correspond to the position on the tic-tac-toe board below. The values in the column represent whether an “X”, “O”, or “blank” occupies the space. The last column represents whether X wins (Xwin) or Y wins (Ywin).



Your first task for this lab is to plan a way to modify the network structure to accommodate this new information. Work with your partner to propose a structure. Once you have proposed a way to adjust BOTH your network and your input strategy, explain your plan to the instructor before moving forward to avoid implementing something that will waste your time.

Next, you should work in your pairs to implement your plan. Make sure you document your data set, accuracy measurements, robustness calculations (validation, training split), etc.

If you are not able to get your plan to work, you are expected to adjust it accordingly and try again as many times as you need to get it working. Make sure you document not only what your original plan was, but why you think it didn’t work and what you ultimately did to get it to work.

You should document your strategy, network structure, training approach, as well as the results of your training in a lab report. Your lab report must clearly show the results of the training accuracy as well as some example test cases where it mis-classified the outputs. Please include reaction statements regarding the accuracy of your trained network compared with the week 6 lab that didn’t include blanks.

Grades:

|  |  |
| --- | --- |
| Code/Lab Report Style | 20% |
| Checkpoint | 10% |
| Implementation | 70% |

Due date:

5/5/2020 11:59PM

Submission Instructions:

Submit both your lab report as well as your nn.py method for this lab on Blackboard.