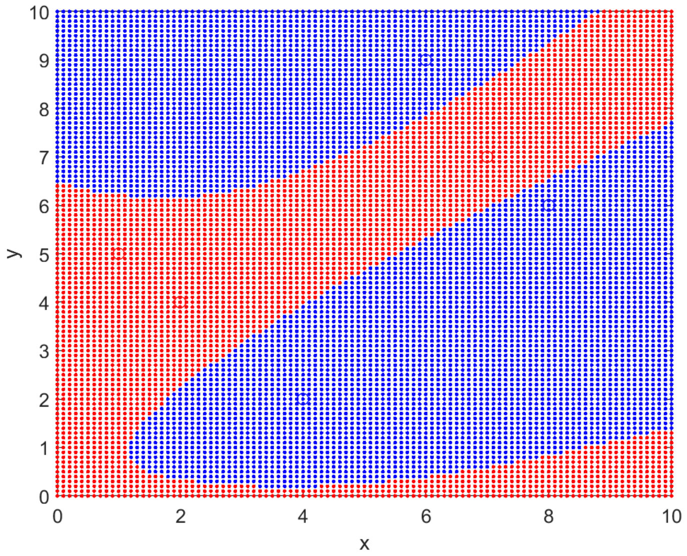
MTRE4300 Machine Learning for Robot Perception

Project #3

Due by 11:59 pm on 02/24/21 (Wednesday)

In this project, you are required to develop a Python program with the keras and tensorflow packages to implement a neural network to classify the data points in a 2D region (0<x<10 and 0<y<10) into two classes (red or blue) , as shown below:



In particular, the training samples are below:

(1) The coordinates of the red dots: (1, 5); (2, 4); (7, 7); (4,6); (6,4)
(2) The coordinate of the blue dots: (6, 9); (4, 2); (8, 6); (5, 5); (3, 8)

The project requirements are below:

1. It is a group project. In the first line of the Python code, use a comment line to show all group members' names.
2. Your neural network must have more than 2 hidden layers.
3. Your program needs to display the history of the training loss on the screen when the training is over.
4. After training, your program uses the trained model to predict the class of all data points within the region and plot the decision boundary picture similar to the picture above.
5. Hint: Since this is a classification project, you may need to set your deep learning model as shown below:
   model.compile(optimizer='rmsprop', loss='categorical_crossentropy',
   metrics=['accuracy'])

6. Each group saves your Python code as “classification_region.py”, and uploads it to the D2L drop box. You are not required to submit a report for this project.

Grading Rubrics
10 points: The Python code was submitted correctly.
30 points: The code runs without any syntax errors.
10 points: The code correctly displays the history of the training loss on the screen.
10 points: The final training loss is less than 5%.
20 points: The classification result (decision boundary) is shown as a picture similar to the one above.
10 points: Your neural network has more than 2 hidden layers.
10 points: No obvious classification error (i.e., No training sample is misclassified based on the decision boundary picture).