

Assignment for LAB 2

Note : Please include all pictures and code.

Assignment due: 10/20/2014, Total Point : 20 points (6+6+4+4)

For questions 2 and 3 please do feel free to work together. Questions 1 and 4 must be done alone.

Question 1)

General premise:

Make use of XOR.py code for the following assignment:

Write and, or, not, and nor Boolean operators similar to the XOR operator demonstrated in class

Part a)

Make use of the same code as provided in class and simply change the sample data as given. Does the neural network always get the correct answer? Write a short paragraph explaining how you might improve this.

Part b)

Tune the neural network by playing with the number of hidden layers and the number of hidden nodes in each layer. What improved the effectiveness of your neural network. Provided examples of parameter sets that worked well and parameter sets that did not. These should be python files. Also you should include a write up, with the code merely as a reference for the grader.

Part c)

Write boolean functions for composite boolean operators:

1)

(A AND B) OR C

2)

(NOT (A OR B)) AND C

3)

NOT ((A OR B) AND C)

First work with the original code as provided in class, simply changing the data set. Then tune the code. Was anything different? Did the configuration change or did the "tuning" in part b effectively work for all of the different boolean operators? If so, this would imply there is a boolean function tuning. Does such a tuning exist?

Question 2)

Premise:

Teach py brains to recognize your face.

Make use of https://github.com/EricSchles/neuralnet/blob/master/facial_recognition.py

Take 10 pictures of your own face. Then find 10 other faces, that are not your own and are human.

Make use of the code presented in class for doing facial recognition. You will need to process the images first. You can make use of cleaner.py to do this. However you will have to figure out how to make use of the different functions available to you.

Part a)

Did the tuning I provided in class work for your face? Why do you think this worked or didn't work? If you were not able to successfully use my tuning what did you have to change to get the neural network to work with your face?

Part b)

Now that you have tuned the code to work with your face, try the code when all of the pictures have your face, some of them should be the ten original faces and the other ten should be group photos. Note there should be at least five other faces in all group photos you use. Did the tuning you used in part (a) still work? What did you have to change to get the neural network to recognize your face in all of the pictures. Was it able to recognize your face EVEN with multiple faces?

Part c) - optional extra credit

Note: For this part you will need to work with opencv so you cannot use ubuntu.

You will need to make use of this code: <https://github.com/EricSchles/faceFind>

For this question you'll need to use faceFind to crop all the images in the group photos down to separate single faces (ideally in their own files). Then you'll have the neural network use the 10 original faces of you to predict which face the group picture is yours.

Question 3)

Find faces of giraffe's, gorillas, dogs, cats, and alligators. Use py brains to classify each of these pictures. Your code should be able to consistently tell which animal is which.

Part a)

Initially just use the code from class with the settings I provided. How well does this do? Are you able to identify each of the animals successfully? Your code should print out the following output:

General example:

Picture [name of picture goes here] has a [name of animal goes here]

explicit example:

Picture giraffe.png has a giraffe.

Part b)

Now tune your code so that every animal is guessed correctly, or as many as possible. (Note that a best effort is expected here, if your code does not do well here you will lose points).

Question 4)

Part a)

Now that you are well versed in playing with neural networks compare and contrast neural networks with linear regression and other classical statistics techniques we have covered so far in the course. What's different about them? What's the same? What kinds of information are different?

Part b)

Explain where you think machine learning is heading. Also, do you think classical statistics will always have a place in data science? Why or why not?