Machine Learning (BITS F464)

Assignment 2 (Artificial Neural Networks)

Submission Date: 2330Hrs on 10th Nov 2016

Max Marks: 50

Languages allowed:- C, C++, Java. You are not allowed to use any packages for the assignment, all the functionalities you have to code on your own and do proper documentation so that it is readable.

Introduction:- This assignment gives you an opportunity to apply neural network learning to the problem of face recognition. In this assignment you have to build a sunglasses recognizer, a face recognizer, and a pose recognizer.

Sunglass recognizer:- When given an image as input, indicates whether the face in the image is wearing sunglasses, or not.

Face recognizer:- Given a image you have to tell who is this person out of the 20 persons given in dataset.

Pose recognizer:- Given a image you have to tell the head position of the person, it can have 4 values- straight, left, right and up.

Before you start coding read this whole document carefully.

The Face Images:- The image data can be found here. This directory contains 20 subdirectories, one for each person, named by userid. Each of these directories contains several different face images of the same person.

You will be interested in the images with the following naming convention:

`<userid>_<pose>_<expression>_<eyes>_<scale>.pgm`

- `<userid>` is the user id of the person in the image, and this field has 20 values: an2i, at33, boland, bpm, ch4f, cheyer, choon, danieln, glickman, karyadi, kawamura, kk49, megak, mitchell, night, phoebe, saavik, steffi, sz24, and tammo.
- `<pose>` is the head position of the person, and this field has 4 values: straight, left, right, up.
- `<expression>` is the facial expression of the person, and this field has 4 values: neutral, happy, sad, angry.
- `<eyes>` is the eye state of the person, and this field has 2 values: open, sunglasses.
- `<scale>` is the scale of the image, and this field has 3 values: 1, 2, and 4. 1 indicates a full-resolution image (128 columns x 120 rows); 2 indicates a half-resolution image (64x60); 4 indicates a quarter-resolution image (32x30). For this assignment, you will be using the quarter-resolution images for experiments, to keep training time to a manageable level, this data-set can be found here.

From this link download the training and testing lists of images, if you want to train your network on complete set of images you are welcomed to do so but it will take a lot of time.

Sunglasses Recognizer:- For this task train and test your network on these following files and report the maximum accuracy reached by your model.

training file- `straightrnd_train.list.`

testing files- `straightrnd_test1.list`, and `straightrnd_test2.list.`

Face recognizer:- Implement a 1–of–20 face recognizer; i.e. implement a neural net that accepts an image as input, and outputs the userid of the person. To do this, you will need to implement a different output encoding (since you must now be able to distinguish among 20 people). (Hint: leave learning rate and momentum at 0.3, and use 20 hidden units). For training and testing use these files

training file- `straighteven_train.list.`

testing files- `straighteven_test1.list and straighteven_test2.list.`

Now let's take a closer look at which images the network may have failed to classify?

Do there seem to be any particular commonalities between the misclassified images?

Pose Recognizer:- Implement a neural net which, when given an image as input, indicates whether the person in the image is looking straight ahead, up, to the left, or to the right. You will also need to implement a different output encoding for this task. (Hint: leave learning rate and momentum at 0.3, and use 6 hidden units).

training file- `all_train.list.`

testing files– `all_test1.list and all_test2.list.`

How did you encode your outputs this time? What was the maximum classification accuracy achieved on the training set?

Report:

- Team Members.
- Network structure i.e. no. of layers, units in hidden layers, etc. for all the three networks.
- Maximum accuracy achieved on training and testing set.
- Specify all the parameters used for networks like learning rate, momentum, etc.

Evaluation:

- Final results
- Understanding of results.
- Ability to reason out the derived results.
- Final report and demo.

Submission should be through **CMS** only.

Contact the following Teaching Assistants for any clarification on this assignment.

- Rajitha <2015409@hyderabad.bits-pilani.ac.in >
- DEVENDRA SINGH SHEKHAWAT <f2013204@hyderabad.bits-pilani.ac.in>