Artificial Neural Network Model for Facial Recognition

AIM

To code Artificial Neural Networks to perform the following functions: -

- When given an image as input, indicates whether the face in the image is wearing sunglasses, or not (Sunglasses Recognizer)
- Given an image you have to tell who is this person out of the 20 persons given in dataset (Face Recognizer).
- Given an image you have to tell the head position of the person out of the 4 possible values- straight, left, right and up (Pose Recognizer).

TEAM MEMBERS

- G.V. Sandeep
- Snehal Rajesh Wadhwani
- Tanmaya Shekhar Dabral
- Kushagra Agrawal

TECHNIQUES USED

All the three problems involved using a Multi-layer Perceptron model of Artificial Neural Network. A Multi-layer Perceptron (or MLP) is a feedforward artificial neural network model that maps sets of input data onto a set of appropriate outputs. An MLP consists of multiple layers of nodes in a directed graph, with each layer fully connected to the next one. Except for the input nodes, each node is a neuron (or processing element) with a nonlinear activation function. MLP utilizes a supervised learning technique called backpropagation for training the network. Each of the three problems use MLP with varying number of neurons and use sigmoid activation function as it gives a value between 0 and 1 and classifies better than a linear function. The programming language used is Java. We have also visualized the weights by normalizing the weight values between 0 and 255. This gives us a further insight into the internal workings of the neural networks.

SUNGLASSES RECOGNIZER

NETWORK STRUCTURE

The Artificial Neural Network consisted of 3 layers i.e. Input, Hidden Layer and an Output node. The input node consisted of 960 neurons, the hidden layer consisted of 4 neurons and the output layer consisted of 1 neuron.

MAXIMUM ACCURACY ACHIEVED

```
199 iteration complete.
Percentage accuracy for training data is 100.0
Percentage accuracy for Test1 data is 100.0
Percentage accuracy for Test2 data is 100.0
Average sum of squares error for all training examples is 3.334393210486755E-4
```

PARAMETERS USED

- Learning rate=0.3
- Momentum=0.3
- Output high=0.9
- Output low=0.1

INTERPRETATION OF THE RESULTS

The below two examples are visualized weights of two of the hidden neurons.

Example 1-

The black patch near the center is close to the sunglasses

Example 2-

The black patch near the center is close to the sunglasses

FACE RECOGNIZER

NETWORK STRUCTURE

The Artificial Neural Network consisted of 3 layers i.e. Input, Hidden Layer and an Output node. The input node consisted of 960 neurons, the hidden layer consisted of 6 neurons and the output layer consisted of 4 neurons.

MAXIMUM ACCURACY ACHIEVED

```
199 iteration complete.
Percentage accuracy for training data is 100.0
Percentage accuracy for Test1 data is 97.22222222222221
Percentage accuracy for Test2 data is 95.0
Average sum of squares error for all training examples is 0.014621889077974145
```

PARAMETERS USED

- Learning rate=0.3
- Momentum=0.3
- Output high=0.9
- Output low=0.1

POSE RECOGNIZER

NETWORK STRUCTURE

The Artificial Neural Network consisted of 3 layers i.e. Input, Hidden Layer and an Output node. The input node consisted of 960 neurons, the hidden layer consisted of 20 neurons and the output layer consisted of 20 neurons.

MAXIMUM ACCURACY ACHIEVED

```
199 iteration complete.
Percentage accuracy for training data is 100.0
Percentage accuracy for Test1 data is 89.92805755395683
Percentage accuracy for Test2 data is 93.75
Average sum of squares error for all training examples is 0.002074606044962439
```

PARAMETERS USED

- Learning rate=0.3
- Momentum=0.3

- Output high=0.9
- Output low=0.1

OBSERVATIONS AND CONCLUSIONS

- To implement the Multi-layer Perceptron, we used matrices instead of traditional arrays. This made the code more modular, generic and easier to implement.
- For the Sunglasses Recognizer, we visualized the weights and a dark patch occurs near the sunglasses, hence showing that it recognized them correctly. The dark near glasses mean those weights have negative values. Therefore, if the pixel values are too high (bright) the node will be turned off.
- It is seen that most of the misclassified images have aberrations like improper lighting (too dark), improper zooming etc. The network is unable to generalize the experience to these aberrations.