Classification of CIFAR-100 Database

Shilpashree Rao, shilpasr@usc.edu, USC ID: 5636765972
CS561-Foundations of Artificial Intelligence
University of Southern California

Introduction

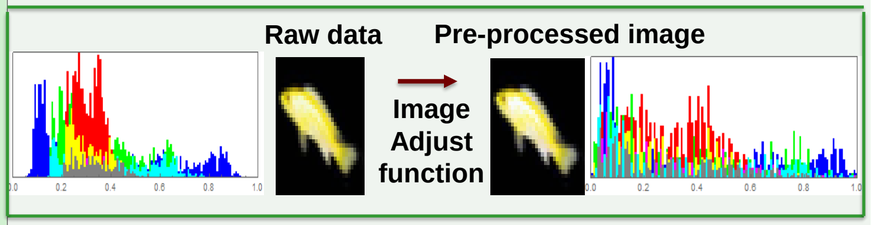
CIFAR-100 Dataset: This dataset consists of *60000 32x32* color images in 100 classes, with 600 images per class. There are *50000 training images* and *10000 test images*.

Problem statement and Implementation:

- Three inductive algorithms are used classify the 50,000 images into 100 possible categories
- *Holdout cross validation* is implemented where in the CIFAR-100 dataset is partitioned as the training and testing
- The learning system is implemented using *Feed Forward Neural Network, Random Forest Decision tree and k-Nearest Neighbors* algorithms here

Data Pre-Processing:

- The images from the dataset are pre-processed using *'Image Adjust'* function
- This function *adjusts the pixel levels* in image, rescaling them to cover the range 0 to 1, i.e., contrast of the image is stretched over *the range 0 to 255* pixels and *rescaled* to fall between *0 and 1*

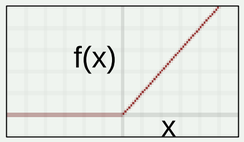


Classification Algorithms

Convolutional Neural Network: Multilayer Feed Forward Network

The experiments were performed on *LENET* and *Multilayer Feed-Forward Network*. The performance of latter was better, and thus has been chosen for implementation.

- The *hidden nodes* in the network, expands the scope of the possible functions the network can utilize to classify the data

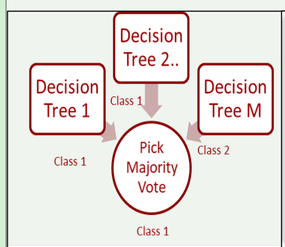


$$f(x) = \begin{cases} 0 \ for \ x < 0 \\ 1 \ for \geq 0 \end{cases}$$

Classifier information	
Method	Neural network
Number of classes	100
Number of features	1
Number of training examples	25 000
L1 regularization coefficient	0
L2 regularization coefficient	0.1
Number of hidden layers	2
Hidden nodes	382, 382
Hidden layer activation functions	Rectified Linear, Rectified Linear
CostFunction	Cost Function

Decision Tree Algorithm: Random Forest

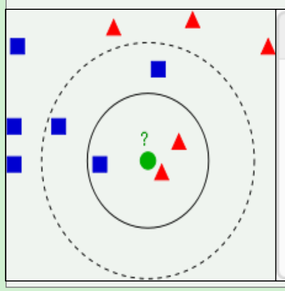
- This algorithm was chosen as the *depth of the tree* can be optimized to avoid over-fitting or fit noise in the data
- Decision Trees are *matrices of random features* collectively creating a forest



Classifier information	
Method	Random forest
Number of classes	100
Number of features	1
Number of training examples	25 000
Number of trees	50

Machine Learning Algorithm: k – Nearest Neighbors

- This algorithm was chosen as there is freedom to play around with the *Number of Neighbors* parameter for performance optimization
- The *Euclidean distance* between points p and q is the length of the line segment connecting them given by the distance formula



Classifier information	
Method	K–nearest neighbors
Number of classes	100
Number of features	1
Number of training examples	25 000
Number of neighbors	10
Distance function	EuclideanDistance

Experimental Results and Visualizations

Pre-processing	Classification type	Accuracy	Training, testing samples
Image Adjust	Multilayer Feed Forward Neural Network	62.3%	50000, 10000
Image Adjust	Multilayer Feed Forward Neural Network	56.38%	25000, 5000
Image Adjust	Random Forest with Variable Sample size = 5	46.69%	50000, 10000
Image Adjust	Random Forest with Variable sample size = 5	44.5%	25000, 5000
Image Adjust	Nearest Neighbors	61.27%	50000, 10000
Image Adjust	10 - Nearest Neighbors	54.98%	25000, 5000

Classification type (training and testing sample size: 25,000, 10,000)	Recall	Precision
Multilayer Feed Forward Neural Network	aquarium fish->0.795918 beaver -> 0.25	aquarium fish->0.619048 beaver -> 0.393939
Random Forest with Variable Sample size = 5	aquarium fish -> 0.894737 beaver-> 0.0384615	aquarium fish -> 0.383459 beaver -> 0.285714
10 - Nearest Neighbors	aquarium fish -> 0.877193 beaver -> 0.269231	aquarium fish -> 0.793651 beaver -> 0.304348

Naïve Bayes	Nearest Neighbors	Support Vector Machine	Logistic Regression
57.25%	61.27%	68.15%	68.97%