1072 Deep Learning – Homework 2

Due: May 24, 2019, 11:55pm

1. (50%) Please use logistic regression and neural net to classify two datasets, “Iris” and “Ionosphere.’ You can check UCI repository to see more details regarding these datasets.

Iris - <http://archive.ics.uci.edu/ml/datasets/Iris> (15%)

Ionosphere - <https://archive.ics.uci.edu/ml/datasets/ionosphere>

To save you some time, you could use “fisheriris.mat” and “ionosphere.mat” released to you.

The “Iris” dataset contains 150 feature vectors in 4 dimensions, where the label contains three different values, {setosa, versicolor, virginica}, each of which refers to a type of iris plant.

The “Ionosphere” dataset contains radar data collected by a system in Goose Bay, Labrador, which consists of a phased array of 16 high-frequency antennas with a total transmitted power on the order of 6.4 kilowatts. It has 34 real-valued features and categorical response: "b" for bad radar returns and "g" for good radar returns.

* 1. First randomly select 80% of the dataset as your training set and the rest 20% as your testing set. (5% for each)
  2. Train logistic regression and neural net classifiers for these two datasets and report accuracy for training and testing datasets as (20% for each)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Classifier | Iris | | Ionosphere | |
| Training | Testing | Training | Testing |
| Logistic Regression |  | |  | |
| Neural Net |  | |  | |

Please detail your models in the report. For neural net, your model must at least have three hidden layers, meaning including the input and output layers, in total, it has at least five layers. You are encouraged to try to use more layers or neurons in your model to see if they help increase the prediction accuracy.

1. (30%) In the mnist example where a classification model based on convolutional neural networks is provided, please add 10%, 20%, and 40% of noise to all the images and report the prediction accuracy for the training and testing datasets.

To add noise to the images, you could use the example code below:

import random

import numpy as np

noise\_lv = 0.1

img\_size = 28\*28

for i in range(len(X\_train)):

ran\_seq = random.sample([n for n in range(img\_size)], np.int(img\_size\*noise\_lv))

x = X\_train[i].reshape(-1, img\_size)

x[0, ran\_seq]=255

1. (20%) Following the previous question, please design a noise removal model based on convolutional neural networks for the images with added noise. Report the mean absolute error for the training and testing datasets.
2. (10% Bonus) Please use the noise removal model to pre-process the images before training the classification model. Would this help improve the prediction accuracy for the images with 10%, 20%, and 40% of noise?