Homework Assignment 4

This assignment includes two problem sets.

PART I

Overview

In this mini project, you will need to apply the feedforward neural network (FNN) on the MNIST dataset for hand written digit recognition.

Dataset

The MNIST database of handwritten digits has a training set of 60,000 digit images, and a test set of 10,000 images. It is a subset of a larger set available from NIST. The digits have been size-normalized and centered in a fixed-size image.

Assignment Project Exam Help

A starter script "main_ha4.py" is provided for downloading MNIST dataset, splitting datasets, and feature normalization. You don't need to changes these coding lines. You might need to install the following packages:

- tensorflow
- keras
- numpy
- scipy
- matplotlib
- sklearn

https://powcoder.com

Add WeChat powcoder

Method

For every image, we will use its pixel-wise intensities as features. To define a FNN model, you will need to specify the following hyper-parameters:

i) number of hidden layers and number of neural units for each layer;
ii) learning rate
iii) activation function (sigm, tanh_opt) and
iv) output function ('sigm','linear', 'softmax');

There is no need for you to code the neural network algorithm from scratch. You can use either third-party libraries or the Tensorflow implementations provided along with this homework (folder named 'code_FNN_TF').

Question 1. Please further split the 60,000 training images (and labels) into two subsets: 50,000 images, and 10,000 images. Use these two subsets for training models and validation purposes. In particular, you will train your FNN model using the 50,000 images and labels, and apply the trained model over the rest 10,000 images for evaluation purposes.

Please specify at least three sets of hyper-parameters (see the above). For each set, call the third-party functions or tensorflow to train a FNN model on the training samples (50,000 images in this case), and apply the learned model over the validation set (10,000 images in this case). For each model and its results, please compute its confusion matrix, average accuracy, per-class Precision and Recall. Report the model that achieves the top accuracy.

A sample function for calculating confusion matrix is provided in 'util.py'

Question 2. Apply the top ranked model over the testing samples (10,000 images). Call the above function to compute the confusion matrix, average accuracy, per-class precision/recall rate. In addition, select and visualize TEN testing images for which your mode made wrong predications. Try to analyze the reasons of these failure cases.

PART II

Assignment Project Exam Help

The purpose of this problem set is to practice the classification model Support Vector Machine (SVM).

https://powcoder.com

Overview

In this project, we try to build a SVM classifier that can identify the gender of a crab from its physical measurements. For every crab, there are six physical features: species, front-allip, rear-width, length, width and depth. You will need to train a binary SVM model from a set of training samples, and apply this model over the testing samples to evaluate its performance.

Add WeChat powcoder

The dataset is provided in the file 'crab.csv', including 200 samples with gender labels (1: male, -1: female). The starter codes are provided in the file 'main_svm.py'. Instructions are available in the .py file.

The starter codes include five major steps. To implement the training & testing, you are recommended to use the provided SVM demo codes.

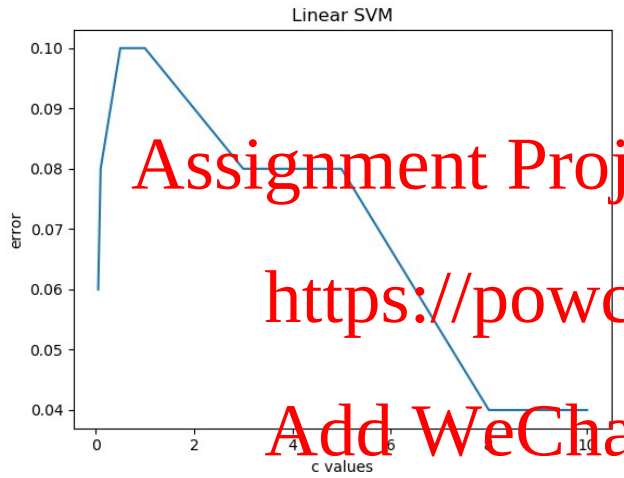
Step-1: Load data from 'crab.csv' to get feature matrix X and label vector Y. X is of 6 by 200 dimension, where each column represents a crab sample. The matrix Y is of 1 by 200 dimension, including the related gender labels (1 or -1). The starter codes will randomly split these 200 samples into two even subsets: use one for training & validation and another for testing. You don't need to change the codes in this step.

Step-2: Randomly divide the training set into two EVEN subsets: use one for training your model, and another for validation. You will need to implement your codes to do such random splitting. This step includes the "placeholder 1: training/validation".

Note that, you will need to use the same training/validation/testing splitting (steps 1 and 2) while evaluating different models in the later steps – you can either store and load the splitting data, or make sure all model selection processes are performed with the exactly same training/validation.

Step-3: Select the optimal model parameters using validation samples. You will need to consider the parameter C (the weighting parameters), kernel types (linear or others) and kernel parameters (if applicable). Please try as many parameters as you can to get the best result. In particular, you will need to generate two figures (in placeholders 2 and 3)

Figure 1: **S**election of Cs. Please use different C (e.g., 2, 4, 6, 10) to train the SVM classifier (with other hyper-parameter fixed). For each classifier, calculate its validation error rate (number of misclassified samples over the total number of validation samples). With these results, please generate the following figure:



Assignment Project Exam Help
https://powcoder.com
Add WeChat powcoder

where the horizontal direction represents different values of C, and the vertical direction represents validation errors. Use at least 3 different values for C.

Figure 2: selection of kernels Plot the validation errors while using linear, RBF kernel, or Polynomial kernel (with other hyper-parameters fixed);

Step-4: Select the best hyper-parameters (C, kernel types, etc.) and apply them over the testing subset. You may write a script to do the selection, or manually pick up the hyper-parameters based on your results. To do the latter, you might run steps 1 to 3 while temporarily commenting step-4 and step-5.

This step includes the "placeholder 4: testing".

Step-5: evaluate your results using confusion matrix and other metrics, including accuracy, precision and recall rates. Analyze your results through visualizing both success and failure examples. Include 5 success examples and 5 failure examples in your report.

This step includes the “placeholder 5: metrics” and “placeholder 6: success and failure examples”.

Submission instructions: what to hand in

- Prepare a Single PDF file to describe your decision process while completing the above questions. Include all figures, tables, intermediate results, or other results that might help understand your efforts. No need to include full paragraphs of your source codes in the report. Coding snippet is allowed.
- Submit your PDF file and source codes through the UCLA system.
- No HARD COPY IS REQURIED.

Assignment Project Exam Help

https://powcoder.com

Add WeChat powcoder