Project Part 1: Density Estimation and Classification

Re-submit Assignment

Due Sep 22, 2019 by 11:59pm **Points** 10 **Submitting** a file upload

Project Overview:

In this part, you need to first perform parameter estimation for a given dataset (which is a subset from the MNIST dataset). The MNIST dataset contains 70,000 images of handwritten digits, divided into 60,000 training images and 10,000 testing images. We use only images for digit “7” and digit “8” in this question.

Therefore, we have the following statistics for the given dataset:

- Number of samples in the training set: "7": 6265 ;"8": 5851.
- Number of samples in the testing set: "7": 1028; "8": 974

You are required to extract the following two features for each image:

1. The average of all pixel values in the image
2. The standard deviation of all pixel values in the image

We assume that these two features are independent, and that each image (represented by a 2-D features vector) is drawn from a 2-D normal distribution.

You may go to the original MNIST dataset (available here http://yann.lecun.com/exdb/mnist/ (Links to an external site.) **(http://yann.lecun.com/exdb/mnist/)**) to extract the images for digit 7 and digit 8, to form the dataset for this project. To ease your effort, we have also extracted the necessary images, and store them in “mnist_data.mat” files. The file can be downloaded **here.** A description of the file can be downloaded **here**.

You may use the following piece of code to read the dataset:

import scipy.io

Numpyfile= scipy.io.loadmat(‘mnist_data.mat’)

The specific algorithmic tasks you need to perform for this part of the project include:

1. Extracting the features and then estimating the parameters for the 2-D normal distribution for each digit, using the training data. Note: You will have two distributions, one for each digit.
2. Use the estimated distributions for doing Naïve Bayes classification on the testing data. Report the classification accuracy for both “7” and “8” in the testing set.
3. Use the training data to train a Logistic Regression model using gradient ascent. Report the classification accuracy for both “7” and “8” in the testing set. **Note that you are not allowed to use package like sklearn to compute the boundary. You need to implement your own version for using gradient**

ascent to find the solution.

Algorithms:

MLE Density Estimation, Naïve Bayes classification, Logistic regression

Resources:

A subset of MNIST dataset, download either from http://yann.lecun.com/exdb/mnist/ (Links to an external site.) **(http://yann.lecun.com/exdb/mnist/)** (requiring you to extract data corresponding to digit 7 and digit 8 only), or from the .mat files provided.

Workspace:

Any Python programming environment.

Software:

Python environment.

Language(s):

Python. (MATLAB is equally fine, if you have access to it.)

Required Tasks:

1. Write code to extract features for both training set and testing set.
2. Write code to implement the Naive Bayes Classifier and use it produce a predicted label for each testing sample.
3. Write code to implement the Logistic Regression and use it produce a predicted label for each testing sample.
4. Write code to compute the classification accuracy, for both the Naive Bayes Classifier and Logistic Regression.
5. Write a short report summarizing the results, including the final classification accuracy.

Note that you are not allowed to use package like sklearn to compute the boundary.

Optional Tasks:

1. Repeat the experiments for different pairs of digits.
2. Consider doing multi-class classification.

Optional tasks are to be explored on your own if you are interested and have extra time for them. No submission is required on the optional tasks. No grading will be done even if you submit any work on the optional tasks. No credit will be assigned to them even if you submit them. (So, please do not submit any work on optional tasks.)

Deliverables and due date(s):

The code and report are due at the end of Week 3 on Sunday, September 22, 2019, 11:59 PM (Arizona time).

What to Submit:

Code:

- Acceptable file types are .py/.m or .zip.
- If you have only one file, name the file to be main.py or main.m for matlab users, and submit it.
- If you have multiple code files, please name the main file as main.py and name other files properly based on its content; Similarly, for matlab users, you should have only one main.m and other relevant .m files. Next, zip all the files and submit Code.zip file.
- Documentation comment is important and required. Be sure to read through the directions carefully to ensure you have included all necessary parts in your code.

Report:

- Acceptable File types: .pdf
- Length: 2-5 A4 pages
- Content: Include the formula that you used to estimate the parameters, the estimated values for the parameters, the expression for the estimated normal distributions, an explanation for how the distributions are used in classifying a testing sample, and the final classification accuracy for both "7" and "8" for the testing set.