Machine Learning Homework 7

PCA, LDA & SNE

Due Date 23:55 3th July.

I. Kernel Eigenfaces/Fisherfaces (60%)

In this section, you are going to do face recognition using eigenface and fisherface.
Reference: https://www.csie.ntu.edu.tw/~mhyang/papers/fg02.pdf

- Data
  - The **Yale_Face_Database.zip** contains 165 images of 15 subjects (subject01, subject02, etc.). There are 11 images per subject, one for each of the following facial expressions or configurations: center-light, ~~w/glasses~~, happy, ~~left-light, w/~~no glasses, normal, right-light, sad, sleepy, surprised, and wink.
  - These data are separated into training dataset(135 images) and testing dataset(30 images). You can resize the images for easier implementation.
- What you are going to do
  - (25%) Use PCA and LDA to show the first 25 eigenfaces and fisherfaces, and randomly pick 10 images to show their reconstruction. (please refer to the lecture slides).
  - (10%) Use PCA and LDA to do face recognition, and compute the performance. You should use k nearest neighbor to classify which subject the testing image belongs to.
  - (25%) Use kernel PCA and kernel LDA to do face recognition, and compute the performance. (You can choose whatever kernel you want, but you should try different kernels in your implementation.) Then compare the difference between simple LDA/PCA and kernel LDA/PCA, and the difference between different kernels.

II. t-SNE (40%) http://www.datakit.cn/blog/2017/02/05/t_sne_full.html

Here are nice implementations of t-SNE in different programming languages: https://lvdmaaten.github.io/tsne/

- Data & reference code
  - Download link: https://lvdmaaten.github.io/tsne/code/tsne_python.zip,
  - **mnist2500_X.txt**: contains 2500 feature vectors with length 784, for describing 2500 mnist images.
  - **mnist2500_labels.txt**: provides corresponding labels
  - **tsne.py:** reference code
- What you are going to do
  - (15%) Try to modify the code a little bit and make it back to symmetric SNE. You need to first understand how to implement t-SNE and find out the specific code piece to modify. You have to explain the difference between symmetric SNE and t-SNE in the report (e.g. point out the crowded problem of symmetric SNE).
  - (5%) Visualize the embedding of both t-SNE and symmetric SNE. Details of the visualization:
    - Project all your data onto 2D space and mark the data points into different colors respectively. The color of the data points depends on the label.
    - Use videos or GIF images to show the optimize procedure.
  - (10%) Visualize the distribution of pairwise similarities in both high-dimensional space and low-dimensional space, based on both t-SNE and symmetric SNE.
  - (10%) Try to play with different perplexity values. Observe the change in visualization and explain it in the report.

III. Report

Submit a report in pdf format for showing your code with detailed explanations, giving detailed discussion on experiments as well as your observations. You should explain everything you have done in this homework and show all your results in the report. The report should be written in English.

Noted that if you don't explain your code in the report, you cannot get any point even you show the result.

IV. Turn in

1. Report (.pdf)
2. Source code
3. Videos or GIF images of optimize procedure

You should zip source code and report in one file and name it like ML_HW7_yourstudentID_name.zip, e.g. ML_HW7_0856XXX_王小明.zip.

P.S. If the zip file name has format error or the report is not in pdf format, there will be a penalty (-10). Please submit your homework before deadline, late submission is not allowed.

♦ Packages allowed in this assignment:

You are only allowed to use numpy, scipy.spatial.distance, and I/O related functions (like cv2.imread(), csv, matplotlib etc.). Official introductions can be found online.

Noted that scikit-learn and SciPy is not allowed.