Machine Learning Homework 7

Due Date: 2024/12/31 23:59

I. Kernel Eigenfaces

In this section, you are going to do face recognition using eigenface and fisherface.

Reference: https://faculty.ucmerced.edu/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, rightlight, 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
  - **Part 1:** 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).
  - **Part 2:** 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.
  - **Part 3:** 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

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
  - **Part 1:** 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).
  - **Part 2:** 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.
  - **Part 3:** Visualize the distribution of pairwise similarities in both high-dimensional space and low-dimensional space, based on both t-SNE and symmetric SNE.
  - **Part 4:** 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. The report should be written in English.
- **Please strictly follow the report format.** We will deduct some points according to the situation if you don't follow it.
- Since this homework is mainly graded by report, please spend more time on it. (e.g. well explained & organized) We won't give you any point if you just finish the code.
- Please don't explain the code line by line. You need to explain it clearly and well structured. For example, explain which part you have done in the function and how.

Report Format

1. **Code with detailed explanations (40%)**
   - Expected Content
     - Paste your code snippets in screenshot or in formatted code block with comments and explain your code.
     - Note if you don't explain your code, **you can't get any points in section 2 and 3 either!**
   - Kernel Eigenfaces (25%)
     - Part1 (10%)
       Also, simply explain how you do PCA & LDA (what is the step of it?)
     - Part 2 (5%)
     - Part 3 (10%)
       Also, simply explain how you do Kernel PCA & kernel LDA (what's the step of it?)
   - t-SNE (15%)
     - Part 1 (10%)
       Also, show the formula of t-SNE & SSNE
     - Part 2 (2%)
     - Part 3 (2%)
     - Part 4 (1%)
2. **Experiments and Discussion (50%)**
   - Expected Content
     - Show experiment settings and results, including the figures and the hyperparameters we asked you to show.
     - Note that if you don't explain your code in the above section, **you cannot get any points in this section either.**
   - Kernel Eigenfaces (20%)
     - Part 1 (5%)
     - Part 2 (5%)
     - Part 3 (5%) & (5%) Please discuss the observation in this part (You can compare the result with PCA/LDA)
   - t-SNE (30%)
     - Part 1 (5%) & (5%) Please discuss the observation in this part.
     - Part 2 (5%)

- Part 3 (5%)
- Part 4 (5%) & (5%) Please discuss the observation in this part.

3. Observations and Discussion (10%)

- Anything you want to discuss, such as the meaning of eigenfaces or trying different dimension reduction methods, comparing the advantages and disadvantages of them. (It is noticed that the score of this part is different from the discussion in Section 2. You can summarize the observation or try to discuss more about the project)
- If you need to refer to images or code snippets in previous sections, you can either
  i. Add a duplicate one in this section.
  ii. Add title or numbering system to all images and code blocks and refer to them by their corresponding identifier.
  iii. Put some parts of your discussions in the middle of the previous sections, but you must make it super obvious for us. (e.g. Add title or headings to tell us that the paragraph is part of "Observations and Discussion")

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).

- Submit your homework in time.
  - After the deadline, you can still submit in the following 7 days, you will get only 70% of the original score.
  - Starting from the seventh day after the deadline, you cannot submit your homework and you will get 0 score.
  - Whenever you submit your homework, the latest submission will be used for grading. (so **don't accidentally submit something after the deadline**, you will get 30% discount no matter what)

Note that if you miss report or source code, you cannot get any score!

◆ 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.

Important: scikit-learn and SciPy is not allowed.