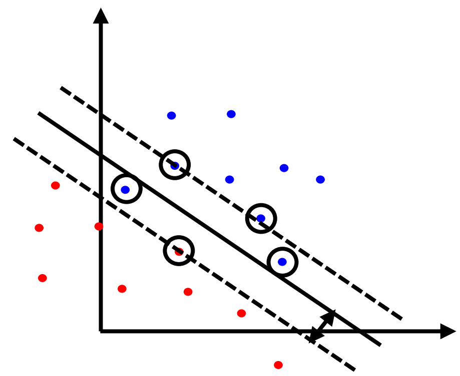
EE468/EE9SO29/EE9CS729

Pattern Recognition

Coursework on PCA, PCA-LDA Ensemble, and SVM for face recognition [50% mark]



Release on 12 Nov 2017, the report due on 17 Dec 2017 (midnight)

The course work requires Matlab programming. Use the provided face data (face.mat). In all questions, you can use any existing toolbox/code, unless specified.

Submission instructions:

One joint report by each pair

Page limit: 4-6 A4 pages per report with 10 font size (use the IEEE standard double column paper format, either in MS word or latex).

http://www.pamitc.org/cvpr16/files/egpaper_for_review.pdf

http://www.pamitc.org/cvpr16/files/cvpr2016AuthorKit.zip

Give insights, discussions, and reasons behind your answers, on the scope of lectures. ***Quality and completeness of discussions within the page limit*** will be marked.

Source code is not mandatory, unless specified. Optionally, this can go to appendices, which do not count for the page limit.

Submit the report **in pdf** through the Blackboard system. No hard copy is needed. Write your full names and CID numbers on the first page.

Do the question 1, and either the question 2-1 or question 2-2. The question 3 carries no mark, should go to an appendix of your report.

If you have questions, please contact

Dr. Guillermo Garcia-Hernando (g.garcia-hernando@imperial.ac.uk)

Useful toolboxes:

You may use the statistical pattern recognition toolbox (STPR) or LibSVM, which is downloaded at

http://cmp.felk.cvut.cz/cmp/software/stprtool/

https://www.csie.ntu.edu.tw/~cjlin/libsvm/

If you encounter any trouble in using the STPR toolbox in college machines, please contact g.garcia-hernando@imperial.ac.uk.

Q1.

[10] Eigenfaces

a. Partition the provided face data into your training and testing data, in a way you choose. Explain briefly the way you partitioned. Apply PCA to your training data, by computing the eigenvectors and eigenvalues of the covariance matrix $S=(1/N)AA^T$ directly. Show and discuss the results, including: the eigenvectors, the eigenvalues, and the mean image, how many eigenvectors with non-zero eigenvalues are obtained and how many eigenvectors are to be used for face recognition. Give insights and reasons behind your answers.

b. Apply PCA to your training data, using the eigenvectors and eigenvalues of $(1/N)A^TA$. Show and discuss the results in comparison to the above, including: if the eigenvectors and eigenvalues obtained are identical, what are the pros/cons of each method. Show respective measurements for your answers.

[15] Application of Eigenfaces

Hereinafter, we use a more efficient PCA technique among the two methods in the above. Use the data partition, which you used in Q1, into training and testing.

a. Perform the face image reconstruction using the PCA bases learnt. Show and discuss the results, while varying the number of bases to use, including: if the reconstruction error (or the distortion measure) obtained is same as in the theory, how good the reconstruction results are for at least 3 images of your choice (e.g. from both the training and testing dataset).

b. Perform the PCA-based face recognition by either the NN classification method or alternative method learnt in the PCA lecture. Report and discuss, including: the recognition accuracy (success rates), example success and failure cases, the confusion matrices, time/memory (and any other aspects you observe), by varying the parameter values/experimental settings you used. Give insights and reasons behind your answers.

Select one of the two questions below, either Q2-1 or Q2-2.

Q2-1. [25] Multi-class SVM for Face Recognition

Use the provided face data, and the same data partition into training and testing as in Q1.

Feature vectors **x** are the raw-intensity vectors (obtained by raster-scanning pixel values of face images) or PCA coefficients. Try both and compare the results below.

Train and test multi-class SVM using the feature vectors **x**. You can use any existing toolbox for two-class (or binary-class) SVMs. Note, write your own lines of code for the multi-class extensions of SVM (both one-versus-the-rest and one-versus-one), and provide your code in an appendix of your report. Compare the results of the two multi-class extensions of SVM.

Show, measure and discuss the results, including:

- setting the SVM parameters, i.e. kernel type, kernel parameters, C (underfitting/overfitting),
- recognition accuracy and confusion matrix
- time-efficiency of SVM training/testing
- examples of support vectors and success/failure images
- margin etc.

Discuss the results in comparison to those of **Q1**.
Give insights and reasons behind all your answers.

Q2-2. [25] LDA Ensemble for Face Recognition

Use the provided face data, and the same data partition into training and testing as in Q1.

Try LDA and its ensemble learning, along with PCA and NN classifier. Compare and discuss face recognition results.

PCA-LDA

Perform the PCA-LDA based face recognition with the NN classifier. Report and discuss, including:

- recognition accuracies by varying the parameter values, M_pca and M_lda
- ranks of the scatter matrices,
- the confusion matrix, example success and failure cases

Explain your observations and reasons, and discuss the results in comparison to those of **Q1**.

PCA-LDA Ensemble

Show, measure and discuss the results, including:

- randomisation in feature space
- randomisation on data samples (i.e. bagging)
- the number of base models, the randomness parameter,
- the error of the committee machine vs the average error of individual models
- fusion rules
- recognition accuracy and confusion matrix

Observe and discuss the above by varying the parameter values/architectures you used. Give insights and reasons behind all your answers.

Q3. Generative and Discriminative Subspace Learning

(This part carries no mark, should go to an appendix of your report.)

PCA is a generative model, by which input images or data can be reconstructed. LDA is a discriminative model, which extracts better features for classification. Say we are interested in subspace learning that fulfils both aspects or controls a balance between the two aspects.

Mathematically formulate the problem (i.e. the objective or goal function to optimise) that learns the subspace for reconstruction and discriminative features at the same time.

And mathematically derive the solution that optimises the defined problem. If needed, you may use Lagrange multiplier formulation, gradient-based optimization, eigenvector-eigenvalues, and/or generalized eigenvector-eigenvalues.

Discuss foreseeable behaviours, and pros and cons, of your method. No programming is needed.