Imperial College
London

COURSEWORK

IMPERIAL COLLEGE LONDON

DEPARTMENT OF COMPUTING

# 496 Mathematics for machine learning

*Author:*
Jinwei Zhang (CID: 01540854)

Date: November 29, 2018

# Part I

You are given a facial dataset with identity information per sample (the dataset is part of PIE facial database). You are also given six Matlab code files. In particular, demo PCA.m, demo wPCA.m and demo LDA.m run a face recognition protocol and report the results in the end (in the form of recognition error). Demo files may not be modified. PCA.m, wPCA.m and LDA.m are utilised by demo PCA.m, demo wPCA.m and demo LDA.m, respectively, and are to be completed matlab func- tions that should perform dimensionality reduction techniques (PCA, whitened PCA and LDA, respectively) on Small Sample Sized (SSS) problems (number of samples significantly less that the number of features), according to the notes. You should attach in your report plots of the recognition error versus the number of components kept for each of the methods. Briefly explain which method is the best and why.

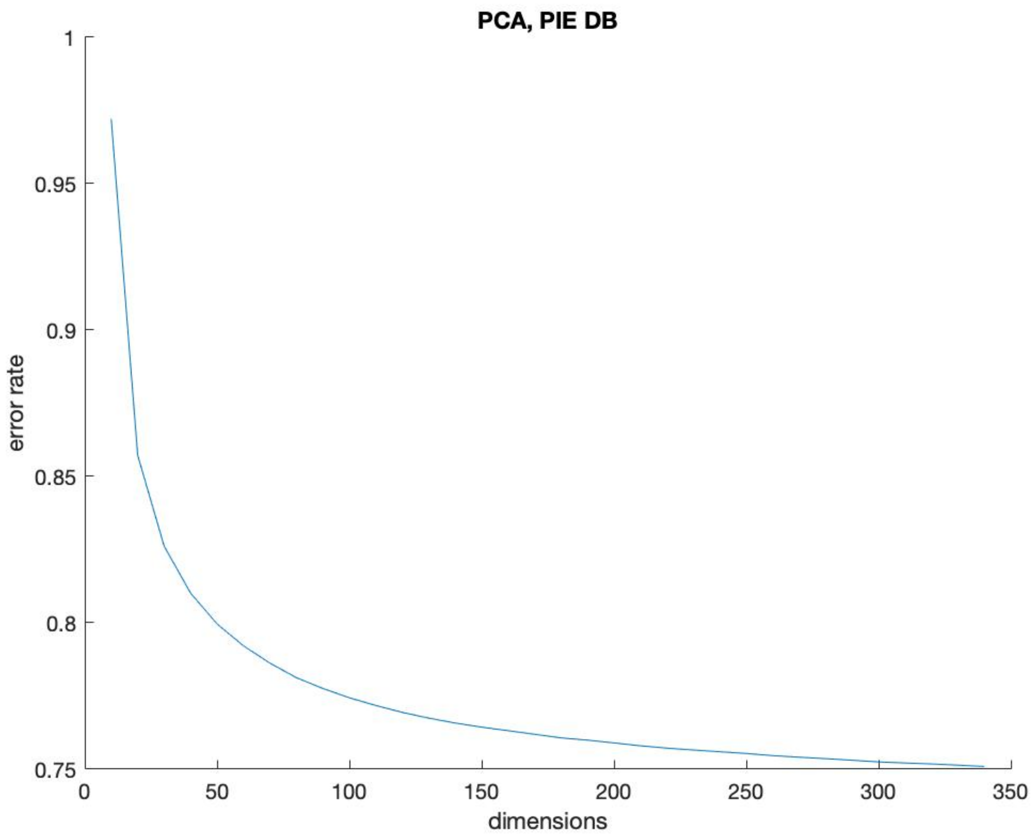


**Figure 1:** error rate by PCA

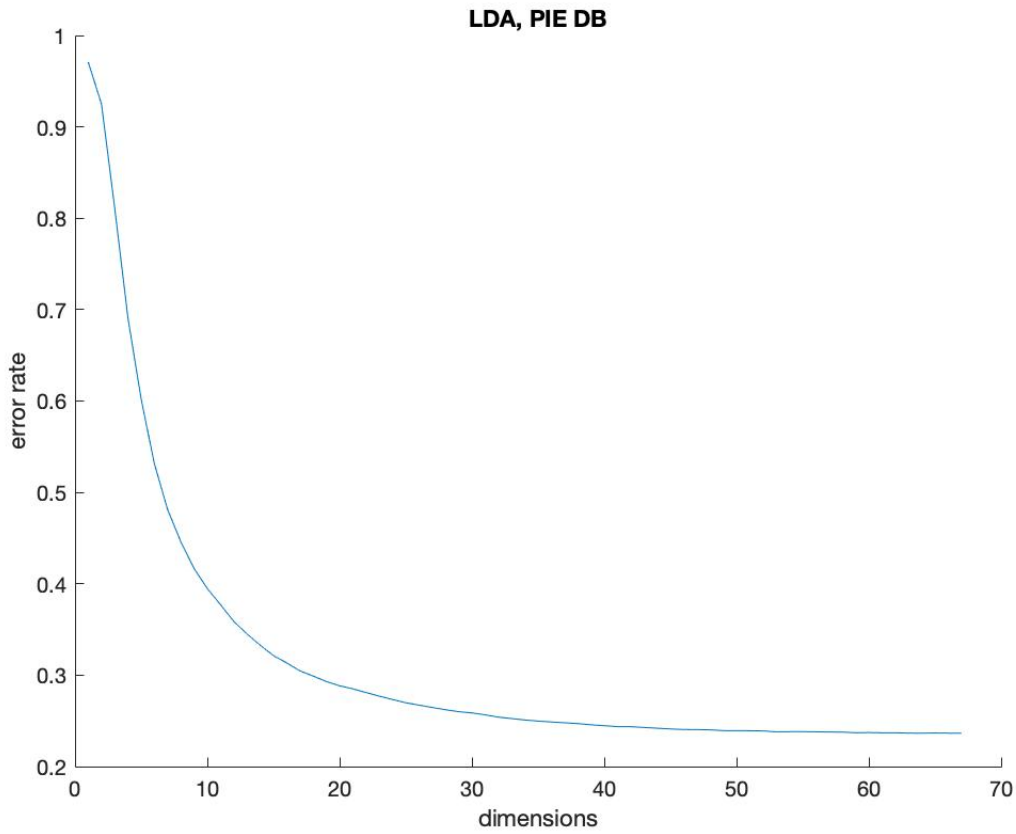


**Figure 2:** error rate by LDA

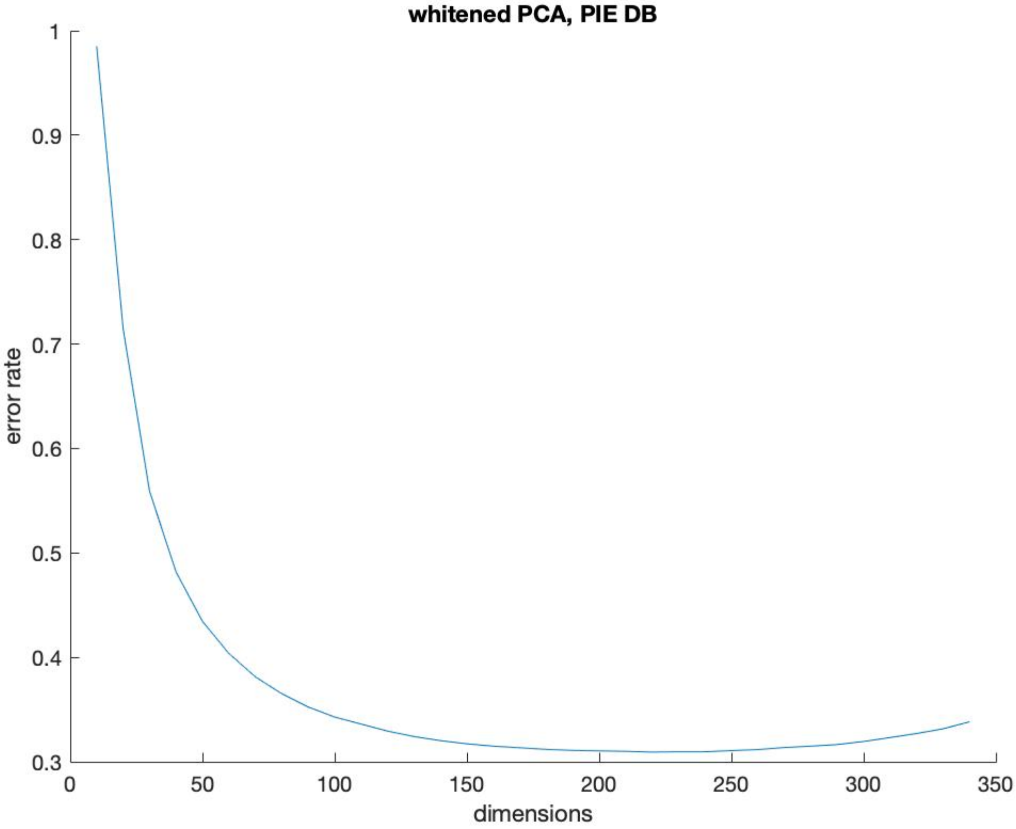


**Figure 3:** error rate by wPCA

Answer:
The best is LDA then wPCA, the worst is PCA. The reason is whitened PCA use another whitening process so that the result is better. LDA combines the label class, it does not need that large amount of data and can achieve a lower error rate. The reason why the error of wPCA goes up a little bit is maybe because the amount of the data is not big enough.

# Part II

Assume you are given a set of n data samples $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$ . The minimum enclosing hyper-sphere of the above data samples can be found by the following constrained optimisation problem

$$\min_{R, \boldsymbol{a}, \xi_i} \quad R^2 + C \sum_{i=1}^{n} \xi_i \tag{1}$$

$$\text{subject to} \quad (\boldsymbol{x}_i - \boldsymbol{a})^T (\boldsymbol{x}_i - \boldsymbol{a}) \leq R^2 + \xi_i, \forall \xi_i \geq 0 \tag{2}$$

where R is the radius of the hyper-sphere, a is the center of the hyper-sphere and the variable C gives the trade-of between simplicity (or volume of the sphere) and the number of errors. You should perform the following tasks

1. Formulate the Lagrangian of the above optimisation problem and derive its dual. Attach your answers in the report. Show the steps that you followed, do not only report the final results.

   Answer:

$$L(R, \boldsymbol{a}, \xi_i, \lambda, r) = R^2 + C\sum_{i=1}^{n}\xi_i + \sum_{i=1}^{n}\lambda_i[(\boldsymbol{x}_i - \boldsymbol{a})^\top(\boldsymbol{x}_i - \boldsymbol{a}) - R^2 - \xi_i] - \sum_{i=1}^{n} r_i\xi_i \tag{3}$$

   To find the dual, we need to maximise the Lagrangian function by finding the first order derivative with respect to $R$, $\boldsymbol{a}$, and $\xi_i$.
   We thus differentiate the Lagrangian function as follow:

$$\frac{\partial L}{\partial R} = 2R - 2R\sum_{i=1}^{n}\lambda_i = 0, (R \neq 0) \tag{4}$$

$$\Rightarrow 1 - \sum_{i=1}^{n}\lambda_i = 0 \tag{5}$$

$$\Rightarrow \sum_{i=1}^{n}\lambda_i = 1 \tag{6}$$

$$\frac{\partial L}{\partial \boldsymbol{a}} = \sum_{i=1}^{n}\lambda_i[-2(\boldsymbol{x}_i - \boldsymbol{a})] = 0 \tag{7}$$

$$\Rightarrow \sum_{i=1}^{n}\lambda_i(\boldsymbol{x}_i - \boldsymbol{a}) = 0 \tag{8}$$

$$\Rightarrow \sum_{i=1}^{n}(\lambda_i\boldsymbol{x}_i - \boldsymbol{a}\lambda_i) = 0 \tag{9}$$

$$\xRightarrow{\text{using (6)}} \sum_{i=1}^{n}(\lambda_i\boldsymbol{x}_i) - \boldsymbol{a} = 0 \tag{10}$$

$$\Rightarrow \boldsymbol{a} = \sum_{i=1}^{n}\lambda_i\boldsymbol{x}_i \tag{11}$$

$$\frac{\partial L}{\partial \xi_i} = C - \lambda_i - r_i = 0 \tag{12}$$

$$\Rightarrow C = \lambda_i + r_i \tag{13}$$

using (6), (8), (11), (13). we can write the Lagrangian fucntion as follow:

$$L(R, \boldsymbol{a}, \xi_i, \lambda, r) = R^2 + C\sum_{i=1}^{n}\xi_i + \sum_{i=1}^{n}\lambda_i[(\boldsymbol{x}_i - \boldsymbol{a})^\top(\boldsymbol{x}_i - \boldsymbol{a}) - R^2 - \xi_i] - \sum_{i=1}^{n} r_i\xi_i \tag{14}$$

$$= R^2 + \sum_{i=1}^{n}(\lambda_i + r_i)\xi_i + \lambda_i\sum_{i=1}^{n}(\boldsymbol{x}_i - \boldsymbol{a})^\top(\boldsymbol{x}_i - \boldsymbol{a}) - \sum_{i=1}^{n}\lambda_i R^2 - \sum_{i=1}^{n}\lambda_i\xi_i - \sum_{i=1}^{n} r_i\xi_i \tag{15}$$

$$= R^2 + \sum_{i=1}^{n}(\lambda_i + r_i)\xi_i + \sum_{i=1}^{n}[\lambda_i(\boldsymbol{x}_i - \boldsymbol{a})^\top(\boldsymbol{x}_i - \boldsymbol{a})] - R^2 - \sum_{i=1}^{n}(\lambda_i + r_i)\xi_i \tag{16}$$

$$= \sum_{i=1}^{n}[\lambda_i(\boldsymbol{x}_i^\top - \boldsymbol{a}^\top)(\boldsymbol{x}_i - \boldsymbol{a})] \tag{17}$$

$$= \sum_{i=1}^{n}\lambda_i[\boldsymbol{x}_i^\top\boldsymbol{x}_i - \boldsymbol{x}_i^\top\boldsymbol{a} - \boldsymbol{a}^\top\boldsymbol{x}_i + \boldsymbol{a}^\top\boldsymbol{a}] \tag{18}$$

$$= \sum_{i=1}^{n}[\lambda_i\boldsymbol{x}_i^\top\boldsymbol{x}_i - \lambda_i\boldsymbol{x}_i^\top\boldsymbol{a} - \lambda_i\boldsymbol{a}^\top\boldsymbol{x}_i + \lambda_i\boldsymbol{a}^\top\boldsymbol{a}] \tag{19}$$

$$= \sum_{i=1}^{n}[\lambda_i\boldsymbol{x}_i^\top\boldsymbol{x}_i - \lambda_i\boldsymbol{x}_i^\top\boldsymbol{a} - \lambda_i\boldsymbol{a}^\top(\boldsymbol{x}_i - \boldsymbol{a})] \tag{20}$$

$$= \sum_{i=1}^{n}[\lambda_i\boldsymbol{x}_i^\top\boldsymbol{x}_i - \lambda_i\boldsymbol{x}_i^\top\boldsymbol{a}] - \boldsymbol{a}^\top\sum_{i=1}^{n}\lambda_i(\boldsymbol{x}_i - \boldsymbol{a}) \tag{21}$$

$$\xRightarrow{\text{using (8)}} \sum_{i=1}^{n}[\lambda_i\boldsymbol{x}_i^\top\boldsymbol{x}_i - \lambda_i\boldsymbol{x}_i^\top\boldsymbol{a}] \tag{22}$$

by using (11) $\quad \boldsymbol{a} = \sum_{j=1}^{n}\lambda_j\boldsymbol{x}_j$ we formalise the equation as follow

$$L(\lambda) = L(R, \boldsymbol{a}, \xi_i, \lambda, r) \tag{23}$$

$$= \sum_{i=1}^{n}[\lambda_i\boldsymbol{x}_i^\top\boldsymbol{x}_i - \lambda_i\boldsymbol{x}_i^\top\boldsymbol{a}] \tag{24}$$

$$= \sum_{i=1}^{n}(\lambda_i\boldsymbol{x}_i^\top\boldsymbol{x}_i) - \sum_{i=1}^{n}\lambda_i\boldsymbol{x}_i^\top(\sum_{j=1}^{n}\lambda_j\boldsymbol{x}_j) \tag{25}$$

$$= \sum_{i=1}^{n}(\lambda_i\boldsymbol{x}_i^\top\boldsymbol{x}_i) - \sum_{i=1}^{n}\sum_{j=1}^{n}\lambda_i\boldsymbol{x}_i^\top\boldsymbol{x}_j\lambda_j \tag{26}$$

since $\boldsymbol{K}_x = [\boldsymbol{x}_i^\top\boldsymbol{x}_j]$. we can write the dual as:

$$= \mathrm{diag}(\boldsymbol{X}^\top\boldsymbol{X})^\top\lambda - \lambda^\top(\boldsymbol{X}^\top\boldsymbol{X})\lambda \tag{27}$$

$$= \mathrm{diag}(\boldsymbol{K}_x)^\top \boldsymbol{\lambda} - \boldsymbol{\lambda}^\top (\boldsymbol{K}_x)\boldsymbol{\lambda} \tag{28}$$

consider, $0 \leq \lambda_i \leq C$, for $i = 1, \ldots, n$. and also $\sum_{i=1}^n \lambda_i = 1$ (6) as the constrains.

$$\begin{aligned} \min_{\lambda} \quad & L(\lambda) = -\mathrm{diag}(\boldsymbol{K}_x)^\top \boldsymbol{\lambda} + \boldsymbol{\lambda}^\top (\boldsymbol{K}_x)\boldsymbol{\lambda} \\ \text{subject to} \quad & 0 \leq \lambda_i \leq C, \quad \text{for } i = 1 \ldots, n \\ & \sum_{i=1}^n \lambda_i = 1 \quad \Rightarrow \quad \boldsymbol{\lambda}^T \mathbf{1}_n = 1 \end{aligned}$$

2. Perform the above when using arbitrary positive definite kernels (i.e., $k(\boldsymbol{x_i}, \boldsymbol{x_j}) = \phi(\boldsymbol{x_i})^T \phi(\boldsymbol{x_j})$) , i.e. find the dual of the following optimisation problem

$$\min_{R, \boldsymbol{a}, \xi_i} \quad R^2 + C \sum_{i=1}^n \xi_i \tag{29}$$

$$\text{subject to} \quad (\phi(\boldsymbol{x_i}) - \boldsymbol{a})^T (\phi(\boldsymbol{x_i}) - \boldsymbol{a}) \leq R^2 + \xi_i, \forall \xi_i \geq 0 \tag{30}$$

Attach your answers in the report. Show the steps that you followed, do not only report the final results.
Answer:
the previous equation can be changed as follow: We also note that, by using the kernal: $k(\boldsymbol{x_i}, \boldsymbol{x_j}) = \phi(\boldsymbol{x_i})^T \phi(\boldsymbol{x_j})$, and $\boldsymbol{K}_x = [k(\boldsymbol{x}_i, \boldsymbol{x}_j)]$ we have

$$\begin{aligned} \min_{\lambda} \quad & L(\lambda) = -\mathrm{diag}(\boldsymbol{K}_x)^\top \boldsymbol{\lambda} + \boldsymbol{\lambda}^\top (\boldsymbol{K}_x)\boldsymbol{\lambda} \\ \text{subject to} \quad & 0 \leq \lambda_i \leq C, \quad \text{for } i = 1 \ldots, n \\ & \sum_{i=1}^n \lambda_i = 1 \quad \Rightarrow \quad \boldsymbol{\lambda}^T \mathbf{1}_n = 1 \end{aligned}$$

3. Produce data from the two classes and find their optimal enclosing hyper-sphere (i.e., calculate the centres and the radii) by solving the dual optimisation problem of (2) in Matlab (to solve the quadratic constrained optimisation problem you may use quadprog()). To solve this coding part, you should fill in the function calcRandCentre in the Matlab file partII.m. Attach the final plot with the centres and the radii in your report.

Radius:
By calculating all the parametres and using the quadprog() function, we get an optimal $\boldsymbol{\lambda}^*$ matrix, output by the function. We can then substitute the $\boldsymbol{\lambda}^*$ back into the dual form, to get the value of function. The negative sign converting maximisation problem to a minimisation

$$L(\lambda) = -\operatorname{diag}(\boldsymbol{K}_x)^\top \boldsymbol{\lambda} + \boldsymbol{\lambda}^\top (\boldsymbol{K}_x)\boldsymbol{\lambda}$$

$$R^2 + C\sum_{i=1}^{n} \xi_i$$

Furthermore, the optimal solution to the optimisation is equal to the optimal solution to the dual. when we want to get the optimal solution, the slack variables, $\xi_i$ become 0. By by taking the square root of $L(\lambda)$. We can thus find the optimal radius of the hypersphere

Centre a:

By using the fact (11):

$$\boldsymbol{a} = \sum_{i=1}^{n} \lambda_i \boldsymbol{x}_i$$

where $\boldsymbol{a} \in \mathbb{R}^2$ is the 100 times sum of the $\lambda_i$row and $\boldsymbol{x}_i$column 4.

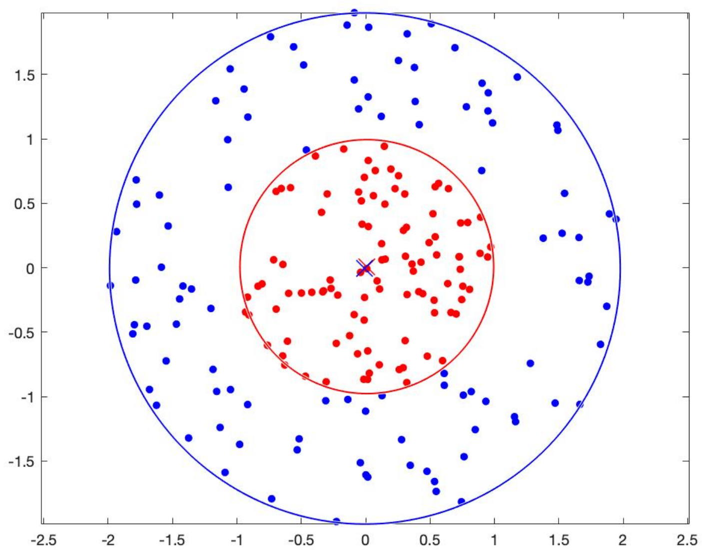


**Figure 4:** data points and optimal enclosing hyper spheres

a1(1),a1(2) 0.0036964 0.0036964 radius1: 0.98658
a2(1),a2(2) -0.0036376 -0.0036376 radius2: 1.9835