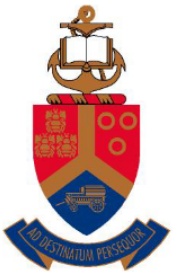

UNIVERSITEIT VAN PRETORIA
UNIVERSITY OF PRETORIA
YUNIBESITHI YA PRETORIA

COS 314: Artificial Intelligence
Assignment 1: Eigenfaces for feature extraction and classification
Due Date: 7 May 2021

1 Assignment Outline

One of the most valuable applications for PCA is feature extraction. A great example of an application where feature extraction is helpful is with images. Images are made up of pixels, usually stored as red, green, and blue (RGB) intensities. Objects in images are usually made up of thousands of pixels, and only together are they meaningful.

This assignment involves using PCA to extract representative features of image datasets in a lower-dimensional feature space and using Neural Networks(NN) to classify the images. You will be working with the face images from the Labeled Faces in the Wild dataset. This dataset contains face images of celebrities downloaded from the Internet, and it includes faces of politicians, singers, actors, and athletes from the early 2000s.

There are 3,023 images, each 87×65 pixels large, belonging to 62 different people. In the example notebook attached on ClickUp, there is a sample code on extracting and visualizing the data. The X_people array is a 2D array where each row represents an individual, and each column represents a pixel, while the y_people array contains the class labels.

In this assignment you are required to do the following:

1. Eigenfaces:
   (a) Using the X_people array, use PCA to obtain the top k principal components of the faces dataset
   (b) Visualise the top 15 principal components of the faces dataset and explain the results
   (c) Visualise a few images of reconstructed faces using 10, 30, 50, 100 and 500 components and explain the quality of the visualisations you see
   (d) You are allowed to use the PCA scikit-learn library
2. Classification:
   (a) You will be required to implement the backpropagation learning algorithm for pattern classification. The single layer Neural Networks must be trained on the transformed lower dimensional PCA representation. The NN must classify faces to the respective individual. You need to use a training and test set. You can also use a validation set in addition to the training and test sets.

(b) Due to the size of the dataset you are only required train and test your model on the following individual faces with the corresponding index in y_people in brackets:

i. Hugo Chavez (21)
ii. Ariel Sharon (5)
iii. Junichiro Koizumi (35)
iv. Serena Williams (53)
v. Tony Blair (58)
vi. George W Bush (14)
vii. Colin Powell (10)
viii. Gerhard Schroeder (15)
ix. Donald Rumsfeld (12)
x. Tony Blair (58)

(c) You are required to create a new dataset from X_people with only the above individuals and their class labels are required to be remapped as follows: [Hugo Chavez: 0, Ariel Sharon: 1, Junichiro Koizumi: 2 , Serena Williams: 3, Tony Blair: 4, George W Bush: 5, Colin Powell: 6, Gerhard Schroeder: 7 , Donald Rumsfeld: 8, Tony Blair: 9]

(d) You are not allowed to use Keras/Pytorch/scikit-learn or any deep learning library to construct your network

2 Submission document/code

You must submit a report, a jupyter notebook that contains your Eigenface analysis code and a *.py* file which contains your NN classifier.

Please adhere to the following submission instructions:

1. The *.py* file with the classifier must output the following to a text file:
   - The weight matrix
   - The biases
   - The accuracy for the training set
   - The accuracy for the test set
   - The confusion matrix
2. The *.py* file must take the following input parameters:
   - A numpy array for the training set
   - A numpy array for the test set
   - The learning rate of the model
3. In addition to the program and notebook please submit a report (in PDF) specifying:
   - Eigenfaces
     - Visualisation of the top 15 principal components of the faces dataset and your observations of the visualisations
     - Visualisation of the images of reconstructed faces using 10, 30, 50, 100 and 500 components and observations on the quality of the visualisations
     - Explanation of how the number of components where chosen

- Classification
  - Number of hidden layers
  - Number of nodes in each hidden layer
  - Activation functions used
  - Learning rules used (if these differ from the those discussed in class)
  - How you chose your parameters
- Conclusion: Overall findings and future recommendations to improve results

3 Mark breakdown

Report: 20 marks Algorithm: 30 marks Total 50