ARISTOTLE UNIVERSITY OF THESSALONIKI
DEPARTMENT OF INFORMATICS

PATTERN RECOGNITION
Programming Exercise 2022
Points: 2

This exercise is about handwritten digit clustering and classification using the public dataset MNIST. MNIST consists of 70000 images of resolution 28×28 pixel, categorized in 10 classes (0-9). Each pixel has a value between 0 and 255. The dataset is split in 60000 training images and 10000 test images. All data along with their corresponding ground truth labels are available at http://yann.lecun.com/exdb/mnist/.

Tasks:

1. Download the dataset and load only a subset of it, consisting of the classes (digits) $i = 1, 3, 7, 9$. Let M_i, N_i be the number of training and test images for class i, respectively. Eventually, you are asked to load four matrices: $\mathbf{M}$ (size $\left(\sum_{i=0}^{3} M_i\right) \times 784$) containing the training data, $\mathbf{N}$ (size $\left(\sum_{i=0}^{3} N_i\right) \times 784$) containing test data, as well as $\mathbf{L}_{tr}, \mathbf{L}_{te}$, (size $\left(\sum_{i=0}^{3} M_i\right) \times 1$ and $\left(\sum_{i=0}^{3} N_i\right) \times 1$, respectively) containing the corresponding class labels i.

2. Reshape each row of $\mathbf{M}$ (that is, each training image) in a square image matrix 28×28 pixel. Then, from each image matrix, extract a two-dimensional feature vector $\mathbf{m}$ using the following process: the first feature component is the mean pixel value of all image matrix rows with odd index, while the second feature component is calculated as the mean pixel value of all image matrix columns whose index is an even number. After calculating the two-dimensional features for each sample in $\mathbf{M}$, use them to create a matrix $\hat{\mathbf{M}}$, of size $\left(\sum_{i=0}^{3} M_i\right) \times 2$. Use a scatter plot to visualize all rows of $\hat{\mathbf{M}}$, considering that the horizontal axis corresponds to their first feature component and the vertical axis to their second feature component. Use $\mathbf{L}_{tr}$ to assign different colors for different class samples (e.g., red for $i = 1$, green for $i = 3$, blue for $i = 7$ and yellow for $i = 9$).

3. Implement and execute the K-Means algorithm, using as samples the rows of $\hat{\mathbf{M}}$. Set $= 4$ and use your implementation of the Maximin algorithm to initialize the cluster centers in the K-Means algorithm. Visualize the rows of $\hat{\mathbf{M}}$ in a scatter plot, this time using the clustering results to assign different colors to different cluster samples. Use $\mathbf{L}_{tr}$ to calculate the purity of the clustering results.

4. Implement the Principal Component Analysis (PCA) algorithm. Aplly the algorithm to reduce the dimension of rows of $\mathbf{M}$, in order to get a new matrix $\tilde{\mathbf{M}}$, of size $\left(\sum_{i=0}^{3} i\right) \times V$, where $V = 2$ the new number dimensions. Visualize the rows of $\tilde{\mathbf{M}}$ in a scatter plot, using again $\mathbf{L}_{tr}$ to assign different colors to different class samples. Then, after using Maximin on the rows of $\tilde{\mathbf{M}}$ to initialize the cluster centers ($K = 4$), execute the K-Means algorithm (on $\tilde{\mathbf{M}}$). Visualize the new clustering results and calculate the clustering purity. Repeat the above (without visualization) for $V = 25, 50, 100$. Find the value of V for which the clustering purity is maximized (V_{max}).

5. Implement a Gaussian Naive Bayes Classifier. Use the rows of $\tilde{\mathbf{M}}$ for $V = V_{max}$ and the ground truth labels of $\mathbf{L}_{tr}$ to train the classifier. Then, use the same dimensionality reduction

process on the test data samples (rows of $\mathbf{N}$), in order to obtain $\tilde{\mathbf{N}}$. Use the trained classifier on the new test samples (rows of $\tilde{\mathbf{N}}$). Use the classification results and $\mathbf{L}_{te}$ to calculate the classification accuracy.

Submission in E-Learning until: 20/05/2022