INT 410- Assessment 3

Assessment Number	3
Contribution to Overall Marks	70%
Submission Deadline	14-Janunary -2021

Assessment Objective

This assessment aims at evaluating students' ability to exploit the advanced pattern recognition knowledge, which is accumulated during lectures, and after-class study, to analyze, design, implement, develop, test and document the pattern classification methods with Neural Networks (MLP), Discriminant Function (MQDF), and Kernel Methods (Support Vector Machine) typically on the image data.

General Guidelines

- An introduction of each model, a description of the hyperparameters and structure of each framework, an interpretation of each algorithm, clipped running images of the training process, and analysis of the results are required to be included in your report.
- Descriptions in the Problem Specifications are required to be analyzed with mathematic equations, combined with the explanations of all elements in each equation.
- A concise description of your results and observations should be included in your report.
- The modified part of the source codes particularly for SVM is required to be included in the report.
- The report is strongly suggested to be written with the latex typesetting language.
- Submit the report and the source code electronically into LearningMall.
- The report in pdf format and source code of your implementation should be zipped into a single file. The naming of report is as follows:
  e.g. StudentID_LastName_FirstName_LabNumber.zip (123456789_Einstein_Albert_1.zip)

Handwritten Numeral Image Classification

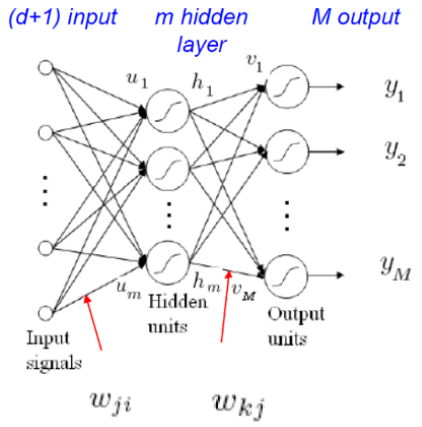
Overall Description:

The assessment focuses on comparing three classic classifiers, i.e. Multi-Layer Perceptron (MLP), Support Vector Machine (SVM) and Modified Quadratic Discriminant Function (MQDF), on the image classification dataset. You are required to design and optimise the models to achieve accurate classification performance on the specified dataset. The assessment includes the programming codes and the report. The programming codes should be implemented correctly, run successfully, and lead to sensible results. Moreover, the code quality will also be considered, such as efficiency, comments, robustness. You are required to follow the general guidelines and answer all the questions in the report clearly.

Tasks:

1. Implement Multilayer Perceptron (MLP) on MNIST **(25 marks)**

   An artificial neural network is composed of many artificial neurons that are linked together according to a specific network architecture. The model transforms the inputs into meaningful outputs by iterating a combination of linear and nonlinear functions.



$$W = \arg\min_{W} \frac{1}{2} \sum_{n=1}^{N} \sum_{k=1}^{M} [y_k(x^n) - t_k^n]^2$$

$$y_k(x) = g[v_k(x)]$$

$$v_k(x) = \sum_{j=1}^{m} w_{kj} h_j + w_{k0}$$

$$h_j(x) = g[u_j(x)]$$

$$u_j(x) = \sum_{i=1}^{d} w_{ji} x_i + w_{j0}$$

Design an MLP model to achieve the classification task on MNIST. Show the training process and classification results.

Many factors will affect the performance, such as, the number of nodes in hidden layers, the selection of the activation functions, the normalization method of the samples, the loss function and its regularization term, the learning rate and the stopping policy of training, etc.

2. Implement Modified Quadratic Discriminant Function (MQDF) on MNIST **(25 marks)**

The MQDF model is a classifier based on Bayesian decision theory. Its discriminant function is given as below:

$$g_0(\mathbf{x},\omega_i) = -(\mathbf{x}-\mu_i)^T \Sigma_i^{-1}(\mathbf{x}-\mu_i) - \log|\Sigma_i|$$
$$= -[\Phi_i^T(\mathbf{x}-\mu_i)]^T \Lambda_i^{-1}\Phi_i^T(\mathbf{x}-\mu_i) - \log|\Sigma_i|$$
$$= -\sum_{j=1}^{d}\frac{1}{\lambda_{ij}}[(\mathbf{x}-\mu_i)^T\phi_{ij}]^2 - \sum_{j=1}^{d}\log\lambda_{ij}$$

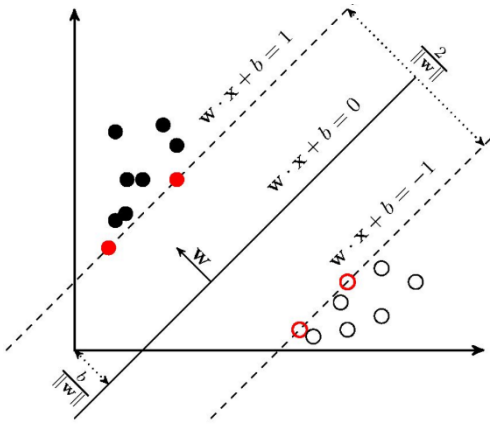
$$\Sigma_i = \Phi_i \Lambda_i \Phi_i^{\mathrm{T}}$$

Design and implement a MQDF model to achieve the classification task on MNIST. Show the training process and classification results. The work in [1] can be used as reference for your work.

[1] F. Kimura, et zl., Modified quadratic discriminant functions and the application to Chinese character recognition, IEEE Trans. PAMI, 9(1): 149-153, 1987.

3. Implement Support Vector Machine (SVM) on MNIST **(20 marks)**

A Support Vector Machine constructs a hyperplane or set of hyperplanes in a high- or infinite-dimensional space, which can be used for classification, regression, or other tasks.



$$\max.\ W(\boldsymbol{\alpha}) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1,j=1}^{n} \alpha_i \alpha_j y_i y_j K(\mathbf{x}_i, \mathbf{x}_j)$$

$$b = 1 - \frac{1}{|i : 0 < \alpha_i < C|} \sum_{i:0<\alpha_i<C} \sum_{j=1}^{s} \alpha_j y_j k(\mathbf{x}_j, \mathbf{x}_i)$$

$$C \geq \alpha_i \geq 0, \sum_{i=1}^{n} \alpha_i y_i = 0$$

Polynomial Kernel: $k(\mathbf{x}_1, \mathbf{x}_2) = (\mathbf{x}_1^{\mathrm{T}}\ \mathbf{x}_2 + 1)^{\mathrm{d}}$

RBF Kernel: $k(\mathbf{x}_1, \mathbf{x}_2) = \exp(-||\mathbf{x}_1 - \mathbf{x}_2||^2 / 2s^2)$

Github is a good choice for searching codes (https://github.com). Download the python code available in https://github.com/prashantkh19/MNIST_SVM. Adjust and modify the codes to implement two SVM models with Polynomial Kernel and RBF Kernel respectively on MNIST. Show the training process and classification results.

4. Analysis and Comparison **(30 marks)**

Analyse and compare the advantages and disadvantages of these three models in classification from different aspects, such as effectiveness, efficiency, complexity. Your analysis and conclusion should be well justified theoretically and/or empirically.

Hints:

Adjust the hyperparameters of all the three models to achieve as higher accuracies as possible based on cross-validation or certain reasonable criteria. The trained model parameters or training seeds should be provided for a quick verification of the experimental results. Moreover, you may consider to perform dimensionality reduction in order to attain better performance.

Dataset:

The MNIST dataset can be download from the website, http://yann.lecun.com/exdb/mnist/. This dataset is used for handwritten digits classification, which contains four files, training images, training labels, testing images and testing labels. Training set contains 60,000 examples and test set contains 10,000 examples. The original format of four files is binary. You can convert it to any file format you want.

Programming Requirements:

Students are strongly encouraged to use python to implement the classification tasks. Highly integrated libraries (e.g. pytorch, tensorflow) for deep learning are forbidden.

Marking Scheme:

70%-100% Essentially complete the tasks with sufficient and correct analysis.

60%-69% Shows understanding but contains a small number of errors or gaps.

50%-59% Clear evidence of a serious attempt at the work, showing some understanding, but with important gaps.

30%-49% Scrappy work, bare evidence of understanding or significant work omitted.

<30% No understanding or little real attempt made.