Indian School of Business
AMPBA Batch-20
Machine Learning Supervised Learning-2
Individual Assignment

1 GENERAL INSTRUCTIONS

- This is an **Individual Assignment** and has **45% weightage** in the Total Score.

1.1 Assignment Deliverables

- **A single Jupyter NB (.ipynb)** with
  - **a.** All four section questions solved.
  - **b.** Code executed,
  - **c.** Relevant plots generated.
  - d. Conclusions written. (After each question, provide a summary that explains/interprets the results. Ensure that summary should be very short).
  - e. Retain the outputs in the notebook and give proper points/comments/explanations for the code and output.

- **Assignment Submission form**.
  Note: The Assignment submission form should be submitted as well, as a separate copy. Your submission will not be considered without the Assignment Submission form being submitted.

1.2 Instructions for the assignment:

- All the Questions/Assignment details are available along with these instructions.
- Please follow the steps mentioned to solve the problems.
- The **honour code** for this submission is **3N-b**.
- Please look through the honour code restrictions carefully. There will be strong consequences for violating them.

1.3 Submission Guidelines

- Any **late submission will attract a penalty** as mentioned in the course outline.
- All the submissions must be made only on the LMS.
- **Email** submissions, **zip files** are **NOT** allowed.
- Code files rendered/exported as pdfs will strictly not be considered for evaluation.
- The files submitted must be **named** as "**Name -nn**" where nn is your PGID.
- Clearly mention each question number and sub-question number as comments
- Upload your submissions to the "**MLSL2 Assignment"** folder **on LMS**.

1.4 Please adhere to the given instructions, otherwise,

- your submission will not be accepted, or a severe penalty will be applied.

Due Date: 28th April 2024, 11:55 PM

2 DATASET

We will use the following dataset for the problems:

TMNIST Dataset

- Data set link : tmnst DATA SET.csv
- This dataset contains 26 characters (classes) in different fonts.
- We will work with all the 26 capital letters A to Z.
- Remove all columns where all values are zero.
- Normalize each pixel value from [0,1] range instead of [0,255] now.
- Please split each class into 70% train and 30% test split

PROBLEM 1 [20 points] Neural Network Classifier

- We will try different neural network architectures to build this model.
- **Input** = number of features in the data
- **Output** = 26 class classifier
- We will use **Soft-Max Activation on the output layer**
- This gives us a distribution as an output: P(c|**x**)
- We will try the following architectures:
    - **One hidden** layer with **5, 10, 20, 25** neurons
    - **Two hidden** layers with (5, 5), (5, 10), (10, 5), (10, 10) neurons in (layer 1, layer 2).
- For each architecture,
    - Compute the number of parameters (x-axis)
    - Compute the accuracy (y-axis)
- Submit the table with
    - Architecture (e.g. 1-5, 1-10, 1-20,1-25, 2-5-5, 2-5-10, ...) as column 1
    - Number of parameters as second column
    - Accuracy on test set as the third column
- Draw the scatter plot with column 1 and column 3.
- Do you see any trends?

PROBLEM 2 [20 points] SVM Classifier

Write an **svm_explore**(train_data, test_data, c1, c2) function that

- takes any **two classes** (c1, c2) out of 26 and does the following:

(1) **[5 points]** Builds the Linear SVM classifiers with C = 5 to 50 in increments of 5
- We will call these Linear/C=5, ..., Linear/C=50

(2) **[5 points]** Builds Polynomial SVM classifier with d = 2, d = 3, d = 4, d = 5 and C=10
- We will call these Poly/d=2, Poly/d=5 (keep c = 1 in these cases)

(3) **[5 points]** Builds the RBF classifier with sigma = 2 to 10 in increments of 2
- We will call these RBF/sigma=2, RBF/sigma=4, ...

Create a 4-column file:

- Column 1 = Name of the classifier above
- Column 2 = Number of support vectors
- Column 3 = Training accuracy of this classifier
- Column 4 = Test accuracy of this classifier

Plot the following:

- **[5 points]** Three plots one for each type of SVM with their training and test accuracies vs. complexity parameters (one for linear, polynomial, and RBF)
- **[5 points]** Plot the scatter plot between number of support vectors vs. training accuracy.
- Do you see any trends in the above two plots?

PROBLEM 3 [15 points] Random Forrest

Write an **random_forrest_explore**(train_data, test_data, c1, c2) function that

- takes any **two classes** (c1, c2) out of 26 and does the following:
- Pick any two classes of your choice. Keep data from only those two classes.
- Choose number of trees in the random forest from 5 to 100 in increments of 5
- Choose the max depth of the tree from 3 to 10 in increments of 1
- Plot train vs. test accuracy plots to show changes w.r.t. these two hyper-parameters
    a. Plot 1 shows depth on x axis with one line for training and one for test for different number of trees
    b. Plot 2 shows number of trees on x axis with one line for training and one for test for different depths
- Draw your conclusions about the tradeoff between these parameters on generalization.

PROBLEM 4 [15 points] Pair-wise Classifier

- For each pair of classes (1, 2), ..., (25, 26)
- Pick the top 30 Fisher dimensions that discriminate that PAIR of classes.
- Build a Linear SVM with a reasonable C value on this 30-dimensional data
- Generate the file with three columns:
    a. Class-1, Class-2, Validation-Set-Accuracy

- See if this makes sense for more “difficult” vs. “easy” pairs to classify.
- Also see if the top 30 for one pair of classes is same or different from the top 30 of another pair of classes.