Sri Sivasubramaniya Nadar College of Engineering, Kalavakkam – 603 110
(An Autonomous Institution, Affiliated to Anna University, Chennai)

UCS2612 Machine Learning Laboratory

Academic Year: 2023-2024 Even **Batch: 2021-2025**
Faculty In-charges: Y.V. Lokeswari & Nilu R Salim **VI Semester A & B**

A. No. : 4 . **Classification of Email spam and MNIST data using Support Vector Machines**

4.a. Download the Email spam dataset from the link given below:

https://www.kaggle.com/datasets/somesh24/spambase

The "spam" concept is diverse: advertisements for products/websites, make money fast schemes, chain letters, pornography. Our collection of spam e-mails came from our postmaster and individuals who had filed spam. Our collection of non-spam e-mails came from filed work and personal e-mails, and hence the word 'george' and the area code '650' are indicators of non-spam. These are useful when constructing a personalized spam filter. One would either have to blind such non-spam indicators or get a very wide collection of non-spam to generate a general purpose spam filter.

Develop a python program to classify Emails as Spam or Ham using Support Vector Machine (SVM) Model. Visualize the features from the dataset and interpret the results obtained by the model using Matplotlib library. **[CO1, K3]**

4.b. Download the MNIST dataset from the link given below:

https://archive.ics.uci.edu/dataset/683/mnist+database+of+handwritten+digits

THE MNIST DATABASE:
http://yann.lecun.com/exdb/mnist/

Kaggle:
https://www.kaggle.com/datasets/hojjatk/mnist-dataset/data

This is a database of 70,000 handwritten digits (10 class labels) with each example represented as an image of 28 x 28 gray-scale pixels.

Develop a python program to recognize the digits using Support Vector Machine (SVM) Model. Visualize the features from the dataset and interpret the results obtained by the model using Matplotlib library. **[CO1, K3]**

Use the following steps to do implementation:

1. Loading the dataset.
2. Pre-Processing the data (Handling missing values, Encoding, Normalization, Standardization).
3. Exploratory Data Analysis.
4. Feature Engineering Techniques.
5. Split the data into training, testing and validation sets.
6. Train the model.
7. Test the model.
8. Measure the performance of the trained model.
9. Represent the results using graphs.

Upload the code in GitHub and include the GitHub main branch link in the assignment PDF.

Hints to do the assignment:

Do the following:

1. Load the Email Spam (text) and MNIST (Image) data.
   For loading MNSIT Dataset use the following.

```
import numpy as np
from tensorflow.keras import datasets
(x_train, y_train), (x_test, y_test) = datasets.mnist.load_data()
```

2. Apply Image Pre Processing techniques (Image Enhancement, Smoothing, Filtering, denoising, segmentation and feature extraction – whichever is applicable.)
   Use **skimage** python library for processing images.
   Refer to : https://www.analyticsvidhya.com/blog/2023/02/lets-start-with-image-preprocessing-using-skimage/
   OR
   https://www.kaggle.com/code/rimmelasghar/getting-started-with-image-preprocessing-in-python

3. Exploratory Data Analysis – Draw Histogram Equalization
   Refer to https://docs.opencv.org/4.x/d5/daf/tutorial_py_histogram_equalization.html
4. Build the SVM model with different kernel functions (use linear, rbf, polynomial, sigmoid)
   Refer to below resources:
   https://scikit-learn.org/stable/modules/svm.html
   https://www.datacamp.com/tutorial/svm-classification-scikit-learn-python
   https://www.geeksforgeeks.org/implementing-svm-and-kernel-svm-with-pythons-scikit-learn/
   https://www.educative.io/answers/how-to-implement-svm-in-python-using-scikit-learn

5. Obtain the Training and Test Accuracy.
6. Compare the results of using different kernel functions. Write your inference about the results.
7. Upload python project in GitHub and explore all git commands. Git Commands Tutorial : https://git-scm.com/docs/gittutorial

   Upload VSCode to GitHub
   https://www.youtube.com/watch?v=vRxfnHtCxEo

   Additional Reference:
   https://www.youtube.com/watch?v=ixszqWHYmC0
   or
   https://www.youtube.com/watch?v=JrCC66R_EhQ

4.c. Classification of Email Spam or Ham using Naïve Bayes Algorithm

Build the Naïve Bayes model for the classification of Email as Spam or Ham.
Refer to following resources
https://scikit-learn.org/stable/modules/naive_bayes.html
https://www.datacamp.com/tutorial/naive-bayes-scikit-learn

https://www.tutorialspoint.com/how-to-build-naive-bayes-classifiers-using-python-scikit-learn
https://www.analyticsvidhya.com/blog/2021/11/implementation-of-gaussian-naive-bayes-in-python-sklearn/

Upload the code in GitHub and include the GitHub main branch link in the assignment PDF.

Hints to do the assignment:

Upload python project in GitHub and explore all git commands. Git Commands Tutorial : https://git-scm.com/docs/gittutorial

Upload VSCode to GitHub
https://www.youtube.com/watch?v=vRxfnHtCxEo

Additional Reference:
https://www.youtube.com/watch?v=ixszqWHYmC0
or
https://www.youtube.com/watch?v=JrCC66R_EhQ