ARIZONA STATE UNIVERSITY

Data Mining

Mini Project 2

Submitted By:

Komal Pandhariwal (1211045521)

Utkarsh Pandey

(1211230069)

Diptanshu Purwar

(1210865965)

Question 1
Problem Statement:

The VidTIMIT dataset consists of video and audio recordings of 43 subjects reciting short sentences. In this assignment, we will use a subset of the dataset with 25 subjects. We will also use only the video modality. The videos were sliced into images and the discrete cosine transform function was used to extract feature vectors of dimension 100 from each image. The training set contains 3,500 samples and the test set contains 1,000 samples. Our objective is to recognize a subject from a given image. Train an SVM classifier with a polynomial kernel with parameter 2 on the training set and test on the test set. You need to train one SVM for each class; for predicting a test sample, use the maximum of the values returned by all the SVMs to decide the final class. Report the percentage accuracy on the test set

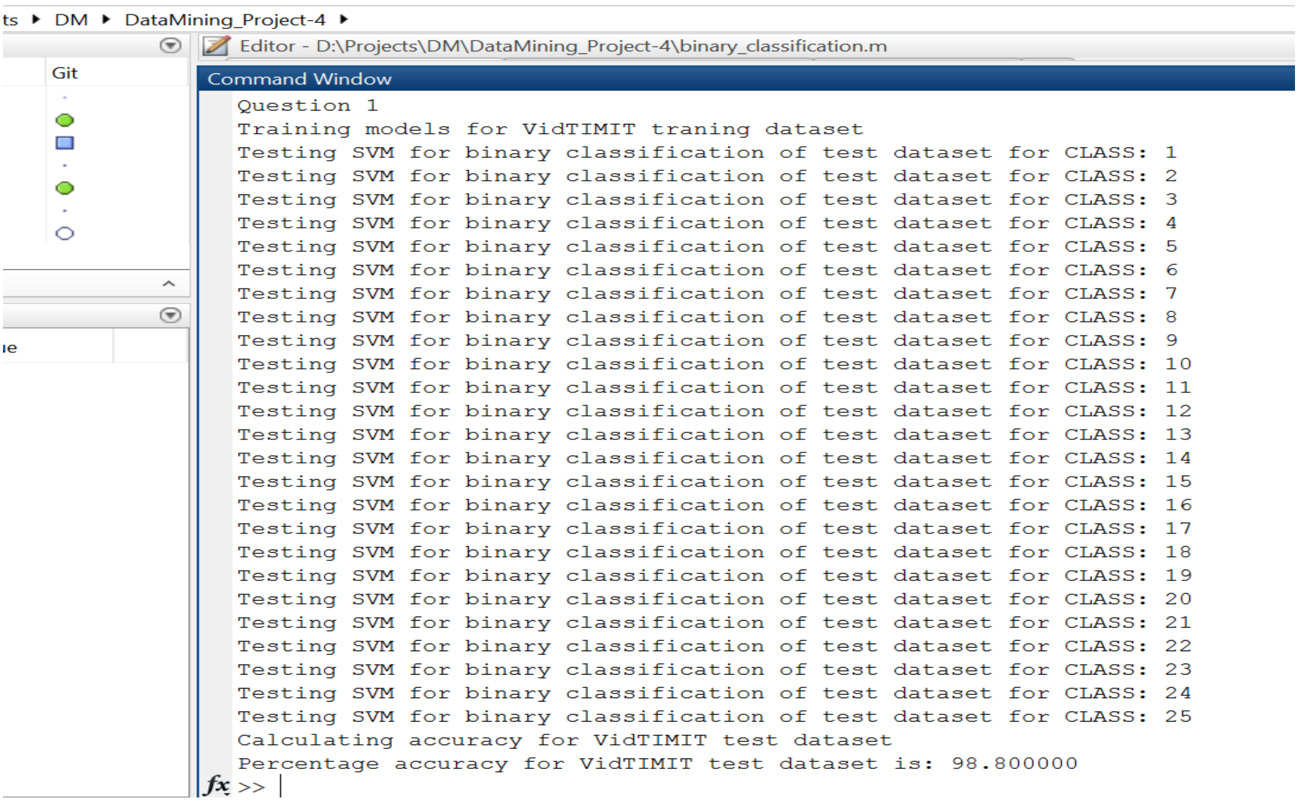
Assumption:

- The data set given for testing and training is clean.
- Every test sample belongs to only 1 class.

Steps:

- Read X_train, y_train, X_test and y_test files from the given VidTIMIT dataset
- Train the svm models using the given training samples in X_train, one by one for each class (25 classes) in y_train using **‘fitcsvm’** method.
- Parameters given to the fitcsvm method are **‘Kernel function**’ and **‘Polynomial of Order 2’**
- Use ‘**predict**’ function to test samples in X_test on the trained SVM model for each of those classes
- Make a final array that stores class label for every observation in X_test.
- Compare the predicted labels with actual labels in y_test and calculate percentage accuracy for VidTIMIT test dataset

Output:
Percentage = 98.8%



Question 2
Problem Statement:

The previous problem is an example of a multi-class classification problem, where there are multiple classes in the dataset, but each data sample can belong to only one class. Multi-label classification is a generalization of multi-class classification, where each data sample can belong to multiple classes simultaneously. For instance, consider the problem of classifying an outdoor image of a scene. Suppose the possible classes are beach, mountain, field and sunset. It is possible for a particular image to contain both beach and mountain or beach, mountain and sunset all together. The objective of multi-label learning is to predict all the classes present in a data sample. The Scene dataset consist of 2407 images of an outdoor scene, where each image is represented by a feature vector of dimension 294. Also, there are 6 classes in the problem and an image can belong to one or more of the 6 classes. The dataset has been divided into a training set (with 1500 samples) and a test set (with 907 samples). Each row of X_train and X_test denotes a sample and each column denotes a feature. Each row of y_train (y_test) denotes the labels of the corresponding training (testing) sample, where 1 means the class is present and 0 means the class is absent. For instance, in the training set, sample 462 belongs to classes 4 and 5. One strategy to solve a multi-label learning problem is to train an SVM separately for each class. To predict a test sample, each SVM is applied separately on it. A positive output indicates that the corresponding class is present and a negative output indicates that it is absent. The accuracy is computed using the following expression:

$$A = \frac{|T \cap P|}{|T \cup P|}$$

where T is the true class label vector of a test sample and P is the predicted class label vector. Train an SVM classification model on the training set and test on the test set. Report the percentage accuracy on the test set using the following classification models: (i) SVM with polynomial kernel with parameter 2 and (ii) SVM with Gaussian kernel with parameter 2.

Steps:

- Read X_train, y_train, X_test and y_test files from the given Scene dataset.
- Train the SVMs for samples in X_train using the **'fitcsvm'** method for each class (6 classes) in y_train.
- Use the fitcsvm method with the following parameters:
  - For Polynomial kernel:
    - Polynomial order '2'
    - Kernel Scale as 'auto'
  - For Gaussian kernel:
    - Kernel Scale as 'auto'
    - BoxConstraint as 1.
- Predict the labels for the Scene_data test samples in X-test using **'predict'** function, by using trained models for each class.
- Compare the predicted classes with actual classes in y_test using

$$A = \frac{|T \cap P|}{|T \cup P|}$$

- Calculate percentage accuracy for predicted labels for test dataset

Output:
Accuracy for Polynomial = 65.45%
Accuracy for Gaussian=64.7%

Projects ▸ DM ▸ DataMining_Project-4 ▸

Editor - D:\Projects\DM\DataMining_Project-4\multilabel_classification.m

Command Window

```
Question 2
 Enter 1: Polynomial 2: Gaussian1
Polynomial kernel selected
Training models for Scene_Data traning dataset
Testing SVM for binary classification of test dataset for CLASS: 1
Testing SVM for binary classification of test dataset for CLASS: 2
Testing SVM for binary classification of test dataset for CLASS: 3
Testing SVM for binary classification of test dataset for CLASS: 4
Testing SVM for binary classification of test dataset for CLASS: 5
Testing SVM for binary classification of test dataset for CLASS: 6
Calculating accuracy for Scene_Data test dataset
Percentage accuracy for Scene_Data test dataset is: 65.453877
>> multilabel_classification
Question 2
 Enter 1: Polynomial 2: Gaussian2
Gaussian Kernel Selected
Training models for Scene_Data traning dataset
Testing SVM for binary classification of test dataset for CLASS: 1
Testing SVM for binary classification of test dataset for CLASS: 2
Testing SVM for binary classification of test dataset for CLASS: 3
Testing SVM for binary classification of test dataset for CLASS: 4
Testing SVM for binary classification of test dataset for CLASS: 5
Testing SVM for binary classification of test dataset for CLASS: 6
Calculating accuracy for Scene_Data test dataset
Percentage accuracy for Scene_Data test dataset is: 64.700478
>> multilabel_classification
Question 2
 Enter 1: Polynomial 2: Gaussian3
Enter the correct value
fx >>
```

Value
2