EE5904/ME5404 Part II

Project 1: SVM for Classification of Spam Email Messages

Project Description and Requirement

Dr. Peter C. Y. Chen
Associate Professor
Department of Mechanical Engineering
National University of Singapore
Email:mpechenp@nus.edu.sg

Report due on 26 April 2024, 23:59 Singapore time

I. Objective

This project is designed for the student to demonstrate (through independent learning):

1. Competence in implementing SVMs, and
2. Understanding of the principles and issues of SVM for classification.

II. Data

The data used in this project is the Spam Data Set[1], which contains a total of 4,601 examples. Each example has a feature vector with 57 attributes that represent the selected key features of an email message, and a label indicating whether the associated email message is spam or not. (Detailed description of these attributes can be found on the source webpage of the dataset.) One feature vector is shown below for illustration:

```
0.00000 0.01043 0.00000 0.00000 0.00000 0.00000 0.00000 0.00000
0.00000 0.00000 0.01043 0.01043 0.02105 0.00000 0.00000 0.00000
0.00000 0.03166 0.06332 0.00000 0.02105 0.00000 0.00000 0.00000
0.00000 0.00000 0.00000 0.00000 0.00000 0.00000 0.00000 0.00000
0.00000 0.00000 0.00000 0.00000 0.00000 0.00000 0.00000 0.00000
0.00000 0.00000 0.00000 0.00000 0.00000 0.00000 0.00000 0.00000
0.00000 0.00000 0.00000 0.00196 0.00000 0.00000 0.02601 0.12811
0.98827
```

For this project, three sub-datasets, namely, the *training* set (with the filed name train.mat), the *test* set (test.mat), and the *evaluation* set (eval.mat), have been created from the Spam Data Set. They are in the MATLAB MAT-file format. The training set and the test set are included in the zipfile that also contains this document. In these two MAT-files, the feature vectors are held in a variable with the name: <file_name_without_extension>_data, while the labels (either "+1" for spam or "−1" for non-spam) associated with the individual feature vectors are held in a variable with the name: <file_name_without_extension>_label. Thus in train.mat, the two variables are *train_data* and *train_label*. Similarly, in test.mat the variables are *test_data* and *test_label*.

The third sub-dataset, namely, eval.mat, is formed using a subset of the remaining examples (after train.mat and test.mat have been chosen) in the Spam Data Set. This third dataset (not included in the zipfile) will be used for the assessment of the program that you will submit, as described in Section V.

[1]http://archive.ics.uci.edu/ml/datasets/spambase.

III. Requirement

A. What to be done

The main tasks involved in this project are:

Task 1: Write a MATLAB (M-file) program to compute the discriminant function $\mathbf{g}(\cdot)$, if one exists, for the following SVMs, using the training set provided:

(i) A hard-margin[2] SVM with the linear kernel

$$\mathbf{K}(\mathbf{x}_1, \mathbf{x}_2) = \mathbf{x}_1^T \mathbf{x}_2 \tag{1}$$

(ii) A hard-margin SVM with a polynomial kernel

$$\mathbf{K}(\mathbf{x}_1, \mathbf{x}_2) = \left(\mathbf{x}_1^T \mathbf{x}_2 + 1\right)^p \tag{2}$$

where the values of p are listed in Table 1.

(iii) A soft-margin SVM with a polynomial kernel as given in Equation (2) above, and with the values for p and C as listed in Table 1.

Note that a MATLAB function quadprog (available in the Optimization Toolbox) can be used to solve constraint optimization problems.

Task 2: Write a MATLAB (M-file) program to implement the SVMs with the discriminant functions obtained in **Task 1**. Apply these SVMs to classify the given training set and test set, and report the classification results in Table 1 by filling the entries indicated by "?". Discuss the results and their implications, including issues related to the admissibility of the kernels and the existence of optimal hyperplanes for the three types of SVMs listed in **Task 1** above.

Task 3: Design a SVM of your own. This SVM can be one of the three types specified in **Task 1** above (i.e., hard-margin with linear kernel, hard-margin with polynomial kernel, and soft-margin with polynomial kernel), or one with your own choice of kernel. Using the given training set, compute the discriminant function $\mathbf{g}(\cdot)$ of the SVM. Implement the resulting SVM in a MATLAB M-file program. This program will be used to classify the evaluation set as part of the assessment discussed in Section V.

[2]From an implementation perspective, a hard-margin SVM can be approximated by a soft-margin SVM with a very large C value.

Type of SVM	Training accuracy				Test accuracy			
Hard margin with Linear kernel	?				?			
Hard margin with polynomial kernel	$p = 2$	$p = 3$	$p = 4$	$p = 5$	$p = 2$	$p = 3$	$p = 4$	$p = 5$
	?	?	?	?	?	?	?	?
Soft margin with polynomial kernel	$C = 0.1$	$C = 0.6$	$C = 1.1$	$C = 2.1$	$C = 0.1$	$C = 0.6$	$C = 1.1$	$C = 2.1$
$p = 1$	?	?	?	?	?	?	?	?
$p = 2$	?	?	?	?	?	?	?	?
$p = 3$	?	?	?	?	?	?	?	?
$p = 4$	?	?	?	?	?	?	?	?
$p = 5$	?	?	?	?	?	?	?	?

TABLE I: Results of SVM classification.

B. What to submit

1. A report (in a PDF file) describing the implementation and the results. It must contain a cover page showing:
   (i) *student's name,*
   (ii) *student number,*
   (iii) *student's email address,*
   (iv) *name of course, and*
   (v) *project title.*

   The report should be in PDF format and no more than ten pages (excluding the cover page). The name of the PDF file must be in the format:

   StudentNumber_SVM.pdf

2. The M-file programs as specified in the description of **Task 1**, **Task 2**, and **Task 3** in Section III-A above.

C. How to submit

Only softcopy of the report (in PDF) and the MATLAB M-file programs are to be submitted. Please put the report and the M-file programs in a folder. Use your student number as the folder name. Generate a non-password-protected zipfile of this folder (again, with your student number as the filename of the zipfile) and upload this zipfile onto CANVAS in the folder *Part 2: SVM project report submission*, under the *Assignments* section of the course *EE5904/ME5404*. Make sure to upload your report and code into the correct folder as specified above.

IV. Demo Session

A demo session will be conducted by the teaching assistant on the use of MATLAB for implementing the required SVMs. Please check the lecture slides about this demo.

V. Assessment

The project will be assessed based on the following criteria:

1. *Discussion* (with supporting argument) on the results of classifications reported in Table 1. Specifically, it is suggested that (at least) the following items be covered:
   - Data pre-processing,
   - Admissibility of the kernels,
   - Existence of optimal hyperplanes,
   - Table 1 - Comments on results (with supporting arguments), and
   - Task 3 – Discussion on design decisions.
2. *Presentation*. This includes good report style, clarity, and conciseness.
3. *Performance of your SVM M-file program for* **Task 3** *(as described in Section III-A) in classifying the evaluation set* eval.mat. Your M-file program must be workable in MATLAB under the Windows environment. During the assessment, eval.mat is first loaded into MATLAB, and your M-file program will then be run in MATLAB. Thus, before your M-file is loaded into MATLAB, the MATLAB workspace will have the variable "eval_data" that holds a 57×600 matrix, in which each column represents one feature vector. Your M-file program must be able to process these 600 features vectors and generate (as output) a vector with the name eval_predicted, whose n^{th} element is the computed label of the n^{th} feature vector (i.e., the n^{th} column) in eval_data. When writing your M-file program, you can test its execution on your own by making up an eval.mat file containing dummy sample values.