Optimization Methods for Machine Learning - Fall 2018

PROJECT # 2
Support Vector Machines

Laura Palagi

Posted on November 23, 2018 - due date December 22, 2018

Instructions

Homework will be done in groups formed by 1 to 3 people: each group must hand in their own answers. We will be assuming that, as participants in a graduate course, every single student will be taking the responsibility to make sure his/her personal understanding of the solution to any work arising from such collaboration.

Homework must be sent by an email both to the teaching assistant Ing. Ruggiero Seccia (ruggiero.seccia@uniroma1.it) and to laura.palagi@uniroma1.it with subject **[OMML-2018] Project 2**. After you submit, you will receive an acknowledgement email that your project has been received. If you have not received an acknowledgement email within 2 days after you submit then contact the instructors.

The mail must contain as attachment a .zip or .tar.gz file with both a typed report in English and the source code following instructions in the text of the project. **The report must be of at most 4 pages excluded figures that must be put at the end.**

Evaluation criteria

The grade are Italian style namely in the range [0,30], being 18 the minimum degree to pass the exam.
You may reach the max score by correctly answering to all questions using the DATASET 2 (handwritten numerical digits). Correct answers to all the points of Question 1 using the DATASET 1 (two dimensional set) allow to obtain only up to 26.
Homework is due at latest at midnight on the due date. For late homework, the score will be decreased. It is worth 85% for the next 48 hours. It is worth 70% from 48 to 120 hours after the due date. It is worth 50% credit after 120 hours delay.
The second homework accounts for 35% of the total vote of the exam.
For the evaluation of the first homework the following criteria will be used:

1. 60% check of the implementation
2. 40% quality of the overall project as explained in the report.

In this assignment you will implement optimization methods for training Support Vector Machines (supervised learning) applied to classification problems.

Data sets.

You may choose among two different data sets (DATASET 1 = DS1 and DATASET 2 = DS2). The choice of the data set influences the score. Indeed each data set has a difficulty coefficient DC_i (DC_1=$\frac{13}{15}$ and DC_2=1) and each question has a maximum score (MSQ_j). The maximum score of each question is given by $DC_i \times MSQ_j$.

In particular the maximum score obtainable by using DS1 is 26 ($\frac{13}{15} \times 30$).

- DATA SET 1 (difficulty coefficient $DS_1 = \frac{13}{15}$) The two-dimensional input samples are reported in the table below. Training set is made up of the pairs (x^i, y^i) with $x^i \in \mathbb{R}^2$ and $y^i \in \{-1, 1\}$. (Hint plot the points)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
x_1	2.	2.2	1	3.5	3.5	4.5	.5	2.	8.	8.	5.5	4	4	7.5	9.5
x_2	5	3	7	3.5	5	5	5.	10.	2.5	4	2	7.5	1.5	6.5	4
y	-1	-1	-1	-1	-1	-1	1	1	1	1	1	1	1	1	1

- DATA SET 2 (difficulty coefficient DS_2=1) Classification of normalized handwritten digits, automatically scanned from envelopes by the U.S. Postal Service. The images here have been deslanted and size normalized, resulting in 16 x 16 grayscale images (Le Cun et al., 1990).

  The training data are available as separate files per digit (0 1 2 3 4 5 6 7 8 9). Each line consists of the 256 grayscale values. You are receiving only the data set for the numbers 2 8.

  The training and test observations are distributed as follows:

number	0	1	2	3	4	5	6	7	8	9	Total
Train	1194	1005	731	658	652	556	664	645	542	644	7291
Test	359	264	198	166	200	160	170	147	166	177	2007

  The test set can be notoriously "difficult", and a 2.5% error rate is excellent.

Question 1. Consider a nonlinear SVM with kernel $k(\cdot,\cdot)$ and determine the nonlinear decision function

$$y(x) = sign\left(\sum_{p=1}^{P} \alpha_p y^p k(x^p, x) + b\right)$$

where α, b are obtained as the optimal solution of the dual nonlinear SVM problem.

As kernel function you may choose either a RBF kernel

$$k(x,y) = e^{-\gamma\|x-y\|^2}$$

where γ is an hyper parameter or a polynomial kernel

$$k(x,y) = (x^T y + 1)^p$$

with hyper parameter $p \geq 1$.

Answer to the following questions.

1. **(max score up to "DC$_i\times$ 24")** Write a program which implements the dual quadratic problem and use a standard QP algorithm for its solution (you must use a routine which uses the gradient of the objective function).
2. **(max score up to "DC$_i\times$27")** Write a program which implements a decomposition method for the dual quadratic problem with any value $q \geq 2$ (use the same kernel and parameter setting as in Question 1). You must define the selection rule of the working set, construct the subproblem at each iteration and uses a standard QP algorithm for its solutions; the gradient of the objective function of the subproblem $\nabla_{\alpha_W} f(\alpha)$ must be available to the routine. A stopping criterion on optimality condition must be implemented.
3. **(max score up to "DC$_i\times$30")** Fix the dimension of the subproblem $q = 2$ and implement a most violating pair (MVP) decomposition method which uses the analytic solution of the subproblems.
4. **Additional bonus exercise. Score up to 30 cum laude (or 1 point bonus)**
   Consider a three class problem made up by the numbers 1, 2, 8. Implement any SVM strategy for multiclass classification (one against one or one against all).

In the report you must state:

- the chosen data set.
- the chosen kernel;
- the final setting for the hyper parameter C and of the hyper parameter of the kernel in Question 1; how do you choose them and if you can identify values that highlight over/under fitting;
- For each Question
  - which optimization routine do you use for solving the quadratic minimization problem and the setting of its parameters, if any (both for Question 1 and Question 2);
  - machine learning performance: report the value of the accuracy on the training and test set;
  - optimization performance: report the number of iterations, the number of function /gradient evaluations, the initial and final value of the objective function of the dual problem

Further the comparison among all the implemented methods - in term of accuracy in learning and computational effort in training - must be gathered into a final table (a minimal sample below)

Ex	settings		Training	test	optimization performance	
	$C=?$	$\gamma/p=?$	error	error	its	number of function eval.
Q1 Full QP						
Q2 Decomposed QP						
Q3 MVP						

Instructions for python code

For each question you must provide a file called `run_i.py` for $i = 1,...,4$ which must print:

- classifcation rate on the training set (% instances correctly classified)
- classifcation rate on the test set
- time for finding the KKT point
- number of optimization iterations
- value of γ

Furthermore, in question 2 and 3 you must print also:

- value of q (not needed for question 3)
- diffenrece between $m(\alpha)$ and $M(\alpha)$

In question 4 please print also which strategy have you implemented (one against one/all OAO/OAA) and optimization time and iterations must be the sum of the two optimization processes.

You can find more information on the Excel file