Erasmus Mundus Master in Data Mining and Knowledge Management

Optimization: Laboratory assignment # 2

Support Vector Machines.

Yue He and Francisco Rodríguez – ULY2

Part 1. The model file.

```
param n; #  The size of our training set.

param c; # the tradeoff parameter;

param x{i in{1..n}, j in{1..4} }; # The value of the features.

param y{i in{1..n}}; #  The label of the class.

var w{i in{1..4}}; # The normal vector to the plane.

var b; # The offset of the plane.

var s{i in{1..n}} ; # % the slacks

# objective function

minimize obj: 1/2 * sum{i in{1..4}}( w[i]^2 ) + c * sum{i in {1..n}}
s[i] ;


subject to c1 {i in {1..n}}:  y[i] * ( sum{j in {1..4}} ( w[j] *
x[i,j] ) + b ) >= 1 - s[i];
subject to c2 {i in{1..n}}: s[i]>=0; # The slacks must be
non-negative.
```

Part 2. The data file.

```
data;
param n := 100;

param c:= 100; # We changed this and we tried C=.1,1,10,100,1000000

param x : 1 2 3 4 :=
1     0.558 0.497 0.246 0.582
2     0.473 0.408 0.433 0.093
3     0.938 0.374 0.129 0.71
4     0.306 0.879 0.146 0.512
5     0.745 0.862 0.006 0.862
... #  For the sake of saving space, we do not show every line. The
# missing lines are the rest of the points in the data set.
99    0.011 0.882 0.967 0.47
100   0.768 0.995 0.398 0.629;

param y :=
1     -1
2     -1
3     1
4     -1
5     1
... #  For the sake of saving space, we do not show every line. The
# missing lines are the rest of the points in the data set.
99    1
100   1;
```

Part 3. The solution obtained and tests.

Training

When training the model, different values of the trade-off parameter ' ν where used. ' ν is the trade-off parameter in this equation:

$$\min_{(w,\gamma,y)\in\mathbb{R}^{n+1+m}} \quad \nu e^T y + \tfrac{1}{2}\|w\|_2^2$$
$$\text{subject to} \quad D(Aw - e\gamma) + y \geq e, \quad y \geq 0,$$

This was done to observe how the value of this parameter affects the results of our model. Five values were used: 0.1,1,10,100 and 1000.000. A training set of size 100 was considered enough after trial and error. Here are the results obtained:

v	Number of iterations	Value of the objective function	Value of the variables	Number of badly classified points (out of 100).
0.1	70	7.602530905	w= 0.66,0.84,0.52,0.75 ; b = -1.95	86
1	111	42.8696011	w= 2.25, 2.44,1.97,2.32 ; b = -4.69	51
10	120	258.0667544	w= 5.30, 4.88,4.33,4.97 ; b = -9.89	29
100	127	2003.441667	w= 7.18, 6.49, 6.04, 6.22 ; b = -13.08	20
1000000	128	19066707.32	w=8.29, 7.38, 7.12, 7.26; b = -15.28	16

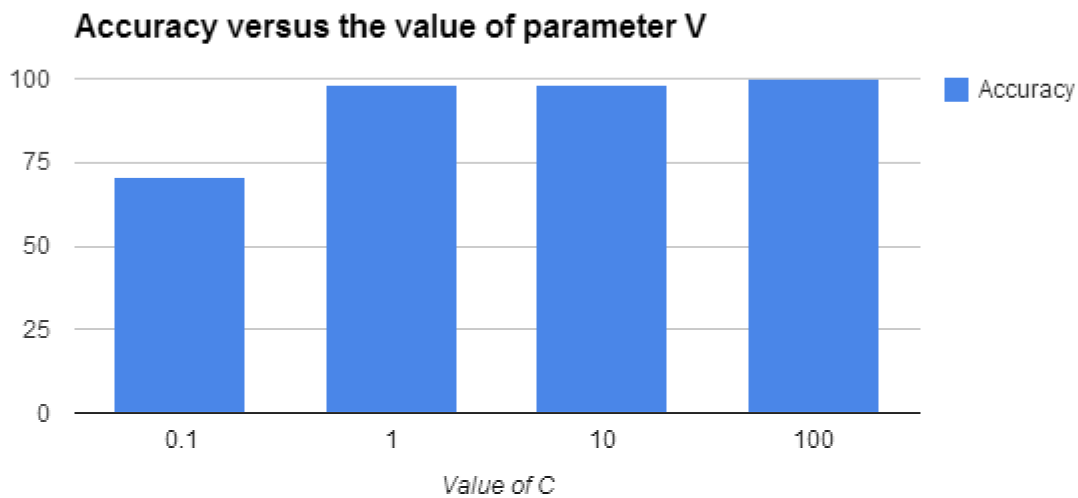
It can be said that as v grew bigger, the model predicted less badly classified points and had a higher objective function. This means that the model is better as v is bigger. The value of the objective function is bigger because v multiplies the slacks in the objective function. However, this does not affect the quality of the model.

Test

To test the model, a test set of size 48 was used. A script in Freemat was written to find the accuracy of the model. The accuracy was defined as the percentage of well classified points. Below are the results found:

V	Accuracy
0.1	70.833
1	97.92
10	97.92
100	100
1000000	100

A graph was made to illustrate the effect of C in the accuracy:



It can be seen that as v grows bigger a higher accuracy is obtained. With a very high v the accuracy of the model is very satisfactory: 100%.

This is the script used to test the model:

```
%% test
function [accuracy, predict_class]= test_svm(x,y,w,b)
right_num = 0;
n = size(x,1);
m = size(x,2);
for i = 1: n
      t = 0;
predict_class(i) = -1;
      for j = 1: m
            t = t + w(j)* x(i,j);
      end
      predict_y(i) = t + b;
if predict_y(i) >= 0
predict_class(i) = 1;
      end
```

```
if predict_class(i) == y(i)
right_num = right_num +1;
end
end
accuracy = right_num / n;
```

Examining the values of the functional margin or the geometrical is an alternative way to measure how good a prediction is. These values measure how far the points are from the boundary line, so they give an intuition of how confident the prediction of a particular data point is. Nonetheless, they give information about a particular prediction. This is why accuracy, defined as the percentage of good predictions, was used to determine how good the model obtained was.

Part 4. Comments.

The effect of the value v has already been commented: the bigger the value of v, the better . There are some other relevant comments.

Before building this model, another was made under the assumption that the points were linearly separable. However, to do this the badly classified points in the training data had to be eliminated so that the assumption that the points were linearly classifiable could hold.The model obtained is an instance of the final model used. The main difference is that in the model that assumes that the elements are linearly separable the slacks are set to 0. This model was tested and an accuracy of 100% was achieved. Of course that the final model, the model that does not assume that they are linearly separable, can also be used in case the points are linearly separable. In fact, in general it is not good to assume that the elements are linearly separable, so it is better not to assume that the slacks are equal to zero.