FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

Machnine Learning (PDEEC0049 : 15-782PP)
Homework 4

António Damião das Neves Rodrigues (200400437 : 700098386)

February 20, 2014

1 PROBLEM 1

1.1

Starting from the dual representation for both $k_1(\mathbf{x}_1;\mathbf{x}_2)$ and $k_2(\mathbf{x}_1;\mathbf{x}_2) = 1 + k_1(\mathbf{x}_1;\mathbf{x}_2)$ we have expressions 1.1 and 1.2 respectively:

$$\arg\max_{\lambda} \sum_{n=1}^{N} \lambda_n - \frac{1}{2} \sum_{n=1}^{N} \sum_{m=1}^{N} \lambda_n \lambda_m y_n y_m \lambda_n k_1(\mathbf{x}_1;\mathbf{x}_2)$$

$$\begin{aligned} &\text{subject to} \\ &0 \le \lambda_n \le C \\ &\sum_{n=1}^{N} \lambda_n y_n = 0 \end{aligned} \tag{1.1}$$

$$
\begin{aligned}
&\arg\max_{\lambda} \sum_{n=1}^{N} \lambda_n - \frac{1}{2}\sum_{n=1}^{N}\sum_{m=1}^{N} \lambda_n \lambda_m y_n y_m \lambda_n k_2(\mathbf{x}_1;\mathbf{x}_2) \\
&\arg\max_{\lambda} \sum_{n=1}^{N} \lambda_n - \frac{1}{2}\sum_{n=1}^{N}\sum_{m=1}^{N} \lambda_n \lambda_m y_n y_m \lambda_n (1 + k_1(\mathbf{x}_1;\mathbf{x}_2)) \\
&\arg\max_{\lambda} \sum_{n=1}^{N} \lambda_n - \frac{1}{2}\sum_{n=1}^{N}\sum_{m=1}^{N} \lambda_n \lambda_m y_n y_m \lambda_n k_1(\mathbf{x}_1;\mathbf{x}_2) - \frac{1}{2}\sum_{n=1}^{N}\sum_{m=1}^{N} \lambda_n \lambda_m y_n y_m
\end{aligned} \tag{1.2}
$$

subject to

$0 \le \lambda_n \le C$

$\sum_{n=1}^{N} \lambda_n y_n = 0$

The quadratic programming optimization problems in expressions 1.2 and 1.2 differ on to the term $-\frac{1}{2}\sum_{n=1}^{N}\sum_{m=1}^{N} \lambda_n \lambda_m y_n y_m$. As one of the problem constraints is $\sum_{n=1}^{N} \lambda_n y_n = 0$, the term will be reduced to 0 during the solving process, and therefore the solutions will be essentially the same.

1.2

$$
\begin{aligned}
K(x,y) &= \phi_\infty(x).\phi_\infty(y) = \sum_{i=0}^{\infty} \frac{e^{-x^2/2} x^i}{\sqrt{i!}} \frac{e^{-y^2/2} y^i}{\sqrt{i!}} \\
K(x,y) &= e^{-x^2/2} e^{-y^2/2} \sum_{i=0}^{\infty} \frac{x^i y^i}{i!}
\end{aligned}
$$

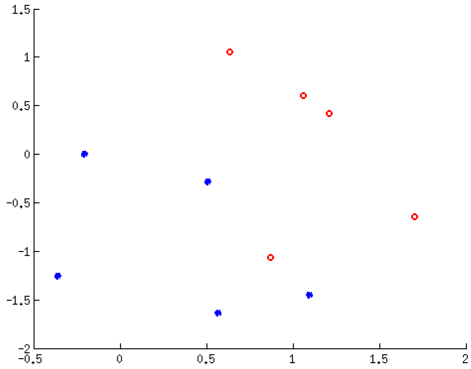
where $\sum_{i=0}^{\infty} \frac{x^i y^i}{i!}$ is the Maclaurin series expansion for the exponential function [1] e^{xy}, i.e. $\sum_{i=0}^{\infty} \frac{x^i y^i}{i!} = e^{xy}$. Therefore we have:

$$
\begin{aligned}
K(x,y) &= e^{-x^2/2} e^{-y^2/2} \sum_{i=0}^{\infty} \frac{x^i y^i}{i!} \\
K(x,y) &= e^{-x^2/2} e^{-y^2/2} e^{xy} \\
K(x,y) &= e^{-x^2/2 - y^2/2 + xy} \\
K(x,y) &= e^{\frac{-x^2+2xy-y^2}{2}} \\
K(x,y) &= e^{\frac{-(x-y)^2}{2}}
\end{aligned}
$$

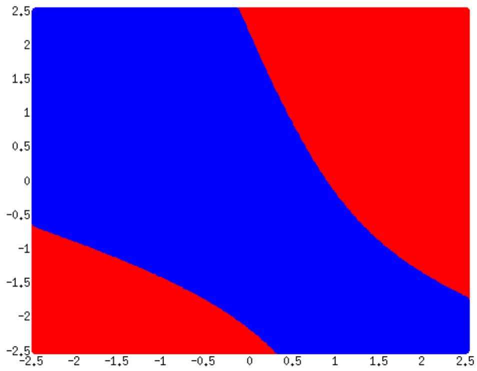
2 Problem 2

Use the MATLAB code given as attachment (`problem2` folder) files `basicSVM.m` for implementation details. Follow the comments on the code for details and reasoning. The MATLAB file `testSVM.m` was used for testing the implementations.

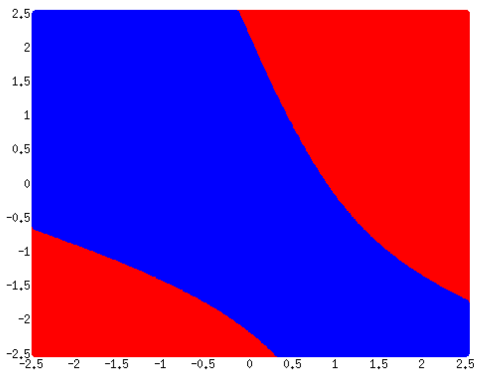
Figures 2.1 to 2.2 compare the results obtained with the functions `svmtrain()` and `svmpredict()` of `libsvm-mat-2.91-1` and those obtained with the implemented `basicSVM()` for different values of training data size T and a constraint value $C = 10$.



(a)

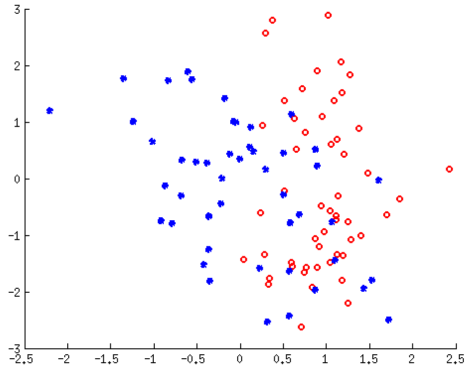


(b)

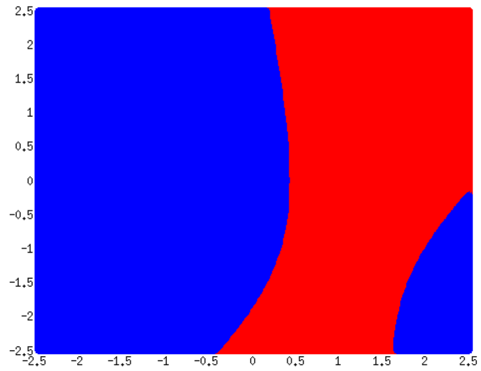


(c)

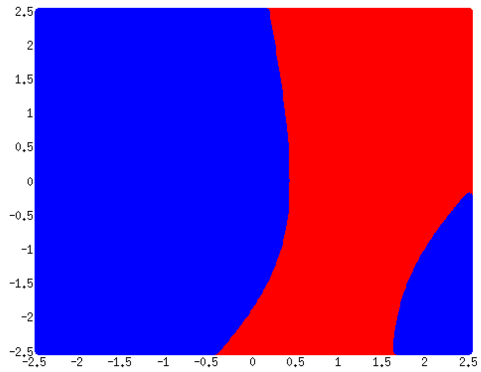
Figure 2.1: Results of SVM classifiers, for $T = 10$ and $C = 10$: training dataset (a), third-party `libsvm-mat-2.91-1` (b), `basicSVM()` (c).



(a)



(b)



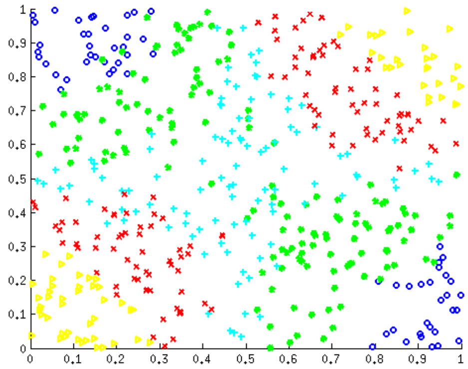
(c)

Figure 2.2: Results of SVM classifiers, for $T = 100$ and $C = 10$: training dataset (a), third-party `libsvm-mat-2.91-1` (b), `basicSVM()` (c).

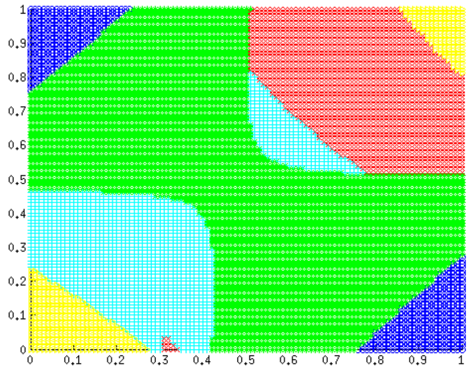
3 PROBLEM 3

Use the MATLAB code given as attachment (`problem3` folder), files `SVM_DAG.m`, `basicSVMtrain.m` and `basicSVMpredict.m` for implementation details. Follow the comments on the code for details and reasoning. The MATLAB file `testDAG.m` was used for testing the implementations.

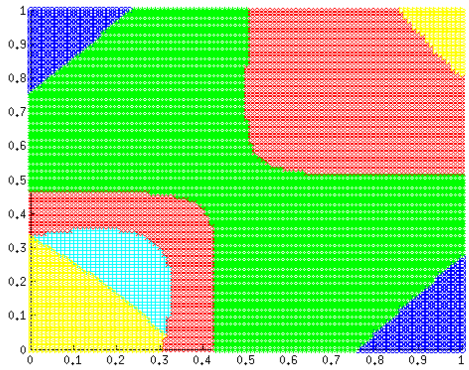
Figures 3.1 to 3.2 compare the results obtained with the functions `svmtrain()` and `svmpredict()` of `libsvm-mat-2.91-1` and those obtained with the implemented `basicSVM()` for different values of training data size T and $C = 1$.



(a)

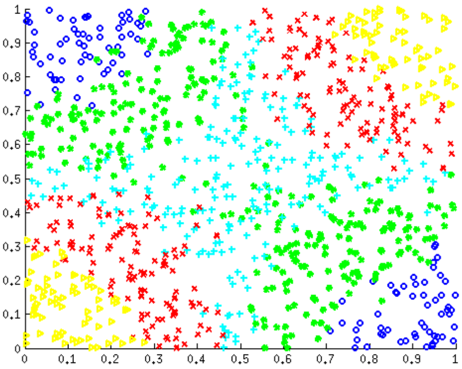


(b)

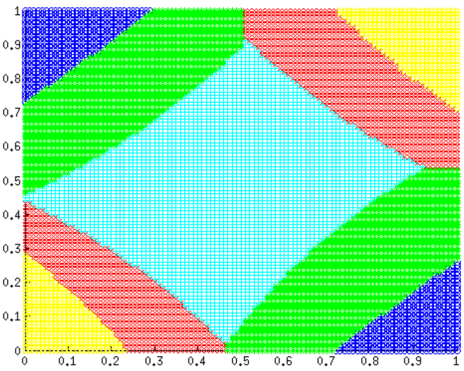


(c)

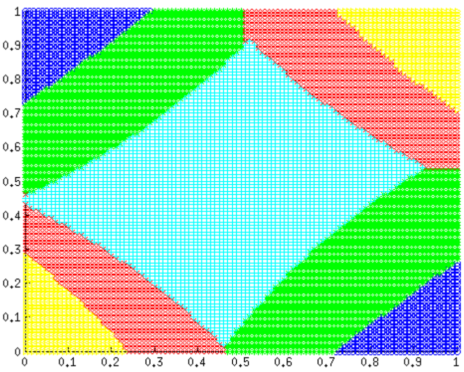
Figure 3.1: Results of SVM classifiers, for $T = 500$ and $C = 1$: training dataset (a), third-party `libsvm-mat-2.91-1` (b), `basicSVM()` (c).



(a)



(b)



(c)

Figure 3.2: Results of SVM classifiers, for $T = 1000$ and $C = 1$: training dataset (a), third-party `libsvm-mat-2.91-1` (b), `basicSVM()` (c).

References

[1] Milton Abramowitz and Irene A. Stegun. *Handbook of Mathematical Functions: with Formulas, Graphs, and Mathematical Tables.* Courier Dover Publications, 2012.