ECE 532
Homework 10

Due Tuesday April 26 at the beginning of class

1. Recall the perceptron algorithm. As we noted in class the solution pair $(\boldsymbol{w}_k, b_k)$ is of the form

$$\boldsymbol{w_k} = \sum_{i=1}^{n} \alpha_{i,k} y_i \boldsymbol{x}_i \quad \text{and} \quad b_k = \sum_{i=1}^{n} \alpha_{i,k} y_i R^2 \ .$$

Therefore the solution is entirely determined by $\boldsymbol{\alpha}_k$ and the data set $\{\boldsymbol{x_i}, y_i\}_{i=1}^n$. Rewrite the perceptron algorithm in terms of this parametrization, that is, your algorithm should return $\boldsymbol{\alpha}_k$ in the end. ***Hint:*** *you just have to plug in the above expressions for* $\boldsymbol{w}_k$ *and* b_k *into the update expressions in the algorithm and see what are the corresponding update equations for* $\boldsymbol{\alpha}_k$*.*

2. The algorithm you constructed in 1. is known as the dual form of the perceptron algorithm. It bears resemblances with the dual form of the Maximum Margin Hyperplane Classifier (MMHC) problem we derived in class, namely that the dependency on the data only enters the algorithm through the *Gram Matrix*

$$\boldsymbol{G} = \begin{pmatrix} < \boldsymbol{x}_1, \boldsymbol{x}_1 > & < \boldsymbol{x}_1, \boldsymbol{x}_2 > & \cdots & < \boldsymbol{x}_1, \boldsymbol{x}_n > \\ < \boldsymbol{x}_2, \boldsymbol{x}_1 > & < \boldsymbol{x}_2, \boldsymbol{x}_2 > & \cdots & < \boldsymbol{x}_2, \boldsymbol{x}_n > \\ \vdots & \vdots & \ddots & \vdots \\ < \boldsymbol{x}_n, \boldsymbol{x}_1 > & < \boldsymbol{x}_n, \boldsymbol{x}_2 > & \cdots & < \boldsymbol{x}_n, \boldsymbol{x}_n > \end{pmatrix} .$$

This allows you to use the "kernel trick", just as in the case of the MMHC.

a. Implement the algorithm derived in Matlab. Experiment the algorithm with a linear kernel (the regular perceptron algorithm) in a set of linearly separable data (**linear.mat**).

b. Use the same algorithm, with a linear kernel, but now on the data set **circle.mat**. Comment on the results.

c. Use polynomial and Gaussian kernels (experiment with different parameters in the kernel functions) in various datasets (**linear.mat**, **circle.mat** and **extra.mat**) and comment on the results.

Note: To make your task easier you should use various functions we provide, namely **draw_ data.m** to draw the data points and proposed classifier decision region, and the various kernel functions that you might need. An example of the usage of **draw_ data.m** is presented in **example.m**.

All the files are available at http://www.cae.wisc.edu/~rcastro/perceptron .