<u>Practice Exercise 7 – SVM Classification (solution)</u>

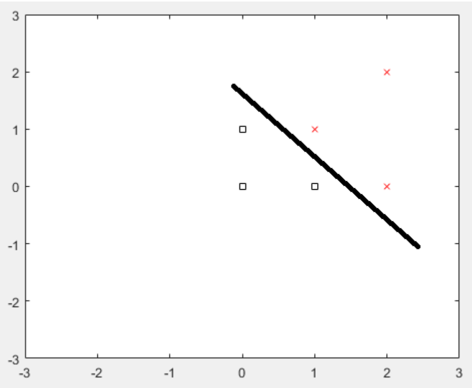
Question 2: The following 2-dimensional data is to be used for training an SVM:

Class one: (1,1), (2,2), (2,0)

Class two: (0,0), (1,0), (0,1)

(i) Plot the training points and, by visual inspection, determine the position of the optimal margin decision boundary.

(ii) List the support vectors.



The optimal decision boundary would look similar to as in the above rough drawing. The support vectors are (1,0), (0,1), (1,1), and (2,0).

Question 3: Consider training data of 1-dimensional points from two classes:

Class 1: -5,5

Class 2: -2,1

(i) Are the two classes linearly separable?

The points are not linearly separable as class 2 points (-2,1) lie in the middle of class 1 points (-5,5) thus making this data non-separable.

(ii) Consider the transformation $\varphi: R \to R^2$, $\varphi(x) = (x, x^2)$. Transform the data and plot these transformed points. Are these linearly separable?

The data, after suggested transformation, would become:

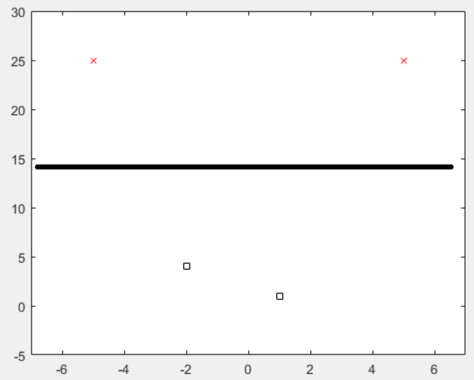
Class 1: (-5, 25), (5, 25)

Class 2: (-2, 4), (1, 1)

The data becomes linearly separable with a horizontal straight line (as shown next).

(iii) Draw the optimal separating hyper-plane in the transformed space, and explain in one or two sentences how this linear boundary helps us to separate the original data points.

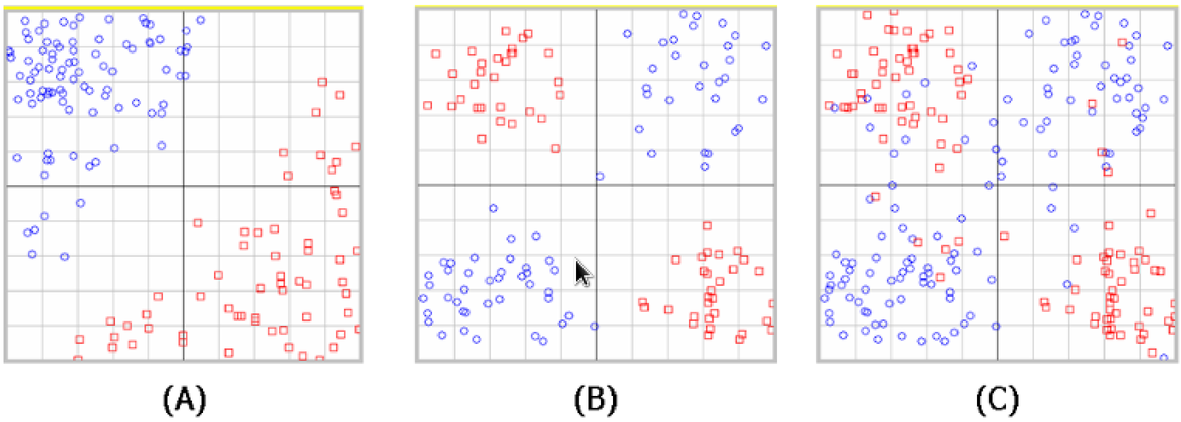
The illustration below clearly shows that the two class data is now linearly separable:



Question 5: How can a SVM classifier work for learning to classify data that is not linearly separable?

SVM is by design a linear classifier and can identify decision in case where the data is linearly separable. However, in cases where the data is not linearly separable, it can work through kernel functions to determine the decision boundary. With the help of kernel function, the data is implicitly transformed to higher dimensional spaces where finding a linear decision boundary becomes possible.

Question 6: Consider the three data sets illustrated below:



Each point has two numeric features (i.e. the x and y coordinates of the points). The circles and the squares represent two different classes.

(i) If you were to use SVM on these data sets, which data set will require a kernel to be used?

The dataset in A is linearly separable and SVM can find decision boundary without the use of a kernel. The datasets in B and C are not linearly separable, and the only way SVM can find a decision boundary for these will be by using kernel function.

(ii) If you were to use SVM with Gaussian kernel on these data sets, how would you set the width parameter of the kernel?

The parameters for SVM (and many other classification algorithms) are determined through cross validation, where the data is split in training and validation sets. The classifier training is conducted on training data by various values of the parameters (gamma, in this instance) and evaluation is performed on the validation data. The parameter which provides highest model classification accuracy is typically chosen as the 'optimal' parameter.

Question 7: The XOR problem is to learn the function from the following input points to their class labels:

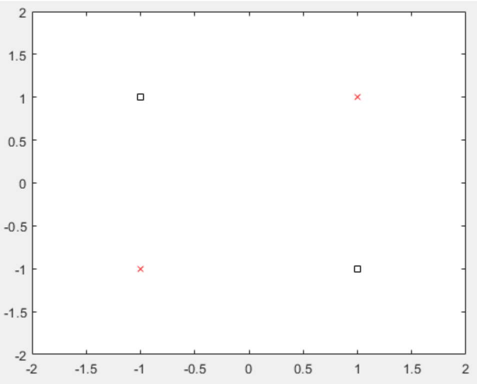
Class : (1,1), (-1,-1)

Class 2: (1,-1), (-1,1)

We know that support vector machine (SVM) with a kernel can solve this problem by mapping the points in a higher dimensional space. But higher dimensional spaces are difficult to visualise, and we would like to construct a support vector machine that classifies these points correctly in a 2-dimensional input space. Is this possible? If so, how? If not, explain why not.

Hint: Try to come up with a feature-transformation that stays in 2D (i.e. maps R^2 into R^2) and makes the classes linearly separable.

A simple visualisation of the datasets illustrates (as below) that this data is not linearly separable.



One possible feature transform (from 2d to 2d) is x,xy which results in following data points:

Class : (1,1), (-1,1)

Class 2: (1,-1), (-1,-1)

These data points results in linearly separable classes as illustrated below: