Decision Boundaries and Classification Performance Of SVM And KNN Classifiers For 2-Dimensional Dataset

S. Shahbudin, A. Hussain, S. M. Mustaza, S. A Samad, H. Husain,
Department of Electrical, Electronics & Systems Engineering,
Department of Electrical, Electronics and System Engineering, Universiti Kebangsaan Malaysia
shaqay@eng..ukm.my or aini@eng.ukm.my

Abstract

Support Vector Machines (SVM) and K-Nearest Neighborhood (k-NN) are two most popular classifiers in machine learning. In this paper, we intend to study the generalization performance of the two classifiers by visualizing the decision boundary of each classifier when subjected to a two-dimensional (2-D) dataset. Four different sets of database comprising of 2-D datasets namely the eigenpostures of human (EPHuman), the breast cancer (BCancer), the Swiss roll (SRoll) and Twinpeaks (Tpeaks) were used in this study. Results obtained confirmed SVM classifier superb generalization performance since it contributed the lower classification error rate when compared to the k-NN classifier during the training for binary classification of all 2-D datasets. This is evident and can be clearly visualized through the plots depicting the decision boundaries of the binary classification task.

1. Introduction

Support Vector Machine (SVM) is a universal machine learning method proposed by Vapnik and co-workers and it is an eminent technique for solving classification problems [1]. The goal of SVM is to determine a classifier that minimizes the empirical risk namely the training set error and the confidence interval which corresponds to the generalization or test set error [2]. Additionally, another classifier known as K-Nearest Neighborhood (k-NN) is also evaluated for comparison purpose. K-NN classifier is a simple but appealing classifier. When a new sample arrives, k-NN finds the k neighbors nearest to the new sample from the training space based on some suitable similarity or distances metrics [3][4] . In term of computing time SVM gave the shortest time when its support vectors have been determined. In [3], the classification accuracy of K-NN is superior when feature selection technique is used to remove redundant and irrelevant features. However, the effectiveness of the classifiers is rarely proven and analyzed through visualization of the decision boundaries especially in cases or problems involving 2-D datasets. Therefore, the purpose of this paper is to analyze and evaluate the generalization ability of the two aforementioned classifiers by means of visualization of the decision boundaries based on the measured values of classification error rate. By observing and analyzing the illustrations of the decision boundaries, conclusion will be drawn to determine the better classifier.

The rest of the paper is organized as follows. First section provides an overview of SVM followed by a brief introduction of the k-NN classifier algorithm in next section. Third section provides brief description of the four datasets used in this study. The experimental results are discussed in forth section and finally, the conclusion is given in last section.

2. Support Vector Machines (SVM)

In general, Support Vector Machine (SVM) is a learning machine for two class classification problems. Given a labeled training dataset, $(x_1.,y_1),\ldots,(x_l\ ,y_l\ )$ where $x_i \in R^N$ is a feature vector and $y_i \in \{-1,1\}$ is a class label, the SVM algorithm seeks to define a decision surface that gives the largest margin or separating between the data classes whilst at the same time minimizing the number of errors. However, this decision surface is not created in the input space, but rather in a very high-dimensional feature space. The resulting model is nonlinear, and is accomplished by the use of kernel functions. The kernel function, K indicates a measure of similarity between a pattern x_i, and a pattern x_j from the stored training set. Using the kernel, the dual Quadratic Programming (QP) problem in term of Lagrange Multipliers, α_i in the feature space is given in equation (1),

$$W(\alpha) = \sum_{i=1}^{l} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{l} \alpha_i \alpha_j y_i y_j K(x_i, x_j) \qquad (1)$$

subject to the following constraint of

$$\sum_{i=1}^{l} \alpha_i y_i = 0 \qquad 0 \le \alpha_i \le C \qquad (2)$$

where i=1,…,l.

After finding the optimal values of α_i, the decision boundary is constructed using the following,

$$f(x) = \sum_{\alpha_i \neq 0}^{l} y_i \alpha_i K(x_i, x) + b. \qquad (3)$$

where the x class is determined from the sign of $f(x)$. The value b is the decision boundary threshold where the x class is determined from the sign of $f(x)$. The x_i corresponding to $\alpha_i \neq 0$ is called the support vector. The regularization parameter, C, is the margin parameter that determines the trade-off between maximizing the margin and minimizing the classification error. It is chosen by means of a validation set [5].

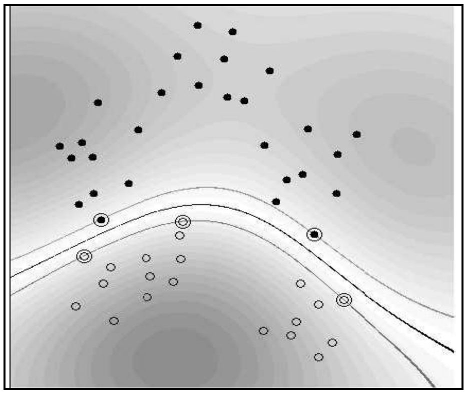


Figure 1: Illustration of decision boundaries of SVM classifier found by using radial basis function (RBF)

An example of the SVM decision boundary for 2D classification generated by Chen et al is as depicted in Fig.1. Both classification boundary and the accompanying soft margins are represented by bold line and timid lines, respectively where as black dotted and white dotted fall on opposite sides of the decision boundary. The circled points are the support vectors that lie closest to the decision boundary.

3. K- Nearest Neighborhood

The k-NN algorithm is amongst the simplest of all machine learning algorithms. The training phase of the algorithm consists only involves storing the feature vectors and class labels of the training samples. In the actual classification phase, the test sample (whose class is not known) is represented as a vector in the feature space. Distances from the new vector to all stored vectors are computed and k closest samples are selected.

There are several ways to classify the new vector to a particular class; one of the most used techniques is to predict the new vector to the most common class amongst the k nearest neighbors. A major drawback to using this technique to classify a new vector to a class is that the classes with the more frequent examples tend to dominate the prediction of the new vector, as they tend to appear in the k nearest neighbors when the neighbors are computed due to their large number.

A way to overcome this problem is to take into account the distance of each k nearest neighbors with the new vector that is to be classified and predict the class of the new vector based on these distances. The best choice of k depends upon the data; generally, larger values of k reduce the effect of noise on the classification, but make boundaries between classes less distinct. A good k can be selected by various heuristic techniques, for example, cross-validation. The special case where the class is predicted to be the class of the closest training sample (i.e. when k = 1) is called the nearest neighbor algorithm. If k = 1, then the object is simply assigned to the class of its nearest neighbor.

An apparent extension of the nearest-neighbor rule is the k-nearest-neighbor rule which classify x by assigning it the label most frequently represented among the k nearest samples as shown in Fig. 2. In other word, a decision is made by examining the labels on the k nearest neighbor by taking a vote. [11].

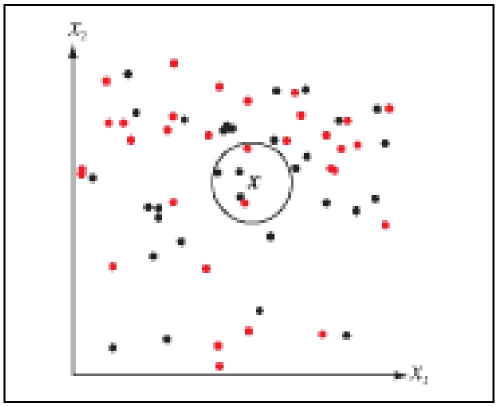


Figure 2: The k-nearest neighbor query forms a spherical region around the test point x until it encloses k training samples, and it labels the test point by a majority vote of these samples. In the case for k = 5, the test point will be labeled as black.

In general, k-NN classifier tend to produce piecewise linear decision boundaries as depicted in Figure 3. The decision boundary in a 1-NN classifier is made of concatenated segments of Voronoi tesselation; in which a set of objects decomposes the space into Voronoi cell.

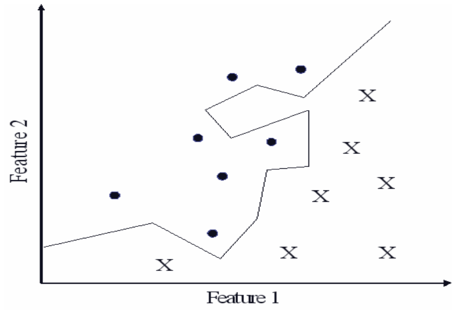


Figure 3: Illustration of piecewise linear decision boundary of k-NN classifier for two different feature vectors

Each object's cell then consists of all points closer to the object than to other objects and as a result all points within that cell are assigned that particular class. Overall the decision boundary is equal to the union of cell boundaries where class decision is different on each side. Details of the k-NN classifier can be found in [11].

4. Brief description of the 2-dimensional dataset

This section provides the description of all four 2-D datasets used in the study. The datasets consist of two types: the naturally generated datasets and artificially obtained datasets. The naturally generated datasets are the Eigenpostures and Breast cancer. The eigenpostures dataset comprises 300 images of human postures [6],[7] in which 200 (100 each for standing and non-standing) are used as training data and another 100 (50 each for standing and non-standing) as testing dataset. Both classifiers were trained to classify the human posture of standing and non-standing. Based on the results obtained from [6] and [7], the best combination involving the second and forth eigenpostures was selected. PCA technique was used to derive the EPHuman data. Meanwhile the BCancer dataset consisted of 200 training data and 167 testing data that was obtained from the UCI Machine Learning Repository.

In this section, the performances for both classifiers are illustrated in terms of their decision boundaries, analyzed and evaluated. Various datasets were used in the analysis so that comparison in terms of the SVM and the k-NN classification accuracies and visualizations of the decision boundaries of the two classifiers can be made and conclusion can be drawn accurately.

5. Experimental Setup

To conduct the experiment, several parameters needed to be set up first. For instances, the k-NN classifier used in this experiment uses a small values of k since the smaller k values will provide a higher variance for this classifier and as such, will make the illustration of decision boundaries more accurate with the highest classification rate. Therefore, in this study, the optimal value of *k* is used to obtain the smallest error of classification rate. In this study, the range of *k* between 1 to 10 was tested to identify the best value. As for the SVM, the Gaussian radial basis function (RBF) kernel and the Sequential Minimal Optimization (SMO) techniques are used as solver for the QP problem.

Cross-validation technique was used to find the optimal value of the kernel parameter of SVM such as regularization parameter, C and kernel width parameter, σ. Having obtained the optimal values would yield the best generalization performance with the smallest values of classification error rate. Both classifiers were implemented using the Statistical Pattern Recognition Toolbox and Matlab 7.0.

6. Results and Discussions

The performance analysis of the SVM and k-NN started with the experiments using the natural datasets and then the artificial data. Typically results are displayed, discussed and analyzed in terms of classification accuracies and tabulated as shown in Table 1. The same will be done here but an additional effort has been made to display the results and discuss the classifier performance by means of their decision boundaries. As expected, our results revealed the superiority of the SVM classifier over the k-NN. For the EPHuman dataset, both classifiers perfectly separate the two classes of standing and non-standing eigenpostures, but with the larger dataset comprising the BCancer data, the SVM outperformed the k-NN by almost half. Next, the artificial datasets are used to analyze and evaluate the classifiers. As tabulated in Table 1, both SVM and k-NN did not perform well in separating the two classes of SRoll dataset. However, SVM performed slightly better (2% more) compared to k-NN. For the Tpeaks dataset, which is smaller than the SRoll, the SVM misclassification error was 1.48% whilst the k-NN obtained a 2.22% error.

Table 1: Performance Comparison Of SVM and k-NN Classifier Based On Classification Rate With Optimal Values Of Each Classifier Parameters

Data Sets	**SVM**		**k-NN**	
	Optimal Values (C, σ)	***Error Rate (%)***	***Optimal Values k***	***Error Rate (%)***
EigenPostures (EPHuman)	(10,0.1)	0.0	1	0.0
Breast Cancer (BCancer)	(1,1)	8.38	1	15.57
Swiss roll (SRoll)	(100,5)	32	1	34
Twinpeaks (Tpeaks)	(100,0.1)	1.48	1	2.22

Selected samples of decision boundaries of the classification results are illustrated in Figure 3(a), 4(a), 5(a) and 6(a) for the SVM classifier and Figure 3(b), 4(b), 5(b) and 6(b) for the k-NN classifier using the EPHuman, BCancer, SRoll and Tpeaks datasets. Even though the classification results for the eigenposture dataset recorded perfect performance, the performance classification of boundaries differs slightly. Similar classification boundaries were noted for the Tpeaks dataset shown in Fig. 6 but for the SRoll datasets in Figure 5 the decision boundaries are much smoother for the SVM compared to the k-NN. It is believed that the smoother decision boundaries contribute to the better performance of SVM.

7. Conclusion

It is evident that based on the classification accuracies and supported by the illustrations of the decision boundaries, SVM has been proven to be the better classifier than the k-NN. Visualization of the decision boundary serves as an aid to better support the attained results and for the researcher to better understand and appreciate the results.

References

[1] V. N. Vapnik, “*The nature of statistical learning theory*,” Springer, New York, 1995.

[2] N. Cristianini, J.Shawe-Taylor, “*An Introduction to Support Vector Machines and other kernel-based learning methods*,”New York: Cambridge University Press, 2000.

[3]Qinghua Hu, Daren Yu,Zongxia Xie “*Neighborhood classifiers*”, Expert System in Applications: An International Journal, Vol. 34 (2), pp 866-876 February 2008,

[4]Tiang Ming, Zhuang Yi and Chen Songcan, “*Improving Support Vector Machine Classifier by Combining it with k Nearest Neighbor Principle Based on the Best Distance Measurement*”, Intelligent Transportation Systems, 2003. Proceedings. 2003 IEEE Vol. 1, 12-15 Oct. 2003 Page(s): 373 – 378

[5] Shiego Abe, “*Support vector machines for pattern classification*” Advances in Pattern Recognition, Springer 2005, pp 22-23.

[6] Nooritawati Md Tahir, Aini Hussain, Salina Abdul Samad, Hafizah Husain & Mohd Marzuki Mustafa, “Eigenposture For Classification” Journal of Applied Sciences, Asian Network for Scientific Information, ANSINET, 6(2), 2006.

[7] Nooritawati Md Tahir, Aini Hussain, Salina Abdul Samad, Hafizah Husain “PCA-based Human Posture Classification ” Jurnal TeknologD.46(D),35-44, 2007. UCI Machine Learning Repository Database http://archive.ics.uci.edu/ml/

[8] Matlab_Toolbox_for_Dimensionality_Reduction, http://www.cs.unimaas.nl/l.vandermaaten/Laurens_van_derMaaten/Matlab_Toolbox_for_Dimensionality_Reduction.html

[9] Statistical Pattern Recognition Toolbox http://cmp.felk.cvut.cz/cmp/software/stprtool/index.html

[10]R.O. Duda, P.E. Hart, and D.G. Stork. “*Pattern Classification.*” John Wiley & Sons, 2nd. edition, 2001, pp 182-183.

[12] Chen, P. -H., C. -J. Lin and B. Schölkopf, “*A tutorial on v-support vector machines*”, Applied Stochastic Models in Business and Industry **21*(2)***, 2005, pp 111-136

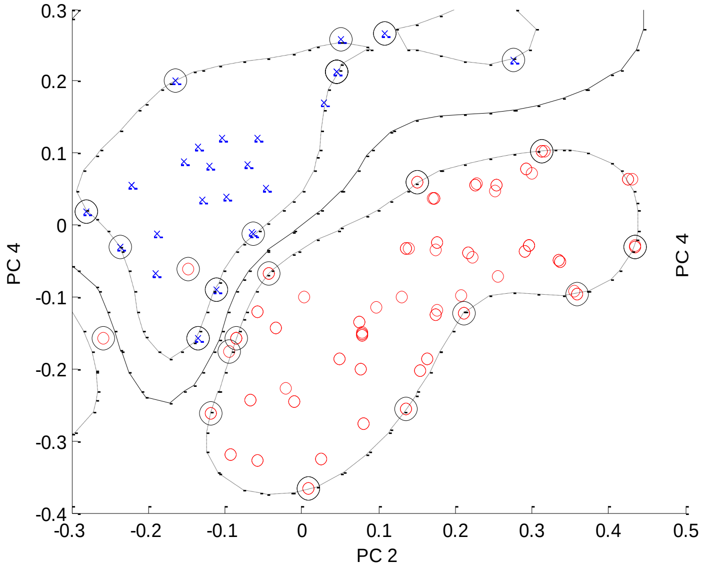


Figure 3a: *SVM classifier Decision Boundary for Eigen-posture dataset*

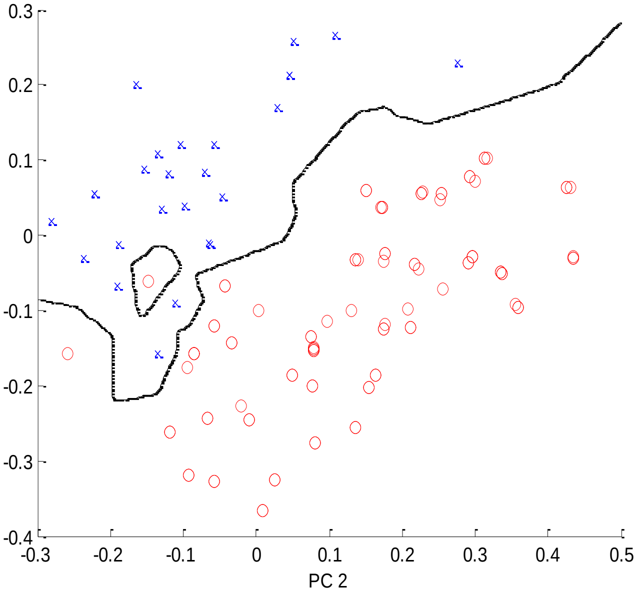


Figure 3b: *k-NN Classifier Decision Boundary for Eigen-postures dataset*

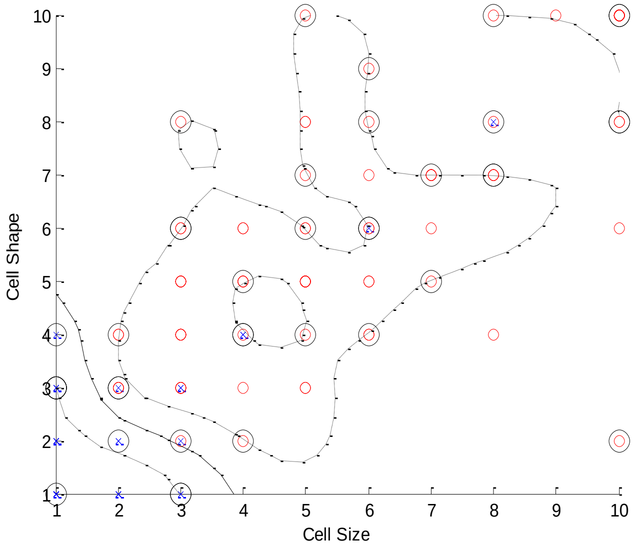


Figure 4a: *SVM Classifier Decision Boundary for Breast Cancer dataset*

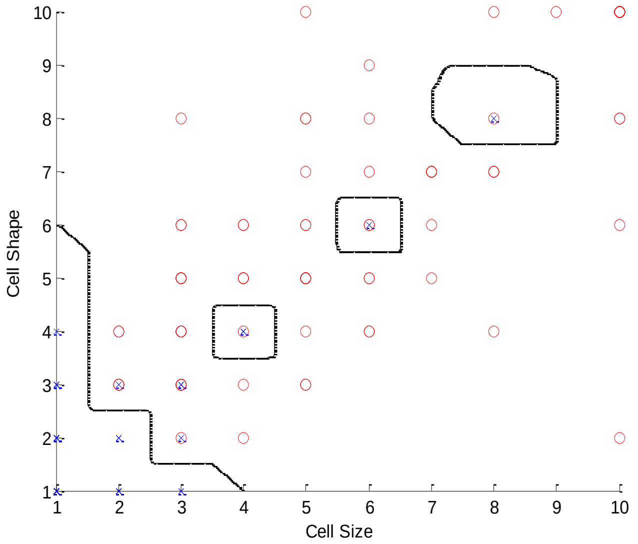
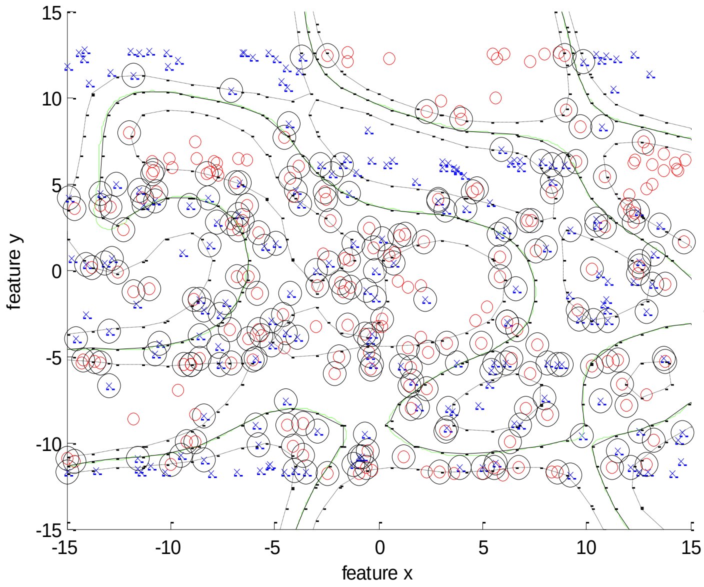


Figure 4b: *k-NN classifier Decision Boundary for Breast Cancer dataset*



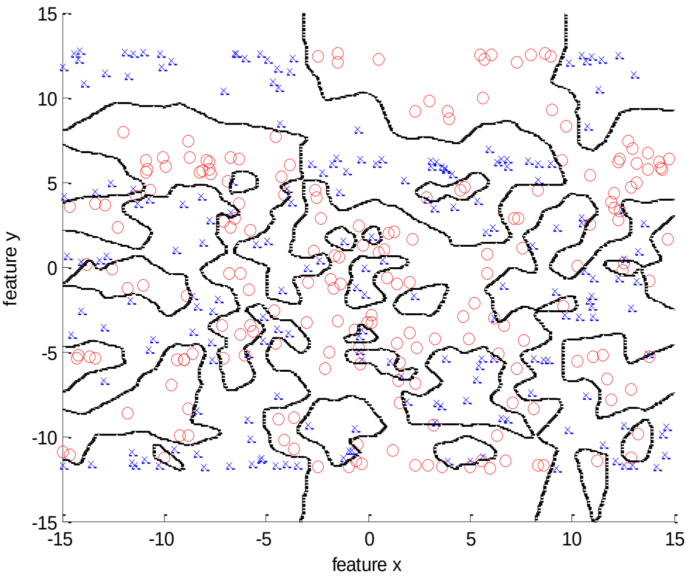
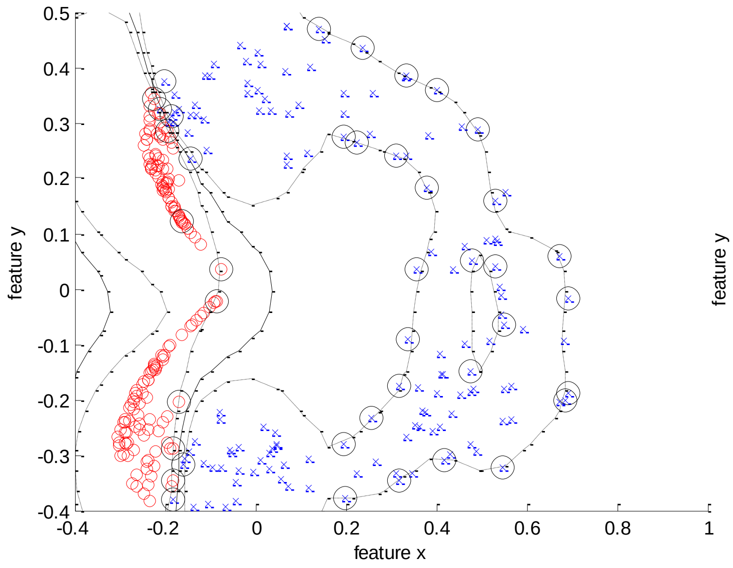


Figure 5a: *SVM Classifier Decision Boundary for Swiss*

Figure 5b: *k-NN Classifier Decision Boundary for Swiss*



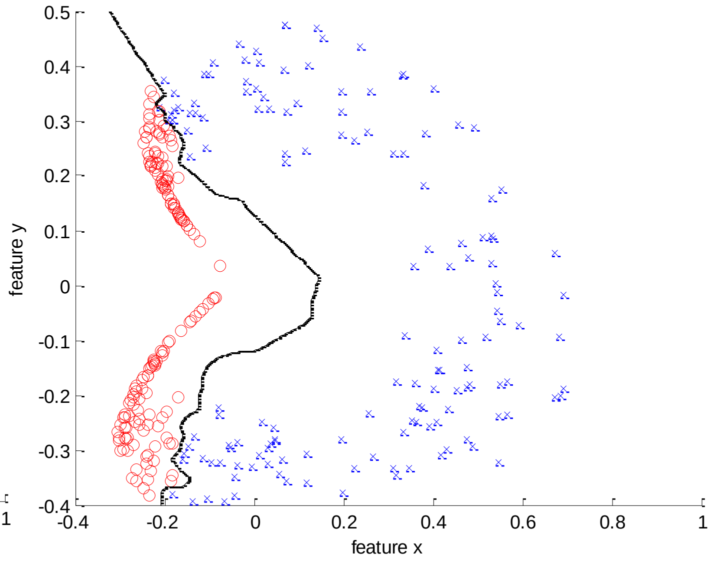


Figure 6a: *SVM Classifier Decision Boundary for Twinpeaks dataset*

Figure 6b: *k-NN classifier Decision Boundary for Twinpeaks dataset*