

Machine Learning Laboratory (INT 407)

PROGRAM 7: Apply and analyse Support Vector Machine algorithm with different kernel functions and parameter values for a linearly separable and non-linearly separable classification Task

AIM:

To Apply and analyse Support Vector Machine algorithm with different kernel functions and parameter values for a linearly separable and non-linearly separable classification Task.

THEORY:

What is a Support Vector Machine(SVM)?

It is a supervised machine learning problem where we try to find a hyperplane that best separates the two classes. Note: Don't get confused between SVM and logistic regression. Both the algorithms try to find the best hyperplane, but the main difference is logistic regression is a probabilistic approach whereas support vector machine is based on statistical approaches.

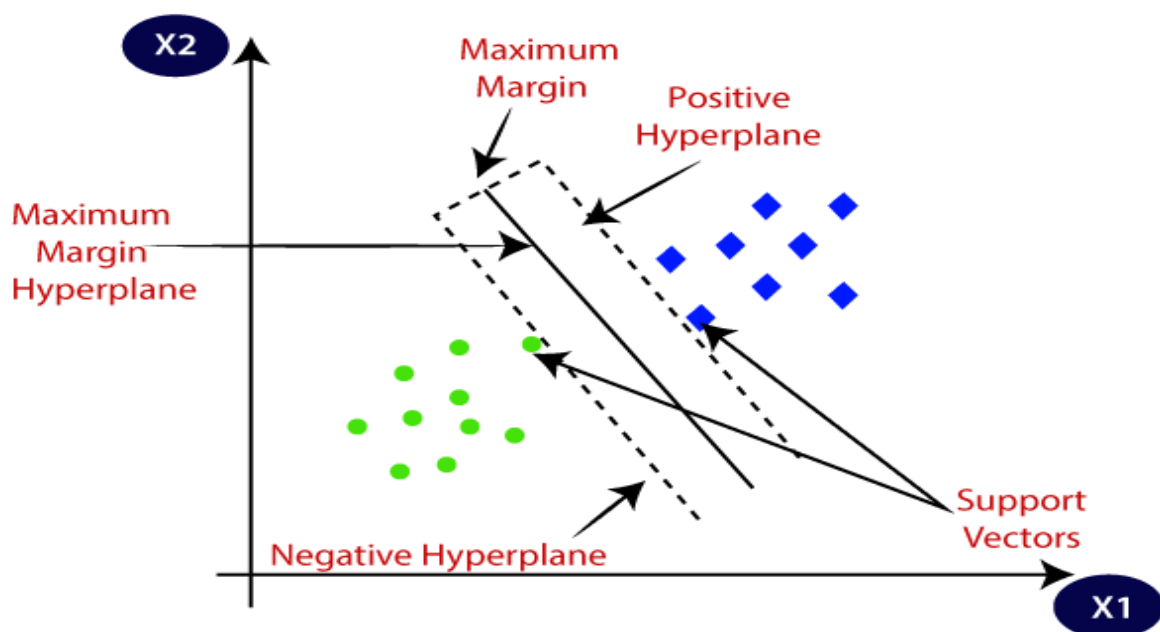
Types of Support Vector Machine (SVM) Algorithms

Linear SVM: When the data is perfectly linearly separable only then we can use Linear SVM. Perfectly linearly separable means that the data points can be classified into 2 classes by using a single straight line(if 2D).

Non-Linear SVM: When the data is not linearly separable then we can use Non-Linear SVM, which means when the data points cannot be separated into 2 classes by using a straight line (if 2D) then we use some advanced techniques like kernel tricks to classify them. In most real-world applications we do not find linearly separable datapoints hence we use kernel trick to solve them.

Support Vectors: These are the points that are closest to the hyperplane. A separating line will be defined with the help of these data points.

Margin: it is the distance between the hyperplane and the observations closest to the hyperplane (support vectors). In SVM large margin is considered a good margin. There are two types of margins hard margin and soft margin.

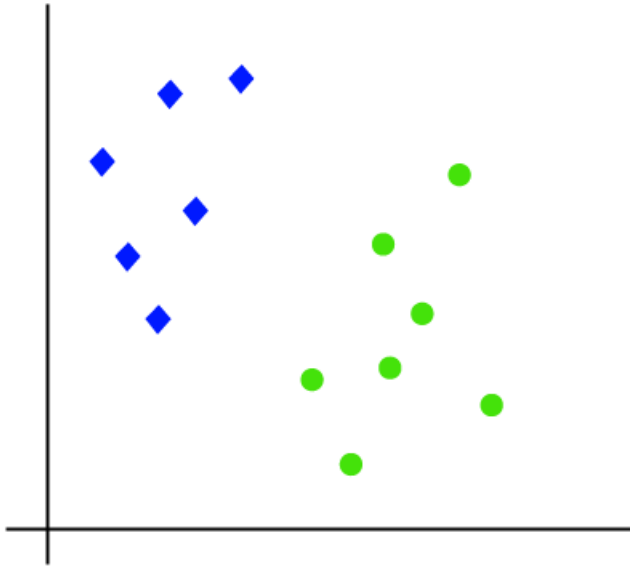


Machine Learning Laboratory (INT 407)

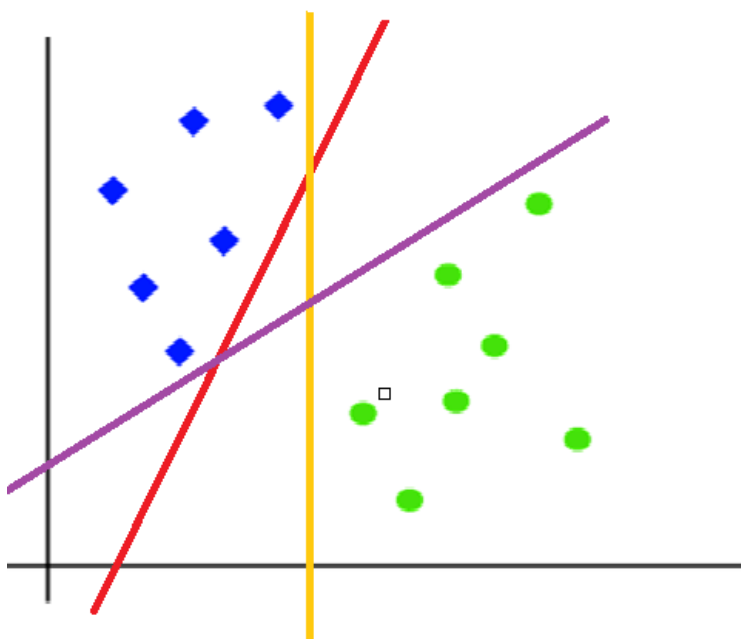
How Does Support Vector Machine Work?

SVM is defined such that it is defined in terms of the support vectors only, we don't have to worry about other observations since the margin is made using the points which are closest to the hyperplane (support vectors), whereas in logistic regression the classifier is defined over all the points. Hence SVM enjoys some natural speed-ups.

Let's understand the working of SVM using an example. Suppose we have a dataset that has two classes (green and blue). We want to classify that the new data point as either blue or green.

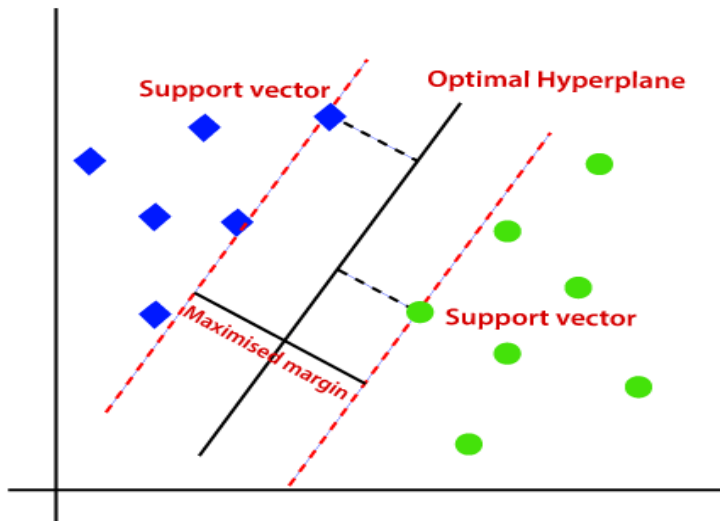


To classify these points, we can have many decision boundaries, but the question is which is the best and how do we find it? NOTE: Since we are plotting the data points in a 2-dimensional graph we call this decision boundary a straight line but if we have more dimensions, we call this decision boundary a "hyperplane".



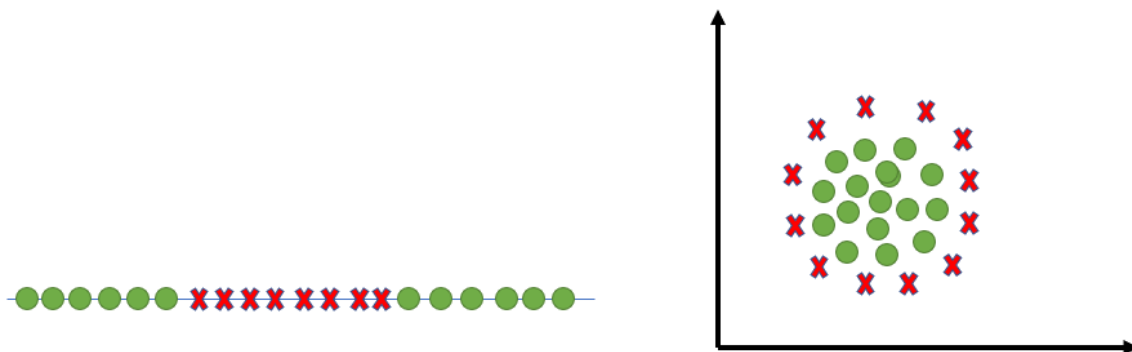
Machine Learning Laboratory (INT 407)

The best hyperplane is that plane that has the maximum distance from both the classes, and this is the main aim of SVM. This is done by finding different hyperplanes which classify the labels in the best way then it will choose the one which is farthest from the data points or the one which has a maximum margin.

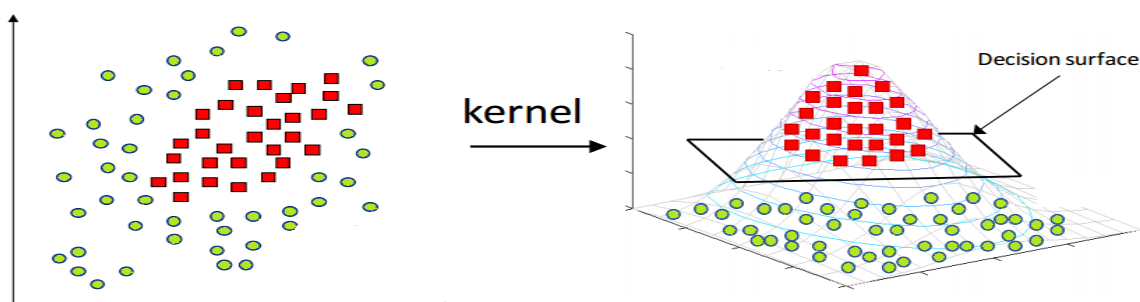


Kernels in Support Vector Machine

The most interesting feature of SVM is that it can even work with a non-linear dataset and for this, we use “Kernel Trick” which makes it easier to classify the points. Suppose we have a dataset like this:



Here we see we cannot draw a single line or say hyperplane which can classify the points correctly. So what we do is try converting this lower dimension space to a higher dimension space using some quadratic functions which will allow us to find a decision boundary that clearly divides the data points. These functions which help us do this are called Kernels and which kernel to use is purely determined by hyperparameter tuning.

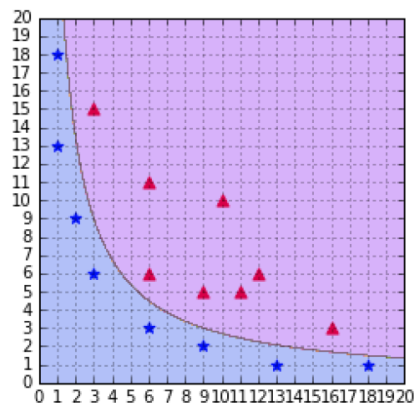


Machine Learning Laboratory (INT 407)

Different Kernel Functions

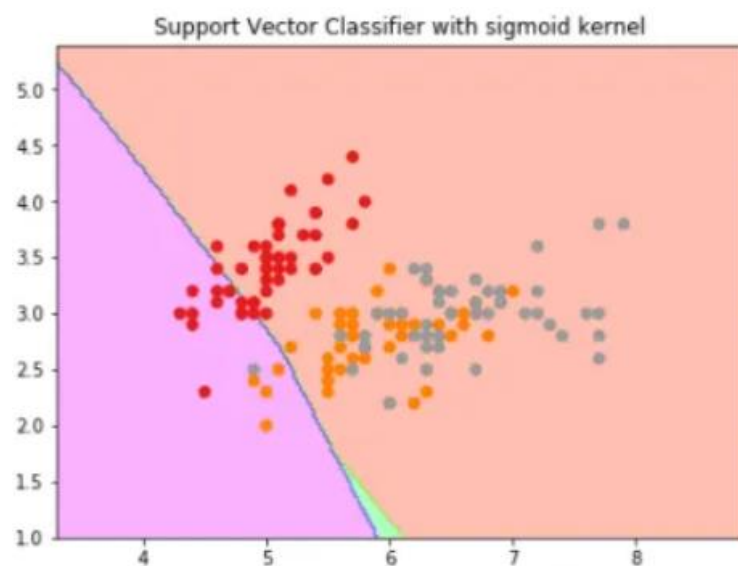
Some kernel functions which you can use in SVM are given below:

1. Polynomial Kernel

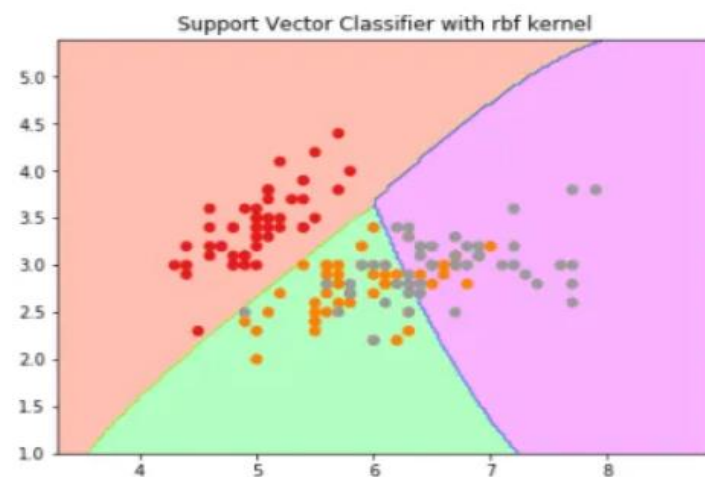


A SVM using a polynomial kernel is able to separate the data (degree=2)

2. Sigmoid Kernel



3. RBF Kernel



Machine Learning Laboratory (INT 407)

4. Bessel function kernel
5. Anova Kernel

How to Choose the Right Kernel?

Choosing a kernel totally depends on what kind of dataset are you working on. If it is linearly separable then you must opt. for linear kernel function since it is very easy to use and the complexity is much lower compared to other kernel functions.

You can then work your way up towards the more complex kernel functions. Usually, we use SVM with RBF and linear kernel function because other kernels like polynomial kernel are rarely used due to poor efficiency.

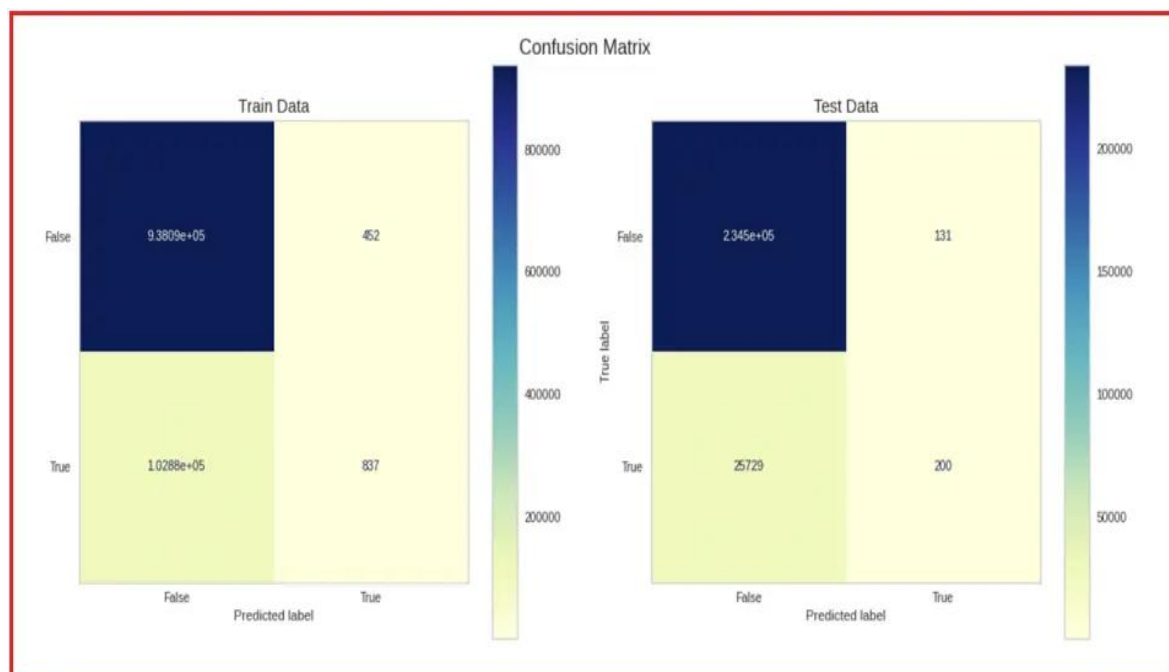
Advantages of SVM

- SVM works better when the data is Linear
- It is more effective in high dimensions
- With the help of the kernel trick, we can solve any complex problem
- SVM is not sensitive to outliers
- Can help us with Image classification

Disadvantages of SVM

- Choosing a good kernel is not easy
- It doesn't show good results on a big dataset
- The SVM hyperparameters are Cost -C and gamma. It is not that easy to fine-tune these hyper-parameters. It is hard to visualize their impact

SAMPLE OUTPUT:



Machine Learning Laboratory (INT 407)

```
print(classification_report(y_test, y_pred_test))
```

	precision	recall	f1-score	support
0	0.99	0.99	0.99	3306
1	0.93	0.84	0.88	274
accuracy			0.98	3580
macro avg	0.96	0.92	0.94	3580
weighted avg	0.98	0.98	0.98	3580

Classification accuracy

```
:
TP = cm[0,0]
TN = cm[1,1]
FP = cm[0,1]
FN = cm[1,0]
```

```
# print classification accuracy

classification_accuracy = (TP + TN) / float(TP + TN + FP + FN)

print('Classification accuracy : {0:0.4f}'.format(classification_accuracy))
```

Classification accuracy : 0.9830

Classification error

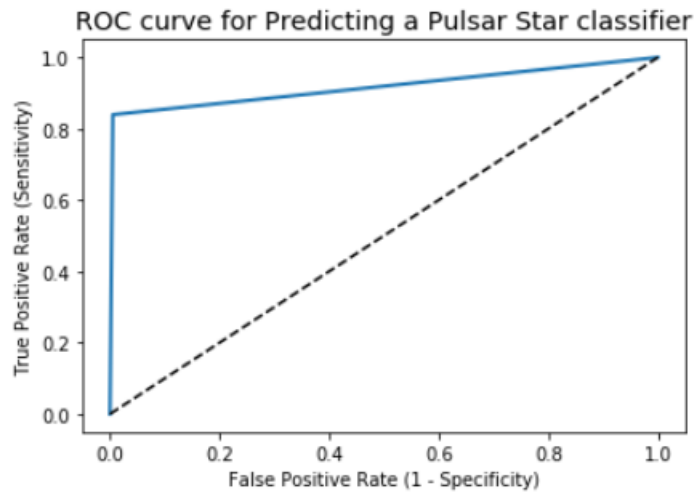
```
# print classification error

classification_error = (FP + FN) / float(TP + TN + FP + FN)

print('Classification error : {0:0.4f}'.format(classification_error))
```

Classification error : 0.0170

Machine Learning Laboratory (INT 407)



SAMPLE VIVA-VOCE QUESTIONS:

1. What is the basic principle of a Support Vector Machine?
2. What are hard margin and soft Margin SVMs?
3. What do you mean by Hinge loss?
4. What is the “Kernel trick”?
5. What is the role of the C hyper-parameter in SVM? Does it affect the bias/variance trade-off?
6. Explain different types of kernel functions.
7. What affects the decision boundary in SVM?
8. When SVM is not a good approach?
9. What is the geometric intuition behind SVM?
10. What is the difference between logistic regression and SVM without a kernel?