**Rashtreeya Sikshana Samithi Trust**

**RV College of Engineering**

**Bangalore – 560 059**

***(Autonomous Institution affiliated to VTU, Belgaum)***

**Department of Computer Science and Engineering**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Student’s Name** | **Giridhar K Shanbhag** | | | **USN** | **1RV18CS056** | | |
| **Semester** | IV | **Branch** | CSE | **Section- B** | | |  |
| **Course Title** | Design and Analysis of Algorithms | | | **Course Code** | | 18CS42 | |

**Certificate**

This is to certify that Giridhar K Shanbhag(**1RV18CS056**),**3rd**Semester **Computer Science Engineering** Branch has satisfactorily completed the **Assignment** prescribed by the Institution in the course

**Design and Analysis of Algorithms** (18CS42) for the academic year 2019 – 2020.

***Title: Support Vector Machine Regression Algorithm***

|  |  |
| --- | --- |
| **Max. Marks** | **Marks Obtained** |
|  |  |

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Signature of the Faculty**

1. **INTRODUCTION:**

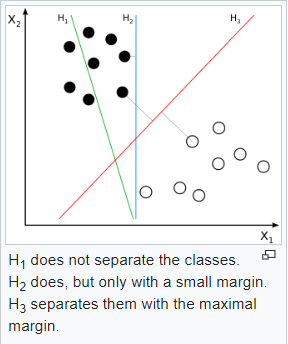
In machine learning, **support-vector machines** (**SVMs**, also **support-vector networks**[[1]](https://en.wikipedia.org/wiki/Support-vector_machine" \l "cite_note-CorinnaCortes-1)) are supervised learning models with associated learning algorithms that analyze data used for classification and regression analysis. Given a set of training examples, each marked as belonging to one or the other of two categories, an SVM training algorithm builds a model that assigns new examples to one category or the other, making it a non-probabilistic binary linear classifier (although methods such as Platt scaling exist to use SVM in a probabilistic classification setting). An SVM model is a representation of the examples as points in space, mapped so that the examples of the separate categories are divided by a clear gap that is as wide as possible. New examples are then mapped into that same space and predicted to belong to a category based on the side of the gap on which they fall.

In addition to performing linear classification, SVMs can efficiently perform a non-linear classification using what is called the kernel trick, implicitly mapping their inputs into high-dimensional feature spaces.

When data are unlabelled, supervised learning is not possible, and an unsupervised learning  approach is required, which attempts to find natural clustering of the data to groups, and then map new data to these formed groups. The **support-vector clustering**[[2]](https://en.wikipedia.org/wiki/Support-vector_machine#cite_note-HavaSiegelmann-2) algorithm, created by Hava Siegelmann and Vladimir Vapnik, applies the statistics of support vectors, developed in the support vector machines algorithm, to categorize unlabeled data, and is one of the most widely used clustering algorithms in industrial applications.

1. **METHODOLOGY AND DEFINITION:**

Classifying data is a common task in machine learning. Suppose in a given data every point belong to one of the two classes. The goal is to decide in which class a new point will belong to. A data point is viewed as a p dimensional vector i.e., a vector with p different values each representing a single dimension. The goal is to know whether we can split the p-dimensional vector into a (p-1)-dimensional hyperplane and this is known as linear regression.



1. **APPLICATIONS:**

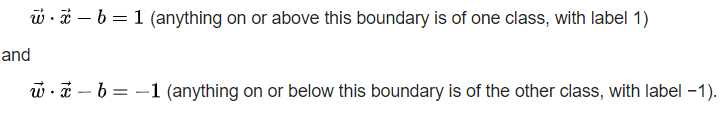
SVMs can be used to solve various real-world problems:

* SVMs are helpful in text and hypertext categorization, as their application can significantly reduce the need for labeled training instances in both the standard inductive and transductive settings. Some methods for shallow semantic parsing are based on support vector machines.
* Classification of images can also be performed using SVMs. Experimental results show that SVMs achieve significantly higher search accuracy than traditional query refinement schemes after just three to four rounds of relevance feedback. This is also true for image segmentation systems, including those using a modified version SVM that uses the privileged approach as suggested by Vapnik.
* Classification of satellite data like SAR data using supervised SVM.
* Hand-written characters can be recognized using SVM.
* The SVM algorithm has been widely applied in the biological and other sciences. They have been used to classify proteins with up to 90% of the compounds classified correctly. Permutation tests based on SVM weights have been suggested as a mechanism for interpretation of SVM models.Support-vector machine weights have also been used to interpret SVM models in the past. Posthoc interpretation of support-vector machine models in order to identify features used by the model to make predictions is a relatively new area of research with special significance in the biological sciences.

1. **LINEAR SVM:**

If the data is linearly separable, we can select two parallel hyperplanes that separate two classes of data so that the distance between them is as large as possible. The region bounded by these two hyperplanes is called as margin. The margin that lies exactly halfway between the two hyperplanes is called the maximum margin hyperplane.

With a normalized dataset or a standardized dataset, these hyperplanes can be described by the equations.



1. **TIME AND SPACE COMPLEXITY ANALYSIS:**

The time complexity of SVM regression algorithm depends on both the number of free/independent vectors R and number of support vectors S.

For training the dataset we choose two arbitrary vectors from R which takes quadratic time and then we choose a subset of independent vectors in this range which takes linear time. Hence the total time required for training is proportional to the product of quadratic and linear power of R.

The time taken to train the dataset directly depends on the third power of R i.e., T=O(R3). Suppose the dimension of the vector is n, i.e., the number of elements in the vector is n, then time needed for the execution is the product of the dimension and the number of support vectors S, as all the individual dimension values of the S support vectors need to be processed. Therefore execution time T=O(n\*S).

Hence total time needed to train the machine with respect to a given dataset and then execute is T=O(R3) + O(n\*S).