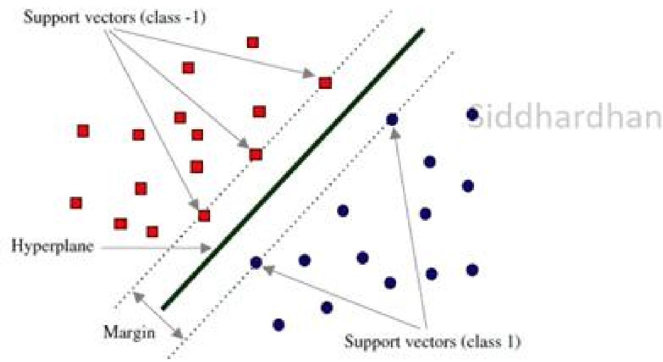
Siddhardhan

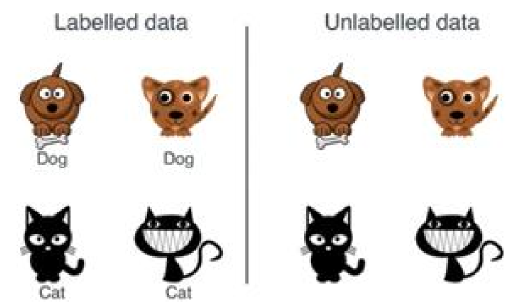
Support Vector Machine (SVM) Classifier - intuition



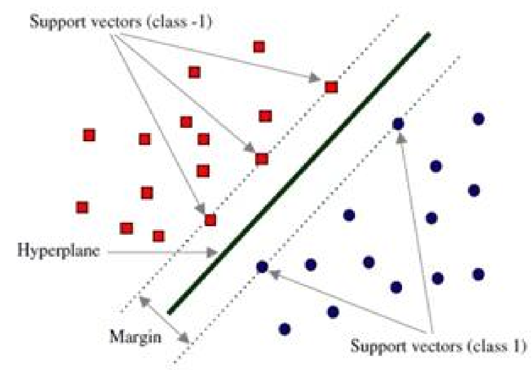
Support Vector Machine

About Support Vector Machine model:

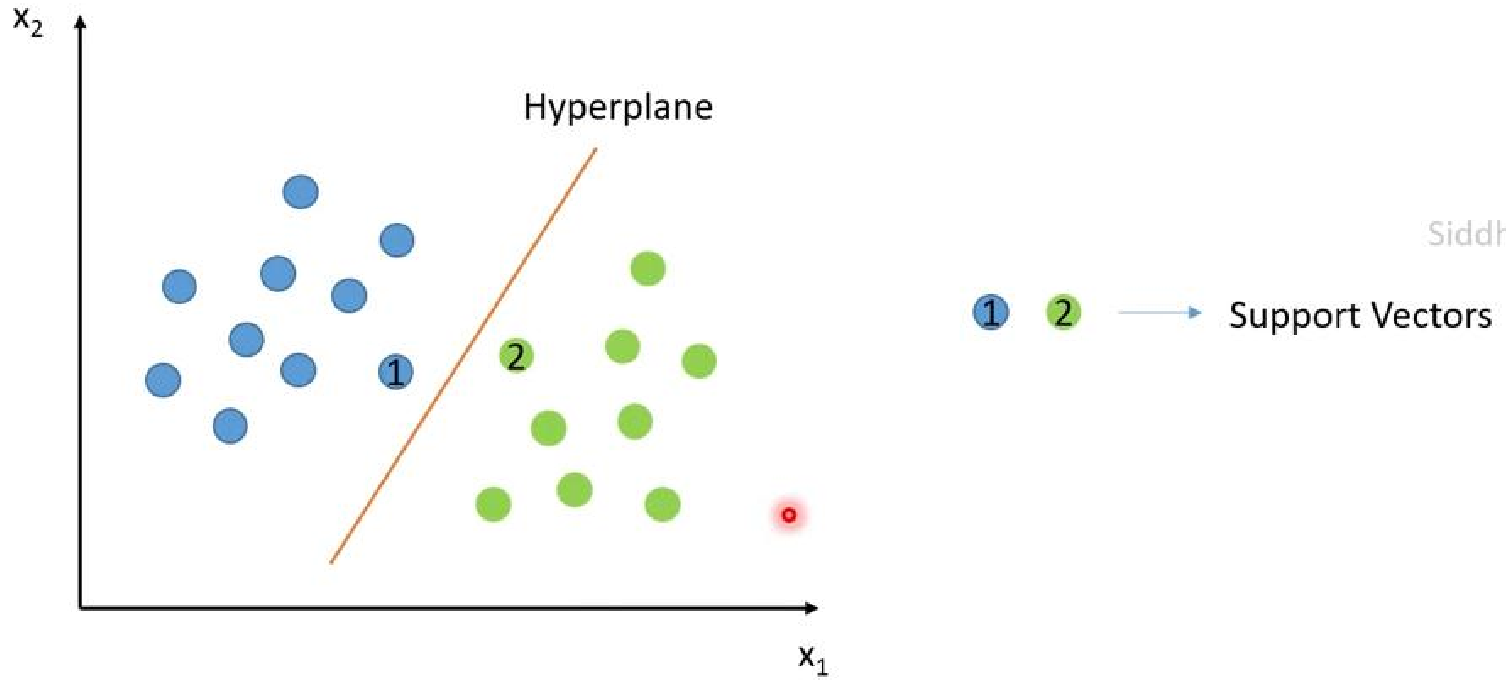
1. Supervised Learning Model
2. Both Classification & Regression
3. Hyperplane
4. Support Vectors



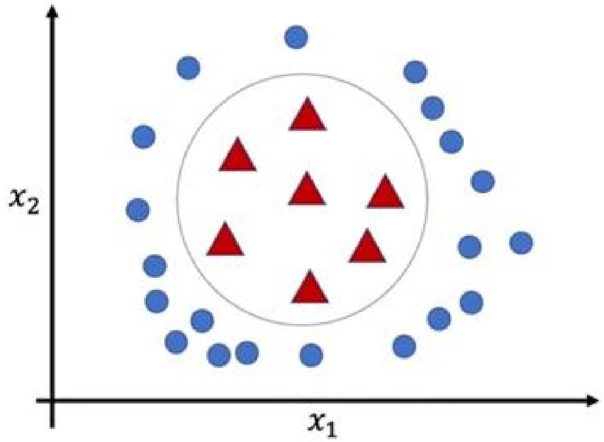
Siddhardhan



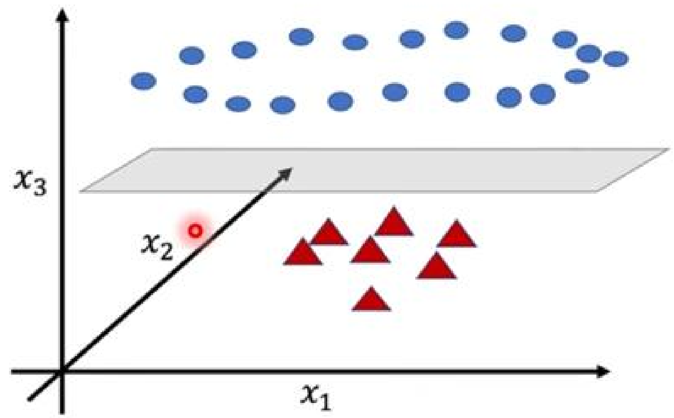
Support Vector Machine Classifier



Support Vector Machine Classifier



SVM in 2 dimensions



SVM in 3 dimensions

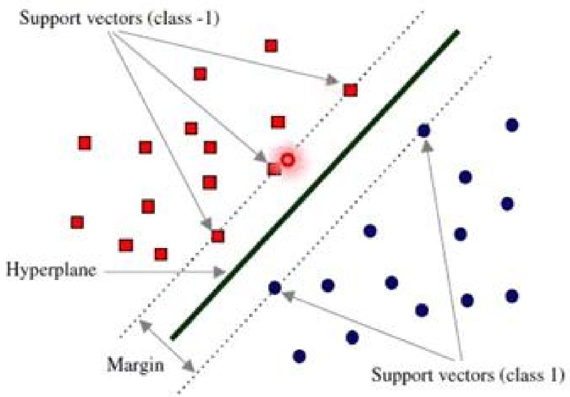
Support Vector Machine Classifier

Hyperplane:

Hyperplane is a line (in 2d space) or a plane that separate the data points into 2 classes.

Support Vectors:

Support Vectors are the data points which lie nearest to the hyperplane. If theses data points changes, the position of the hyperplane changes.



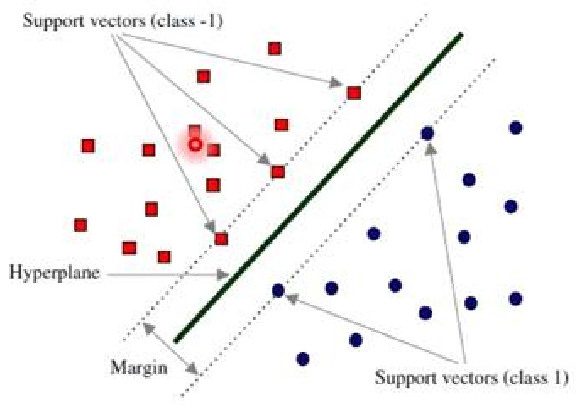
Support Vector Machine Classifier

Hyperplane:

Hyperplane is a line (in 2d space) or a plane that separate the data points into 2 classes.

Support Vectors:

Support Vectors are the data points which lie nearest to the hyperplane. If theses data points changes, the position of the hyperplane changes.



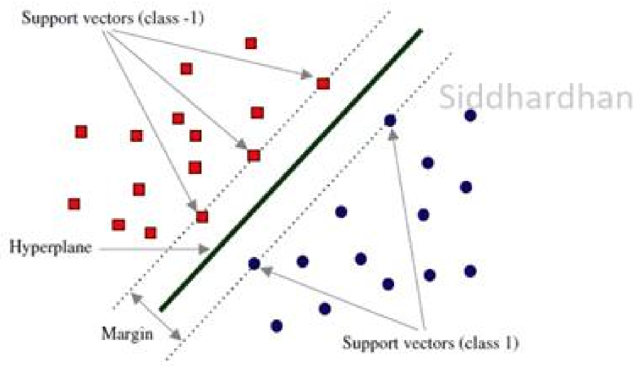
Support Vector Machine Classifier

Advantages:

1. Works well with smaller datasets
2. Works efficiently when there is a clear margin of separation
3. Works well with high dimensional data

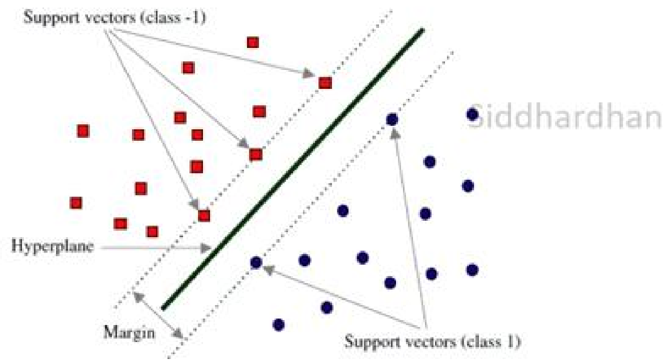
Disadvantages:

1. Not suitable for large datasets as the training time is higher
2. Not suitable for noisier datasets with overlapping classes

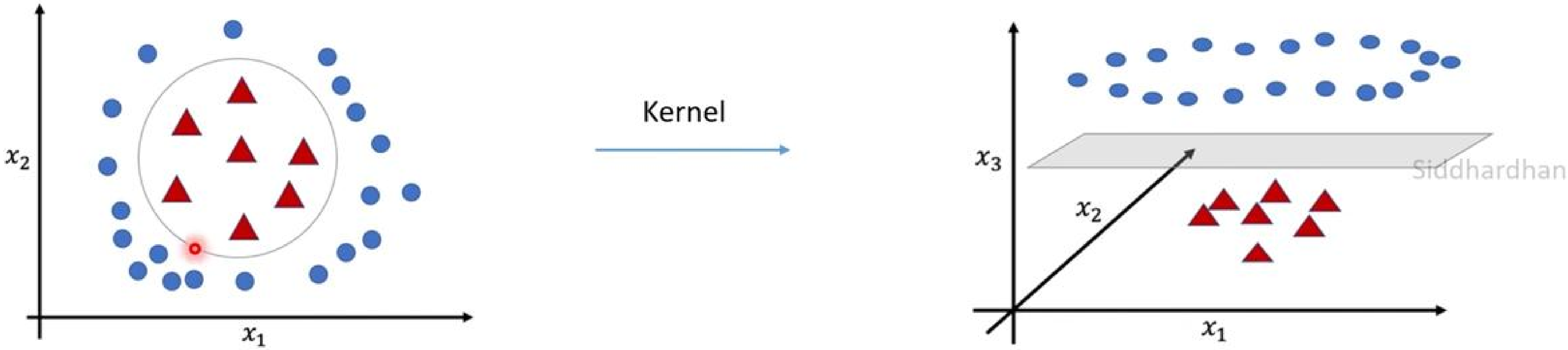


Siddhardhan

Math behind Support Vector Machine (SVM) Classifier



Support Vector Machine Classifier



SVM in 2 dimensions

SVM in 3 dimensions

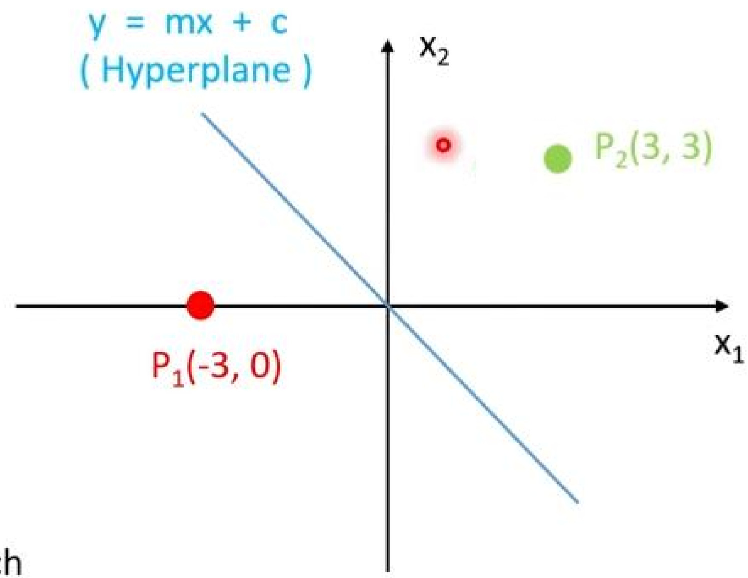
Support Vector Machine Classifier

$P_1(-3, 0)$

$$w^T x = \begin{bmatrix}-1\\0\end{bmatrix}\begin{bmatrix}-3 & 0\end{bmatrix}$$

$$w^T x = 3$$

(Positive)



$P_2(3, 3)$

$$w^T x = \begin{bmatrix}-1\\0\end{bmatrix}\begin{bmatrix}3 & 3\end{bmatrix}$$

$$w^T x = -3$$

(Negative)

Inference: For all the points which lie in the left side of the hyperplane, $w^T x$ value will be Positive

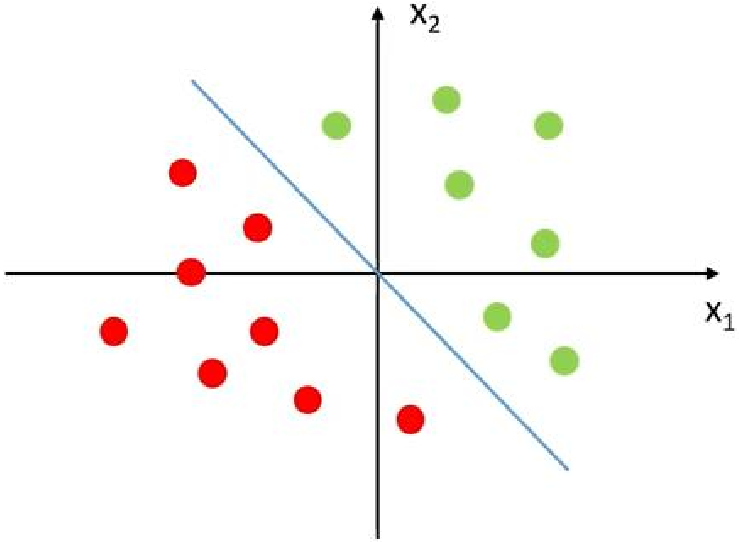
Let slope, m = - 1

Intercept, c = 0

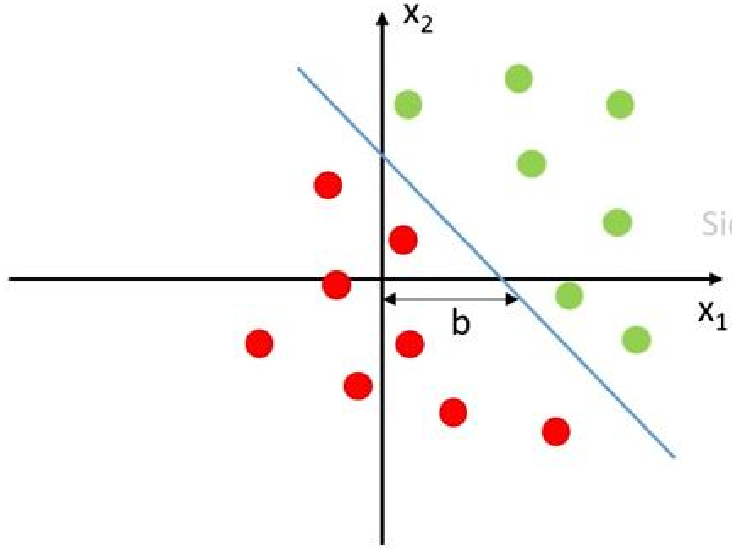
w --> parameters of the line
(m, c) = (-1, 0)

Inference: For all the points which lie in the right side of the hyperplane, $w^T x$ value will be Negative

Support Vector Machine Classifier



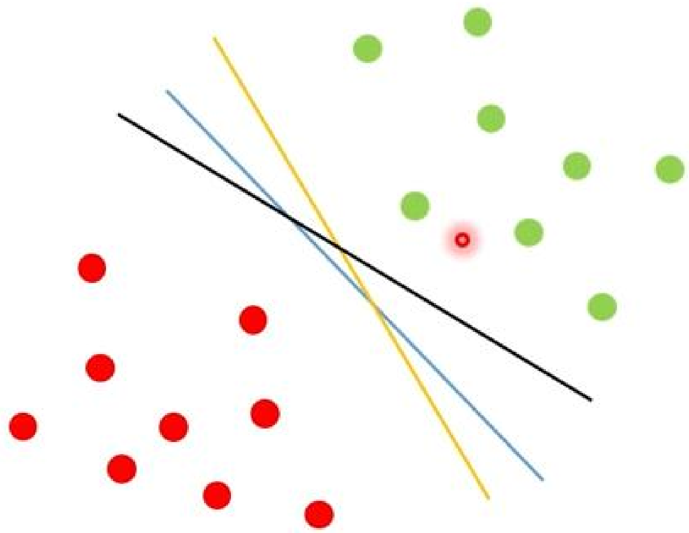
w^Tx = Label



w^Tx + b = Label

Support Vector Machine Classifier

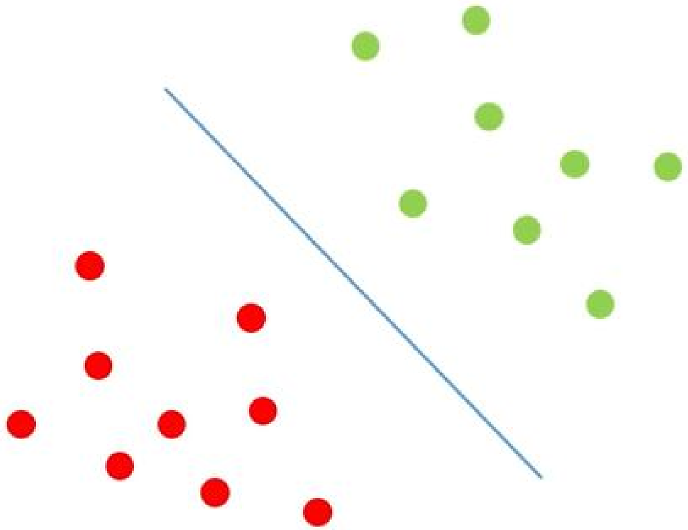
Which is the best Hyperplane?

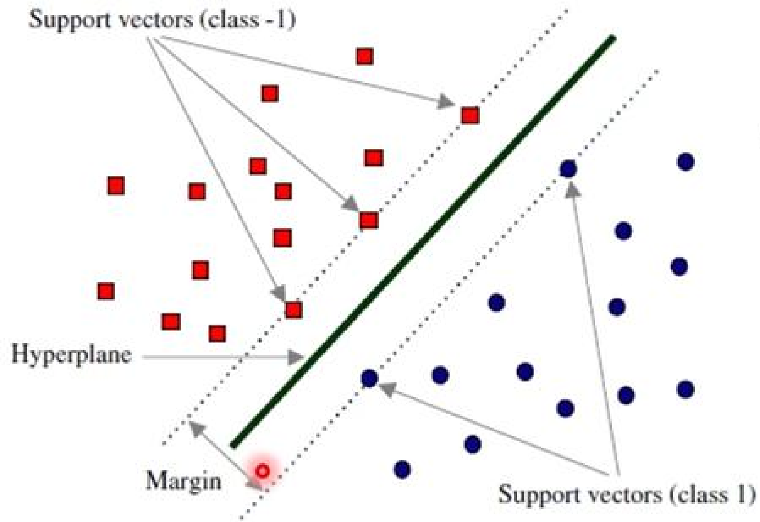


Siddhardhan

Support Vector Machine Classifier

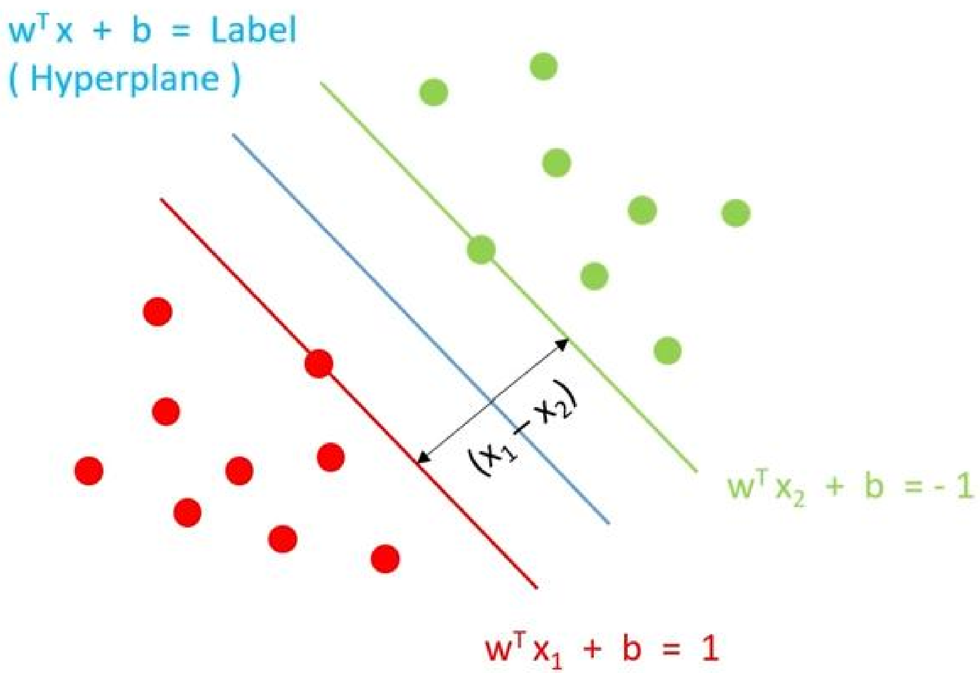
Which is the best Hyperplane?





Support Vector Machine Classifier

Optimization for Maximum margin:



$$w^T x_1 + b = 1$$

$$(-)\quad w^T x_2 \overset{-}{+} b = \overset{+}{-}1$$

$$w^T (x_1 - x_2) = 2$$

Divide by $|| w ||$

(magnitude of the vector)

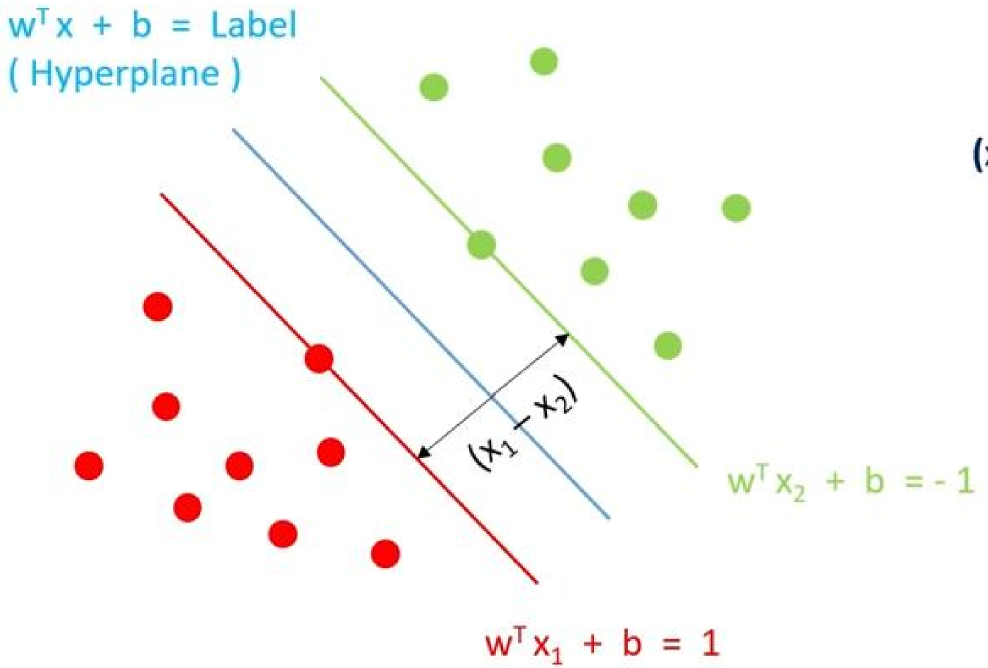
$$\frac{w^T}{|| w ||} (x_1 - x_2) = \frac{2}{|| w ||}$$

$$\mathbf{(x_1 - x_2) = \frac{2}{|| w ||}} \quad \text{(margin)}$$

Support Vector Machine Classifier

Optimization for Maximum margin:

$$y_i = \begin{cases} -1, & w^T x_1 + b \leq -1 \\ 1, & w^T x_1 + b \geq 1 \end{cases} \qquad \text{(Label)}$$



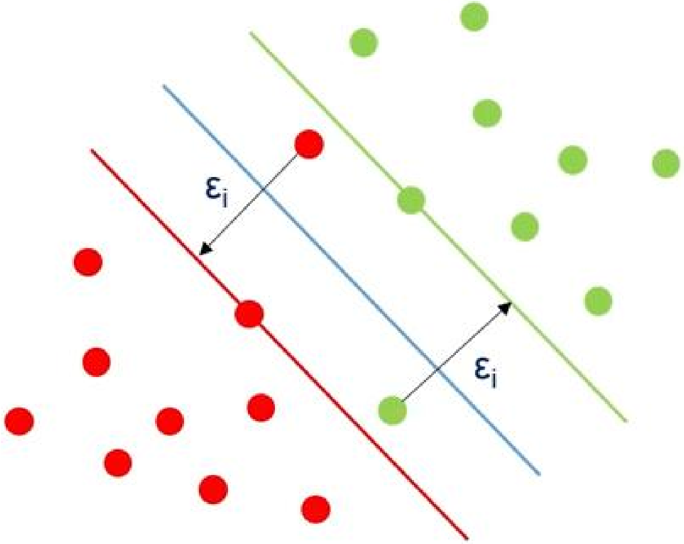
$$(x_1 - x_2) = \frac{2}{||w||} \qquad \text{(margin)}$$

$$\max \left(\frac{2}{||w||} \right) \quad \text{Such that,}$$

$$y_i = \begin{cases} -1, & w^T x_1 + b \leq -1 \\ 1, & w^T x_1 + b \geq 1 \end{cases}$$

Support Vector Machine Classifier

Maximum margin without overfitting:



$$\max \left(\frac{2}{||\mathbf{w}||} \right) \quad \text{Such that,}$$

$$y_i = \begin{cases} -1, & w^T x_1 + b \leq -1 \\ 1, & w^T x_1 + b \geq 1 \end{cases}$$

Siddhardhan

$$\min \left(\frac{||\mathbf{w}||}{2} \right) + c * \sum \varepsilon_i$$

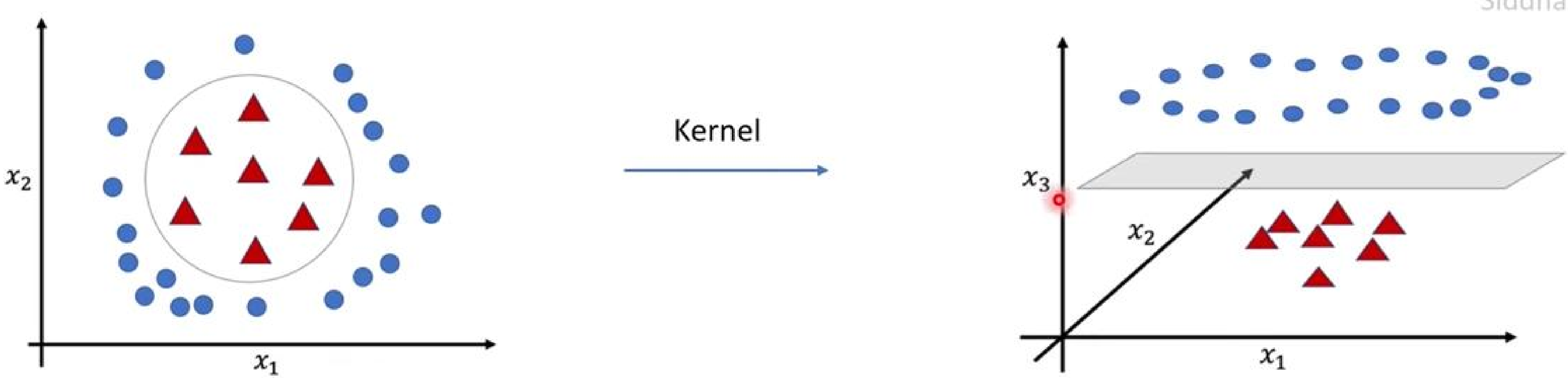
c --> Number of errors

ε_i --> Error magnitude

SVM Kernel

SVM Kernel :

Kernel Function generally transforms the training set of data so that a non-linear decision surface can be transformed to a linear equation in a higher number of dimension spaces. It returns the inner product between two points in a standard feature dimension.



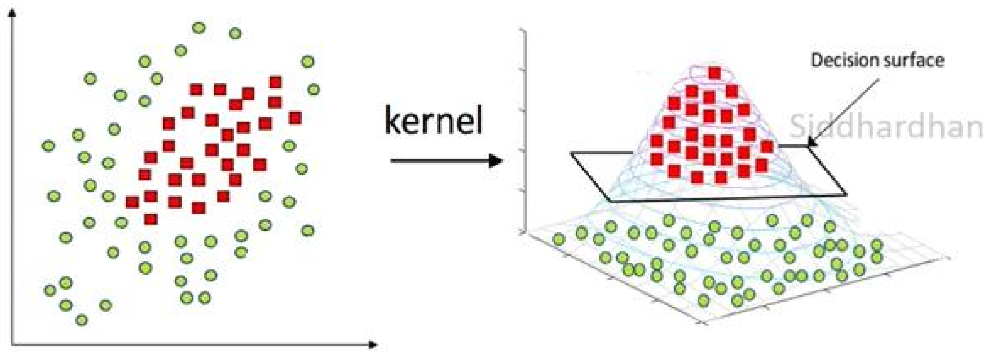
SVM in 2 dimensions

SVM in 3 dimensions

SVM Kernels

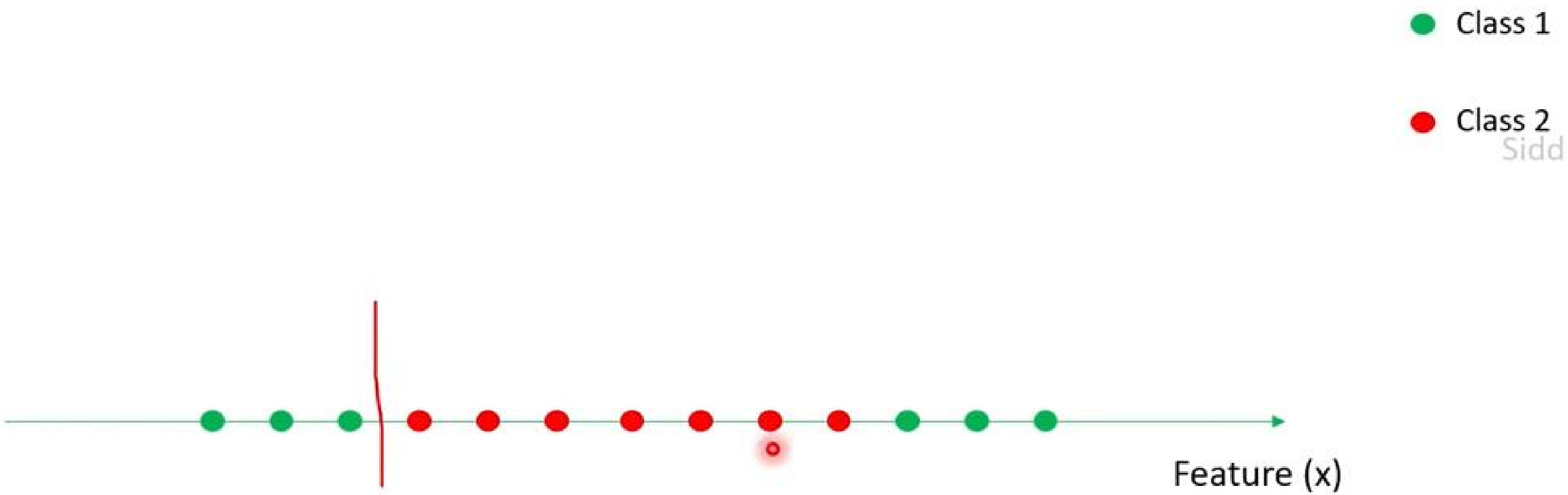
Types of SVM Kernels :

1. Linear
2. Polynomial
3. Radial Basis Function (rbf)
4. Sigmoid



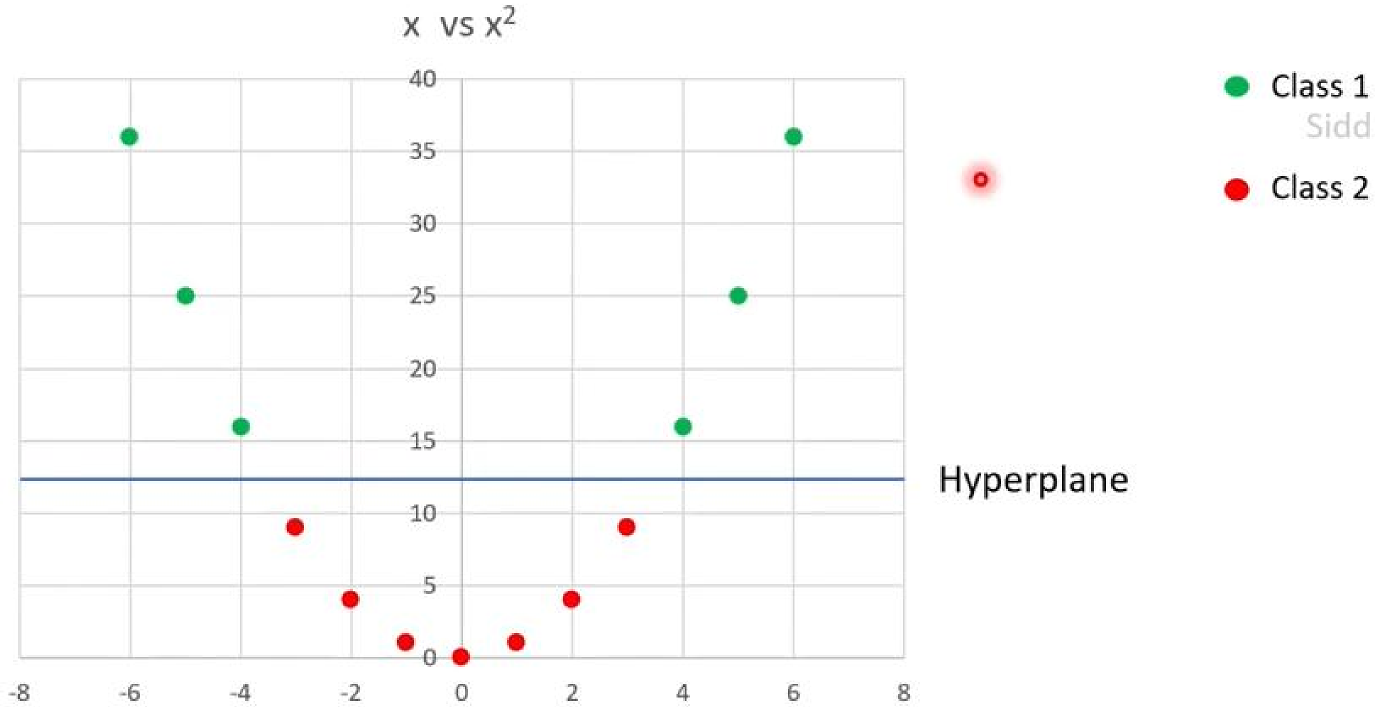
SVM Kernels

Feature (x)	-6	-5	-4	-3	-2	-1	0	1	2	3	4	5	6



SVM Kernels

Feature (x)	-6	-5	-4	-3	-2	-1	0	1	2	3	4	5	6
x^2	36	25	16	9	4	1	0	1	4	9	16	25	36



Types of SVM Kernels

1. Linear Kernel :

$K(x_1, x_2) = x_1^T x_2$

2. Polynomial Kernel:

$K(x_1, x_2) = (x_1^T x_2 + r)^d$

3. Radial Basis Function (rbf) Kernel :

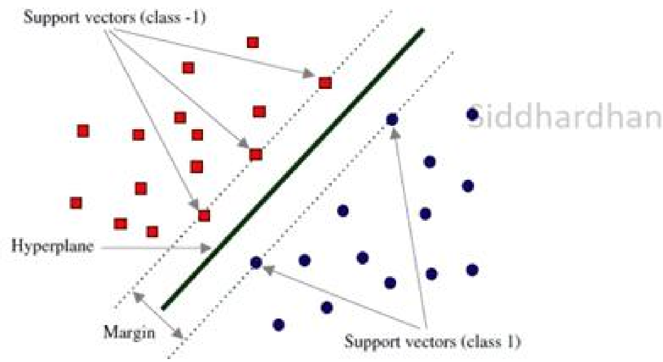
$K(x_1, x_2) = \exp(-\gamma . || x_1 - x_2 ||^2)$

4. Sigmoid Kernel :

$K(x_1, x_2) = \tanh(\gamma . x_1^T x_2 + r)$

Siddhardhan

Loss Function for Support Vector Machine Classifier



Loss Function

Loss function measures how far an estimated value is from its true value.

It is helpful to determine which model performs better & which parameters are better.

$$\textbf{\textit{Loss}} = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2$$

For Support Vector Machine Classifier "Hinge Loss" is used as the Loss Function.

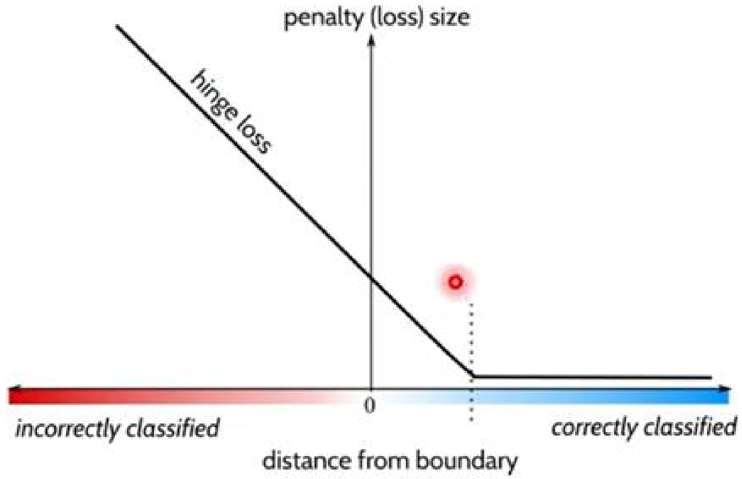
Hinge Loss

Hinge Loss is one of the types of Loss Function, mainly used for maximum margin classification models.

Hinge Loss incorporates a margin or distance from the classification boundary into the loss calculation. Even if new observations are classified correctly, they can incur a penalty if the margin from the decision boundary is not large enough.

$$L = \max(0, 1 - y_i (w^T x_i + b))$$

0 - for correct classification
1 - for wrong classification



Hinge Loss

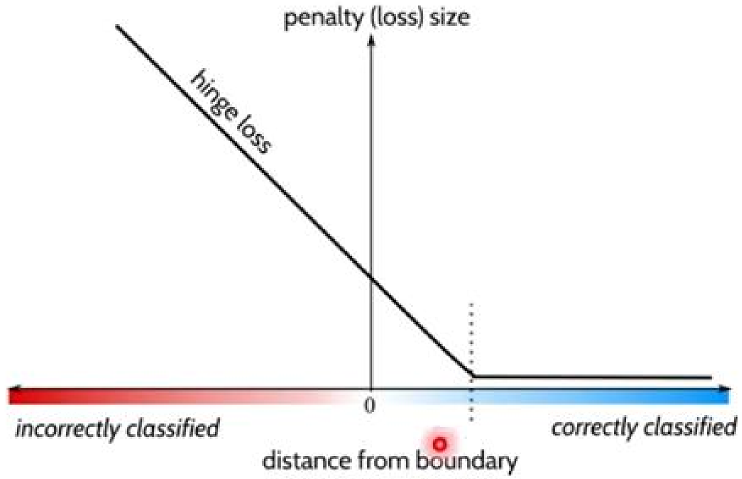
Hinge Loss is one of the types of Loss Function, mainly used for maximum margin classification models.

Hinge Loss incorporates a margin or distance from the classification boundary into the loss calculation. Even if new observations are classified correctly, they can incur a penalty if the margin from the decision boundary is not large enough.

Siddhardhan

$$L = \max(0, 1 - y_i (w^T x_i + b))$$

0 - for correct classification
1 - for wrong classification



Hinge Loss

Misclassification :

$y_i = 1 \quad \hat{y}_i = -1$

$L = (1 - (1)(-1)$

$L = (1 + 1)$

$L = 2$ (High loss Value)

$y_i = -1 \quad \hat{y}_i = 1$

$L = (1 - (-1)(1)$

$L = (1 + 1)$

$L = 2$ (High loss Value)

$$L = \max(0, 1 - y_i (w^T x_i + b))$$

0 - for correct classification

1 - for wrong classification

Correct classification :

$y_i = 1 \quad \hat{y}_i = 1$

$L = (0 - (1)(1)$

$L = (0 - 1)$

$L = -1$ (Low loss Value)

$y_i = -1 \quad \hat{y}_i = -1$

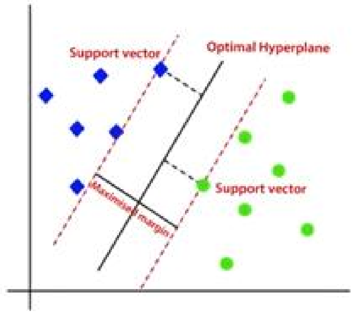
$L = (0 - (-1)(-1)$

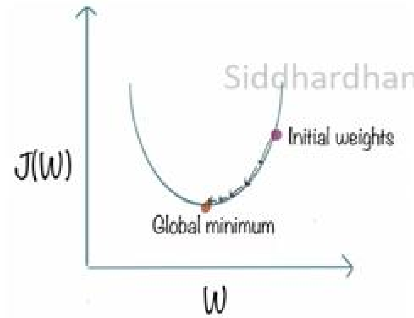
$L = (0 - 1)$

$L = -1$ (Low loss Value)

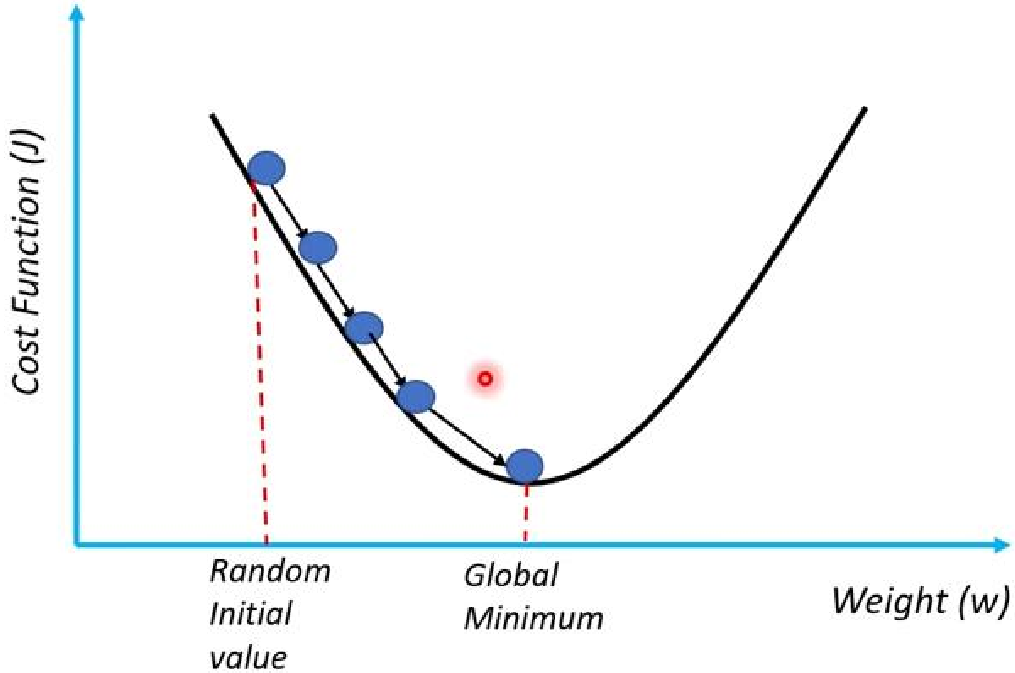
Siddhardhan

Gradient Descent for Support Vector Machine Classifier

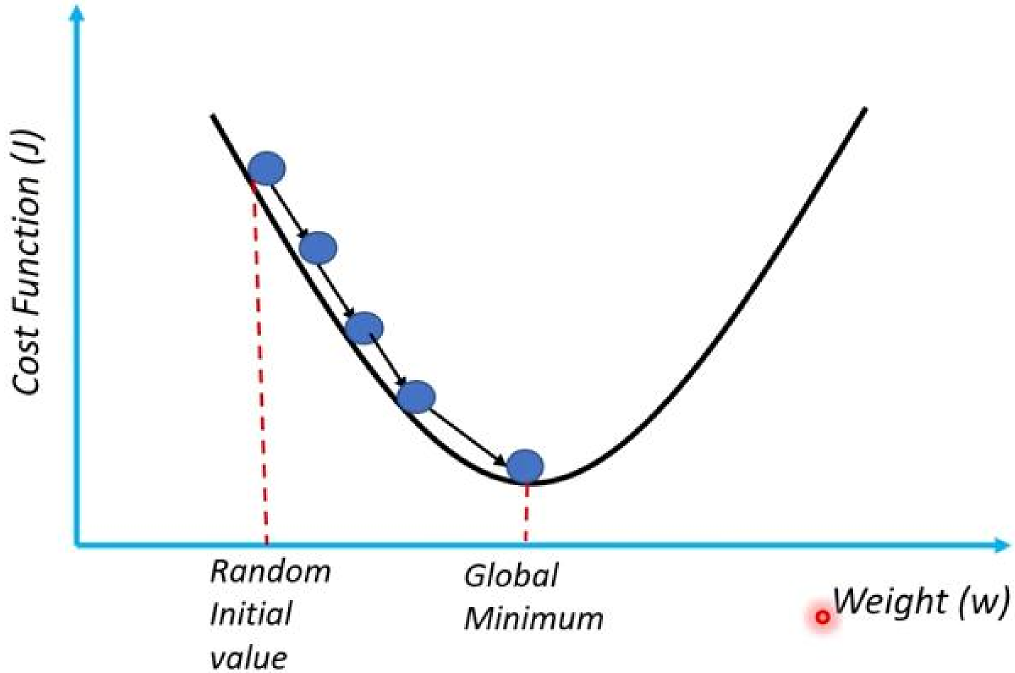




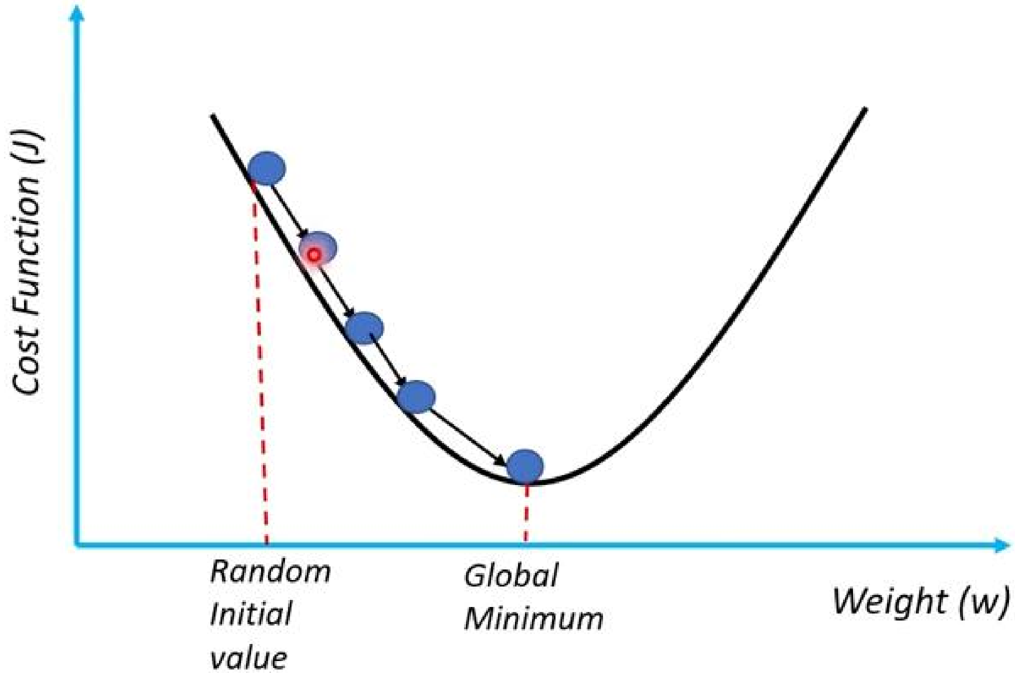
Gradient Descent



Gradient Descent



Gradient Descent



Gradient Descent

Gradient Descent is an optimization algorithm used for minimizing the cost function in various machine learning algorithms. It is used for updating the parameters of the learning model.

$$w_2 = w_1 - L * \frac{dJ}{dw}$$

$$b_2 = b_1 - L * \frac{dJ}{db}$$

w --> weight
b --> bias
L --> Learning Rate

$\frac{dJ}{dw}$ --> Partial Derivative of cost function with respect to w

$\frac{dJ}{db}$ --> Partial Derivative of cost function with respect to b

Gradients for SVM Classifier

if $(y_i . (w.x + b) \geq 1)$ **:**

$$\frac{dJ}{dw} = 2\lambda w$$

$$\frac{dJ}{db} = 0$$

else $(y_i . (w.x + b) < 1)$ **:**

$$\frac{dJ}{dw} = 2\lambda w - y_i . x_i$$

Siddhardhan

$$\frac{dJ}{db} = y_i$$

$$w_2 = w_1 - L * \frac{dJ}{dw}$$

$$b_2 = b_1 - L * \frac{dJ}{db}$$