CS 4104
APPLIED MACHINE LEARNING

Dr. Hashim Yasin

National University of Computer and Emerging Sciences,

Faisalabad, Pakistan.

SVM … PRELIMINARIES

Linear Separable Data

- **Given:** m examples $(x_1, c_1), \dots, (x_m, c_m)$
- **Goal:** Learn classification!
- Most simple case: *binary classification*, where each example shows
  - n-dimensional input data vector $\mathbf{x}_i = (x_{i,1}, \dots, x_{i,n})$ and
  - its binary classification $c_i \in \{+1, -1\}$
- e.g., classification of all web pages into "*related to computer science*" and "*not related to computer science*":
  - **Given:** data vectors $\boldsymbol{x}_i$ with binary elements $\boldsymbol{x}_{ij}$ for appearance or missing appearance of a relevant keyword.
  - **Goal:** classification of new web pages with small prediction error

Linear Separable Data

- Classification of n-dimensional input data

$$\boldsymbol{x} = (x_1, \dots, x_n)$$

is possible using a linear separation function

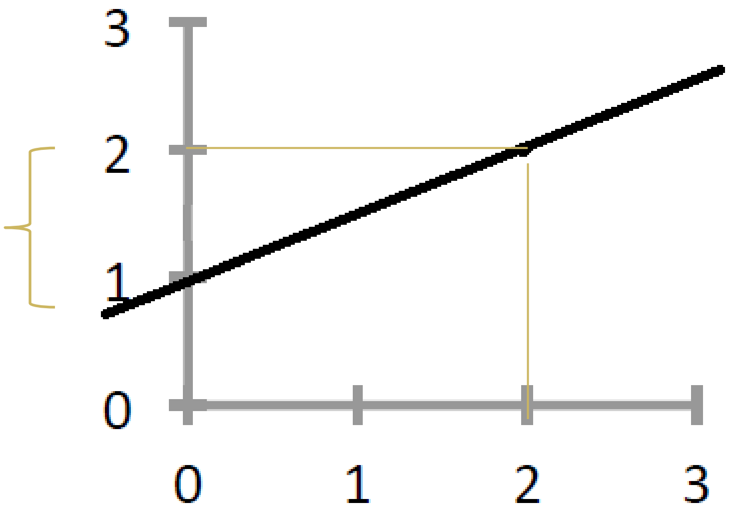
$$f: \mathbb{R}^n \rightarrow \mathbb{R}$$

- $\boldsymbol{x}$ is classified as positive $(c = +1)$, if $f(\boldsymbol{x}) \geq 0.$
- $\boldsymbol{x}$ is classified as negative $(c = -1)$, is $f(\boldsymbol{x}) < 0.$

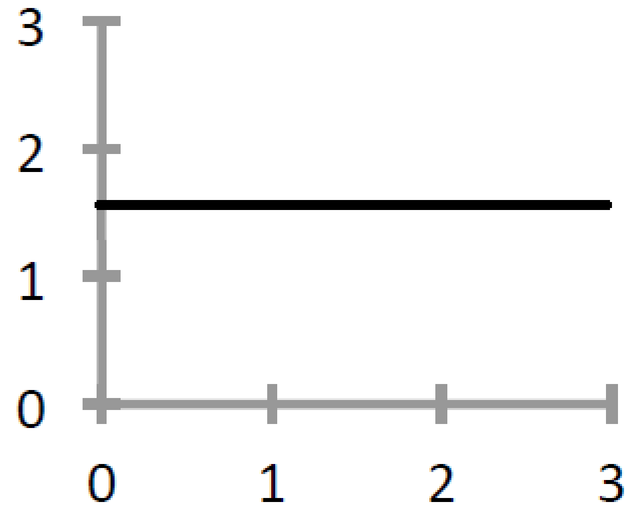
Linear Separable Data

$$y = \theta_0 + \theta_1 x$$

$$\theta = \frac{change\ in\ Y}{change\ in\ X}$$



$$\theta_1 = 0.5, \theta_0 = 1$$

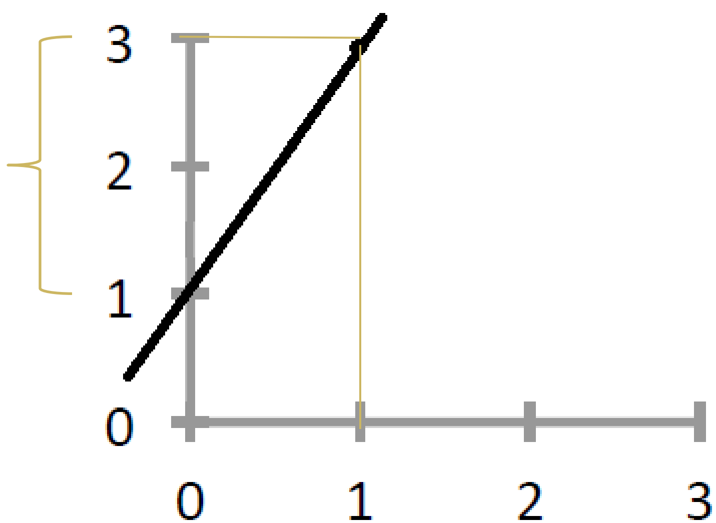


$$\theta_1 = 0, \theta_0 = 1.5$$

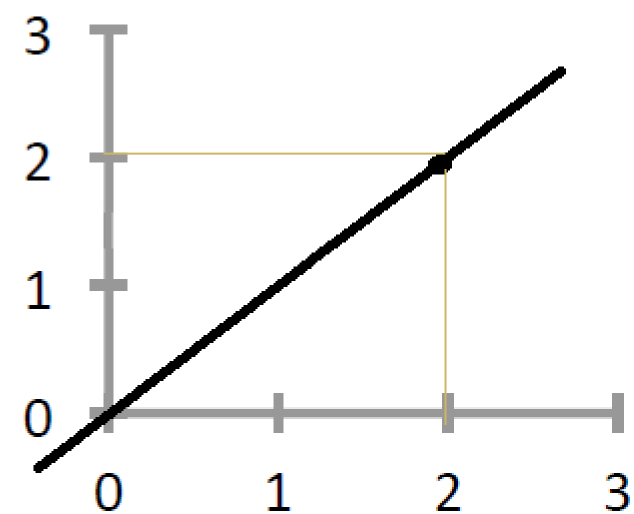
Linear Separable Data

$$y = \theta_1 x + \theta_0$$

$$\theta = \frac{change\ in\ Y}{change\ in\ X}$$



$$\theta_1 = 2, \theta_0 = 1$$



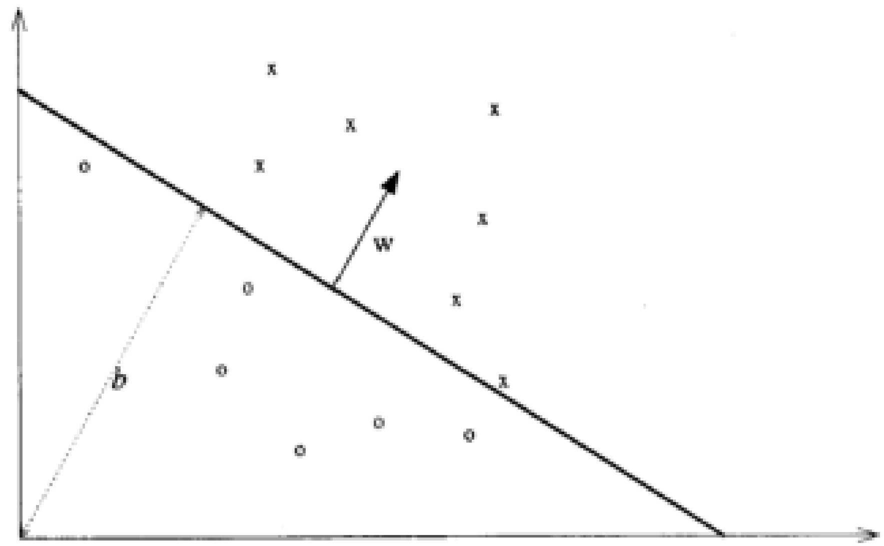
$$\theta_1 = 1, \theta_0 = 0$$

Linear Separable Data

The data can be separated in the n-dimensional data space by a planar hyper plane $\langle \boldsymbol{w},\boldsymbol{x}\rangle - b = 0$.

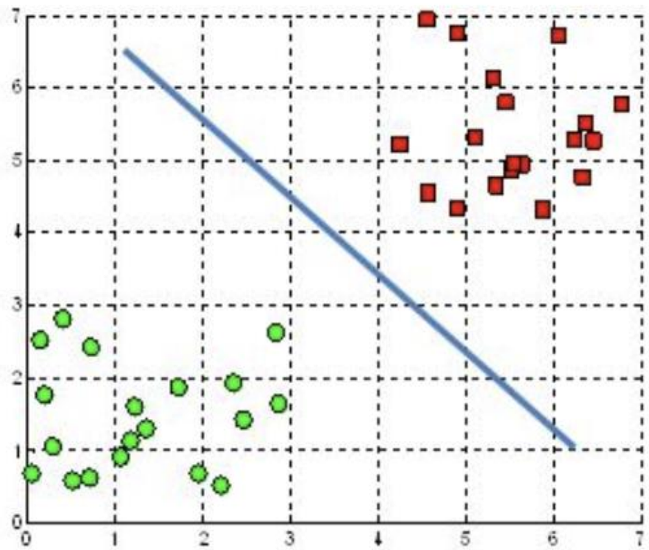
Parameter $\boldsymbol{w}$ defines the normal vector of the hyper plane, parameter b stands for the bias of the hyper plane.

Planer hyper plane
$\langle \boldsymbol{w}, \boldsymbol{x}\rangle - b = 0$

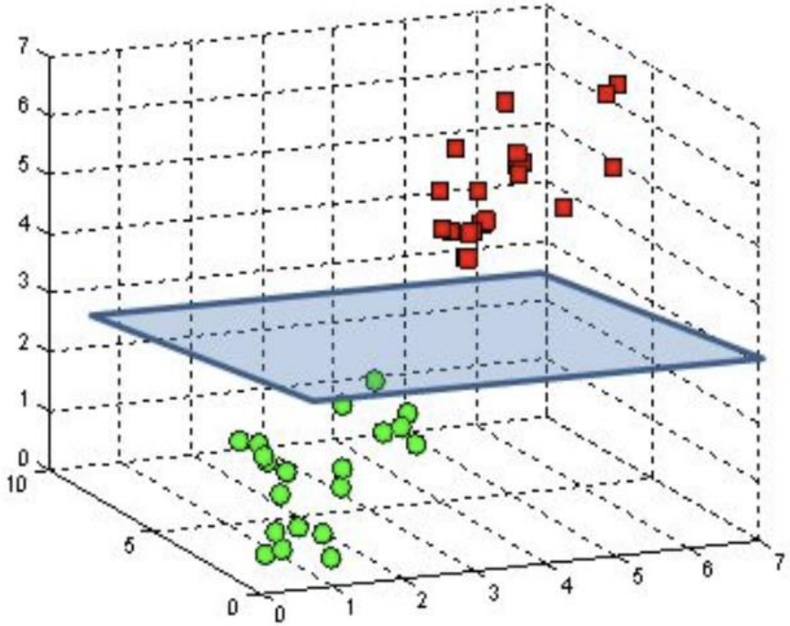


Hyperplane

A hyperplane in $\mathbb{R}^2$ is a line

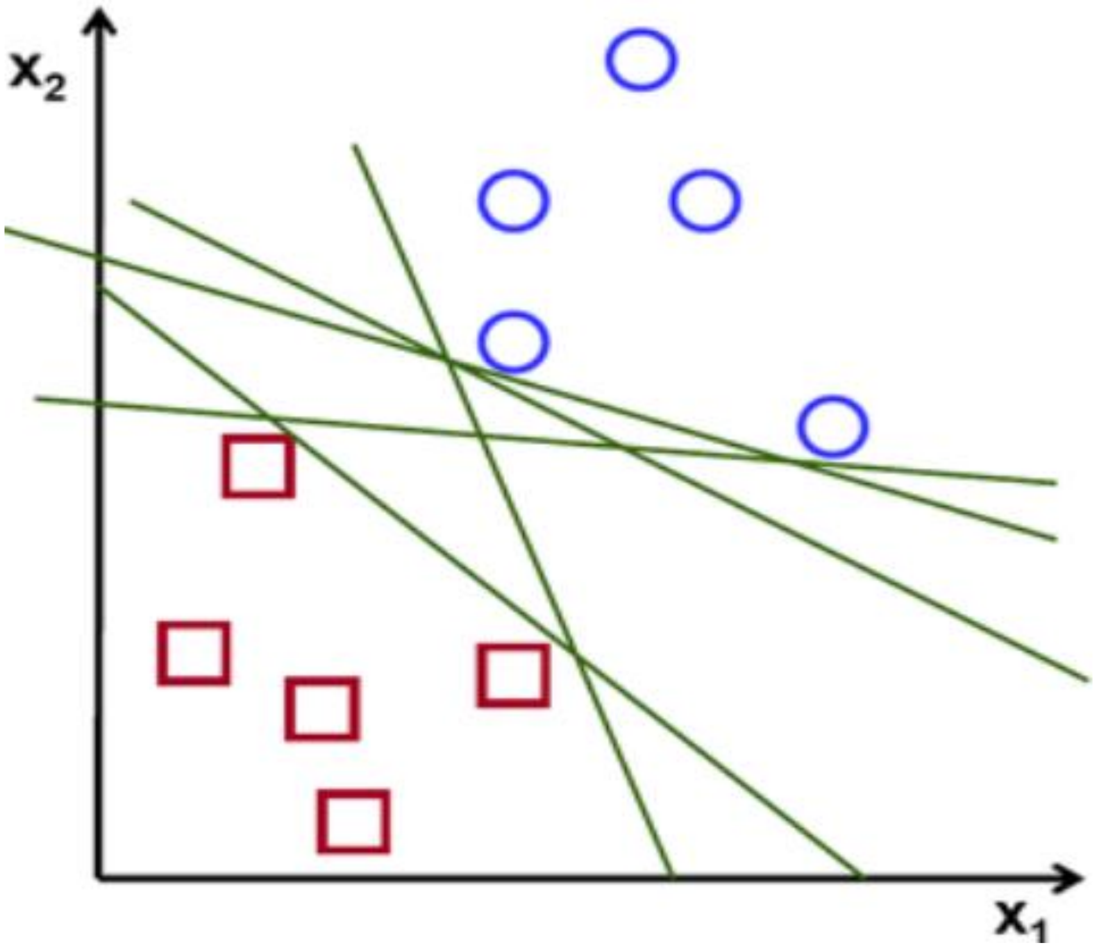


A hyperplane in $\mathbb{R}^3$ is a plane



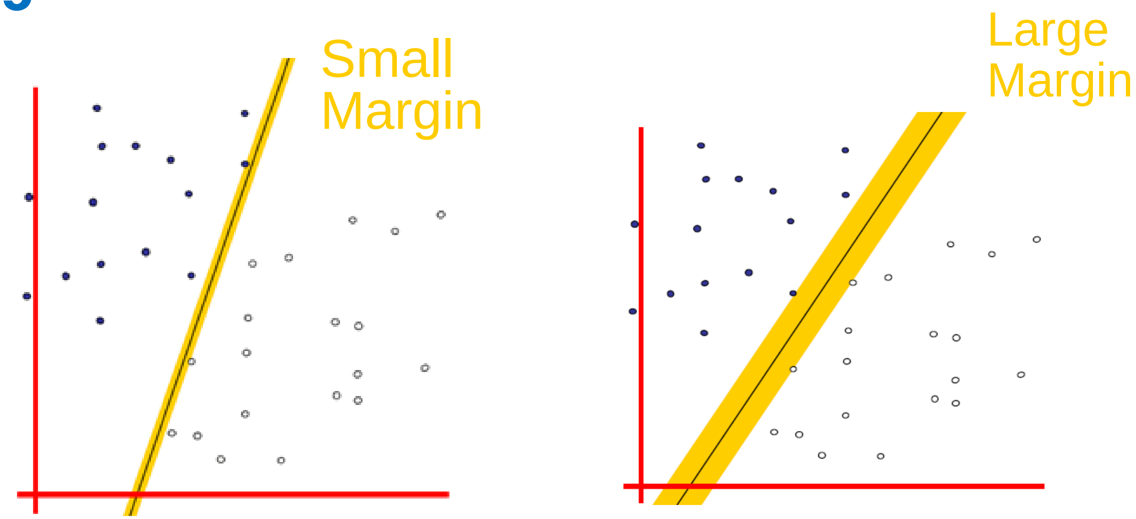
SUPPORT VECTOR MACHINE

Support Vector Machine (SVM)



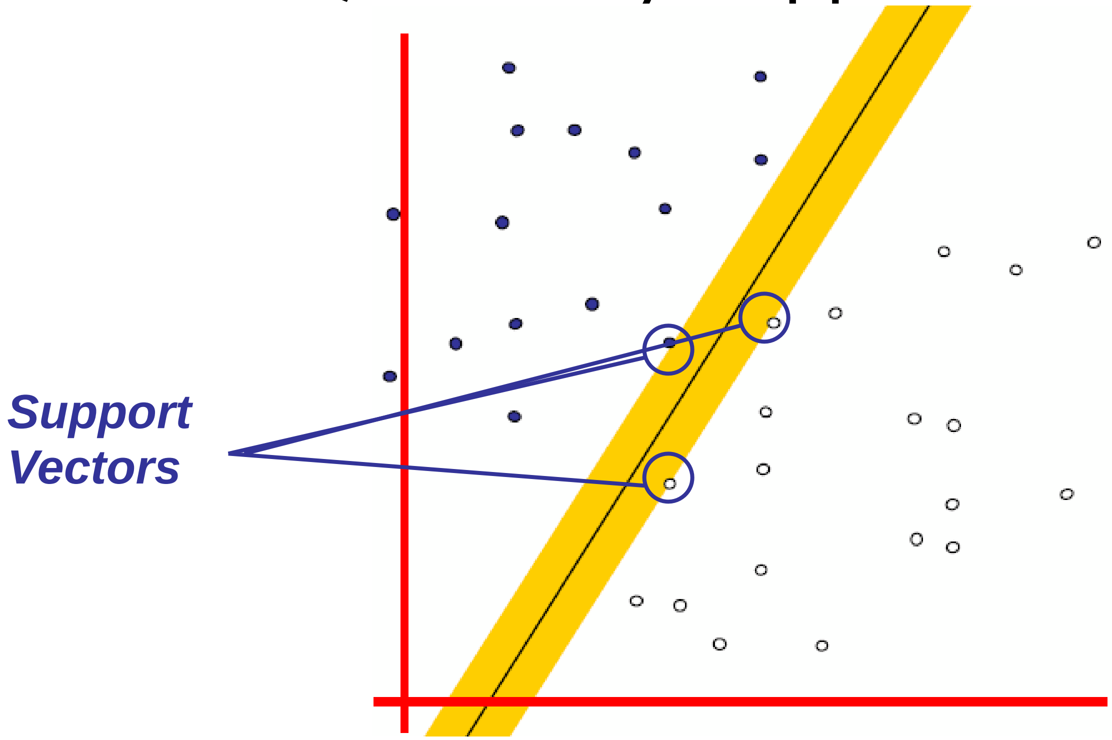
Support Vector Machine (SVM)

- Derives a linear separator
- informally chooses that separating hyper plane that maximizes the so-called *margin*, *i.e., the space between positive and negative examples*
- → ***Maximum Margin Classifier***



Support Vector Machine (SVM)

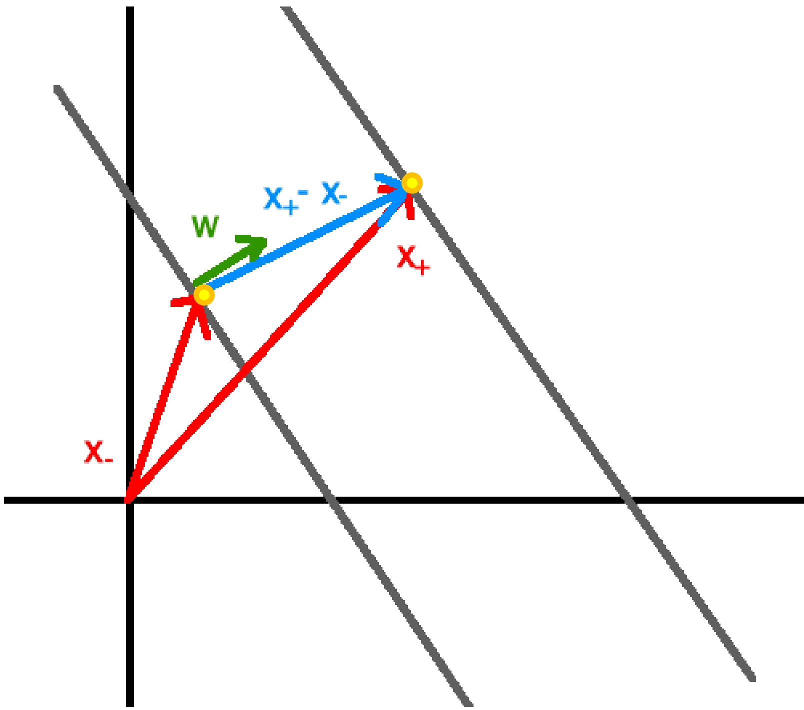
- That points closest to the separator are called ***Support Vectors***, since they "support" the separator.



Support Vector Machine (SVM)

- To get an equation for the width of the margin,
  - subtract the first support vector from the second support vector and
  - multiply the result by the unit vector of $w = \frac{\vec{w}}{\|w\|}$ which is always perpendicular to the decision boundary.

$$width = (\vec{x_+} - \vec{x_-}) \cdot \frac{\vec{w}}{\|w\|}$$



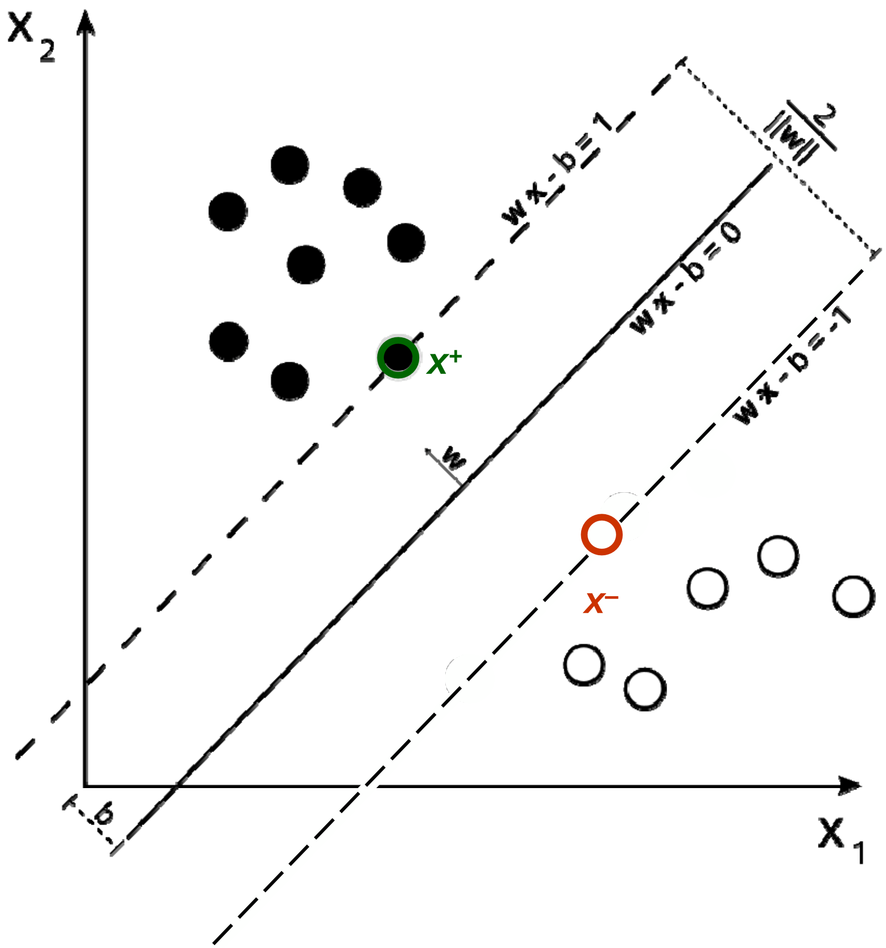
Support Vector Machine (SVM)

- First, we scale $f(\mathbf{x}) = \mathbf{w}\cdot\mathbf{x} - b$ that way, that the values of the support vectors are +1 and –1, respectively.
- Thus, the width M of the margin can be expressed as a function of $\mathbf{w}$.
- $\langle \mathbf{w}, \mathbf{x}^+ \rangle - b = +1 \qquad \langle \mathbf{w}, \mathbf{x}^- \rangle - b = -1$

  $\rightsquigarrow \langle \mathbf{w}, (\mathbf{x}^+ - \mathbf{x}^-) \rangle = 2$

  $\rightsquigarrow M = \langle (\mathbf{w}/\|\mathbf{w}\|), (\mathbf{x}^+ - \mathbf{x}^-) \rangle$

  $= 2/\|\mathbf{w}\|$



Support Vector Machine (SVM)

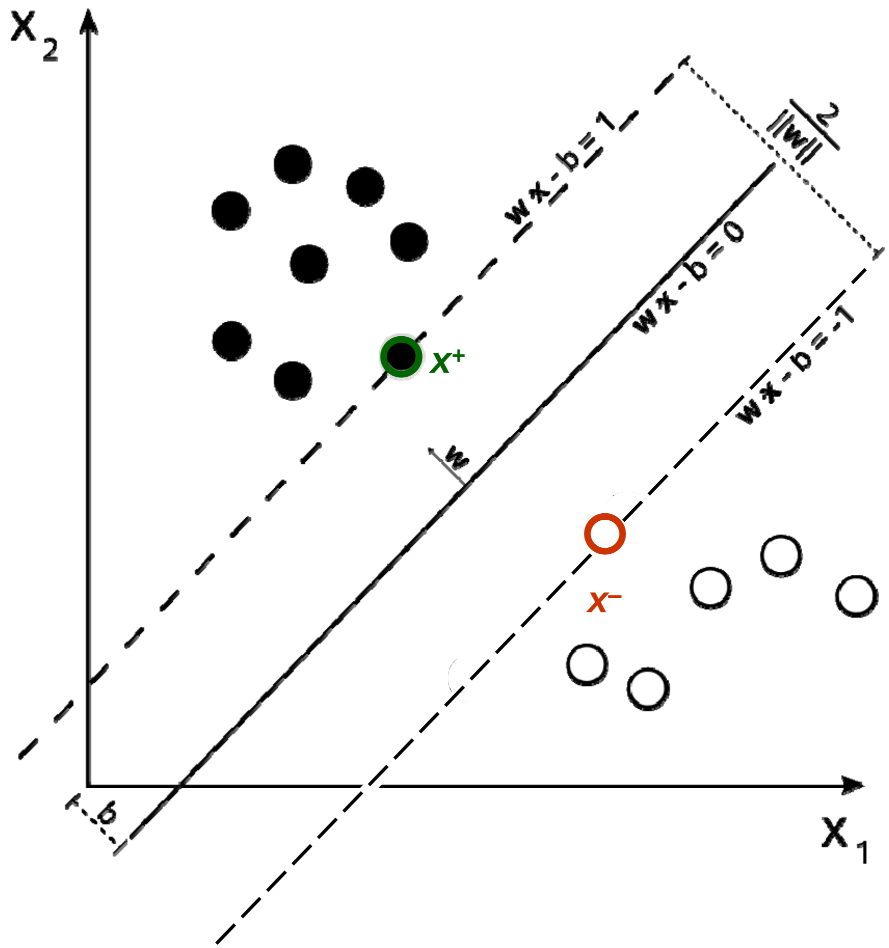
- $\langle \mathbf{w}, \boldsymbol{x}^{+} \rangle - b = +1 \qquad \langle \mathbf{w}, \boldsymbol{x}^{-} \rangle - b = -1$

  $\sim \langle \mathbf{w}, (\boldsymbol{x}^{+} - \boldsymbol{x}^{-}) \rangle = 2$

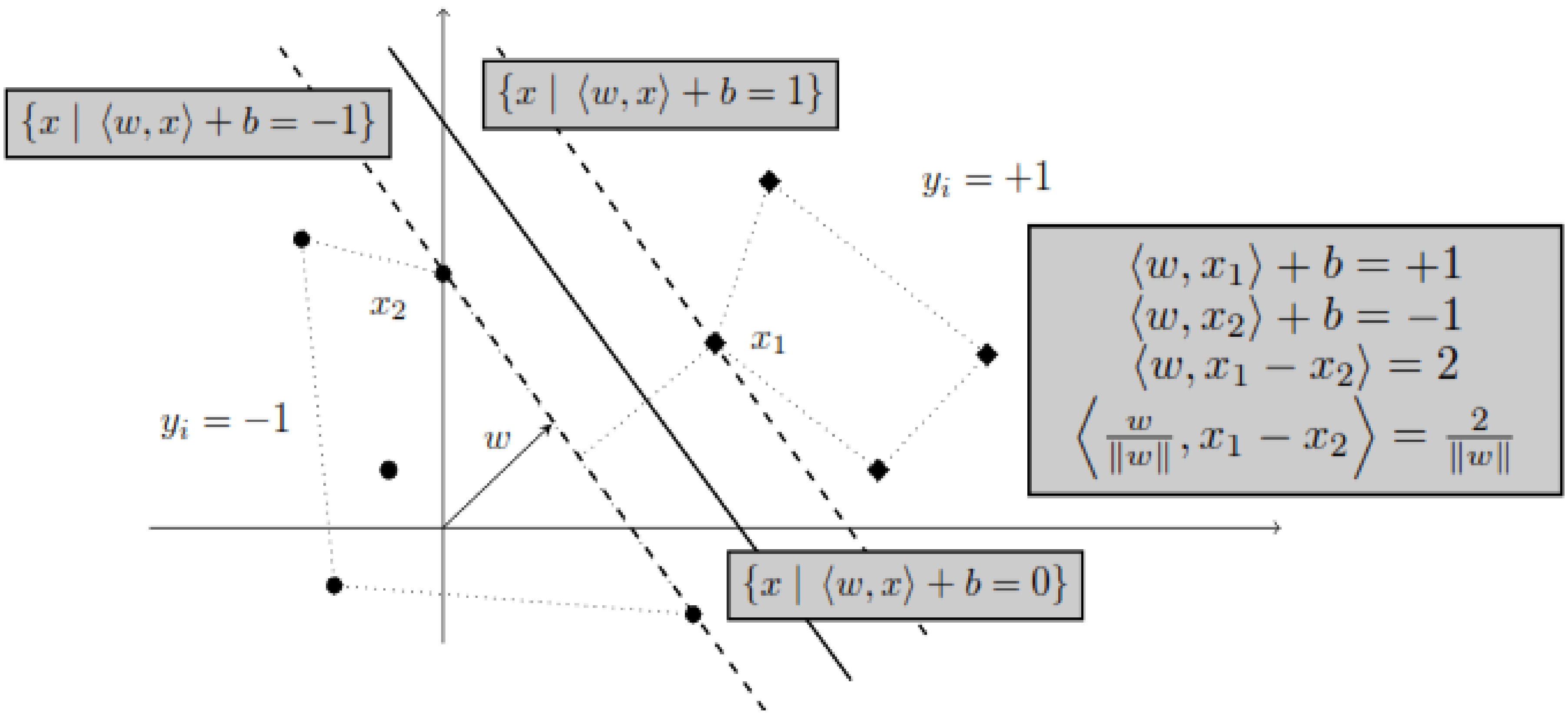
  $\sim M = \langle (\mathbf{w}/\|\mathbf{w}\|), (\boldsymbol{x}^{+} - \boldsymbol{x}^{-}) \rangle$

  $= 2/\|\mathbf{w}\|$

- Maximizing $M = 2/\|w\|$ is equal to minimize $\|\mathbf{w}\|/2$.
- With subject to the constraint, that all examples of the training data are correctly classified.



Support Vector Machine (SVM)



Support Vector Machine (SVM)

- The problem of maximizing the margin therefore reduces to

$$\max_{w,b} \quad \frac{1}{||w||}$$
$$\text{s.t.} \quad y_i(\langle w, x_i \rangle + b) \geq 1 \text{ for all } i,$$

Support Vector Machine (SVM)

- The problem of maximizing the margin therefore reduces to

$$\max_{w,b} \frac{1}{\|w\|}$$
$$\text{s.t.} \quad y_i(\langle w, x_i \rangle + b) \geq 1 \text{ for all } i,$$

- Which is equivalently to

$\boxed{min\ \frac{1}{2}\|w\|^2 \quad \text{since} \quad \frac{\mathrm{d}}{\mathrm{d}x}\frac{1}{2}x^2 = x}$

$$\min_{w,b} \frac{1}{2}\|w\|^2$$
$$\text{s.t.} \quad y_i(\langle w, x_i \rangle + b) \geq 1 \text{ for all } i.$$

Support Vector Machine (SVM)

The picture depicts

- the well classified points in black,

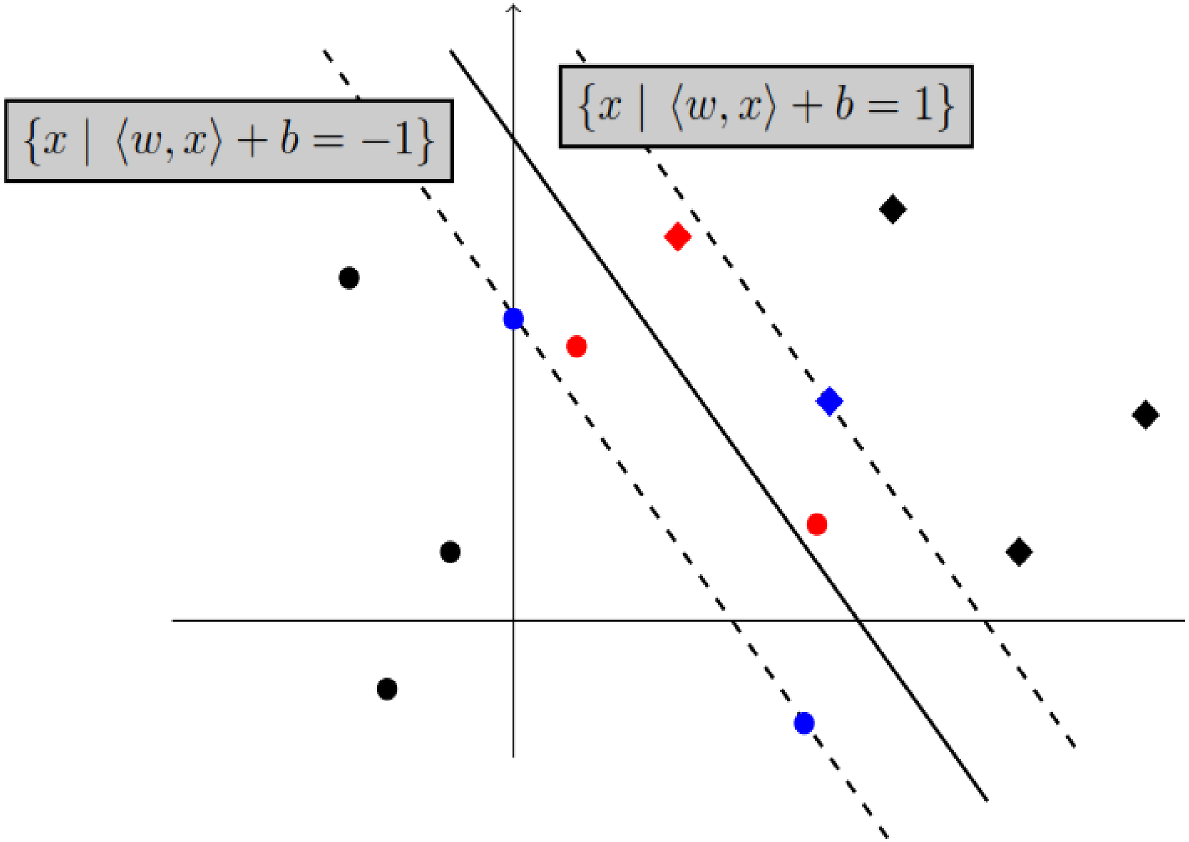
$$y_i(\langle w, x_i \rangle + b) > 1$$

- the support vectors in blue,

$$y_i(\langle w, x_i \rangle + b) = 1$$

- margin errors in red.

$$y_i(\langle w, x_i \rangle + b) < 1$$



Support Vector Machine (SVM)

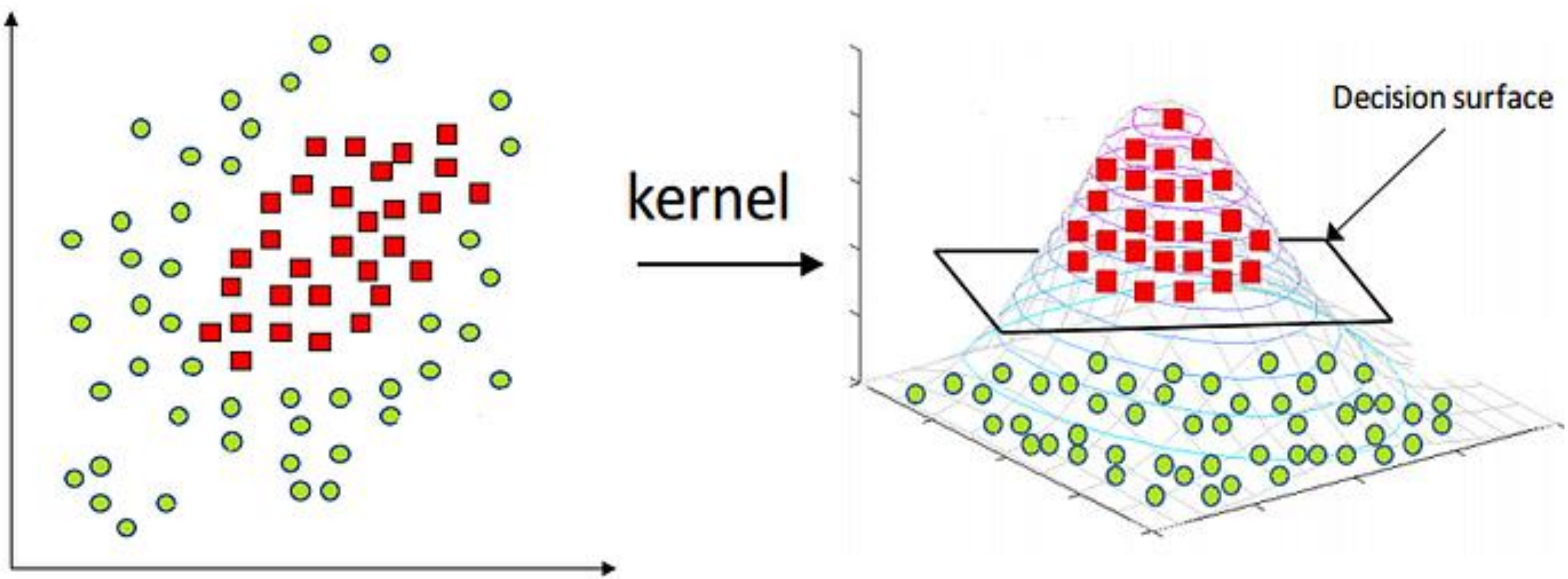
- This is a ***constrained convex optimization problem*** with a quadratic objective function and linear constraints.
- In deriving this equation, we implicitly assume that the ***data is linearly separable***, that is, there is a hyperplane that correctly classifies the training data.
- Such a classifier is called a ***hard margin classifier***.
- If the ***data is not linearly separable***, then does not have a solution.

SVM … KERNEL TRICK

SVM … Kernel Tricks

- Kernel Trick is widely used in the Support Vector Machines (SVM) model to bridge linearity and non-linearity.
- It converts non-linear lower-dimension space to a higher dimension space thereby we can get a linear classification.
  - So, we are projecting the data with some extra features so that it can convert to a higher dimension space.

SVM … Kernel Tricks



Kernel Function:

- A function that takes as its inputs vectors in the original space and returns the dot product of the vectors in the feature space is called a *kernel function*
- More formally, if we have data $\mathbf{x}, \mathbf{z} \in X$ and a map $\phi : X \to \Re^N$ then

$$k(\mathbf{x}, \mathbf{z}) = \langle \phi(\mathbf{x}), \phi(\mathbf{z}) \rangle$$

is a kernel function

SVM … Kernel Tricks

- Let us consider a simple kernel which is:

$$K(x, y) = \langle f(x), f(y) \rangle$$

where,

  - K is the kernel function,
  - X and Y are the dimensional inputs,
  - f is the map from n-dimensional to m-dimensional space and,
  - $\langle x, y \rangle$ is the dot product.

SVM … Kernel Tricks

- Let us say that we have two points,
  - $x = (5, 6, 7)$ and $y = (8, 9, 10)$

- As we have seen, $K(x, y) = \langle f(x), f(y) \rangle$, let us first calculate $\langle f(x), f(y) \rangle$

SVM … Kernel Tricks

f(x) = (x1x1, x1x2, x1x3, x2x1, x2x2, x2x3, x3x1, x3x2, x3x3)

f(y) = (y1y1, y1y2, y1y3, y2y1, y2y2, y2y3, y3y1, y3y2, y3y3)

So,

f(5,6,7) = (25,30,35,30,36,42,35,42,49) and

f(8,9,10) = (64,72,80,72,81,90,80,90,100)

So the dot product is,

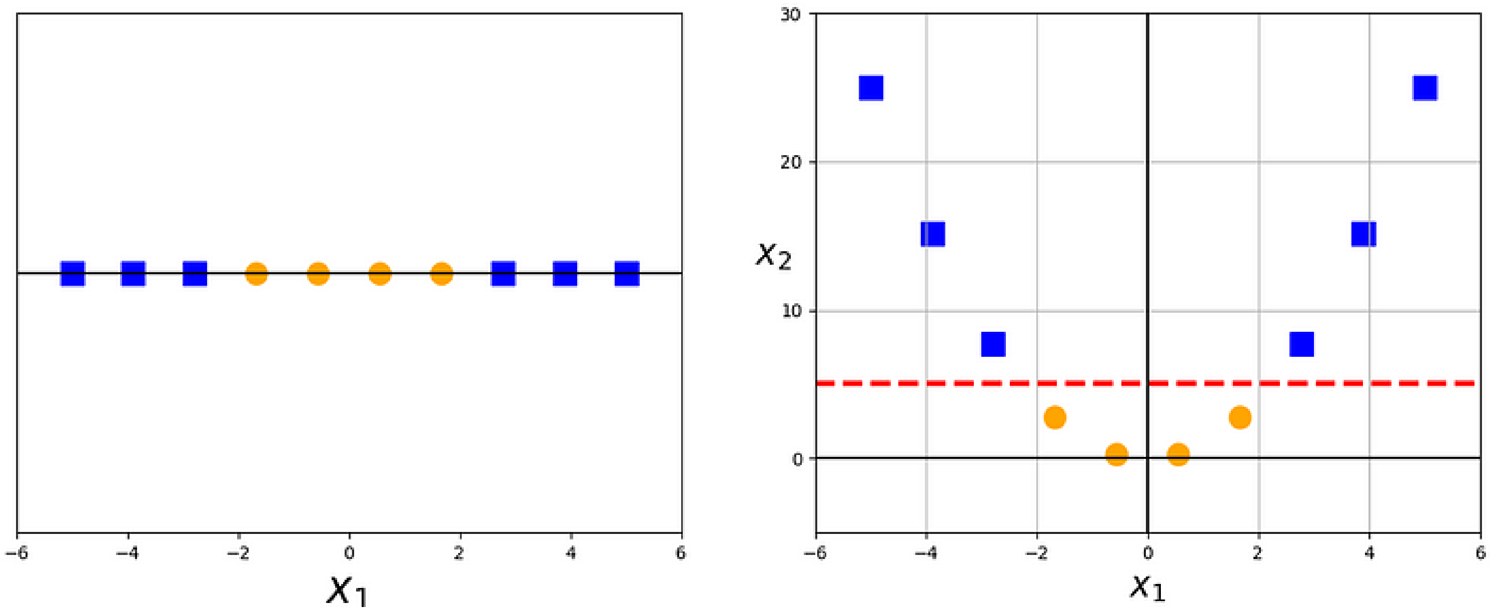
f (x) . f (y) = f(5,6,7) . f(8,9,10) =

(1600 + 2160 + 2800 + 2160 + 2916 + 3780 + 2800 + 3780 + 4900) = 26,896

Using Kernel,

K(x, y) = (5*8 + 6*9 + 7*10) ^ 2 = (40 + 54 + 70) ^ 2 = 164*164 = 26,896

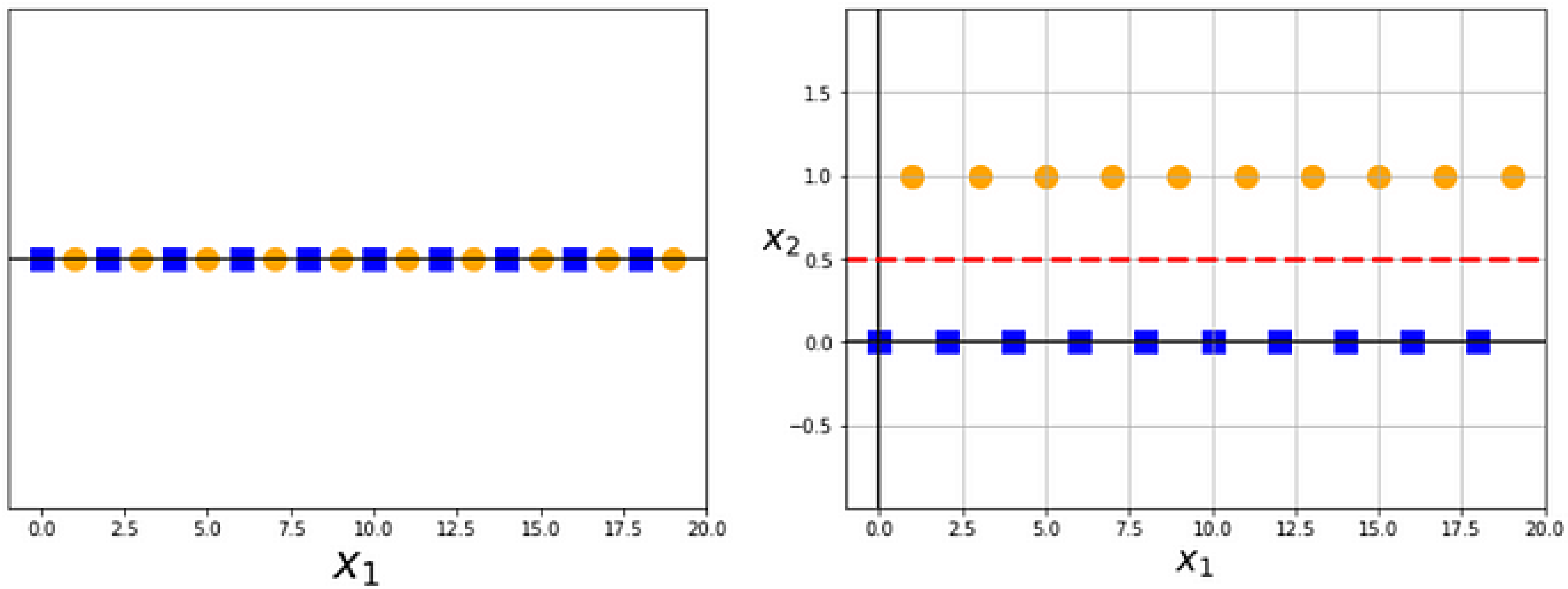
SVM … Kernel Tricks



This data becomes linearly separable after a quadratic transformation to 2-dimensions.

In 1-dimension, this data is not linearly separable, but after applying the transformation $\phi(x) = x^2$ and adding this second dimension to our feature space, the classes become linearly separable.

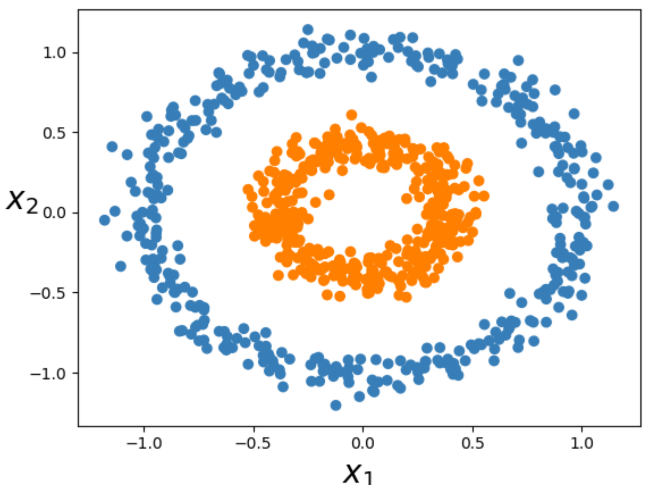
SVM … Kernel Tricks

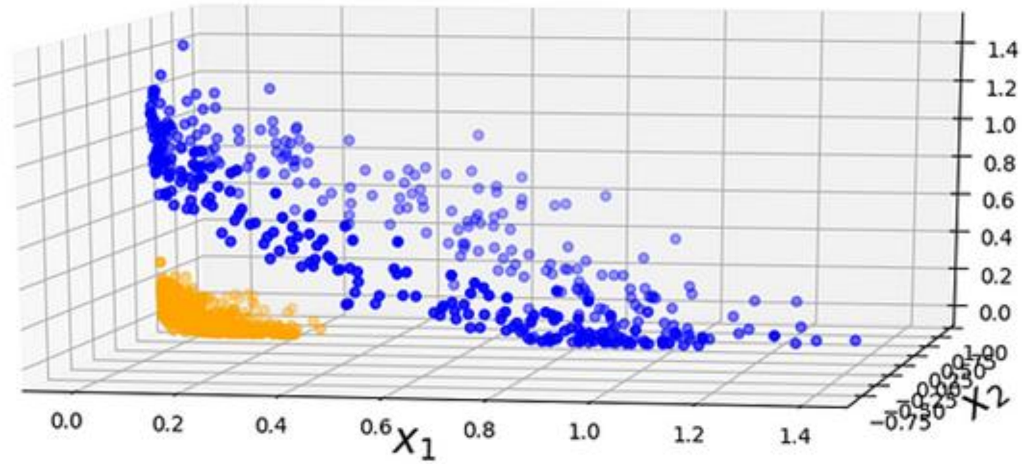


This transformation allows us to linearly separate the even and odd X1 values in 2 dimensions.

In 1-dimension, this data is not linearly separable, but after applying the transformation $\phi(x) = x \bmod 2$ to our feature space, the classes become linearly separable.

SVM … Kernel Tricks



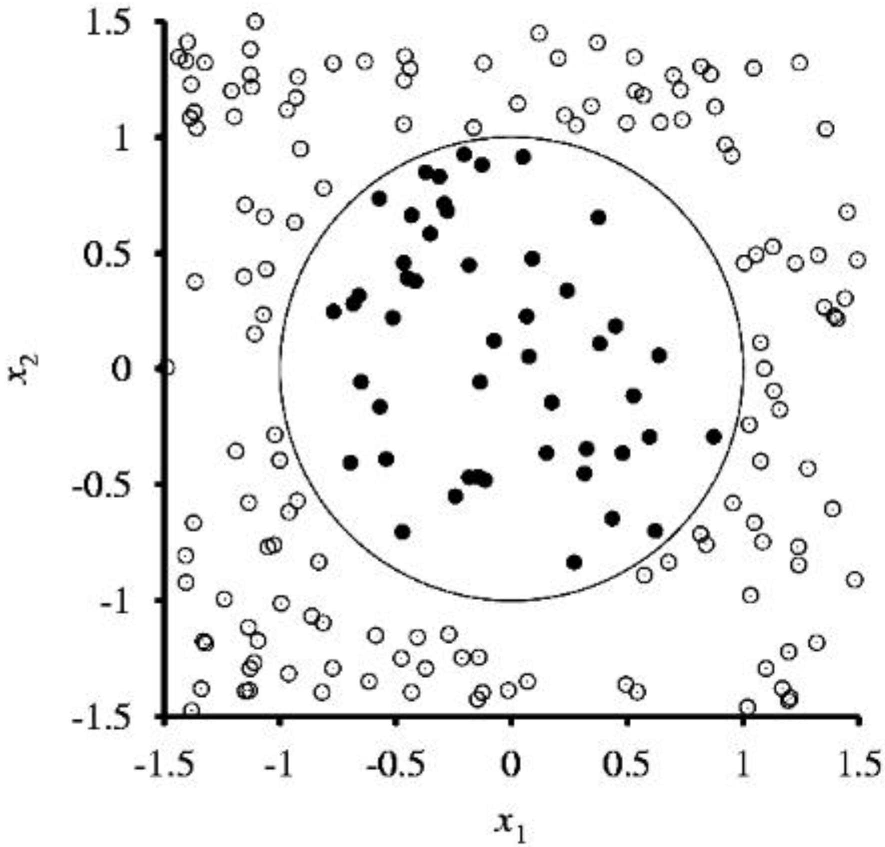


Linearly separable data in 3-d after applying the 2nd-degree polynomial transformation

In 2-dimension, this data is not linearly separable, but after applying the second-degree polynomial transformation, the classes become linearly separable.

$$\phi(\mathbf{x}) = \phi\left(\begin{pmatrix} x_1 \\ x_2 \end{pmatrix}\right) = \begin{pmatrix} {x_1}^2 \\ \sqrt{2}\, x_1 x_2 \\ {x_2}^2 \end{pmatrix}$$

Support Vector Machine (SVM)



Example:

- Given: 2-dimensional input space defined by attributes **x** = (x_1, x_2).
- All positive examples (y=+1) are inside a circular region.
- All negative examples (y=-1) are outside that circular region.
- The separator is: $x_1^2 + x_2^2 \leq 1$.

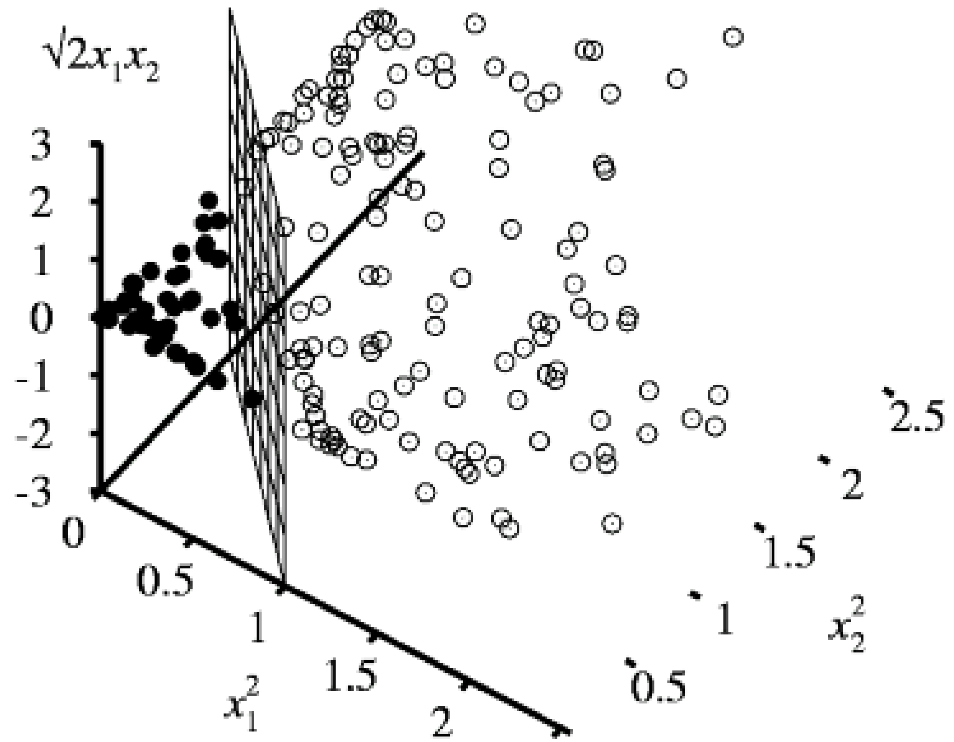
$\Rightarrow$ There is no linear separator!

Support Vector Machine (SVM)

- Suppose we re-express the input data using some computed features – i.e. we map each input vector $\mathbf{x} = (x_1, x_2)$ to a new vector of feature values, $F(\mathbf{x})$.

- In particular, we use the three features $f_1 = x_1^2$, $f_2 = x_2^2$, $f_3 = \sqrt{2}\, x_1 x_2$.

⇒ the new vectors in the three-dimensional so-called feature space are linearly separable!



SVM … Kernel Tricks

Steps involved in SVM:

- Collects the Data and plot it accordingly
- Apply the Kernel Trick
- Learns Linear Line that classifies the data
- Projects back the data

SVM … Kernel Types

Linear Kernel:

- Let us say that we have two vectors, x and y, then the linear kernel is defined by the dot product of these two vectors:

$$K(x, y) = (x \cdot y)$$

SVM … Kernel Types

Polynomial Kernel:

- Let us say that we have two vectors, x and y, then the polynomial kernel is defined by the dot product of these two vectors:

$$K(x, y) = (x \cdot y + 1)^d$$

where d is the degree of the polynomial

SVM … Kernel Types

Gaussian RBF Kernel:

- Let us say that we have two vectors, x and y, then the Gaussian Radial Basis Function kernel is defined by the dot product of these two vectors:

$$K(x,y) = e^{-\frac{\|x-y\|^2}{2\sigma^2}}$$

The given sigma plays a very important role in the performance of the Gaussian kernel and should neither be overestimated nor underestimated, it should be carefully tuned according to the problem.

SVM … Kernel Types

Laplacian Kernel:

- Let us say that we have two vectors with names x and y, then the Laplacian kernel is defined by the dot product of these two vectors:

$$K(x, y) = e^{-\frac{\|x-y\|}{\sigma}}$$

This type of kernel is less prone to changes and is totally equal exponential function kernel

SVM … Kernel Types

Sigmoid Kernel:

- Let us say that we have two vectors, x and y, then the bipolar sigmoid function. The equation for the hyperbolic kernel function is:

$$K(x, y) = \tanh(ax^T y + c)$$

Support Vector Machine (SVM)

- Generally, we have the danger of overfitting!
- Therefore, we use the SVM as a maximum margin classifier.