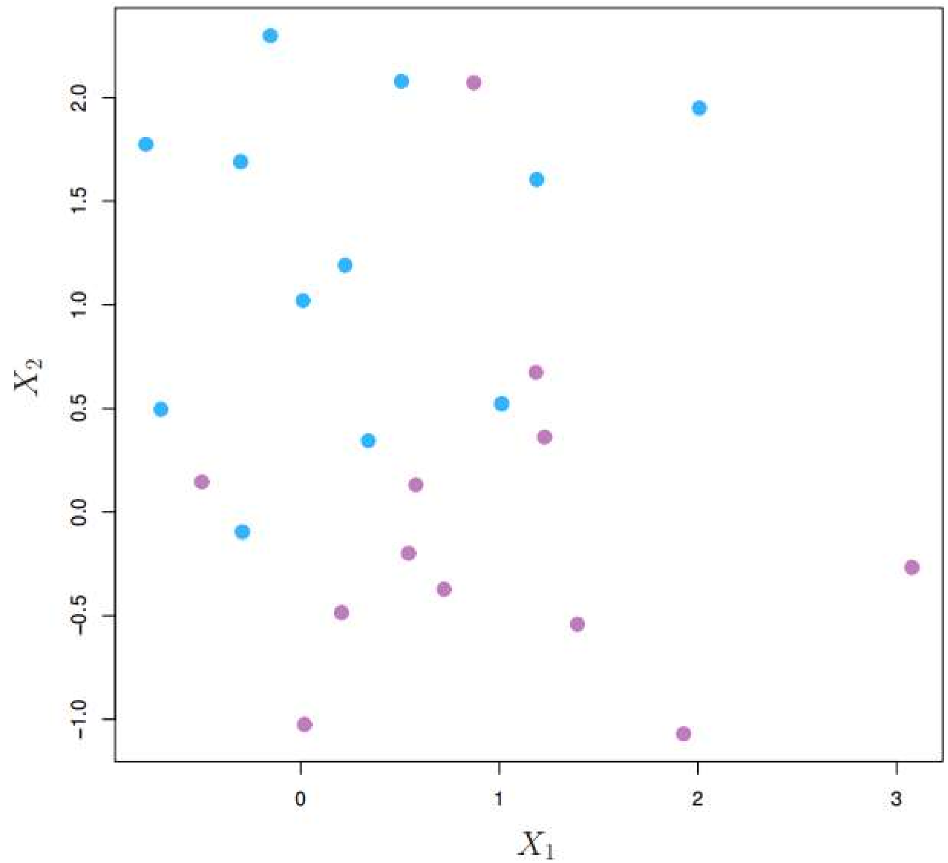
SUPPORT VECTOR MACHINES

Outline

- Decision boundary
- Linearly separable dataset
- Maximal Margin classifier (linear) hard-margin classifier
- Support vector classifier (linear) soft-margin classifier
- Support vector machines (nonlinear) classifier

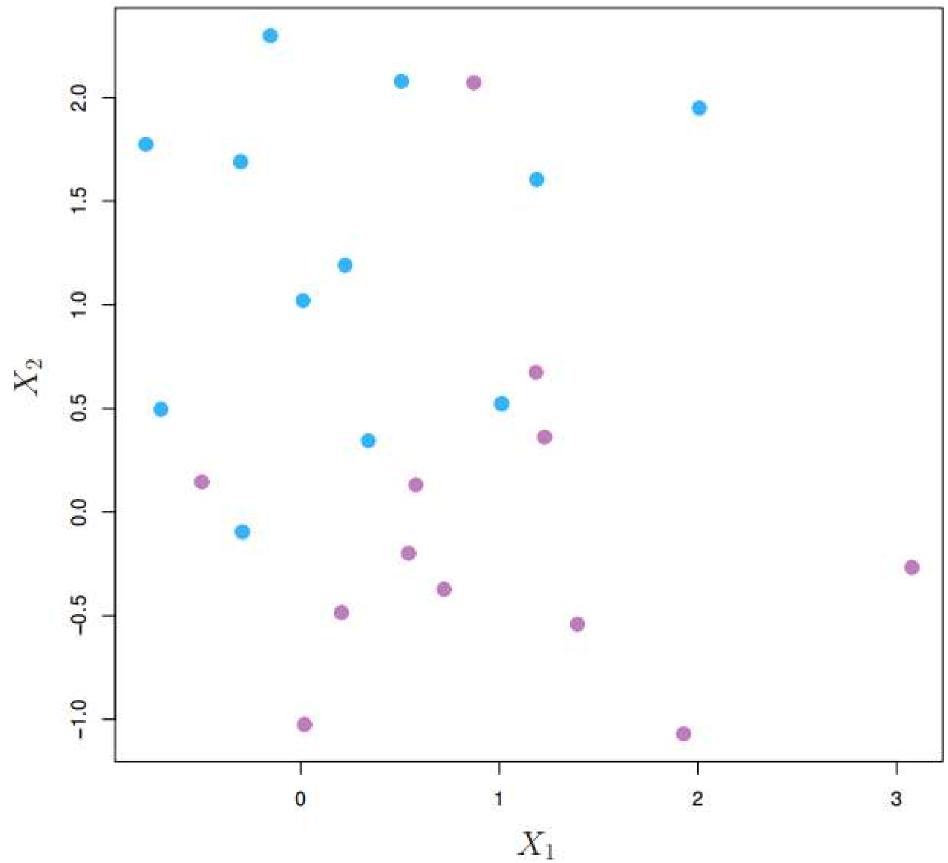
3

Non-Linearly separable dataset



Non-Linearly separable dataset

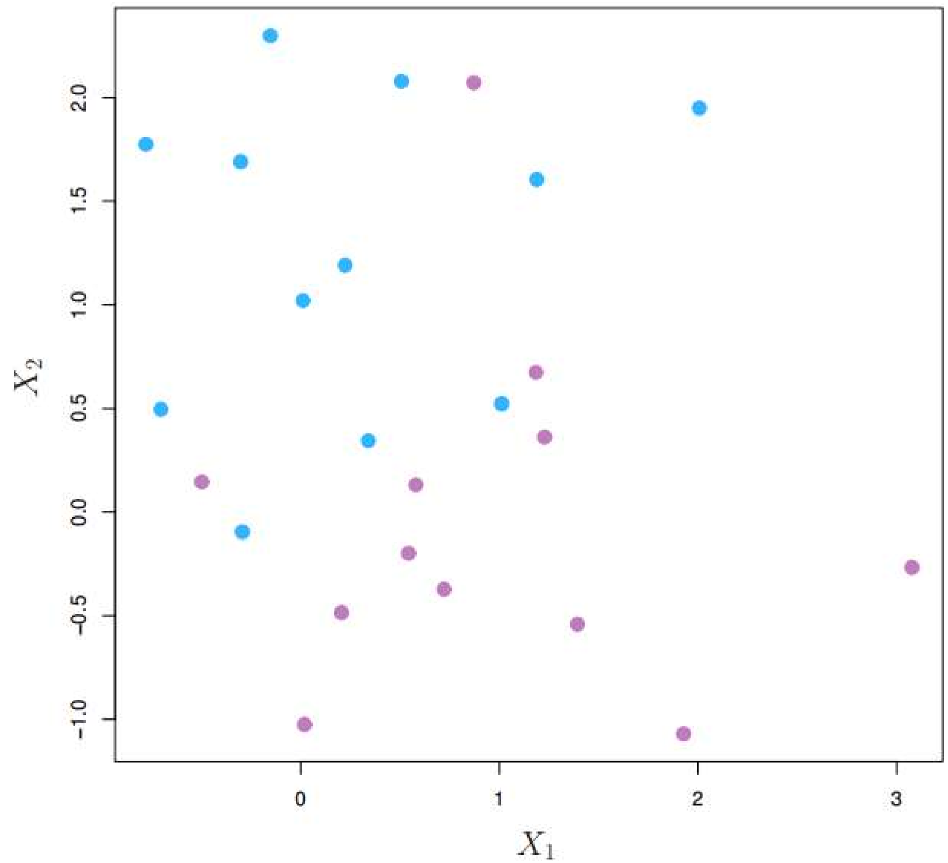
How to deal with non-linearly separable datasets?



Non-Linearly separable dataset

How to deal with non-linearly separable datasets?

Convert dataset into a linearly separable dataset



Non-Linearly separable dataset

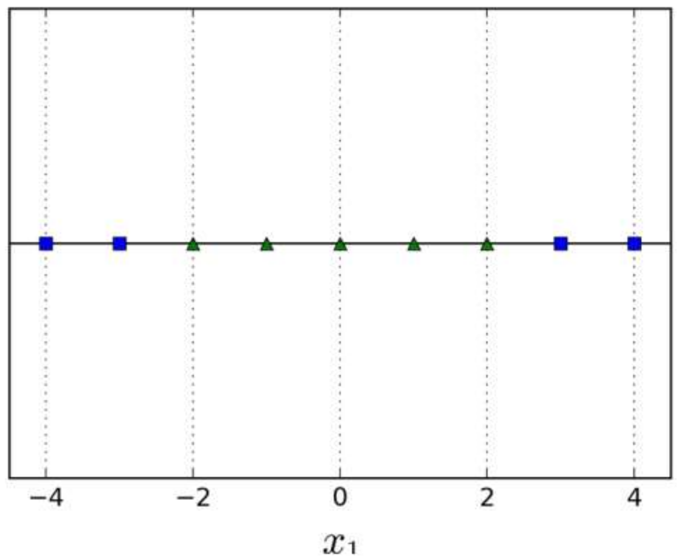
To convert into a linearly separable dataset

add more features (polynomial, exponential, etc.)

Non-Linearly separable dataset

To convert into a linearly separable dataset

add more features (polynomial, exponential, etc.)

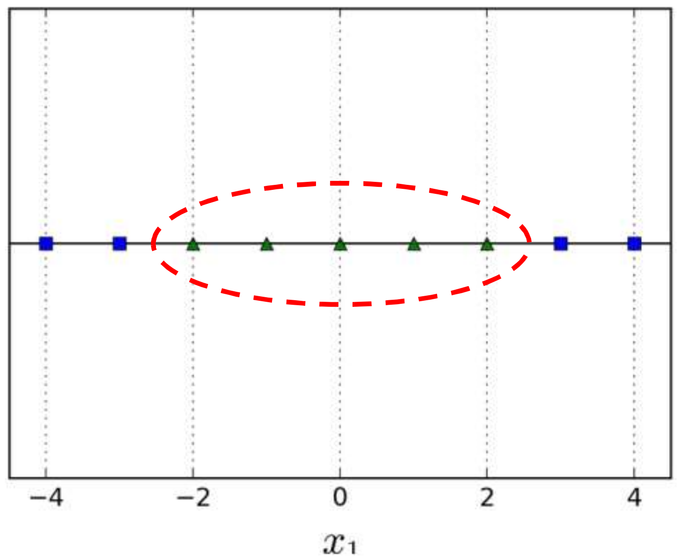


one-feature dataset

Non-Linearly separable dataset

To convert into a linearly separable dataset

add more features (polynomial, exponential, etc.)

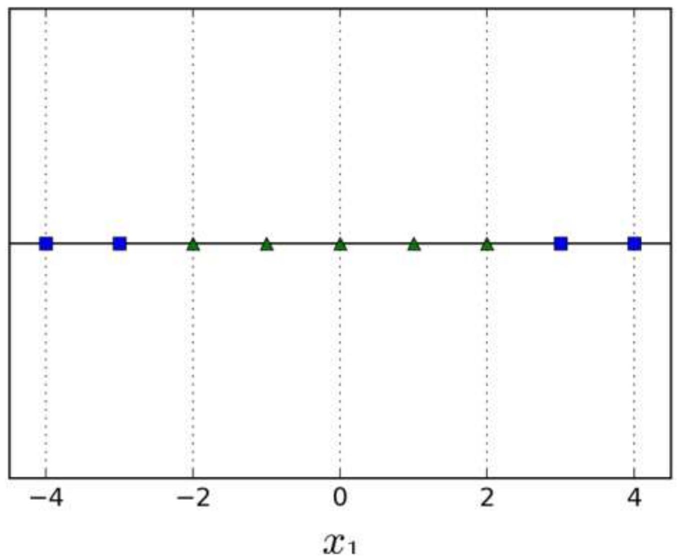


not linearly separable

Non-Linearly separable dataset

To convert into a linearly separable dataset

add more features (polynomial, exponential, etc.)



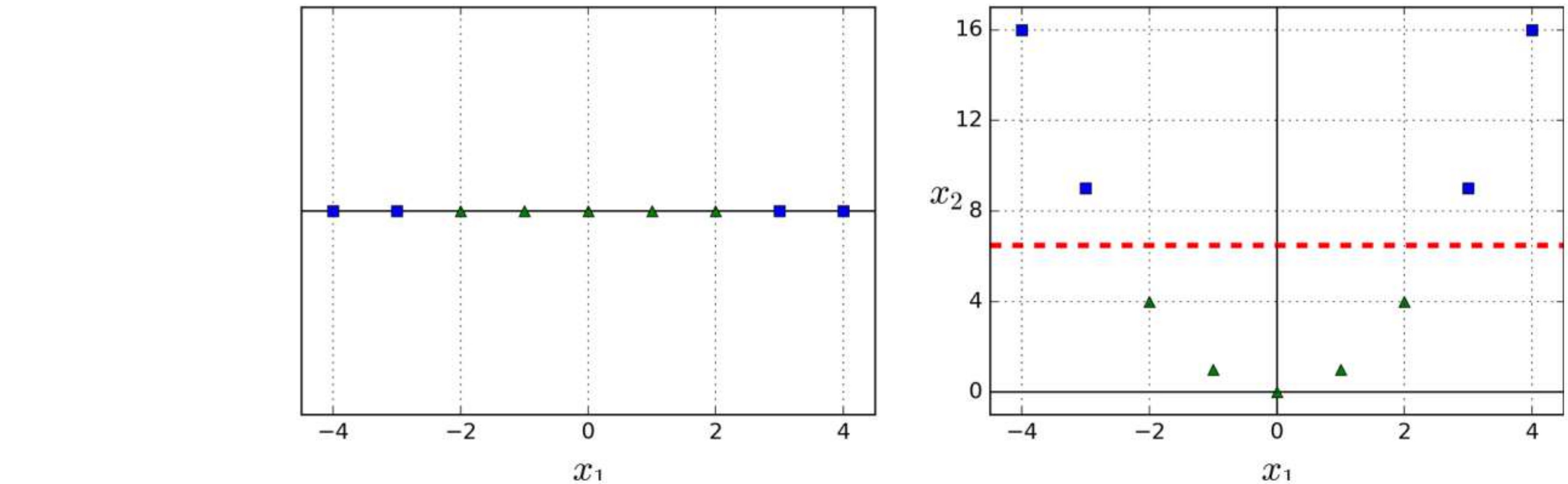
create new feature x_2

$$x_2 = x_1^2$$

Non-Linearly separable dataset

To convert into a linearly separable dataset

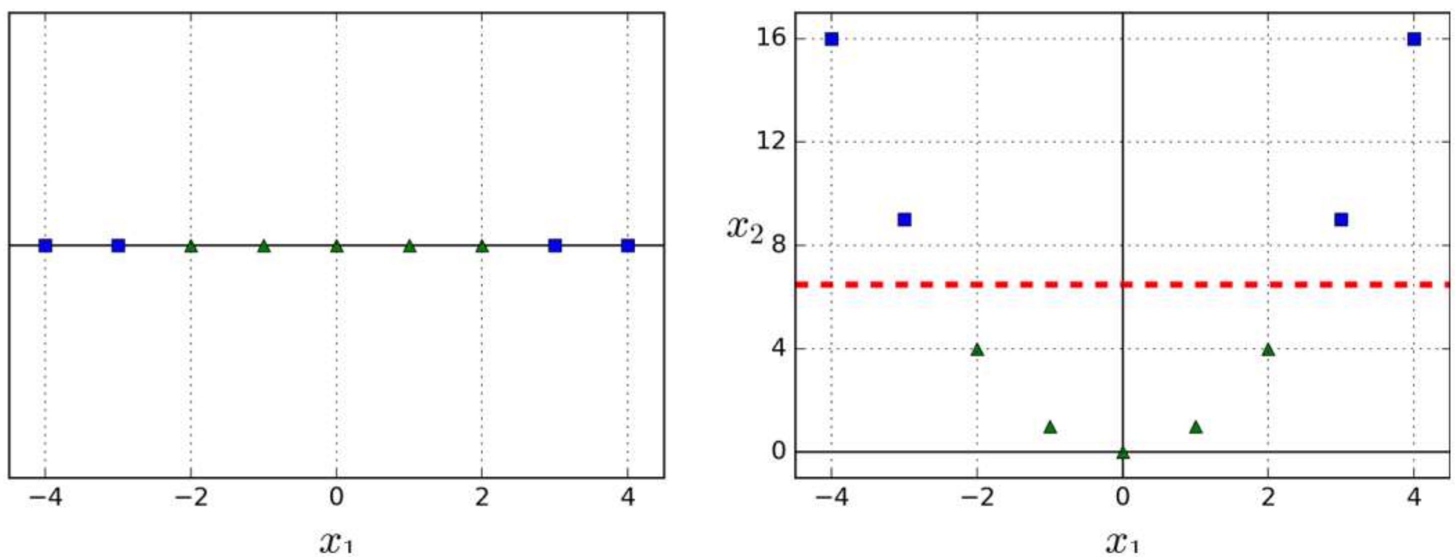
add more features (polynomial, exponential, etc.)



Non-Linearly separable dataset

To convert into a linearly separable dataset

add more features (polynomial, exponential, etc.)

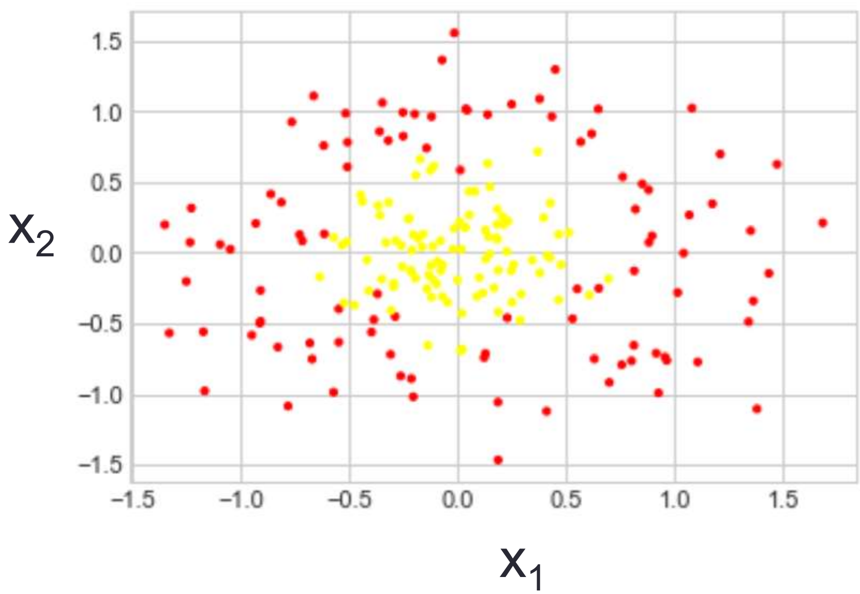


actually x_1 not needed (y is linearly separable on x_2 alone)

Non-Linearly separable dataset

To convert into a linearly separable dataset

add more features (polynomial, exponential, etc.)



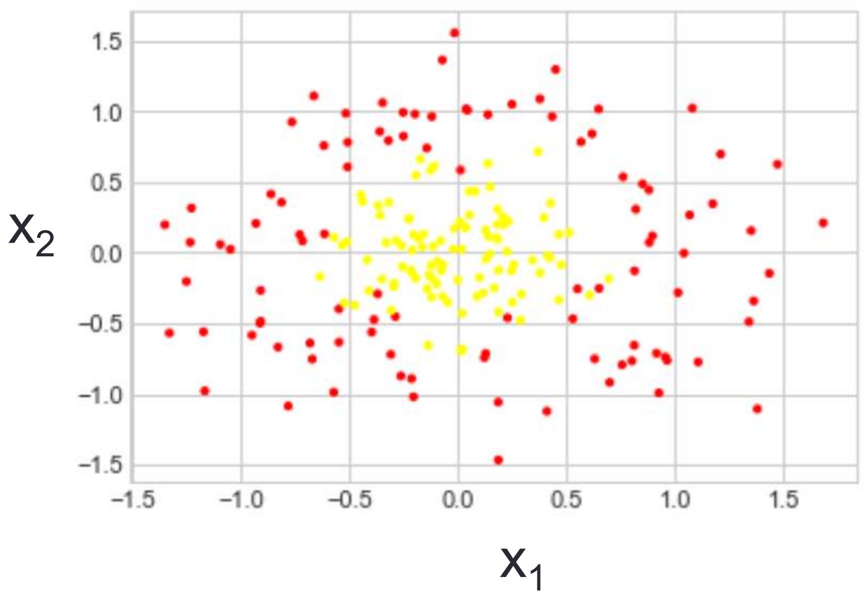
not linearly

separable

Non-Linearly separable dataset

To convert into a linearly separable dataset

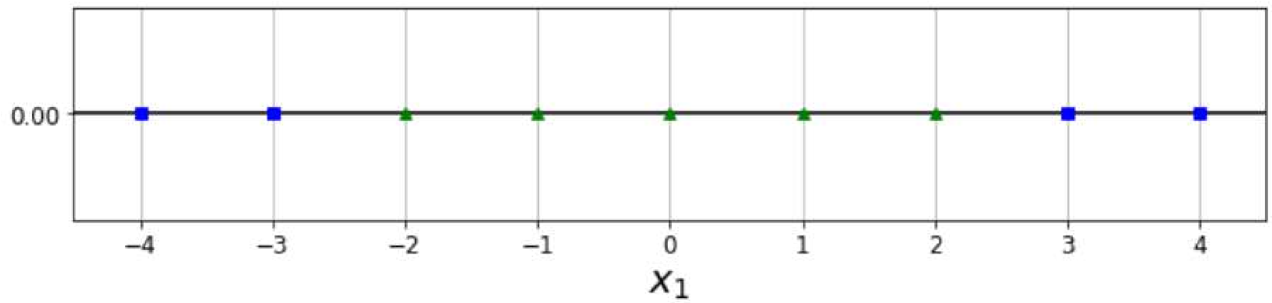
add more features (polynomial, exponential, etc.)



add new feature x_3

$$x_3 = x_1^2 + x_2^2$$

Radial Basis Function (*RBF*)



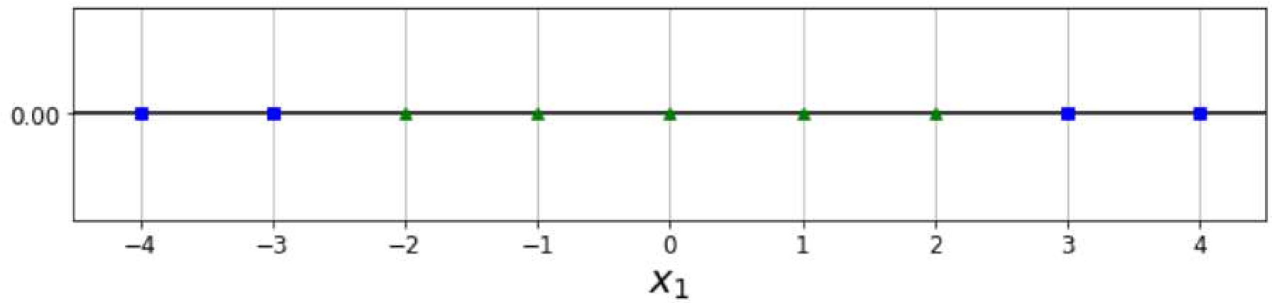
new exponential features

x_2 and x_3

$$x_2 = e^{-\gamma\,(x_1 - a)^2}$$

$$x_3 = e^{-\gamma\,(x_1 - b)^2}$$

Radial Basis Function (*RBF*)



new exponential features

x_2 and x_3

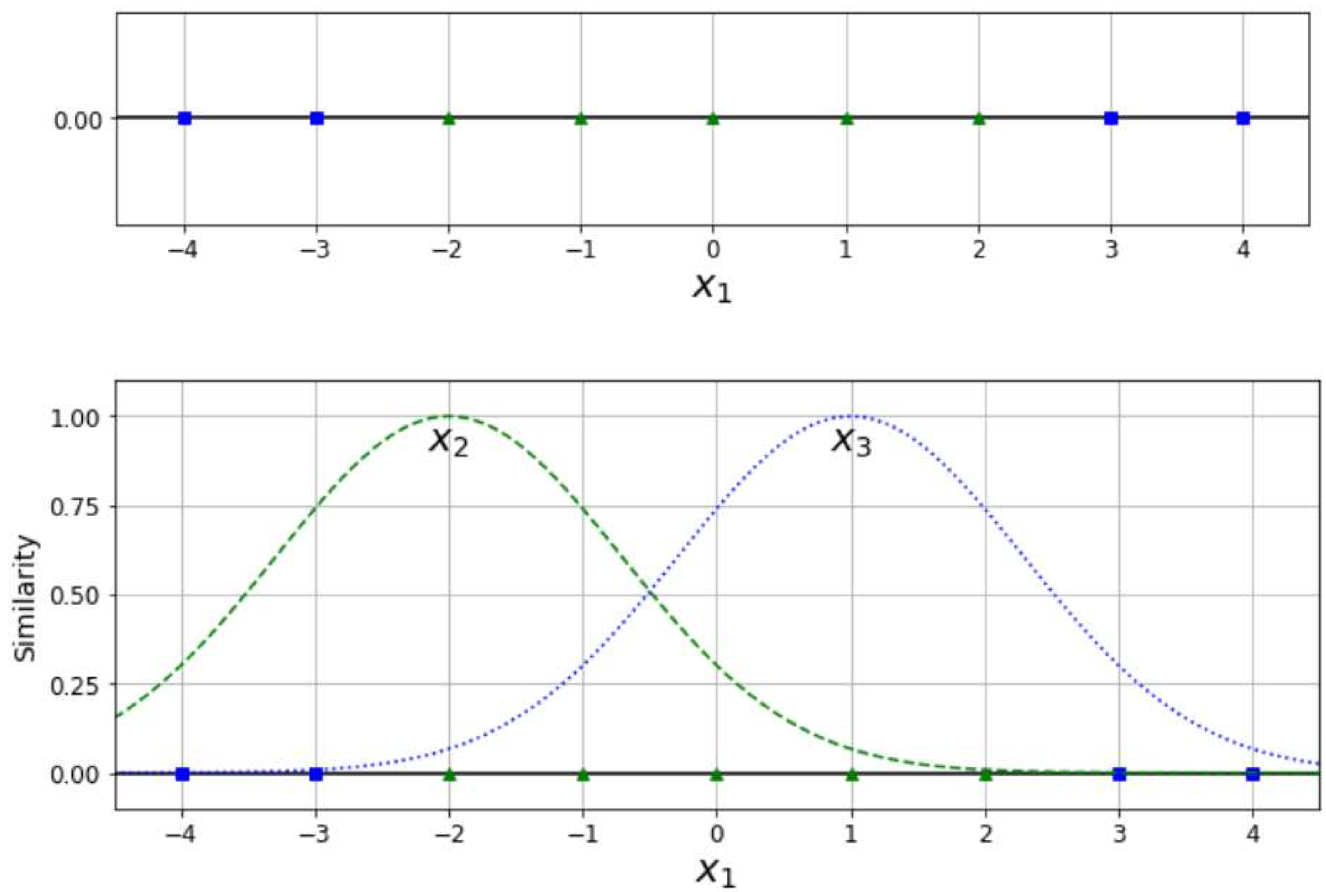
$$x_2 = e^{-\gamma\,(x_1 - a)^2}$$

$$x_3 = e^{-\gamma\,(x_1 - b)^2}$$

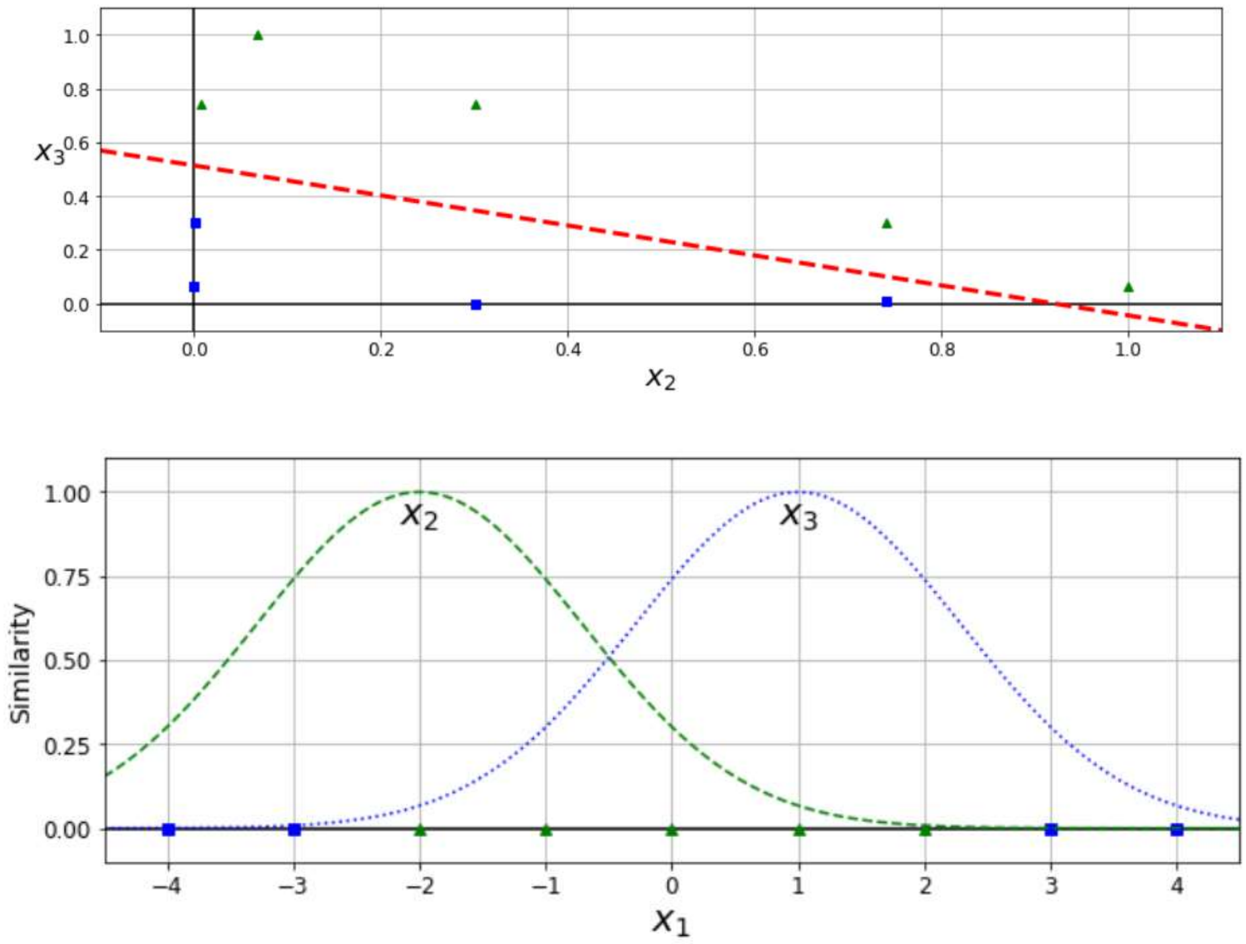
Example: $\gamma=0.3, \quad a = -2, \quad b = 1$

16

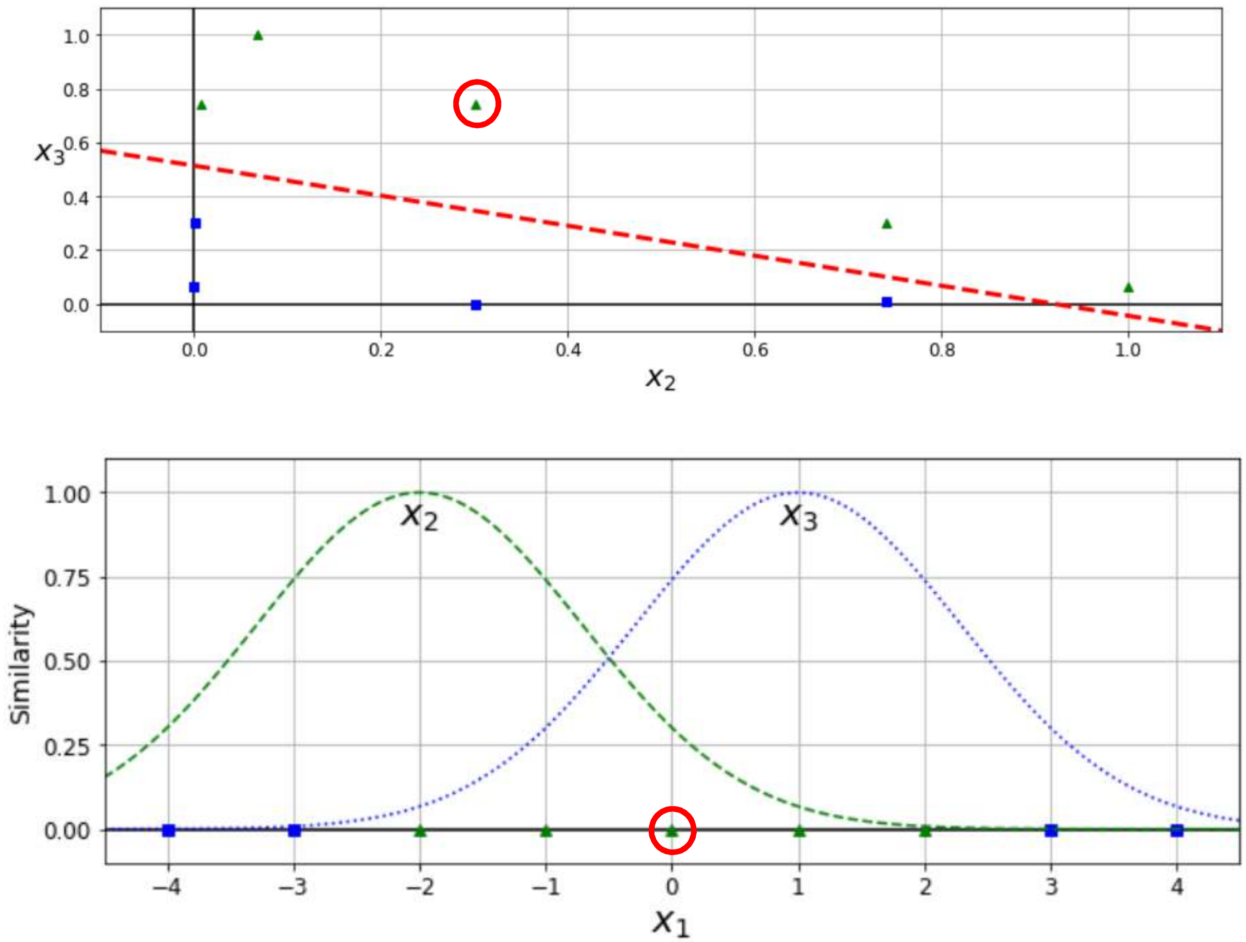
Radial Basis Function (*RBF*)



Radial Basis Function (*RBF*)

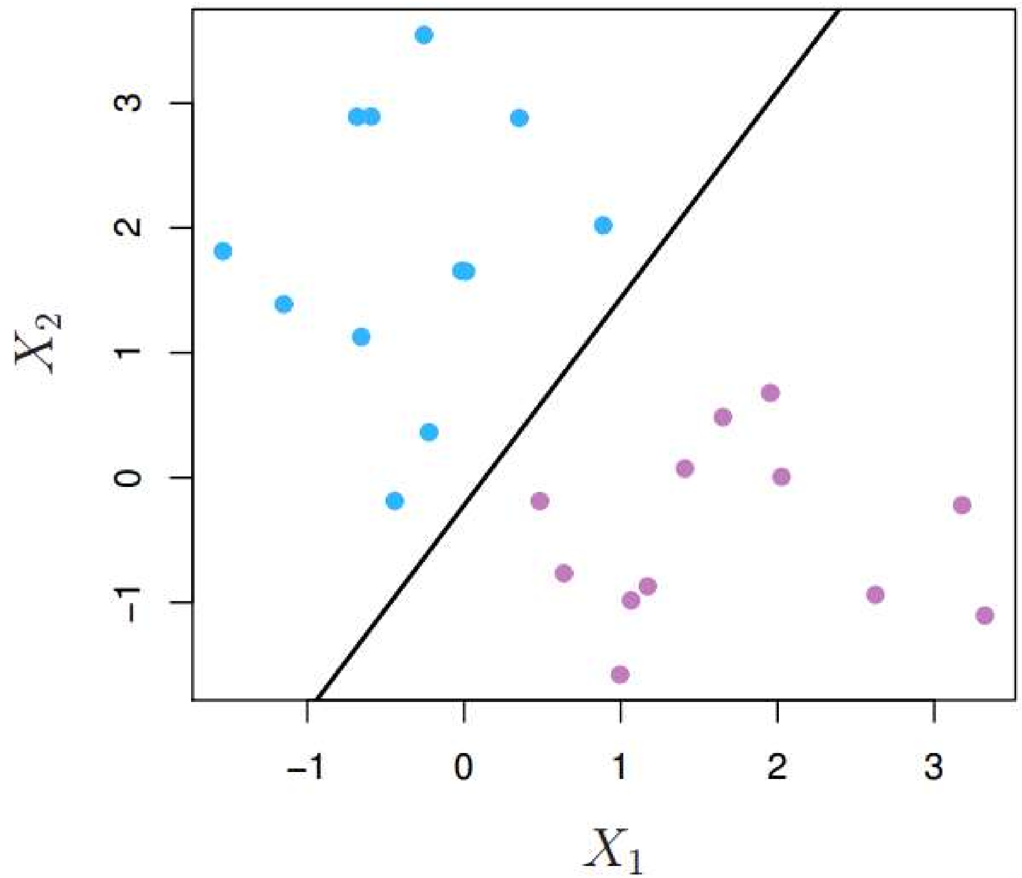


Radial Basis Function (*RBF*)



19

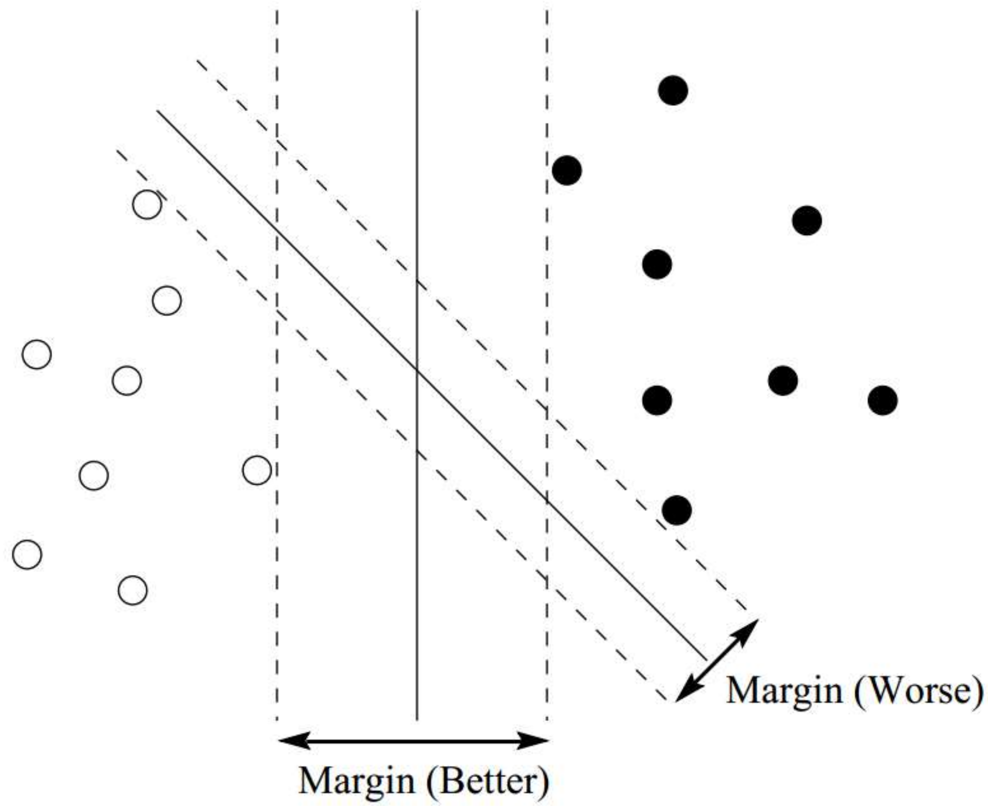
Linearly separable dataset



Linearly separable dataset

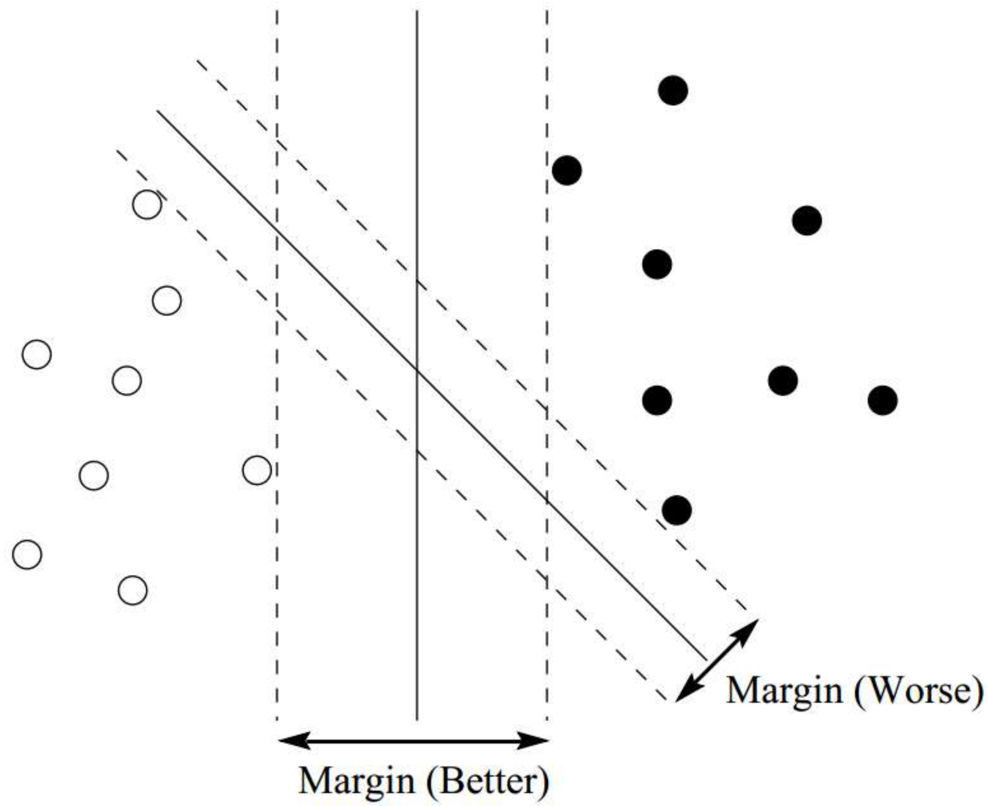
Many linear

boundaries



Maximal Margin Classifier

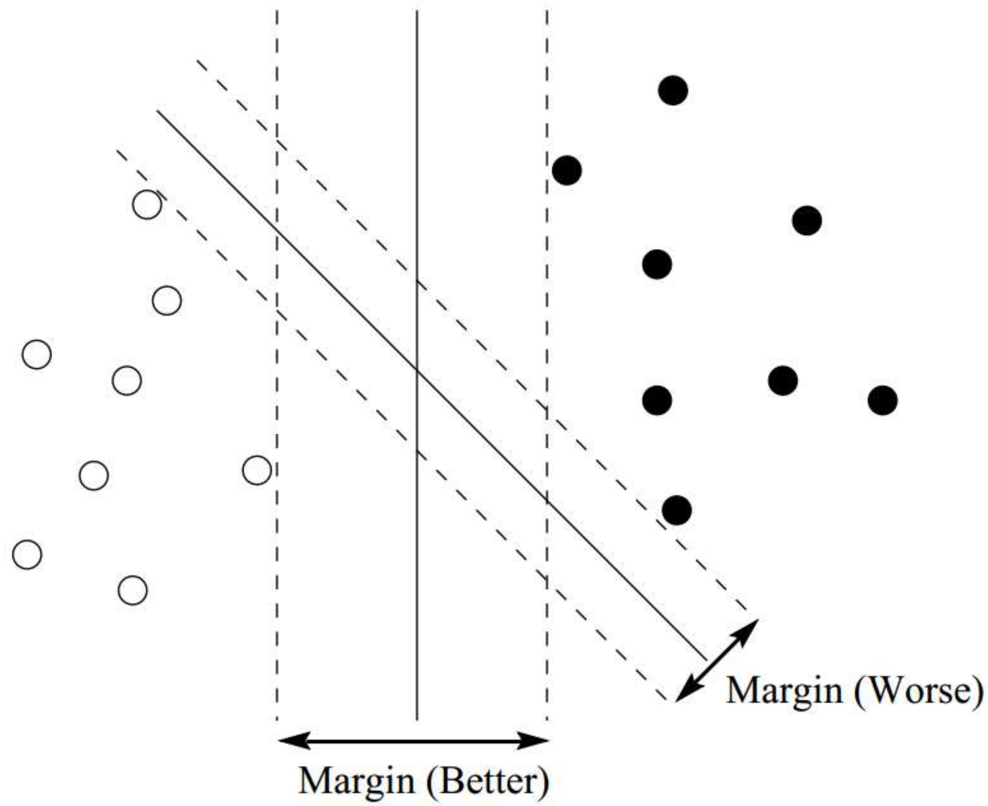
Find the separating hyperplane that makes the biggest gap (or margin) between the two classes



Maximal Margin Classifier

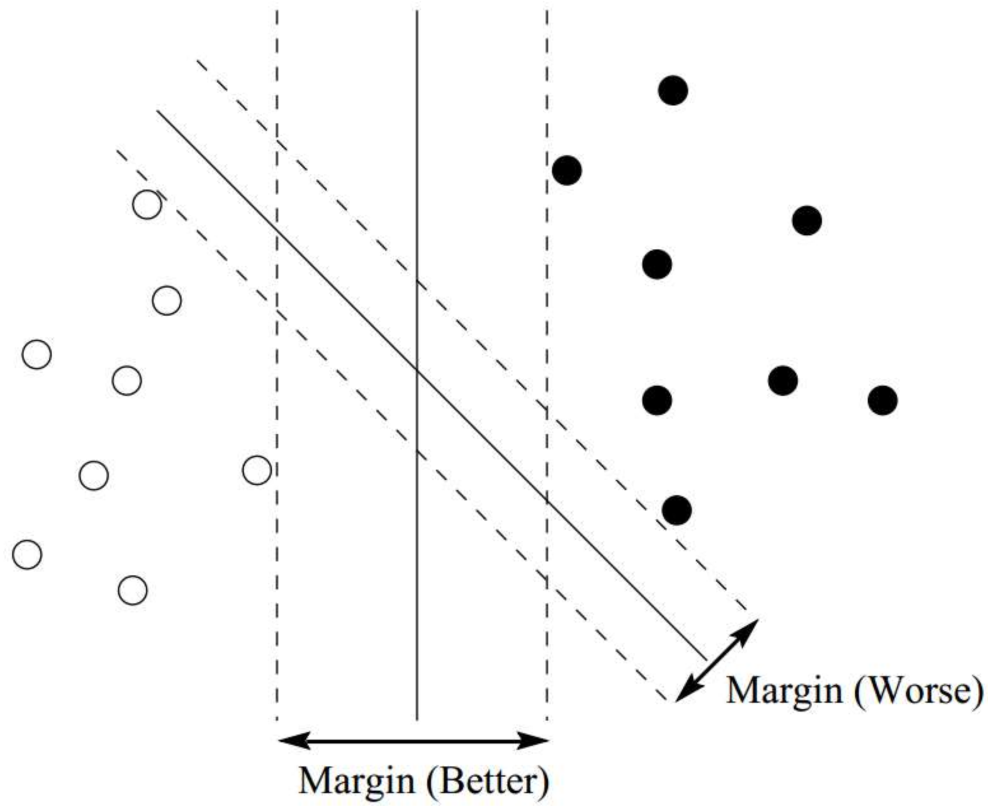
Find the separating hyperplane that makes the biggest gap (or margin) between the two classes

Objective is to maximize the Margin



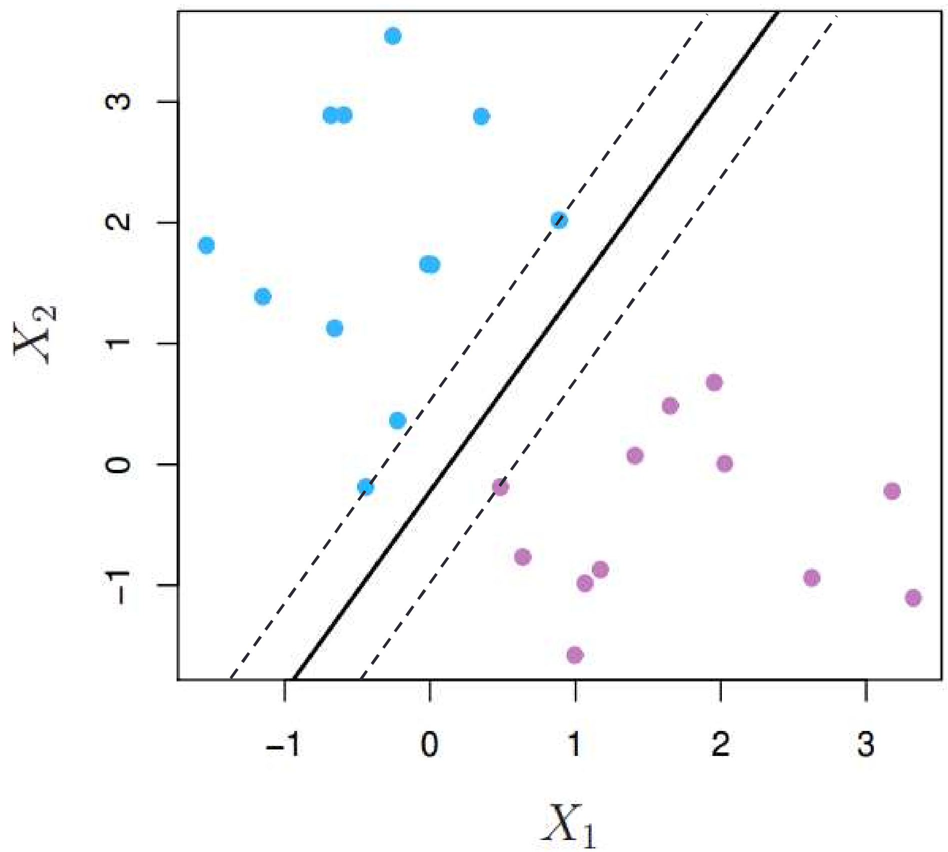
Maximal Margin Classifier

Think of fitting the widest possible street between the classes



Hard Margin Classifier

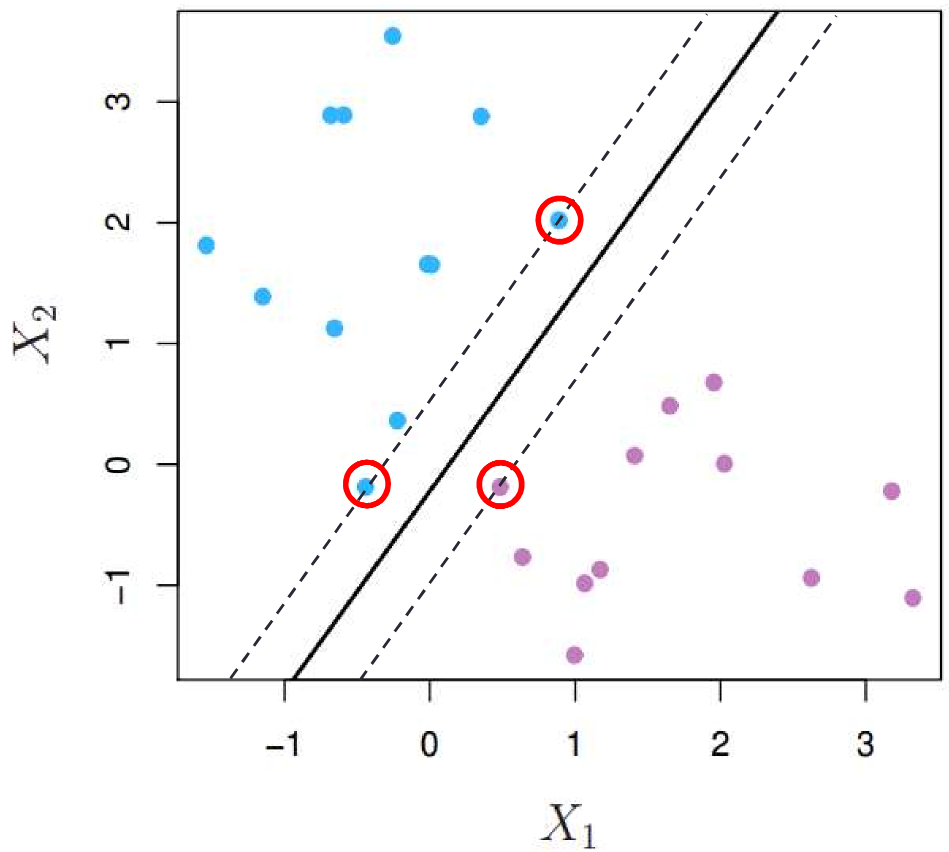
Restrict that all observations must be off the street and on the correct side



Hard Margin Classifier

Restrict that all observations must be off the street and on the correct side

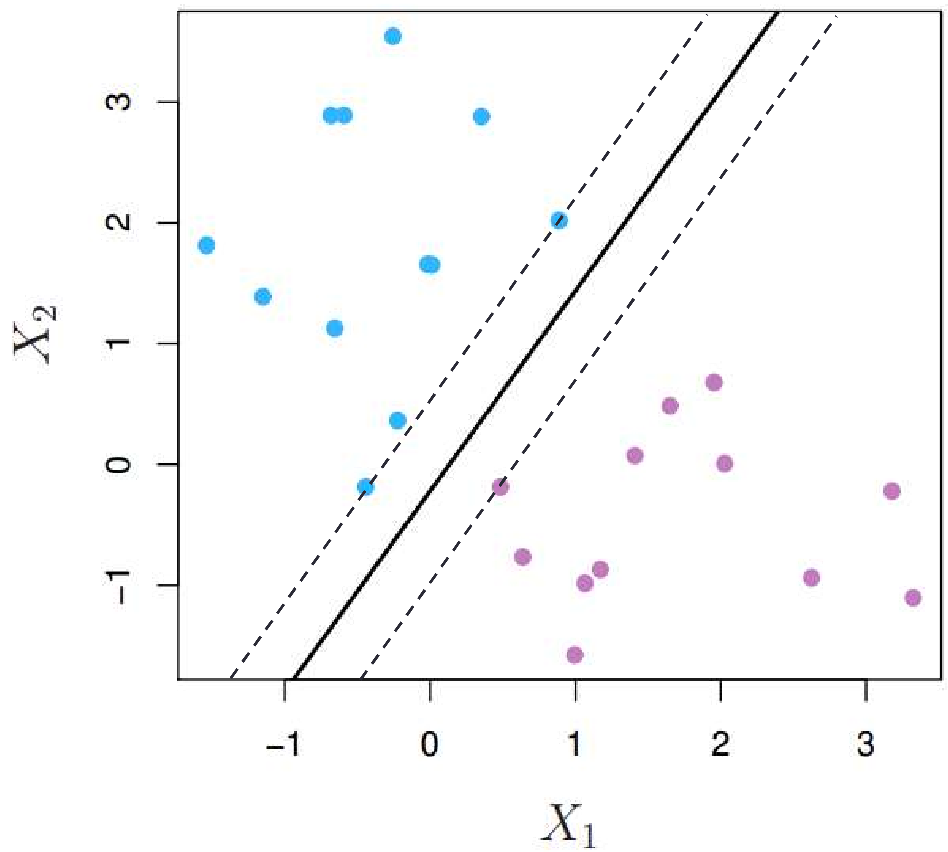
Points on the left and right margins are called support vectors



Hard Margin Classifier

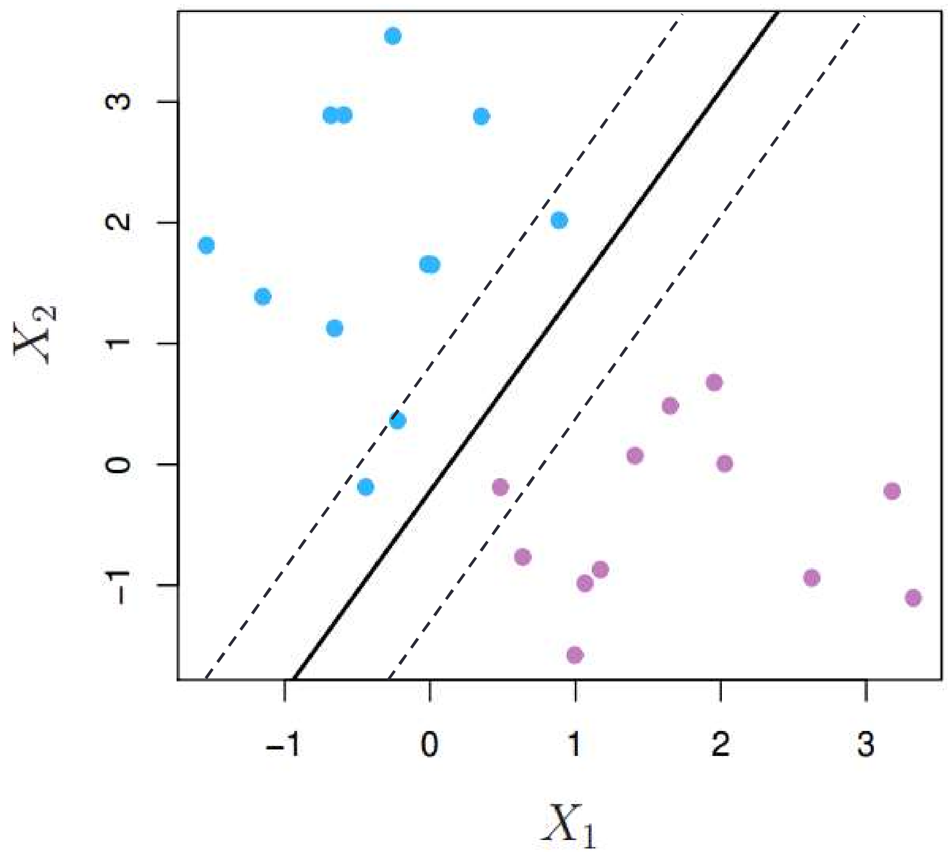
Restrict that all observations must be off the street and on the correct side

Only works if data is linearly separable



Soft Margin Classifier

- Keep the street as wide as possible
- Allowing for some margin violations
- Observations lying on the street (even on the wrong side)



Support Vector Classifier

SVC predicts the class of *y* with a linear decision function *h*

$$h = \mathbf{w}^T \cdot \mathbf{x} + b = w_1 x_1 + \cdots + w_p x_p \ + b$$

$$\hat{y} = \begin{cases} 0 & \text{if} \quad h < 0 \\ 1 & \text{if} \quad h \geq 0 \end{cases}$$

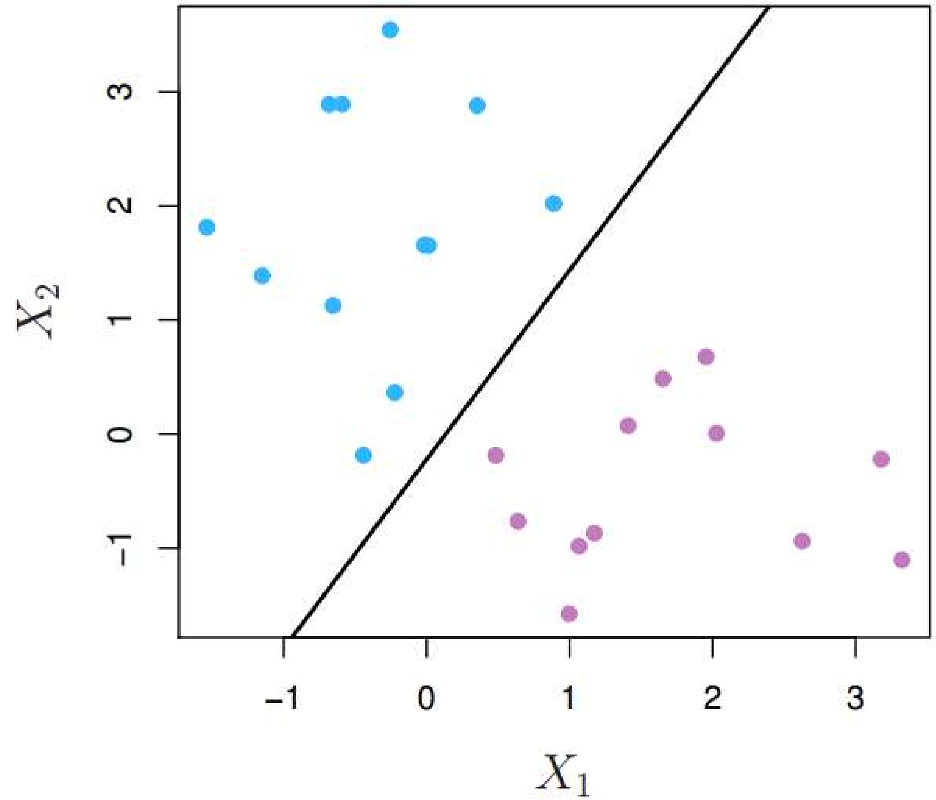
$w_1, \ldots, w_p$ feature weights, b bias term

29

Linearly separable dataset

Suppose boundary is

$$x_2 = 2x_1 + 1$$



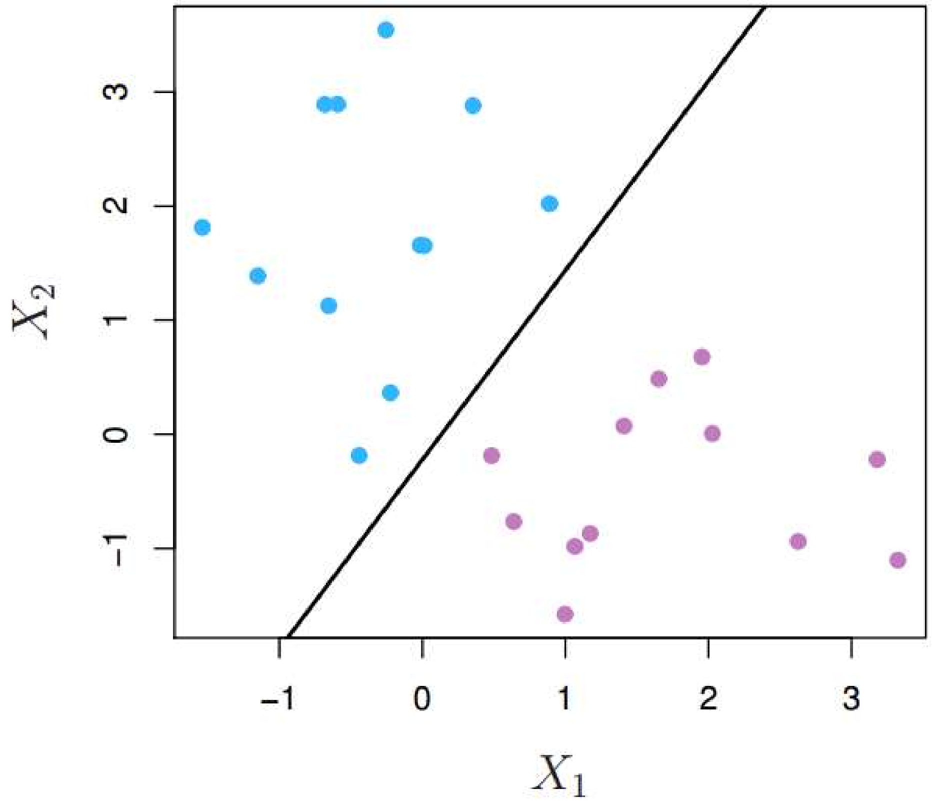
Linearly separable dataset

Suppose boundary is

$$x_2 = 2x_1 + 1$$

or

$$2x_1 - x_2 + 1 = 0$$



Linearly separable dataset

Suppose boundary is

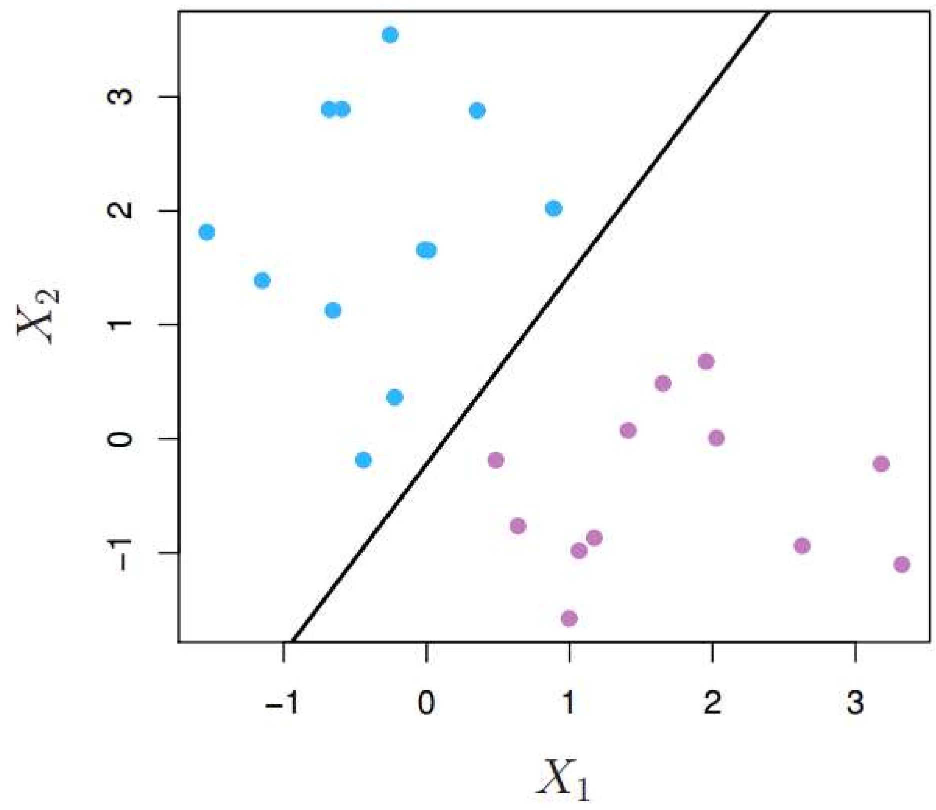
$$x_2 = 2x_1 + 1$$

or

$$2x_1 - x_2 + 1 = 0$$

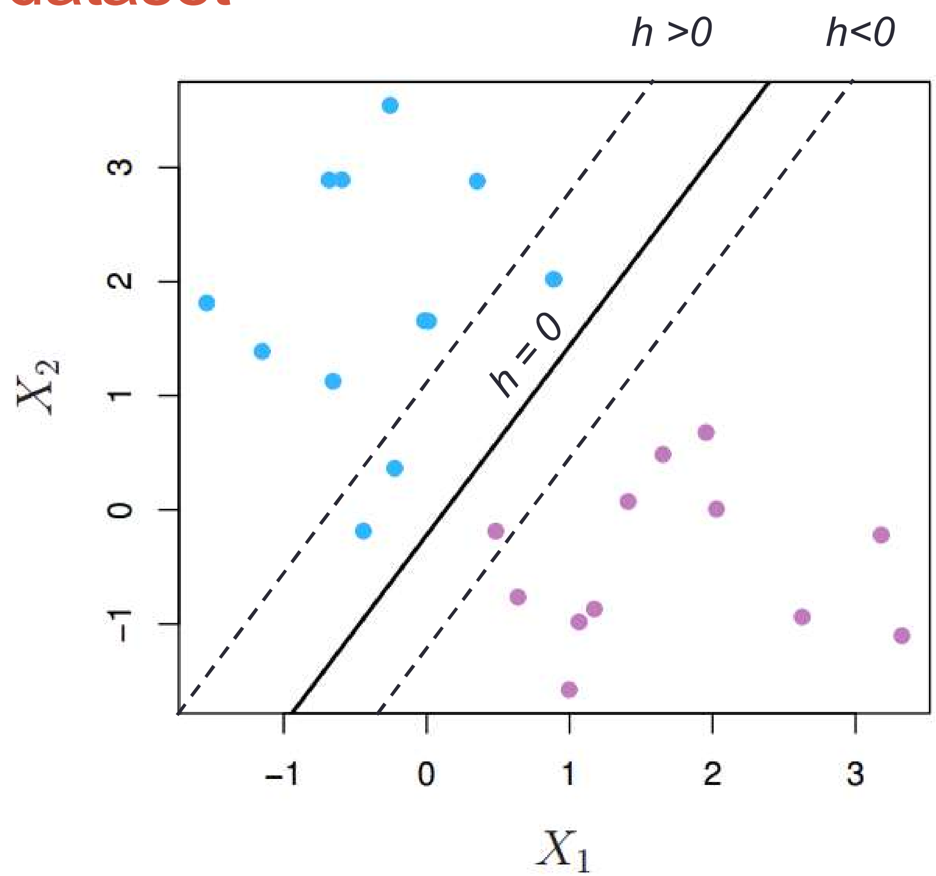
Let

$$h(x_1,x_2) = 2x_1 - x_2 + 1$$



Linearly separable dataset

$h(x_1,x_2) = 2x_1 - x_2 + 1$



Linearly separable dataset

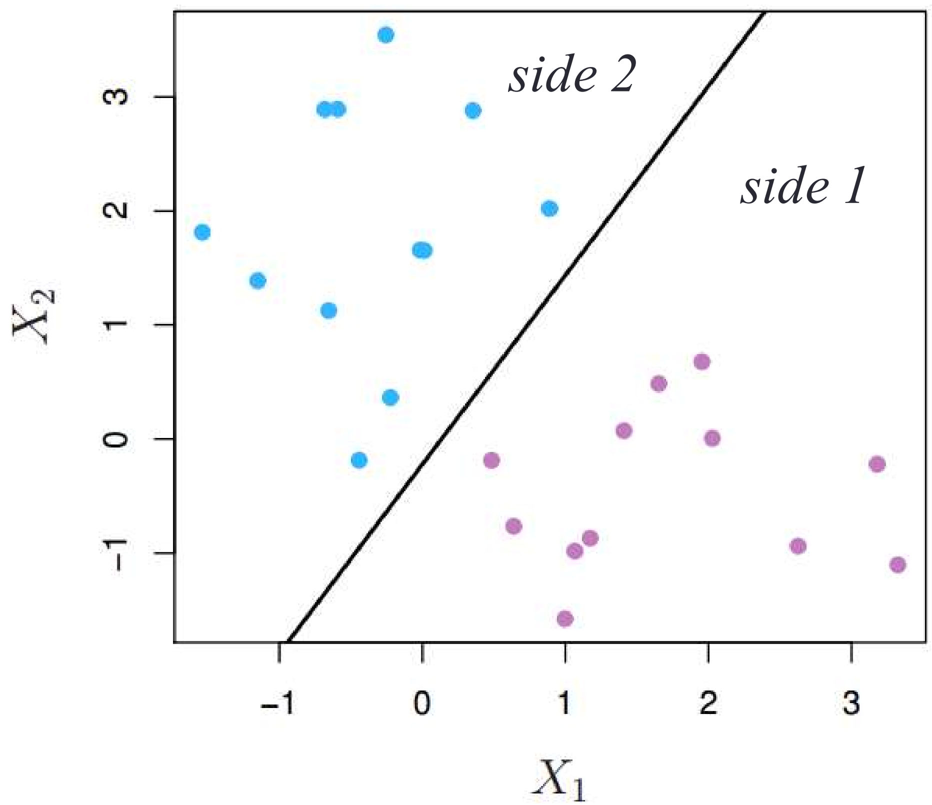
$h(x_1,x_2) = 2x_1 - x_2 + 1$

If

$h = 0$ (x_1,x_2) on bound

$h < 0$ (x_1,x_2) on *side* 1

$h > 0$ (x_1,x_2) on *side* 2



Linearly separable dataset

$h(x_1,x_2) = 2x_1 - x_2 + 1$

If

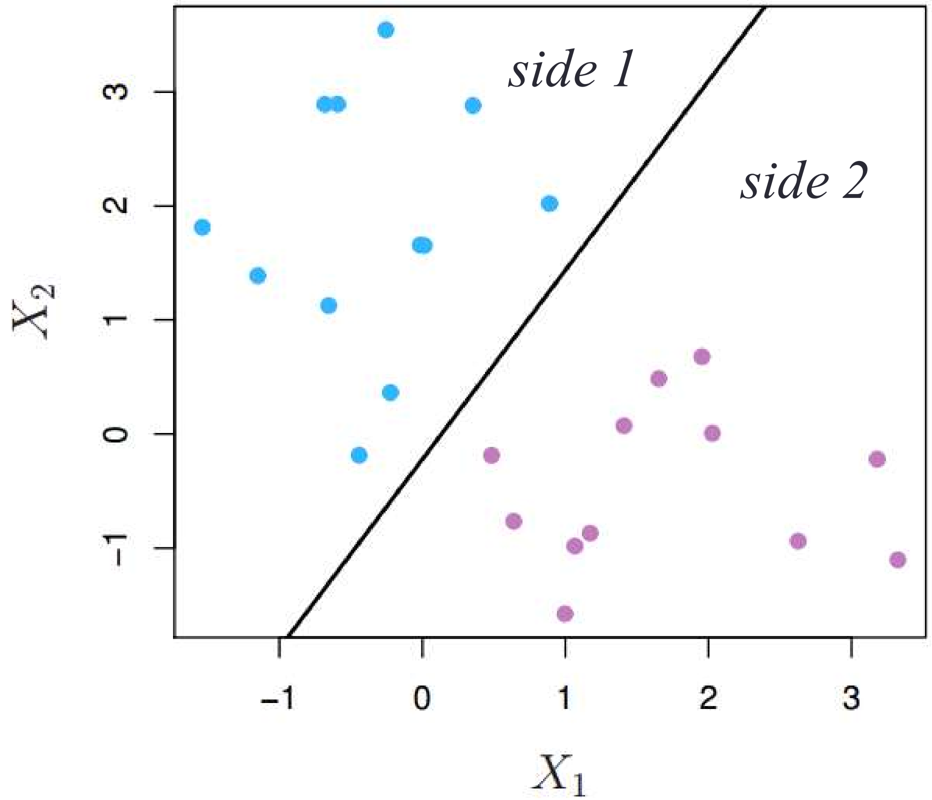
$h = 0$ (x_1,x_2) on bound

$h < 0$ (x_1,x_2) on *side* 1

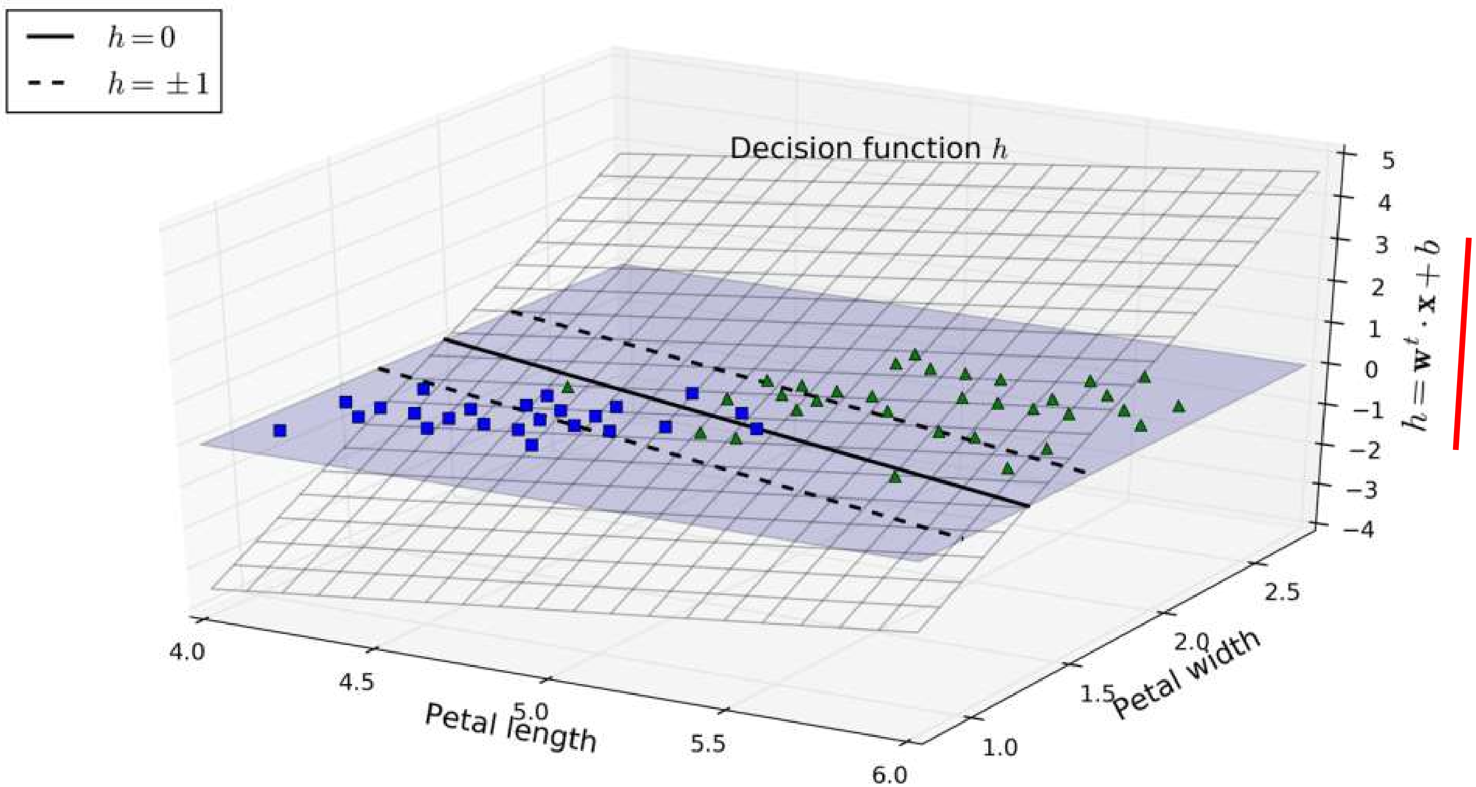
$h > 0$ (x_1,x_2) on *side* 2

Let

$$\hat{y} = \begin{cases} 0 & \text{if} \quad h < 0 \\ 1 & \text{if} \quad h \geq 0 \end{cases}$$



Decision function

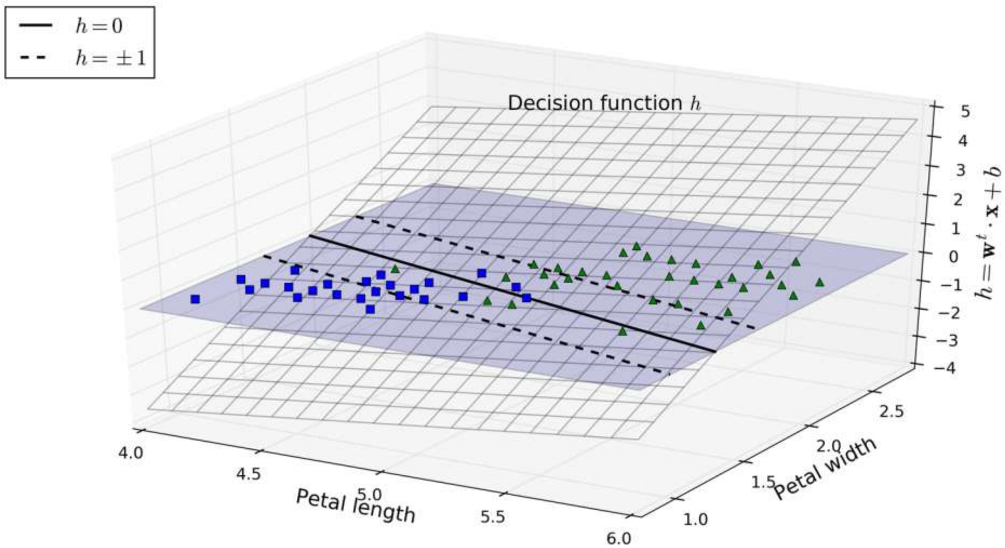


Decision function

Decision boundary

is the set of points

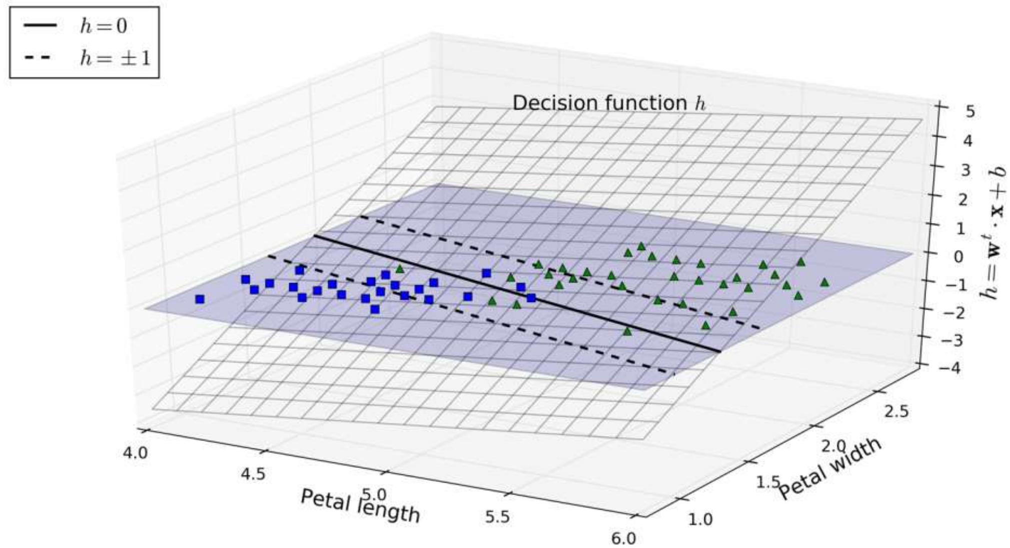
where h = 0



Decision function

Decision boundary is the set of points where h = 0

Dash lines are points where h is equal to -1, or, 1

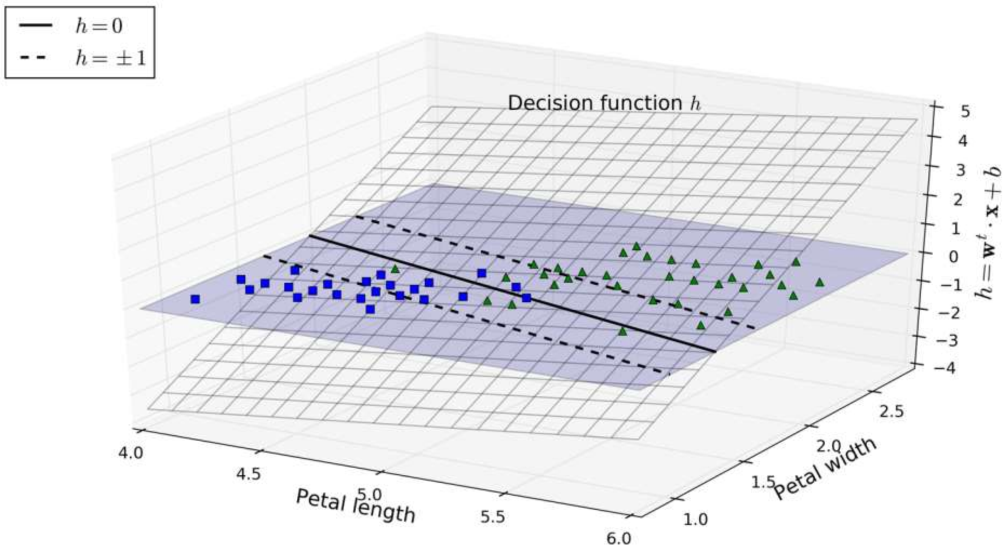


Decision function

Decision boundary is the set of points where h = 0

Dash lines are points where h is equal to -1, or, 1

Margin is between the dash lines



Decision function

Predict

one class when *h > 1*

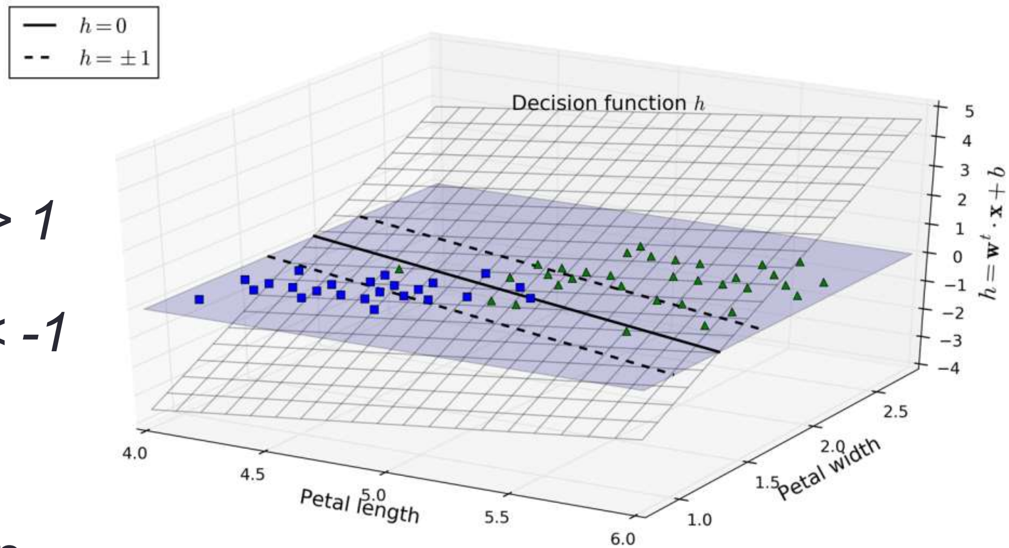
or other class if *h < -1*

The smallest slope

gives widest margin

Slope is

$$\mathbf{w}^T \cdot \mathbf{w}$$



Support Vector Classifier

$$h = \mathbf{w}^T \cdot \mathbf{x} + b = w_1 \, x_1 + \cdots + w_p \, x_p$$

by changing $$y = \begin{cases} 0 & \text{if} \quad h < 0 \\ 1 & \text{if} \quad h \geq 0 \end{cases}$$

to $$y = \begin{cases} -1 & \text{if} \quad h < -1 \\ 1 & \text{if} \quad h \geq 1 \end{cases}$$

we get $y \, h \geq 1$

or $y_i \, (\mathbf{w}^T \cdot \mathbf{x}_i + b) \geq 1 \qquad i = 1, \ldots, n$

Hard Margin Classifier

Constrained optimization problem

Find $b, w_1, \ldots, w_p$ to

$$\text{Min} \quad \frac{1}{2}\,\mathbf{w}^T \cdot \mathbf{w}$$

$$\text{subject to} \quad y_i\left(\mathbf{w}^T \cdot \mathbf{x}_i + b\right) \geq 1 \qquad i = 1, \ldots, n$$

Hard Margin Classifier

Constrained optimization problem

Find $b,\ w_1, \ldots, w_p$ to

$$\text{Min} \quad \frac{1}{2}\,\mathbf{w}^T \cdot \mathbf{w}$$

$$\text{subject to} \quad y_i\left(\mathbf{w}^T \cdot \mathbf{x}_i + b\right) \geq 1 \qquad i = 1, \ldots, n$$

a quadratic optimization problem on $b,\ w_1, \ldots, w_p$

with linear constraints

Soft Margin Classifier

Let $\zeta_i \geq 0$, slack of i^{th} observation

it measures

how much i^{th} observation is allowed to violate the margin

Include ζ_i in the optimization problem

Soft Margin Classifier

Find $b,\ w_1, \ldots, w_p,\ \zeta_1, \ldots, \zeta_n$ to

$$\text{Min} \quad \frac{1}{2}\,\mathbf{w}^T \cdot \mathbf{w} + C \sum_{i=1}^{n} \zeta_i$$

$$\text{subject to} \quad y_i\left(\mathbf{w}^T \cdot \mathbf{x}_i + b\right) \geq 1 - \zeta_i \qquad i = 1, \ldots, n$$

reducing $\mathbf{w}^T \cdot \mathbf{w}$ increases the margin

increasing the margin increases $\sum_{i=1}^{n} \zeta_i$

Soft Margin Classifier

Find $b, w_1, \ldots, w_p, \zeta_1, \ldots, \zeta_n$ to

$$\text{Min} \quad \frac{1}{2}\,\mathbf{w}^T\!\cdot\mathbf{w} + C\sum_{i=1}^{n}\zeta_i$$

$$\text{subject to} \quad y_i\,(\mathbf{w}^T\!\cdot\mathbf{x}_i + b) \geq 1 - \zeta_i \qquad i = 1, \ldots, n$$

reducing $\mathbf{w}^T\!\cdot\mathbf{w}$ increases the margin

increasing the margin increases $\sum_{i=1}^{n}\zeta_i$

trade-off

Soft Margin Classifier

Find $b, w_1, \ldots, w_p, \zeta_1, \ldots, \zeta_n$ to

$$\text{Min} \quad \frac{1}{2}\,\mathbf{w}^T\cdot\mathbf{w} + C\sum_{i=1}^{n}\zeta_i$$

$$\text{subject to} \quad y_i\left(\mathbf{w}^T\cdot\mathbf{x}_i + b\right) \geq 1 - \zeta_i \qquad i = 1, \ldots, n$$

$$\left.\begin{array}{l}\text{reducing } \mathbf{w}^T\cdot\mathbf{w} \text{ increases the margin} \\ \text{increasing the margin increases } \sum_{i=1}^{n}\zeta_i\end{array}\right\} \text{trade-off}$$

Tune C parameter (cross validation)

Soft Margin Classifier

Primal problem

$$\text{Min} \quad \frac{1}{2}\,\mathbf{w}^T\!\cdot\mathbf{w} + C\sum_{i=1}^{n}\zeta_i$$

$$\text{subject to} \quad y_i\left(\mathbf{w}^T\!\cdot\mathbf{x}_i + b\right) \geq 1 - \zeta_i \qquad i = 1,\ldots,n$$

Dual problem is usually used for efficiency

Support Vector Machine *(SVM)*

SVM can predict the class of y

with a nonlinear decision function *h*

Support Vector Machine *(SVM)*

SVM is an extension of the SVC

that results from enlarging the set of

predictors by means of kernels

Support Vector Machine *(SVM)*

Available kernels

- linear
- polynomial
- rbf
- sigmoid

SVM Extension for K classes

Two approaches

One vs. One

One vs. All

SVM for K classes

One vs. One

Fit SVMs (one for each pair of classes)

Classify observation using each SVM

Assign the observation to the class to which it was most frequently assigned

SVM for K classes

One vs. All

Reclassify observations

+1 if belongs to class *i*

-1 otherwise

Fit SVM and classify the observations

Repeat for $i = 1, \ldots, k$ classes

Assign each observation to the class to which it was most frequently assigned