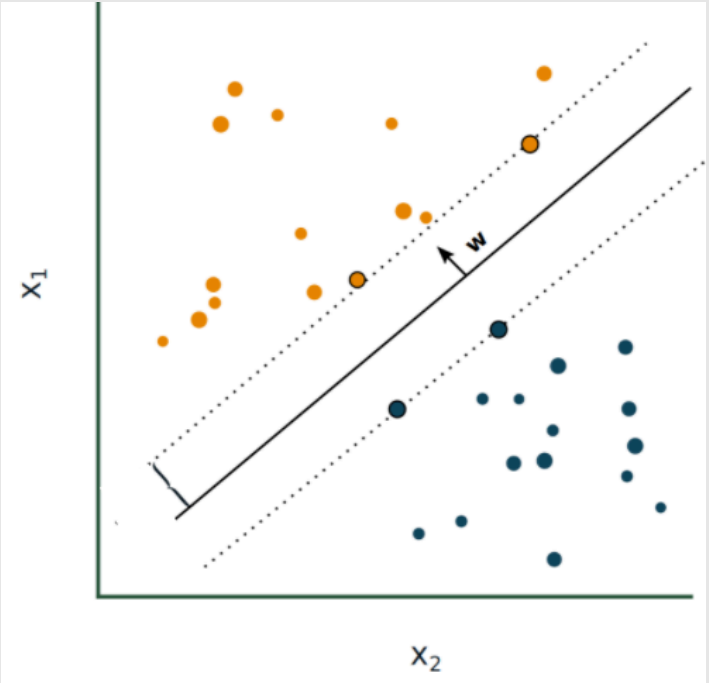
Support Vector Machine

Agenda

- Hard margin SVM
- Soft margin SVM
- Kernel trick

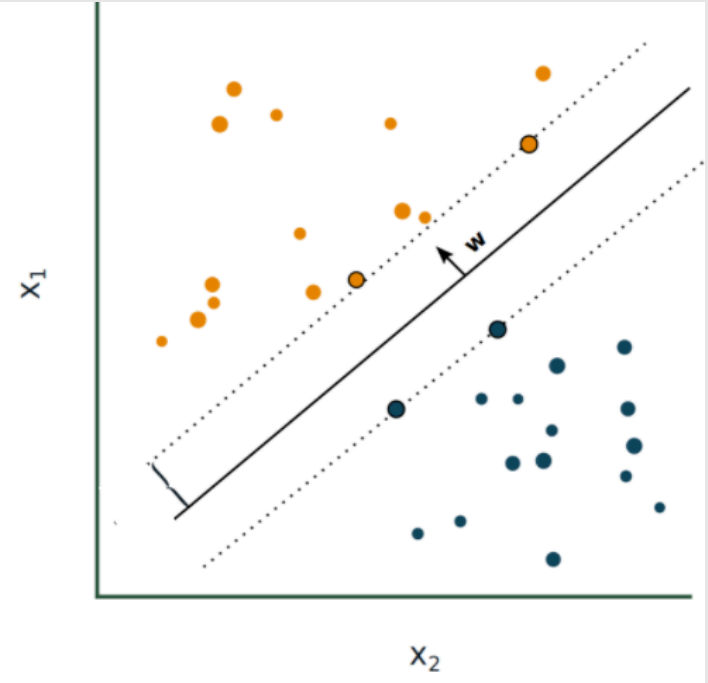
SVM is a discriminative method



Good according to intuition, practice

SVM became famous when, using images as input, it gave accuracy comparable to neural-network in a handwriting recognition task

SVM is a discriminative method



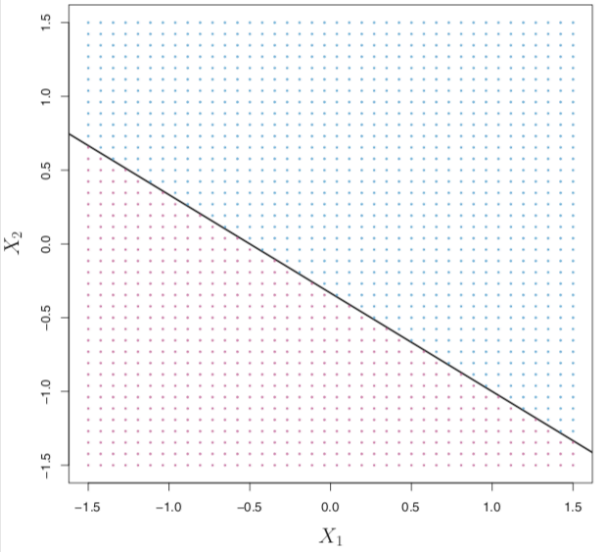
SVM is a binary classifier

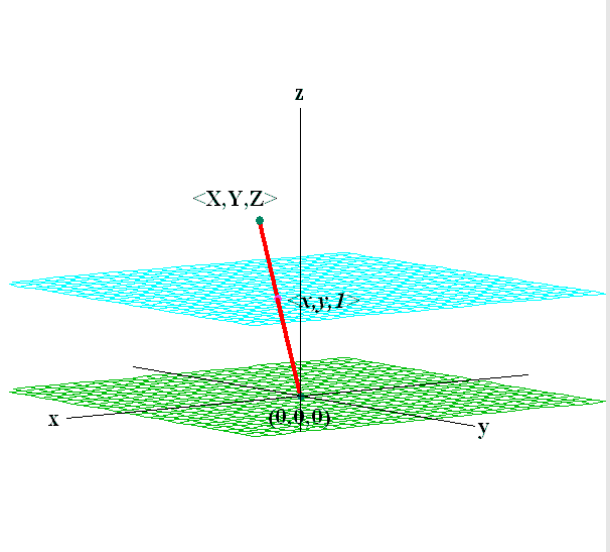
$$G(x) = \begin{cases} -1 & w^T x + w_0 < 0 \\ +1 & w^T x + w_0 \geq 0 \end{cases}$$
$$= sign(w^T x + w_0)$$

Hyperplane

Hyperplane

In a p-dimensional space, a **hyperplane** is a flat affine subspace of dimension p − 1
In two dimension case (p=2), a hyperplane is a one dimensional subspace: a line



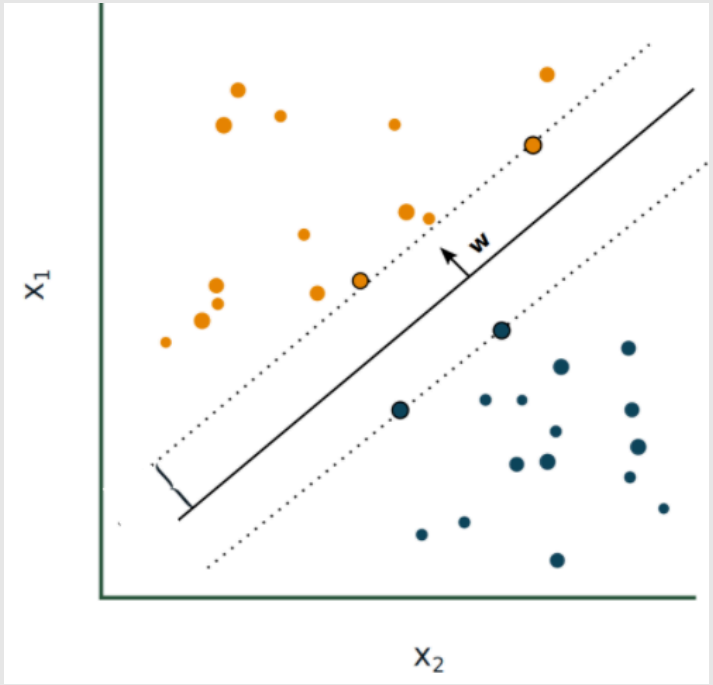


Maximal Margin Classifier

Core ideas:

- The classification must be correct
- The Margin must be maximal

We consider the (perpendicular) distance from each training observation to a given hyperplane. The smallest such distance is known as the margin



Maximal Margin Classifier

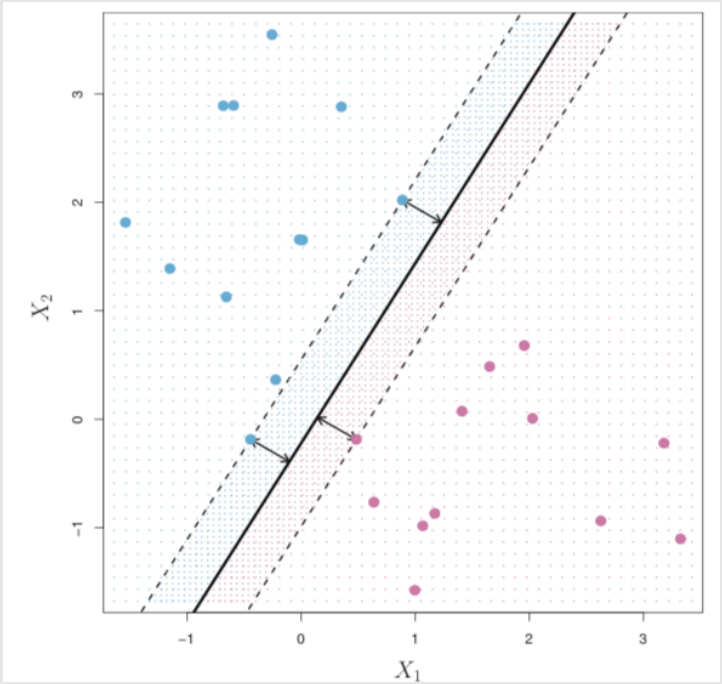
Point may never cross the boundary:

$$G(x) = \begin{cases} -1 & w^T x + w_0 < 0 \\ +1 & w^T x + w_0 \geq 0 \end{cases}$$
$$= sign(w^T x + w_0)$$

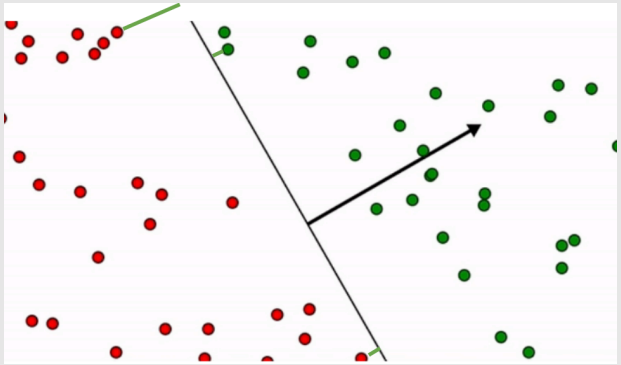
Or equivalently

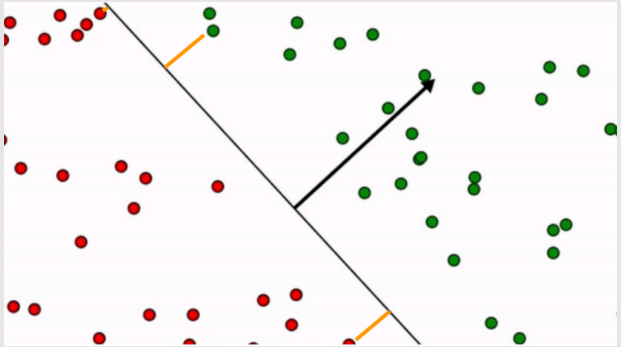
$$y^{(i)}\left(w^T x^{(i)} + w_0\right) \geq 0$$

$y^{(i)} f(x^{(i)}) \geq 0$



Maximal Margin Classifier

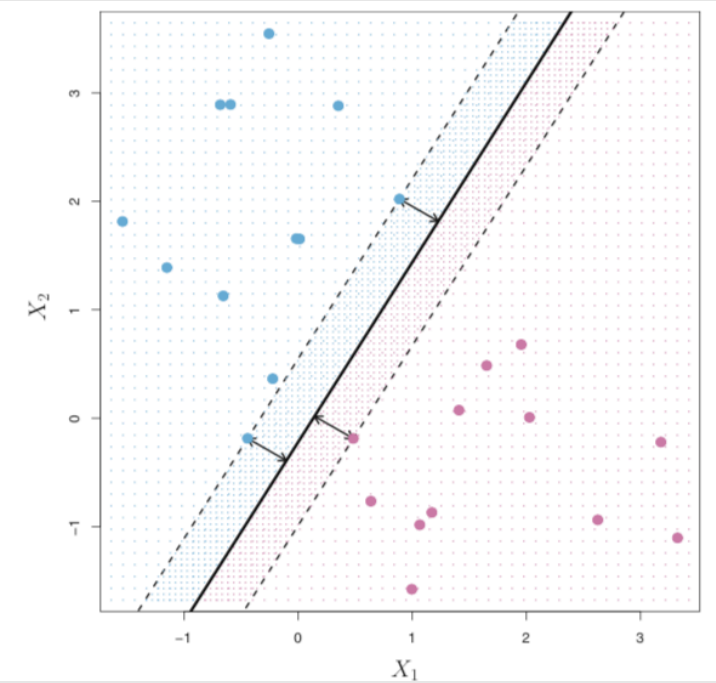




When the data from two classes are separable, there in fact exist an infinite number of separating hyperplanes.

Which of these has maximal margin?
Margin is the minimal distance from the observations to the hyperplane

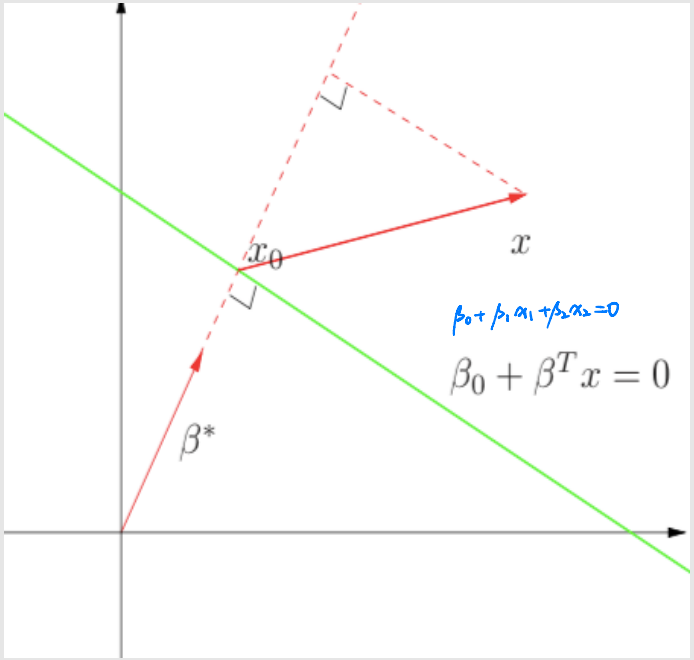
Maximal Margin Classifier



Maximal Margin Classifier is corresponding to the maximal margin hyperplane

Support Vectors: the 3 training observations are equidistant from the maximal margin hyperplane and lie along the dashed lines indicating the width of the margin.

Some linear algebra



The hyperplane L defined by $f(x) = \beta_0 + \beta^T x = 0$

$\beta^* = \beta / \|\beta\|$ is the unit vector orthogonal to the surface L

For any point x_0 in L, we have $\beta^T x_0 = -\beta_0$

The signed distance of any point x to L is given by

$$\beta^{*T}(x - x_0) = \frac{1}{\|\beta\|}(\beta^T x + \beta_0)$$

$$= \frac{1}{\|\beta\|} f(x)$$

Maximal Margin Classifier

Given the training data $\{(x_i, y_i)\}_{i=1}^{N}$

Define a hyperplane L by
$\{x | f(x) = x^T\beta + \beta_0 = 0\}$, where β is a unit vector: $\|\beta\| = 1$

The classification rule is $G(x) = sign(x^T\beta + \beta_0) = \begin{cases} -1 \text{ if } x^T\beta + \beta_0 < 0 \\ 1 \text{ if } x^T\beta + \beta_0 \geq 0 \end{cases}$

The signed distance from a point x to the hyperplane L is $f(x)/\|\beta\| = (x^T\beta + \beta_0)/\|\beta\|$

How can we find the hyperplane that gives the biggest margin?

Maximal Margin Classifier

$f(x) = \beta_0 + \beta^T x$

Maximal Margin Classifier:

$$\max_{\beta, \beta_0} M$$
subject to $\|\beta\| = 1$,
$y_i(\beta_0 + \beta^T x_i) \geq M$ for $i = 1, \dots N$

The distance from observation i to the hyperplane is given by $y_i(\beta_0 + \beta^T x_i)$

$\frac{y_i(\beta_0 + \beta^T x_i)}{\|\beta\|} \geq M$

$\|\beta\| = \frac{1}{M}$

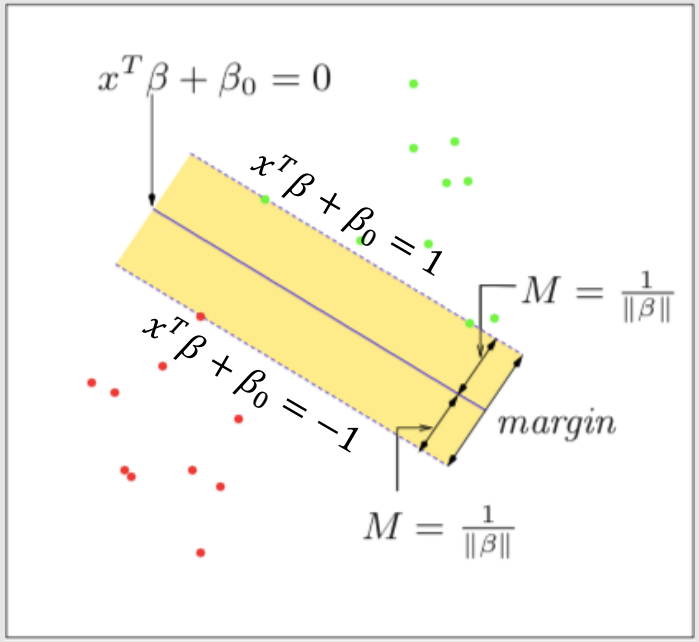
Get rid of $\|\beta\| = 1$ by using $\frac{1}{\|\beta\|} y_i(\beta_0 + \beta^T x_i) \geq M$, the problem is equivalent to

$$\min_{\beta, \beta_0} \frac{1}{2} \|\beta\|$$
$$s.t. \;\; y_i(\beta_0 + x_i^T \beta) \geq 1 \text{ for } i = 1, \dots N$$

$y_i(\beta_0 + \beta^T x_i) \geq 1$

The margin is now $M = 1/\|\beta\|$

Maximal Margin Classifier



Maximize the margin while keep all points well classified

$$\min_{\beta,\beta_0} \frac{1}{2}\|\beta\| \qquad \left(\min_{\beta,\beta_0} \frac{1}{2}\|\beta\|^2\right)$$

$$s.t.\ \ y_i\left(\beta_0 + x_i^T\beta\right) \geq 1 \text{ for } i = 1, \ldots N$$

Lagrange Multipliers

If f(x) is convex and continuously differentiable

$$\boldsymbol{Minimize\ f(x)}$$
$$\textbf{s.t. } \boldsymbol{h(x) = 0}$$

$$L(x, \lambda) = f(x) + \lambda h(x)$$

$$\nabla L_{x,\lambda}(x, \lambda) = 0$$

Dealing with inequality

$$\textbf{Minimize } f(x)$$
$$\text{s.t. } g_i(x) \leq 0, i = 1, \dots, k$$

Lagrangian: $L(x, \alpha) = f(x) + \sum_{i=1}^{k} \alpha_i g_i(x)$ ($\alpha_i \geq 0$)

Primal: $\min_x \max_{\alpha_i \geq 0} L(x, \alpha)$

Dual: $\max_{\alpha_i \geq 0} \min_x L(x, \alpha)$

① Inside feasible region: $g(x^*) < 0$
$\alpha_i = 0$, Reduce to min $f(x)$

② Out of feasible region $g(x^*) > 0$
$g_i(x) = 0$, min $f(x)$ w.r.t $g_i(x) = 0$

When g_i and f are convex Primal and Dual problems have the same solution. And the solutions satisfy the Karush-Kuhn-Tucker (KKT) conditions.

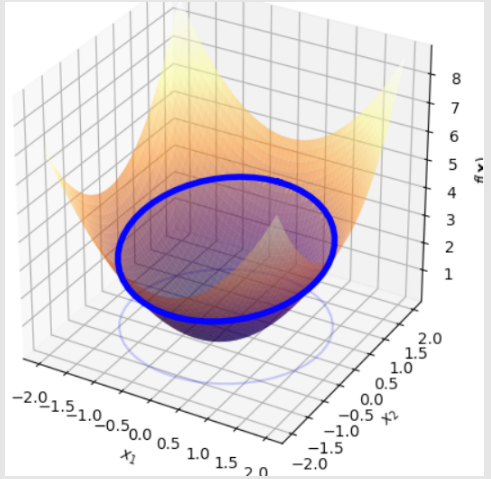
Karush–Kuhn–Tucker conditions

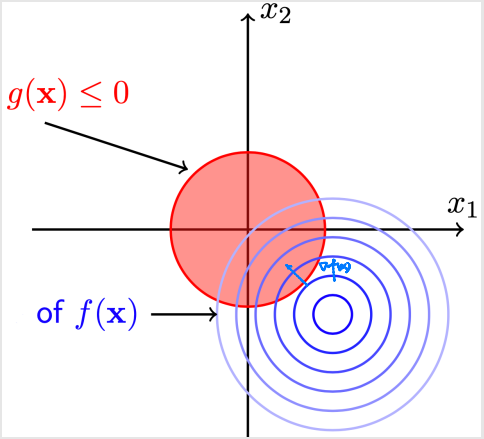
Lagrangian: $L(x, \alpha) = f(x) + \sum_{i=1}^{k} \alpha_i g_i(x)$

$$\begin{cases} \dfrac{\partial L}{\partial x}(x, \alpha) = 0 \\ \alpha_i g_i(x) = 0 \;\; i = 1, \dots k \\ g_i(x) \;\leq 0, i = 1, \dots, k \\ \alpha_i \geq 0, i = 1, \dots, k \end{cases}$$

Two scenarios

- when solution inside the feasible region: $g_i(x) \; < 0$
- when solution at the boundary of the feasible region: $g_i(x)$ =0





Karush–Kuhn–Tucker conditions

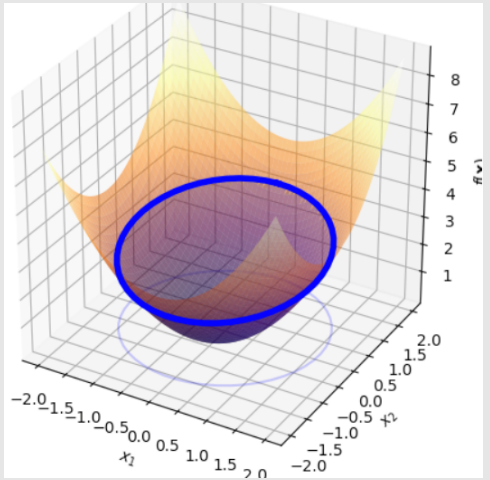
Lagrangian: $L(x, \alpha) = f(x) + \sum_{i=1}^{k} \alpha_i g_i(x)$

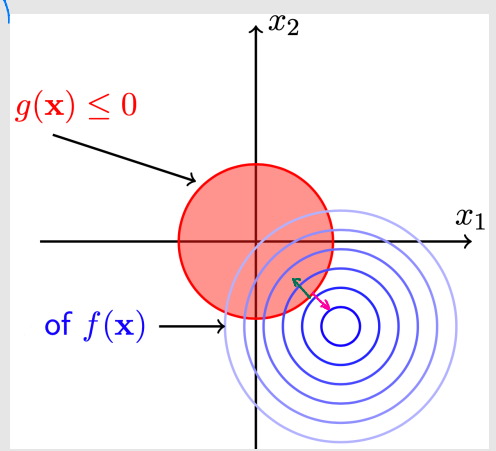
Solution for $\max_{\alpha_i \geq 0} \min_{x} L(x, \alpha)$ satisfies

$$\begin{cases} \dfrac{\partial L}{\partial x}(x, \alpha) = 0 \\ \alpha_i g_i(x) = 0 \;\; i = 1, \dots k \\ g_i(x) \leq 0, i = 1, \dots, k \\ \alpha_i \geq 0, i = 1, \dots, k \end{cases}$$

$-\nabla f(x)$ point to non-feasible region

$\nabla g(x)$ point to non-feasible region





Two scenarios

- At solution, $g_i(x^*) < 0$, the function is optimized in unrestricted way , so $\left.\frac{\partial f}{\partial x}\right|_{x^*}$=0, it's an unconstrainted case with α_i^*=0
- At solution, $g_i(x^*)$ =0, then there exists a α_i^*such that $\frac{\partial L}{\partial x}(x^*; \alpha_i^*) = \left.\frac{\partial f}{\partial x}\right|_{x^*} + \alpha_i^* \left.\frac{\partial g_i}{\partial x}\right|_{x^*} = 0$ and α_i^*>0

Lagrange method

$$\min_{\beta,\beta_0} \frac{1}{2}\|\beta\|^2$$

$$s.t.\ \ y_i(\beta_0 + x_i^T\beta) \geq 1\ for\ i = 1, \dots N$$

Lagrangian: $L(\beta, \beta_0, \alpha) = \frac{1}{2}\|\beta\|^2 - \sum_{i=1}^{N} \alpha_i[y_i(x_i^T\beta + \beta_0) - 1]$

Dual problem: $\max_{\alpha_i \geq 0} \min_{\beta,\beta_0} L(\beta, \beta_0, \alpha)$

Setting derivatives $\nabla_\beta L(\beta, \beta_0, \alpha) = 0$ and $\nabla_{\beta_0} L(\beta, \beta_0, \alpha) = 0$ to zero, we get

$$\begin{cases} \beta = \sum_{i=1}^{N} \alpha_i y_i x_i \\ 0 = \sum_{i=1}^{N} \alpha_i y_i \end{cases}$$

Lagrange method

$$= \frac{1}{2}\|\sum_i \alpha_i y_i \vec{x}_i\|^2 - \sum_i \alpha_i \left[y_i \vec{x}_i^T \sum_j \alpha_j y_j \vec{x}_j - 1 \right]$$
$$= \frac{1}{2}\sum_i \sum_j \alpha_i y_i \alpha_j y_j \vec{x}_i^T \vec{x}_j - \sum_i \sum_j \alpha_i y_i \alpha_j y_j \vec{x}_i^T \vec{x}_j + \sum_i \alpha_i$$
$$= \sum \alpha_i - \frac{1}{2}\sum_i \sum_j \alpha_i \alpha_j y_i y_j \vec{x}_i^T \vec{x}_j$$

Plug $\begin{cases} \beta = \sum_{i=1}^N \alpha_i y_i x_i \\ 0 = \sum_{i=1}^N \alpha_i y_i \end{cases}$ into $L(\beta, \beta_0, \alpha) = \frac{1}{2}\|\beta\|^2 - \sum_{i=1}^N \alpha_i \left[y_i \left(x_i^T \beta + \beta_0 \right) - 1 \right]$ we get

$$L(\beta, \beta_0, \alpha) = \sum_{i=1}^N \alpha_i - \frac{1}{2}\sum_i \sum_j y_i y_j \alpha_i \alpha_j \langle x_i, x_j \rangle$$

$\max_{\alpha_i \geq 0} \min_{\beta, \beta_0} L(\beta, \beta_0, \alpha)$

The dual problem is equivalent to

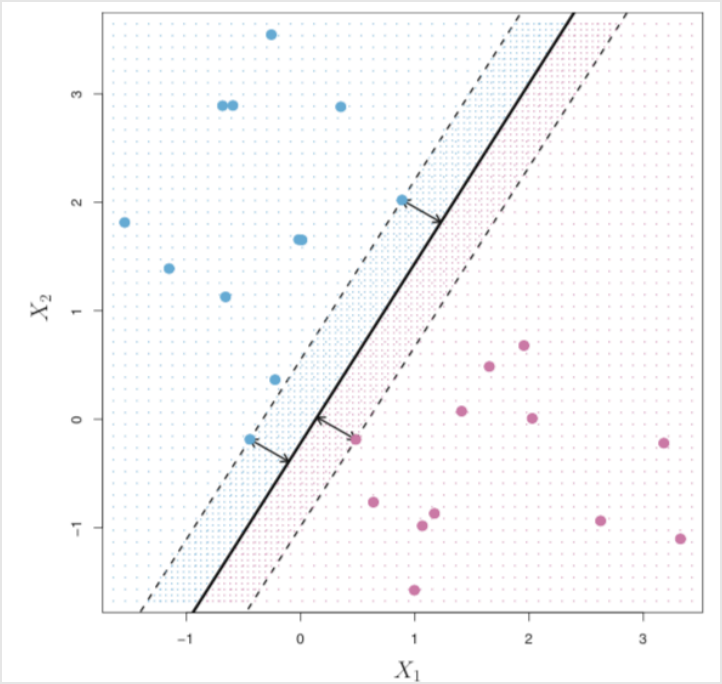
$$\max_{\alpha} \sum_{i=1}^N \alpha_i - \frac{1}{2}\sum_{i=1}^N y_i y_j \alpha_i \alpha_j \langle x_i, x_j \rangle$$
$$s.t. \ \alpha_i \geq 0, i = 1, \ldots, k$$
$$\sum_{i=1}^N \alpha_i y_i = 0$$

Where does the name SVM come from?

From KKT condition

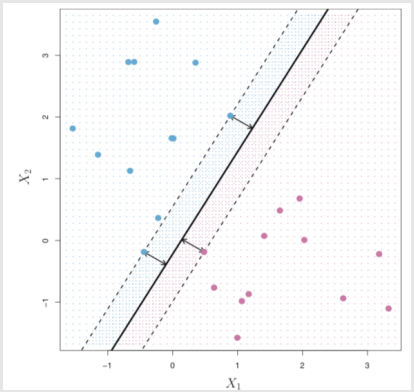
$\alpha_i\left[y_i(x_i^T\beta+\beta_0)-1\right]=0$

- If α_i>0, then $y_i(x_i^T\beta+\beta_0)$ =1, x_i is on the boundary of the slab
- If $y_i\left(x_i^T\beta+\beta_0\right)>1$, x_i is not on the boundary of the slab, and α_i=0



Support vectors

Advantage



$$f(x) = \langle \beta, x \rangle + \beta_0$$

$$= \beta_0 + \sum_{i=1}^{N} y_i \alpha_i \langle x_i, x \rangle$$

Only consider support vectors

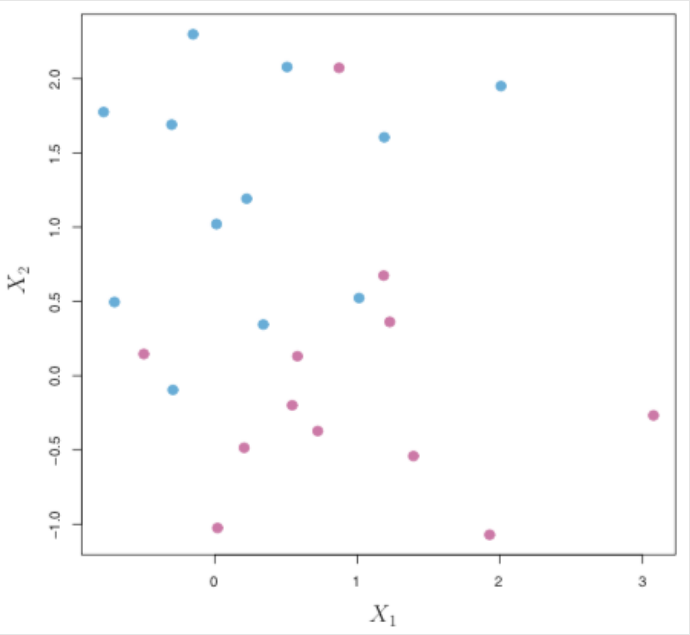
$$\beta = \sum_{i=1}^{N} \alpha_i y_i x_i$$

α is sparse. They are all zero, except for the support vectors!

$\hat{y} = \text{sign}(f(x)) = \text{sign}(\beta^T x + \beta_0)$

Soft Margin SVM

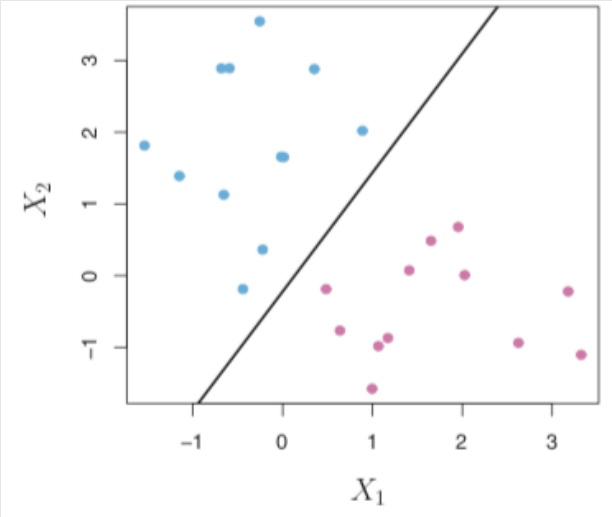
Non separable case



When two classes are not separable by a hyperplane, maximal margin classifier does not exist.

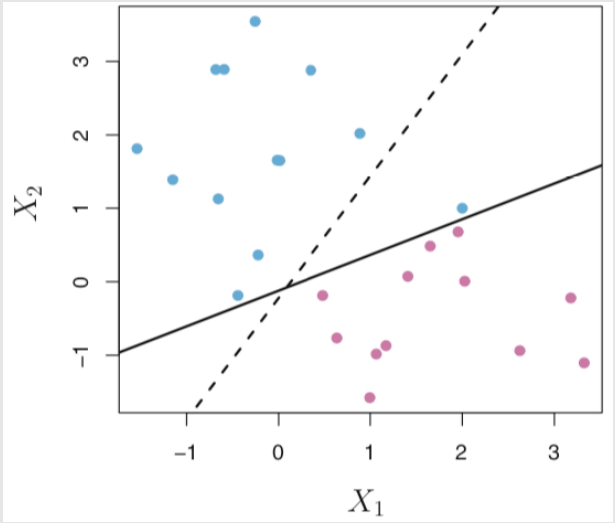
In such cases, we can allow some number of observations to violate the rules so that they can lie on the wrong side of the margin boundaries. We can develop a hyperplane that *almost* separates the classes.

Support Vector Classifier

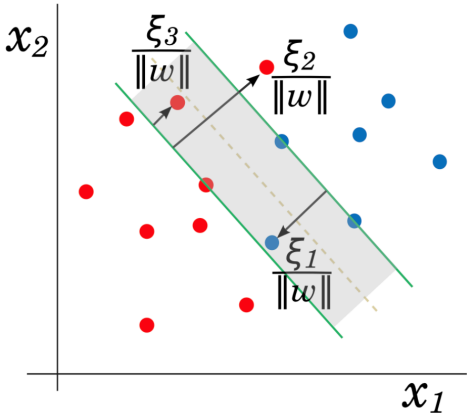


An observation can be on the wrong side of the margin or the wrong side of hyperplane

- Greater robustness to individual observations, and
- Better classification of most of the training observations.



Support Vector Classifier



$$\max_{\beta,\beta_0,\varepsilon_1,\ldots\varepsilon_n} M$$

subject to $\|\beta\| = 1$

$$y_i(\beta_0 + \beta^T x_i) \geq M(1 - \varepsilon_i) \text{for } i = 1, \ldots N$$

$$\varepsilon_i \geq 0$$

$$\sum_{i=1}^{n} \varepsilon_i \leq C$$

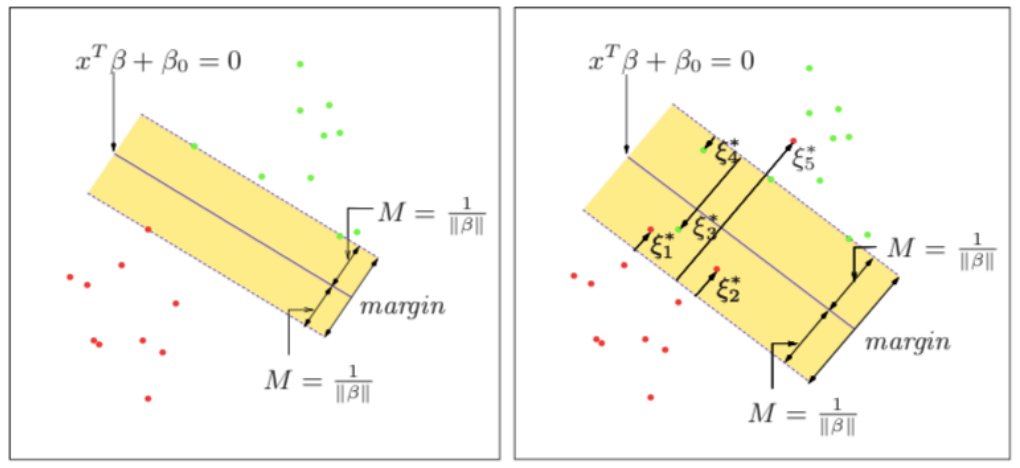
where C is a nonnegative **tuning** parameter, M is the width of the margin.
$\varepsilon_1, \ldots, \varepsilon_n$ allow individual observations to lie on the wrong side of the margin or the hyperplane, they are called **slack variables**.

More on the slack variables

The slack variable ε_i indicates where the i^{th} observation is located

- $\varepsilon_i = 0$: the i^{th} observation is on the correct side of the margin
- $\varepsilon_i > 0$: the i^{th} observation violates the margin

C bound the sum of ε_i's and it determines the number and severity of the violation to the margin



Support Vector Classifier

distance $\frac{y_i(\beta_0+\beta^T x_i)}{\|\beta\|} \geq \frac{1-\varepsilon_i}{\|\beta\|}$

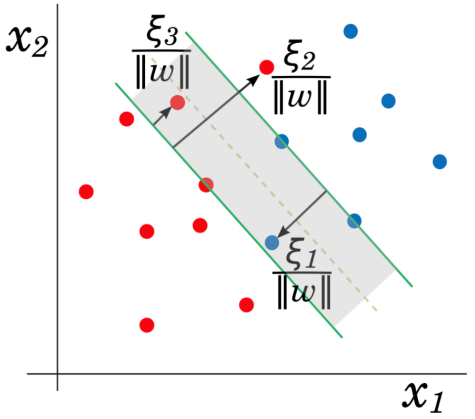
$\varepsilon_i = 0$, signed distance $\geq \frac{1}{\|\beta\|}$

$\varepsilon_i > 0$, $\frac{1}{\|\beta\|} \leq$ signed distance $= \frac{1-\varepsilon_i}{\|\beta\|}$

Voilate the margin

λ small ~ 0, allow more voilation

Large $\lambda \to$ too, No voilation $\to$ Hard margin SVM



$$\min_{\beta,\beta_0} \frac{1}{2}\|\beta\|^2 + \lambda \sum_i \varepsilon_i$$

subject to

$$y_i(\beta_0 + \beta^T x_i) \geq (1 - \varepsilon_i) \text{ for } i = 1, \ldots N$$

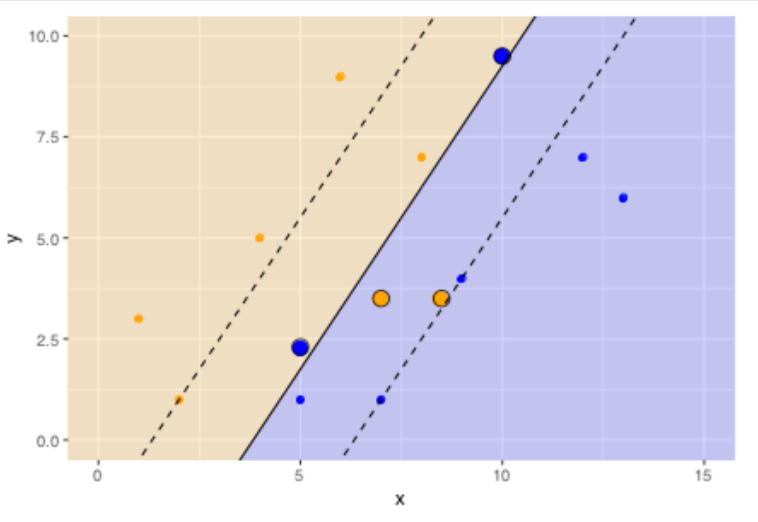
$$\varepsilon_i \geq 0$$

Where λ is a nonnegative **tuning** parameter.

$\varepsilon_1, \ldots, \varepsilon_n$ allow individual observations to lie on the wrong side of the margin or the hyperplane, they are called **slack variables**.

Support Vector Classifier

Once the optimization problem is solves, the **classification** follows instantly by looking at **which side of the hyperplane** the observation lies

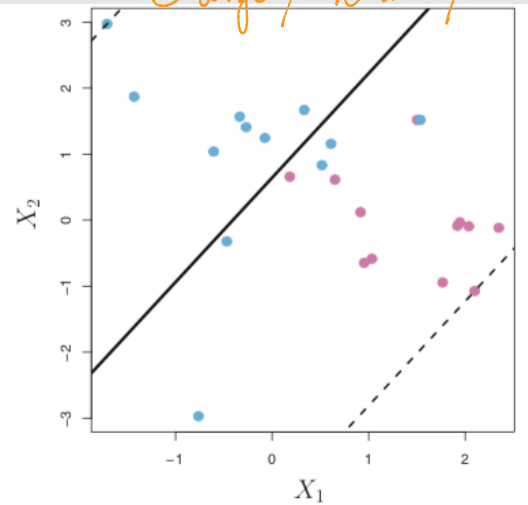


Support vectors

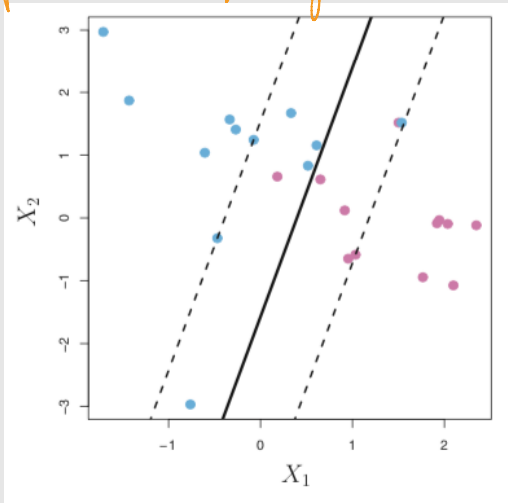
Only observations lies on the margin or violate the margin will affect the hyperplane.

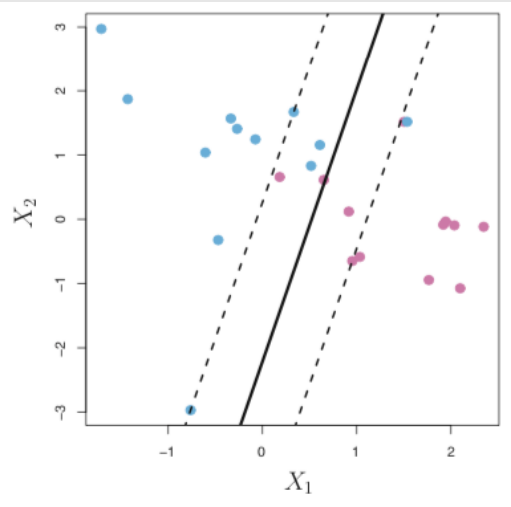
Support Vector Classifier

C large / λ small / underfit



Smaller C / Large λ



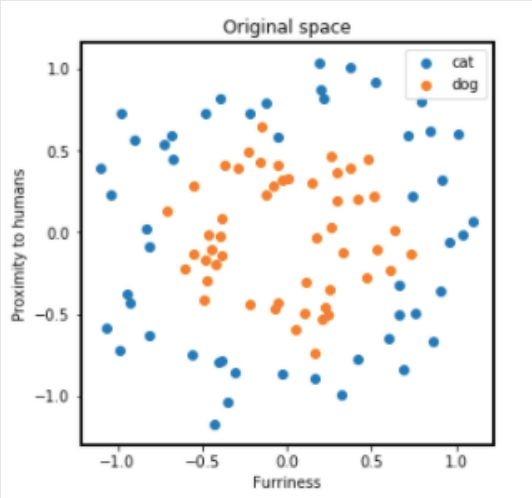


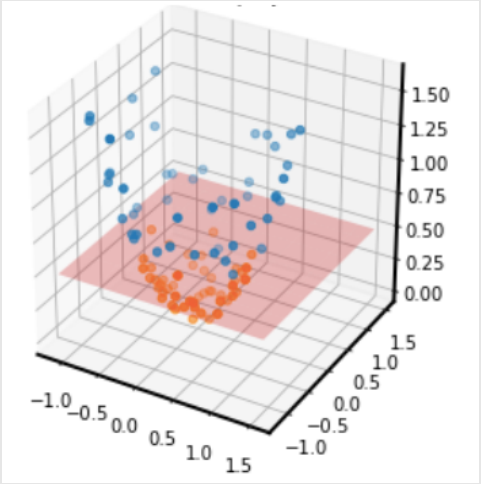
When the tuning parameter C is large, then the margin is wide, many observations violate the margin, and so there are many support vectors.

Kernel Trick

Support Vector Machine

When the classes are not linearly separable, we **convert a linear classifier** into a classifier that produced **non-linear decision boundaries**.





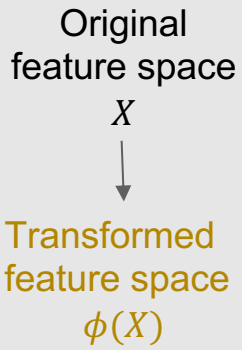
Support Vector Machine

Instead of fitting a support vector classifier using

$$X_1, X_2, \ldots . X_p$$

We can fit a support vector classifier using:

$$X_1, X_1^2, X_2, X_2^2 \ldots . X_p, X_p^2$$



Support Vector Machine

Recall support vector classifier

$f(x) = \beta_0 + \sum_{i=1}^{N} y_i \alpha_i \langle x_i, x \rangle$ where α_i is non zero only for support vectors

Now consider the new feature space $x \to \phi(x)$, if we define

$$k(x_i, x_j) = \langle \phi(x_i), \phi(x_j) \rangle = \phi(x_i)^T \phi(x_j)$$

Then we have the classifier with the new features

$$f(x) = \langle \beta, \phi(x) \rangle + \beta_0$$

$$= \beta_0 + \sum_{i=1}^{N} y_i \alpha_i \, k(x_i, x_j)$$

Support Vector Machine

$k(x_i, x_j)$ is the kernel function, it quantifies the similarity of the two observations.

Linear kernel: $k(x_i, x_j) = x_i^T x_j$ → Support vector classifier

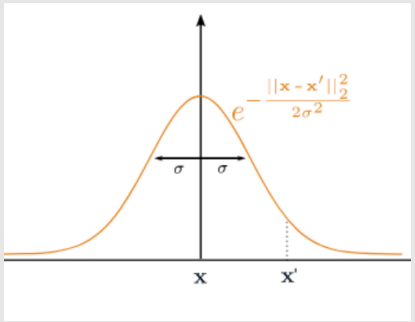
d-degree polynomial: $k(x_i, x_j) = (1 + x_i^T x_j)^d$

Radial: $k(x_i, x_j) = \exp(-\gamma \|x_i - x_j\|^2)$

For the radial kernel, a given test observation x is far from a training observation x_i in terms of Euclidean distance, then $\|x - x_i\|^2$ will be large, and so $k(x, x_i)$ will tiny.

Radial basis function kernel (RBF): $k(x_i, x_j) = \exp(-\gamma \left\| x_i - x_j \right\|^2)$

It's kind of like doing a KNN with a smooth neighborhood



$$
\begin{aligned}
k(\mathbf{x}, \mathbf{x}') &= \langle \phi(\mathbf{x}), \phi(\mathbf{x}') \rangle \\
&= e^{\frac{||\mathbf{x}-\mathbf{x}'||^2}{2}} \\
&\approx C \sum_{n=0}^{\infty} \frac{\langle \mathbf{x}, \mathbf{x}' \rangle^n}{n!}
\end{aligned}
$$

Support Vector Machine