SUPPORT VECTOR MACHINES

CS6140

Predrag Radivojac
KHOURY COLLEGE OF COMPUTER SCIENCES
NORTHEASTERN UNIVERSITY

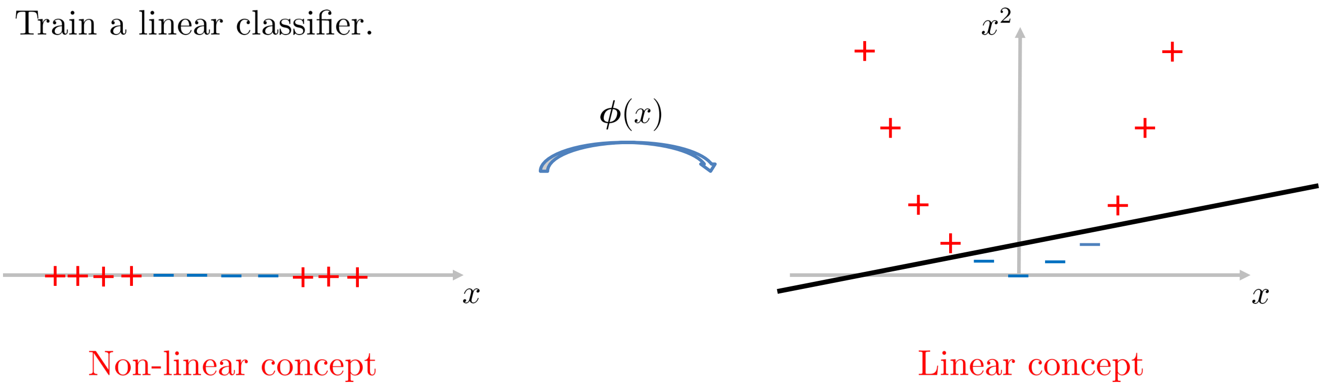
Fall 2024

Main Idea

Given: $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^n$, where $x_i \in \mathcal{X}$ and $y_i \in \mathcal{Y}$.

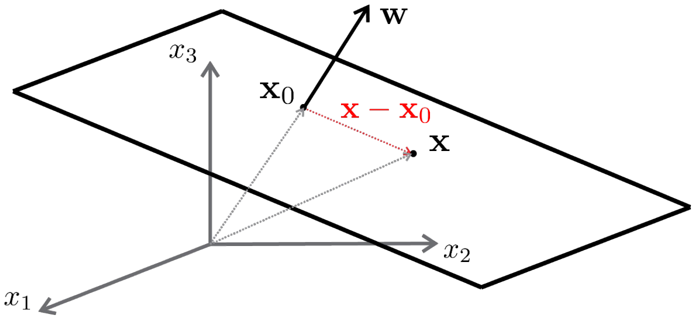
Here, $\mathcal{X} = \mathbb{R}$, $\mathcal{Y} = \{-1, +1\}$.

Goal: Train a linear classifier.



Desiderata: Avoid explicitly finding $\phi : \mathbb{R} \to \mathbb{R}^2$. Allow for $\mathcal{X}$ to extend beyond $\mathbb{R}^d$.
Sparse solution.

Equation of the Plane



A plane is defined using:

1. a point $\mathbf{x}_0$ lying in the plane
2. a vector $\mathbf{w}$ normal to the plane

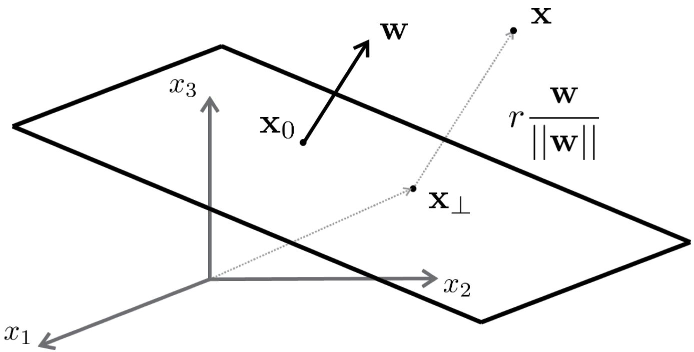
Let $\mathbf{x}$ be on the plane defined by $\mathbf{w}$ and $\mathbf{x}_0$:

$$\mathbf{w}^T(\mathbf{x} - \mathbf{x}_0) = 0$$

$$\mathbf{w}^T\mathbf{x} - \mathbf{w}^T\mathbf{x}_0 = 0$$

$$\boxed{\mathbf{w}^T\mathbf{x} + w_0 = 0}$$

Distance from Point to the Plane



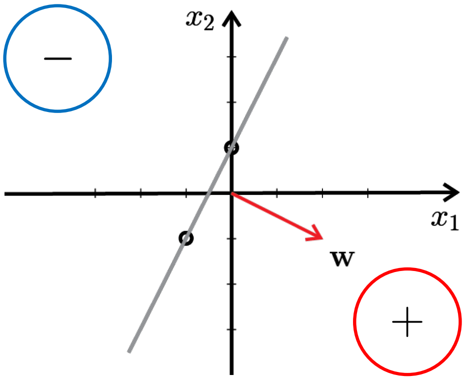
$\mathbf{x}$ = outside the plane

$$\mathbf{x} = \mathbf{x}_\perp + r\frac{\mathbf{w}}{||\mathbf{w}||}$$

$$\mathbf{w}^T\mathbf{x} + w_0 = \underbrace{\mathbf{w}^T\mathbf{x}_\perp + w_0}_{0} + r||\mathbf{w}||$$

$$\boxed{r = \frac{\mathbf{w}^T\mathbf{x} + w_0}{||\mathbf{w}||}}$$

Example



$$x_2 = 2x_1 + 1 \qquad \text{or} \qquad 2x_1 - x_2 + 1 = 0$$

$$\mathbf{x}, \mathbf{w} \in \mathbb{R}^2$$

$$\mathbf{w}^T\mathbf{x} + w_0 = 0$$

where $\mathbf{w} = (2, -1)$ and $w_0 = 1$.

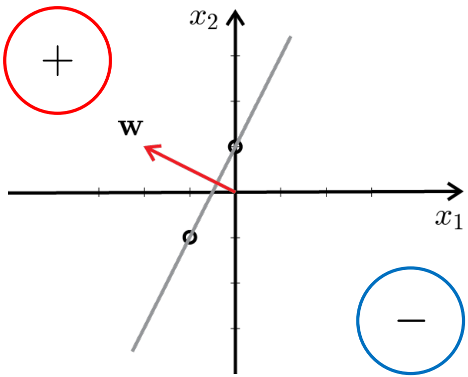
$$r = \frac{\mathbf{w}^T\mathbf{x} + w_0}{||\mathbf{w}||}$$

$$\mathbf{x} = (0, 0) \quad \Longrightarrow r = \tfrac{1}{\sqrt{5}}$$

$$\mathbf{x} = (-1, 1) \quad \Longrightarrow r = -\tfrac{2}{\sqrt{5}}$$

The vector **w** defines what side of the plane is positive.

Example



$$x_2 = 2x_1 + 1$$

What if $\mathbf{w} = (-2, 1)$?

$$\mathbf{x}, \mathbf{w} \in \mathbb{R}^2$$

$$\mathbf{w}^T\mathbf{x} + w_0 = 0$$

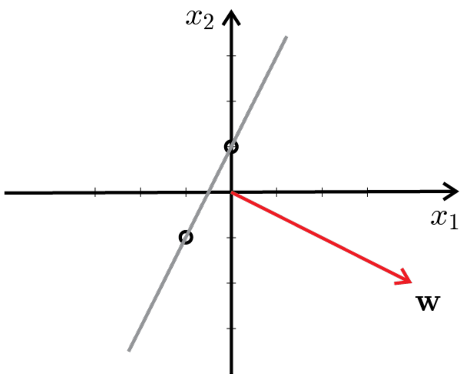
where $\mathbf{w} = (-2, 1)$ and $w_0 = -1$.

$$r = \frac{\mathbf{w}^T\mathbf{x} + w_0}{||\mathbf{w}||}$$

$$\mathbf{x} = (0, 0) \implies r = -\frac{1}{\sqrt{5}}$$

$$\mathbf{x} = (-1, 1) \implies r = \frac{2}{\sqrt{5}}$$

Example



$$x_2 = 2x_1 + 1$$

What if $\mathbf{w} = (4, -2)$ and $w_0 = 2$?

$$4x_1 - 2x_2 + 2 = 0$$

$\mathbf{w}^T\mathbf{x} + w_0$ is “bigger”!!!

$$r = \frac{\mathbf{w}^T\mathbf{x} + w_0}{||\mathbf{w}||}$$

$$\mathbf{x} = (0, 0) \implies r = \frac{1}{\sqrt{5}}$$

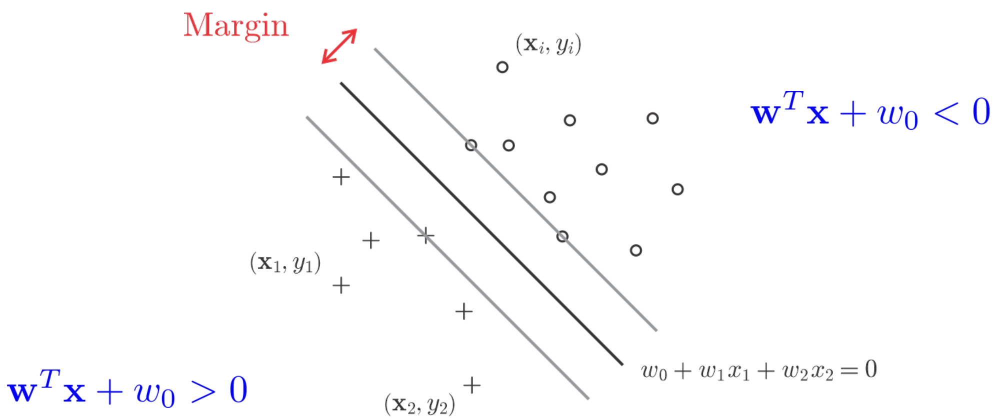
$$\mathbf{x} = (-1, 1) \implies r = -\frac{2}{\sqrt{5}}$$

Distances are unchanged when $\mathbf{w}$ and w_0 are multiplied by a constant!

Problem Formulation

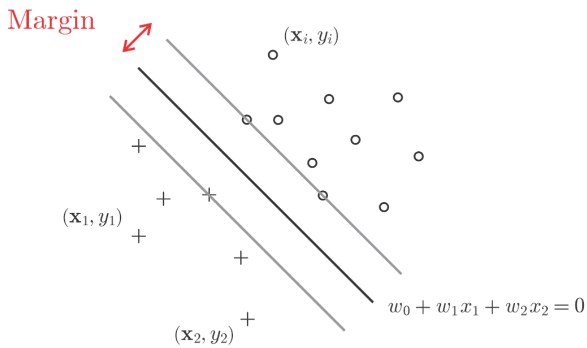
Given: $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{n}$, where $\mathbf{x}_i \in \mathbb{R}^d$ and $y_i \in \{-1, +1\}$. Data is linearly separable.

Objective: Find separating hyperplane such that the minimum distance from any data point to the hyperplane is maximized.



*Margin can also be defined as double of the minimum distance to the separating hyperplane

MAXIMIZING MARGIN



$$\mathbf{w}^T\mathbf{x}_i + w_0 > 0 \quad \Longrightarrow y_i = +1$$
$$\mathbf{w}^T\mathbf{x}_i + w_0 < 0 \quad \Longrightarrow y_i = -1$$

$$y_i(\mathbf{w}^T\mathbf{x}_i + w_0) > 0$$
$$i \in \{1, 2, \ldots, n\}$$

Idea: find $(\mathbf{w}, w_0)$ to maximize minimum unsigned distance $d_i = \dfrac{y_i(\mathbf{w}^T\mathbf{x}_i + w_0)}{||\mathbf{w}||}$

$$(\mathbf{w}^*, w_0^*) = \arg\max_{\mathbf{w}, w_0} \left\{ \frac{1}{||\mathbf{w}||} \min_i \left(y_i(\mathbf{w}^T\mathbf{x}_i + w_0) \right) \right\}$$

Reformulating the Problem

$$(\mathbf{w}^*, w_0^*) = \arg\max_{\mathbf{w}, w_0} \left\{ \frac{1}{||\mathbf{w}||} \min_i \left(y_i(\mathbf{w}^T \mathbf{x}_i + w_0) \right) \right\}$$

Scale $\mathbf{w}$ and w_0 such that $\min_i \left\{ y_i(\mathbf{w}^T \mathbf{x}_i + w_0) \right\} = 1$

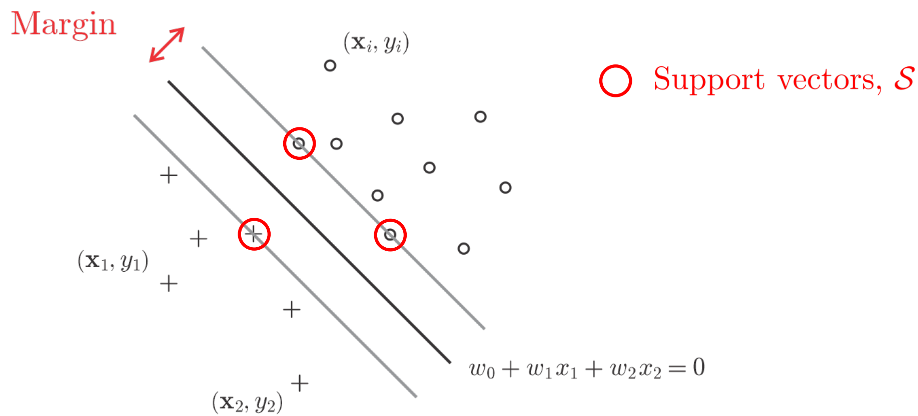
$$\mathbf{w} \leftarrow k \cdot \mathbf{w}$$
$$w_0 \leftarrow k \cdot w_0$$

$$(\mathbf{w}^*, w_0^*) = \arg\min_{\mathbf{w}} \{||\mathbf{w}||\}$$

Subject to:

$$y_i(\mathbf{w}^T \mathbf{x}_i + w_0) \geq 1 \quad \forall i \in \{1, 2, \ldots, n\}$$

Final Problem Formulation



$$(\mathbf{w}^*, w_0^*) = \arg\min_{\mathbf{w}} \left\{ \frac{1}{2} \mathbf{w}^T \mathbf{w} \right\}$$ ⟵ Convex function!

Subject to:

$$y_i(\mathbf{w}^T \mathbf{x}_i + w_0) \geq 1 \quad \forall i \in \{1, 2, \ldots, n\}$$ ⟵ Linear constraints!

How can We Solve it?

$$(\mathbf{w}^*, w_0^*) = \arg\min_{\mathbf{w}} \left\{ \frac{1}{2} \mathbf{w}^T \mathbf{w} \right\}$$

Subject to:

$$y_i(\mathbf{w}^T \mathbf{x}_i + w_0) \geq 1 \quad \forall i \in \{1, 2, \ldots, n\}$$

Solution: use Lagrangians!

$$L(\mathbf{w}, w_0, \boldsymbol{\alpha}) = \frac{1}{2} \mathbf{w}^T \mathbf{w} - \sum_{i=1}^{n} \alpha_i \left(y_i \left(\mathbf{w}^T \mathbf{x}_i + w_0 \right) - 1 \right) \qquad \alpha_i \geq 0$$

Solving it

$$\frac{\partial}{\partial w_j} L(\mathbf{w}, w_0, \boldsymbol{\alpha}) = 0 \qquad \Longrightarrow \qquad w_j = \sum_{i=1}^{n} \alpha_i y_i x_{ij}$$

After d derivatives... $\Longrightarrow$ $$\mathbf{w} = \sum_{i=1}^{n} \alpha_i y_i \mathbf{x}_i$$

$$\frac{\partial}{\partial w_0} L(\mathbf{w}, w_0, \boldsymbol{\alpha}) = 0 \qquad \Longrightarrow \qquad \sum_{i=1}^{n} \alpha_i y_i = 0$$

DUAL PROBLEM

$$L^{\text{dual}}(\mathbf{w}, w_0, \boldsymbol{\alpha}) = \frac{1}{2}\mathbf{w}^T\mathbf{w} - \sum_{i=1}^{n} \alpha_i y_i \mathbf{w}^T \mathbf{x}_i - \sum_{i=1}^{n} \alpha_i y_i w_0 + \sum_{i=1}^{n} \alpha_i$$

For the "duality" theorem see Fletcher. Practical methods of optimization. 1987.

Dual Problem

$$
\begin{aligned}
L^{\text{dual}}(\mathbf{w}, w_0, \boldsymbol{\alpha}) &= \frac{1}{2}\mathbf{w}^T\mathbf{w} - \sum_{i=1}^{n} \alpha_i y_i \mathbf{w}^T \mathbf{x}_i - \sum_{i=1}^{n} \alpha_i y_i w_0 + \sum_{i=1}^{n} \alpha_i \\
&= \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j \mathbf{x}_i^T \mathbf{x}_j - \sum_{i=1}^{n} \alpha_i y_i (\sum_{j=1}^{n} \alpha_j y_j \mathbf{x}_j)^T \mathbf{x}_i + \sum_{i=1}^{n} \alpha_i
\end{aligned}
$$

For the “duality” theorem see Fletcher. Practical methods of optimization. 1987.

Dual Problem

$$
\begin{aligned}
L^{\text{dual}}(\mathbf{w}, w_0, \boldsymbol{\alpha}) &= \frac{1}{2}\mathbf{w}^T\mathbf{w} - \sum_{i=1}^{n} \alpha_i y_i \mathbf{w}^T\mathbf{x}_i - \sum_{i=1}^{n} \alpha_i y_i w_0 + \sum_{i=1}^{n} \alpha_i \\
&= \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i\alpha_j y_i y_j \mathbf{x}_i^T\mathbf{x}_j - \sum_{i=1}^{n} \alpha_i y_i (\sum_{j=1}^{n} \alpha_j y_j \mathbf{x}_j)^T \mathbf{x}_i + \sum_{i=1}^{n} \alpha_i \\
&= \sum_{i=1}^{n} \alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i\alpha_j y_i y_j \mathbf{x}_i^T\mathbf{x}_j
\end{aligned}
$$

For the “duality” theorem see Fletcher. Practical methods of optimization. 1987.

DUAL PROBLEM

$$
\begin{aligned}
L^{\text{dual}}(\mathbf{w}, w_0, \boldsymbol{\alpha}) &= \frac{1}{2}\mathbf{w}^T\mathbf{w} - \sum_{i=1}^{n} \alpha_i y_i \mathbf{w}^T \mathbf{x}_i - \sum_{i=1}^{n} \alpha_i y_i w_0 + \sum_{i=1}^{n} \alpha_i \\
&= \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j \mathbf{x}_i^T \mathbf{x}_j - \sum_{i=1}^{n} \alpha_i y_i (\sum_{j=1}^{n} \alpha_j y_j \mathbf{x}_j)^T \mathbf{x}_i + \sum_{i=1}^{n} \alpha_i \\
&= \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j \mathbf{x}_i^T \mathbf{x}_j \\
&= \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j k(\mathbf{x}_i, \mathbf{x}_j)
\end{aligned}
$$

Subject to:

$$\alpha_i \geq 0 \qquad \forall i \in \{1, 2, \ldots, n\}$$

$$\sum_{i=1}^{n} \alpha_i y_i = 0$$

For the "duality" theorem see Fletcher. Practical methods of optimization. 1987.

SOLVING THE DUAL PROBLEM

Use quadratic programming to solve for $\boldsymbol{\alpha}$

$$\Longrightarrow \qquad \mathbf{w} = \sum_{i=1}^{n} \alpha_i y_i \mathbf{x}_i$$

$$\Longrightarrow \qquad f(\mathbf{x}) = \mathbf{w}^T \mathbf{x} + w_0$$

Solving the Dual Problem

Use quadratic programming to solve for $\boldsymbol{\alpha}$

$$\implies \qquad \mathbf{w} = \sum_{i=1}^{n} \alpha_i y_i \mathbf{x}_i$$

$$\implies \qquad f(\mathbf{x}) = \mathbf{w}^T \mathbf{x} + w_0$$
$$= \sum_{i=1}^{n} \alpha_i y_i \mathbf{x}_i^T \mathbf{x} + w_0$$

Solving the Dual Problem

Use quadratic programming to solve for $\boldsymbol{\alpha}$

$$\implies \qquad \mathbf{w} = \sum_{i=1}^{n} \alpha_i y_i \mathbf{x}_i$$

$$\implies \qquad \begin{aligned} f(\mathbf{x}) &= \mathbf{w}^T \mathbf{x} + w_0 \\ &= \sum_{i=1}^{n} \alpha_i y_i \mathbf{x}_i^T \mathbf{x} + w_0 \\ &= \sum_{i=1}^{n} \alpha_i y_i k(\mathbf{x}_i, \mathbf{x}) + w_0 \end{aligned}$$

Analysis of the Solution

Karush-Kuhn-Tucker (KKT) conditions:

$$\begin{aligned} &\alpha_i \geq 0 \\ &y_i(\mathbf{w}^T\mathbf{x}_i + w_0) - 1 \geq 0 \qquad \forall i \in \{1, 2, \ldots, n\} \\ &\alpha_i\left(y_i\left(\mathbf{w}^T\mathbf{x}_i + w_0\right) - 1\right) = 0 \end{aligned}$$

This means that for $\forall i$, either $\alpha_i = 0$ or $y_i\left(\mathbf{w}^T\mathbf{x}_i + w_0\right) = 1$

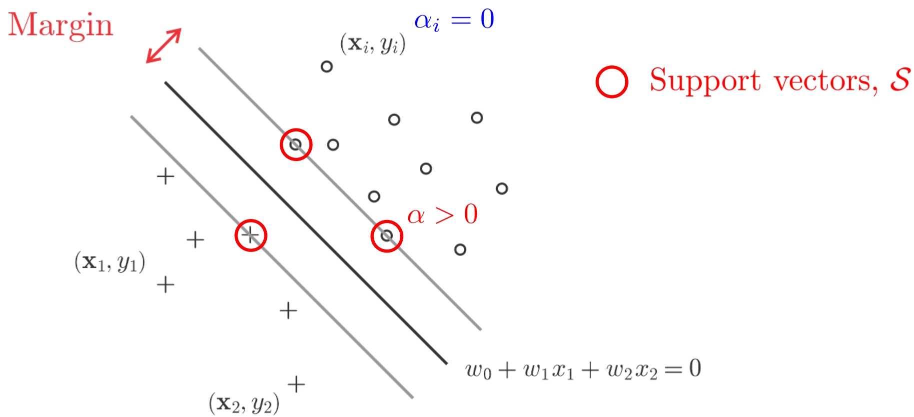
$\implies$ $\alpha_i = 0$ for all vectors that are not support vectors

$$f(\mathbf{x}) = \sum_{\mathbf{x}_i \in \mathcal{S}} \alpha_i y_i k(\mathbf{x}, \mathbf{x}_i) + w_0$$

Pick $\mathbf{x}_s \in \mathcal{S}$ where $y_s = 1$ $\longrightarrow$

$$w_0 = 1 - \textstyle\sum_{\mathbf{x}_i \in \mathcal{S}} \alpha_i y_i k(\mathbf{x}_s, \mathbf{x}_i), \text{ where } \mathbf{x}_s \in \mathcal{S}$$

Lagrange Multipliers for Support Vectors



Popular Kernels

Polynomial kernel

$$k(\mathbf{x}_i, \mathbf{x}_j) = (1 + \mathbf{x}_i^T \mathbf{x}_j)^p, \text{ where } p \geq 1$$

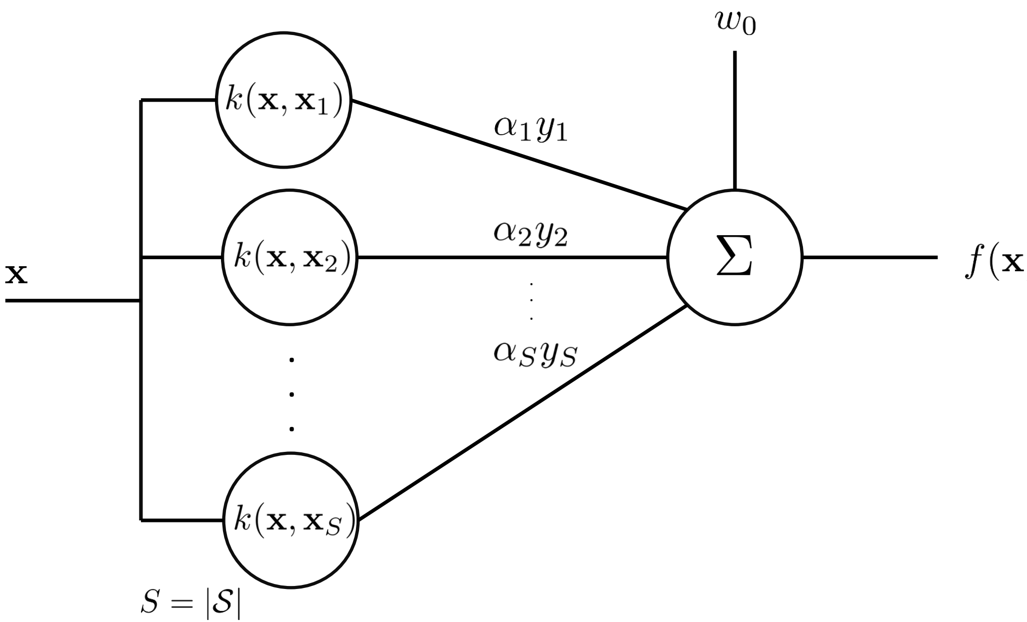
Radial basis function (RBF) kernel

$$k(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\tfrac{1}{2\sigma^2} ||\mathbf{x}_i - \mathbf{x}_j||^2\right), \text{ where } \sigma > 0$$

Hyperbolic tangent function kernel

$$k(\mathbf{x}_i, \mathbf{x}_j) = \tanh\left(\beta_0 \mathbf{x}_i^T \mathbf{x}_j + \beta_1\right), \text{ though not all pairs } (\beta_0, \beta_1) \text{ work}$$

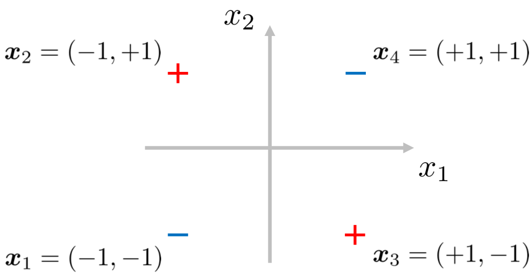
A Support Vector Machine



A support vector machine is a neural network.

Example: XOR

Given: $\mathcal{D} = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^4$ for the XOR concept. **Goal:** train SVM with a quadratic kernel.



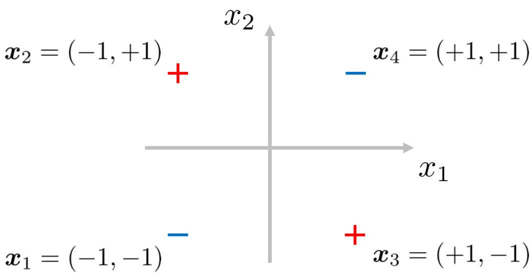
$$\boldsymbol{x}_i = (x_{i1}, x_{i2})$$

$$k(\boldsymbol{x}_i, \boldsymbol{x}_j) = (1 + \mathbf{x}_i^T \mathbf{x}_j)^2$$

Haykin. Neural networks. 1999.

Example: XOR

Given: $\mathcal{D} = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^4$ for the XOR concept. **Goal:** train SVM with a quadratic kernel.



$$\boldsymbol{x}_i = (x_{i1}, x_{i2})$$

$$k(\boldsymbol{x}_i, \boldsymbol{x}_j) = (1 + \mathbf{x}_i^T \mathbf{x}_j)^2$$

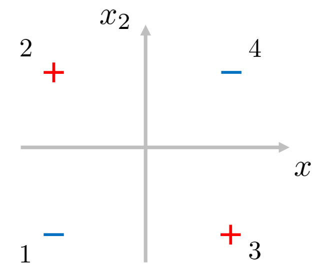
$$k(\boldsymbol{x}_i, \boldsymbol{x}_j) = (1 + x_{i1}x_{j1} + x_{i2}x_{j2})^2$$

Haykin. Neural networks. 1999.

Example: XOR

$$
\begin{aligned}
k(\boldsymbol{x}_i, \boldsymbol{x}_j) &= (1, \sqrt{2}x_{i1}, \sqrt{2}x_{i2}, x_{i1}^2, \sqrt{2}x_{i1}x_{i2}, x_{i2}^2)^T (1, \sqrt{2}x_{j1}, \sqrt{2}x_{j2}, x_{j1}^2, \sqrt{2}x_{j1}x_{j2}, x_{j2}^2) \\
&= \varphi(\boldsymbol{x}_i)^T \varphi(\boldsymbol{x}_j)
\end{aligned}
$$

Kernel matrix $\mathbf{K}$ for the data set $\mathcal{D}$, where $K_{ij} = k(\boldsymbol{x}_i, \boldsymbol{x}_j)$



e.g., $K_{11} = k(\boldsymbol{x}_1, \boldsymbol{x}_1) = (1 + \underset{x_{11}}{(-1)} \cdot \underset{x_{11}}{(-1)} + \underset{x_{12}}{(-1)} \cdot \underset{x_{12}}{(-1)})^2 = 9$

$K_{12} = k(\boldsymbol{x}_1, \boldsymbol{x}_2) = (1 + \underset{x_{11}}{(-1)} \cdot \underset{x_{21}}{(-1)} + \underset{x_{12}}{(-1)} \cdot \underset{x_{22}}{1})^2 = 1$

$$
\mathbf{K} = \begin{bmatrix} 9 & 1 & 1 & 1 \\ 1 & 9 & 1 & 1 \\ 1 & 1 & 9 & 1 \\ 1 & 1 & 1 & 9 \end{bmatrix}
$$

Example: Solving the Dual Problem (Manually)

$$
\begin{aligned}
L(\boldsymbol{w}, \boldsymbol{\alpha}) = \alpha_1 + \alpha_2 + \alpha_3 + \alpha_4 - \tfrac{1}{2}(&9\alpha_1^2 - \alpha_1\alpha_2 - \alpha_1\alpha_3 + \alpha_1\alpha_4 \\
&-\alpha_1\alpha_2 + 9\alpha_2^2 + \alpha_2\alpha_3 - \alpha_2\alpha_4 \\
&-\alpha_1\alpha_3 + \alpha_2\alpha_3 + 9\alpha_3^2 - \alpha_3\alpha_4 \\
&+\alpha_1\alpha_4 - \alpha_2\alpha_4 - \alpha_3\alpha_4 + 9\alpha_4^2)
\end{aligned}
$$

$\frac{\partial L(\boldsymbol{w}, \boldsymbol{\alpha})}{\partial \alpha_i} = 0$

$$
\begin{aligned}
9\alpha_1 - \alpha_2 - \alpha_3 + \alpha_4 &= 1 \\
-\alpha_1 + 9\alpha_2 + \alpha_3 - \alpha_4 &= 1 \\
-\alpha_1 + \alpha_2 + 9\alpha_3 - \alpha_4 &= 1 \\
\alpha_1 - \alpha_2 - \alpha_3 + 9\alpha_4 &= 1
\end{aligned}
\quad \Rightarrow \quad \alpha_1 = \alpha_2 = \alpha_3 = \alpha_4 = \tfrac{1}{8}
$$

All examples are support vectors!

EXAMPLE: SOLUTION TO THE DUAL PROBLEM

$$\begin{aligned}\boldsymbol{w} &= \sum_{i=1}^{4} \alpha_i y_i \varphi(\boldsymbol{x}_i) \\ &= \frac{1}{8}(-\varphi(\boldsymbol{x}_1) + \varphi(\boldsymbol{x}_2) + \varphi(\boldsymbol{x}_3) - \varphi(\boldsymbol{x}_4)) \\ &= \frac{1}{8}(0, 0, 0, 0, -4\sqrt{2}, 0) \\ &= (0, 0, 0, 0, -\frac{1}{\sqrt{2}}, 0)\end{aligned}$$

$$w_0 = 1 - \textstyle\sum_{\boldsymbol{x}_i \in \mathcal{S}} \alpha_i y_i k(\boldsymbol{x}_s, \boldsymbol{x}_i) = 0$$

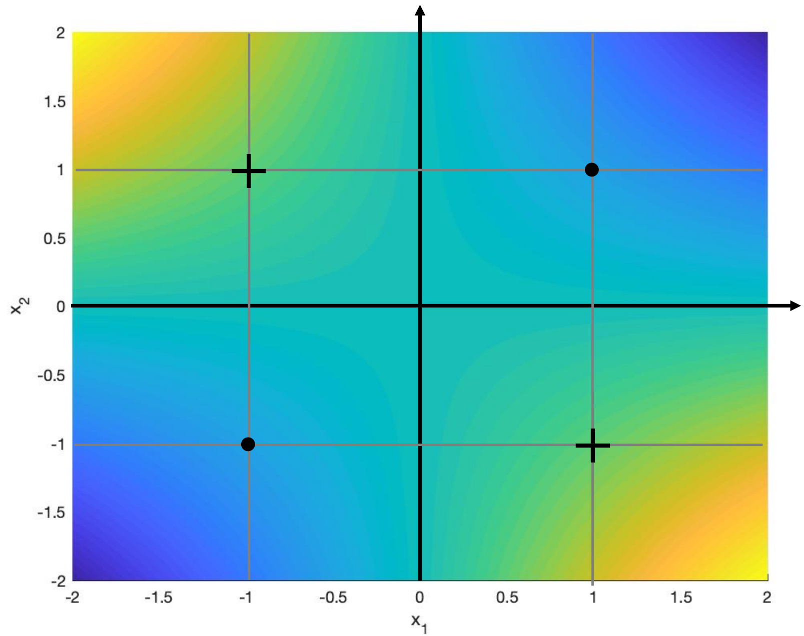
for any $\boldsymbol{x}_s \in \mathcal{S}$ where $y_s = +1$

Example: Visualizing Prediction Scores

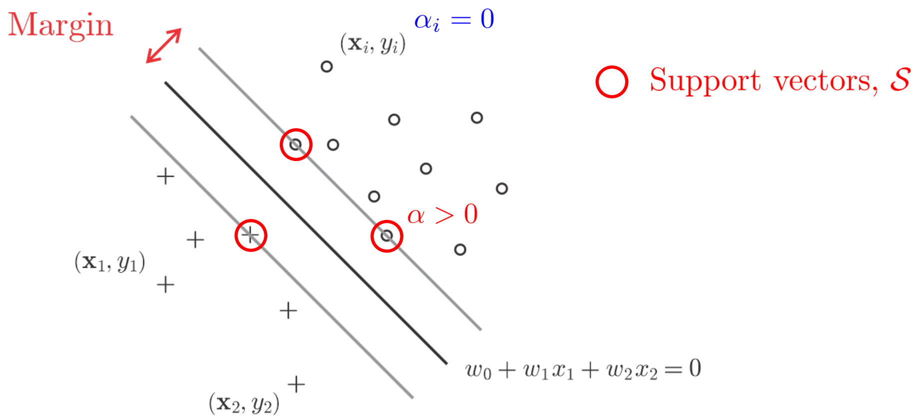
New prediction:

$$f(\boldsymbol{x}) = \sum_{\boldsymbol{x}_s \in \mathcal{S}} \alpha_s y_s k(\boldsymbol{x}_s, \boldsymbol{x}) + w_0$$

$\mathcal{S}$ = support vectors

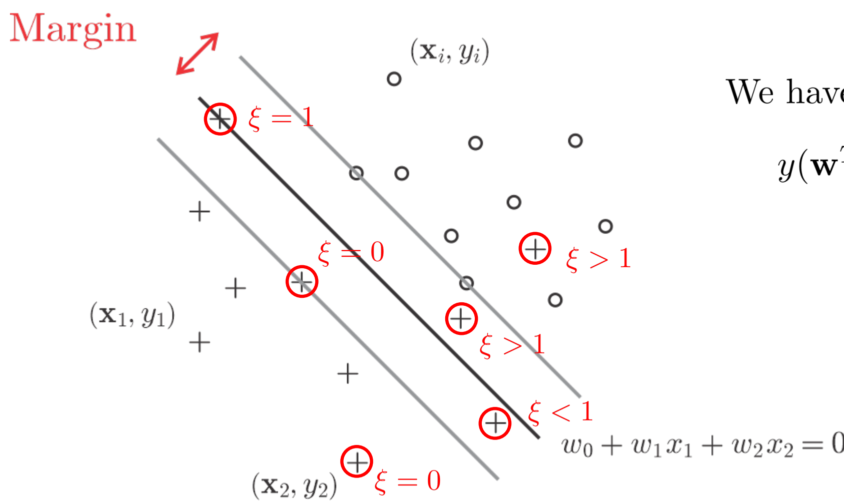


(Hard Margin) Support Vector Machines



Non-Separable Case (Soft Margin)

Introduce “slack” variables $\xi_i \geq 0$, one for each data point $\boldsymbol{x}_i$.



We have:

$$y(\mathbf{w}^T \mathbf{x} + w_0) + \xi \geq 1$$

Non-Separable Case

$\xi_i = 0$: $\mathbf{x}_i$ is on or inside the correct halfspace.

$\xi_i = |y_i - \mathbf{w}^T\mathbf{x}_i - w_0|$: for all other $\mathbf{x}_i$. Examples with $\xi_i > 1$ are misclassified.

New constraints:

old $\qquad$ new

$$y_i(\mathbf{w}^T\mathbf{x}_i + w_0) \geq 1 \quad \rightarrow \quad y_i(\mathbf{w}^T\mathbf{x}_i + w_0) \geq 1 - \xi_i$$

We now minimize:

$C \cdot \sum_{i=1}^{n} I(\xi_i \geq 1)$

$$(\mathbf{w}^*, w_0^*) = \arg\min_{\mathbf{w}, w_0} \left\{ \frac{1}{2}\mathbf{w}^T\mathbf{w} + C \cdot \sum_{i=1}^{n} \xi_i \right\} \qquad C > 0$$

such that $y_i(\mathbf{w}^T\mathbf{x}_i + w_0) \geq 1 - \xi_i$, $\xi_i \geq 0$

Optimization Steps

$$L(\mathbf{w}, w_0, \boldsymbol{\alpha}, \boldsymbol{\xi}, \boldsymbol{\mu}) = \frac{1}{2}\mathbf{w}^T\mathbf{w} + C \cdot \sum_{i=1}^{n} \xi_i - \sum_{i=1}^{n} \alpha_i(y_i(\mathbf{w}^T\mathbf{x}_i + w_0) - 1 + \xi_i) - \sum_{i=1}^{n} \mu_i \xi_i$$

where $\alpha_i \geq 0$, $\mu_i \geq 0$ are Langrange multipliers

KKT conditions are now:

$$\alpha_i \geq 0,\ \mu_i \geq 0,\ \xi_i \geq 0$$
$$y_i(\mathbf{w}^T\mathbf{x}_i + w_0) - 1 + \xi_i \geq 0$$
$$\alpha_i(y_i(\mathbf{w}^T\mathbf{x}_i + w_0) - 1 + \xi_i) = 0$$
$$\mu_i \xi_i = 0$$

Dual Problem

$$L^{\text{dual}}(\mathbf{w}, w_0, \boldsymbol{\alpha}) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j k(\mathbf{x}_i, \mathbf{x}_j)$$

Same as before

Subject to:

$$\alpha_i \geq 0,\ \mu_i \geq 0$$

More constraints

$$0 \leq \alpha_i \leq C \qquad \forall i \in \{1, 2, \ldots, n\}$$

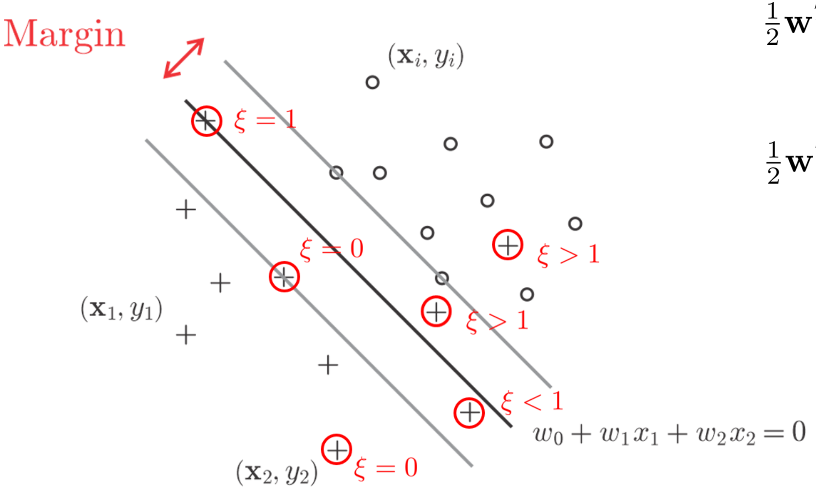
$$\textstyle\sum_{i=1}^{n} \alpha_i y_i = 0$$

Note: $\alpha_i > 0 \quad \rightarrow \quad$ support vector

$\alpha_i < C \quad \rightarrow \quad \mu_i > 0, \xi_i = 0$ points on the margin

$\alpha_i = C \quad \rightarrow \quad \xi_i \leq 1$ (inside margin) or $\xi_i > 1$ (misclassified)

A Different View on Minimization



Objective function to minimize:

$$\frac{1}{2}\mathbf{w}^T\mathbf{w} + C \cdot \sum_{i=1}^{n} \xi_i$$

$$\frac{1}{2}\mathbf{w}^T\mathbf{w} + C \cdot \sum_{i=1}^{n} \underbrace{(1 - y_i(\mathbf{w}^T\mathbf{x}_i + w_0))^+}_{\xi_i}$$

$$x^+ = \max(0, x)$$

A Different View on Minimization

Support vector machine:

$$\sum_{i=1}^{n} \underbrace{(1 - y_i(\mathbf{w}^T \mathbf{x}_i + w_0))^+}_{\xi_i} + \lambda ||\mathbf{w}||^2 \qquad \lambda = \frac{1}{2C}$$

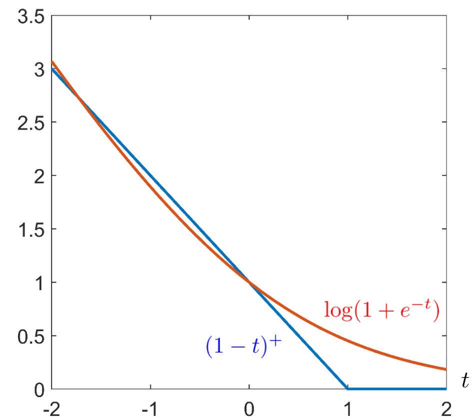
Hinge loss

Logistic regression, regularized:

$$y_i \in \{-1, +1\} \text{ for } \forall i \qquad \Rightarrow \qquad s_i = \frac{1}{1+e^{-y_i(w_0+\mathbf{w}^T\mathbf{x}_i)}}$$

Negative log-likelihood becomes:

$$\sum_{i=1}^{n} \log\left(1 + e^{-y_i(w_0+\mathbf{w}^T\mathbf{x}_i)}\right) + \lambda ||\mathbf{w}||^2$$



*Logistic regression loss is visualized when multiplied by $\frac{1}{\log 2}$

Quadratic Programming (QP)

$$\mathbf{x}^* = \arg\min_{\mathbf{x}} \left\{ \frac{1}{2}\mathbf{x}^T\mathbf{G}\mathbf{x} + \mathbf{x}^T\mathbf{c} \right\}$$

Subject to:

$$\mathbf{a}_i^T\mathbf{x} = \mathbf{b}_i$$

$$\mathbf{a}_j^T\mathbf{x} \geq \mathbf{b}_j$$

Always solvable or shown to be infeasible in finite computation

If $\mathbf{G}$ is positive semi-definite we have a convex QP

If $\mathbf{G}$ is not positive semi-definite we have a multiple minima and stationary points

If $\mathbf{G}$ is positive definite, the optimal solution is also unique

QP falls under the group of problems called linear constraint programming (QP, LP)