SVM + "kernel trick"

$$(**)\quad \boxed{\begin{aligned} &\min_{w} \|w\|^2 + C\sum_{i=1}^{N} s_i \\ &\text{s.t. } w^T\vec{x}_i \geq 1 - s_i, \quad y_i = 1 \\ &\phantom{\text{s.t. }} w^T\vec{x}_i \leq -1 + s_i, \quad y_i = 0 \end{aligned}}$$

Relabel
$$\hat{y}_i = \begin{cases} 1 & y_i = 1 \\ -1 & y_i = 0 \end{cases}$$

Under the hood:

1) Train: solve dual problem of (**)

$$(A)\quad \boxed{\begin{aligned} &\max_{\alpha} \sum_{i=1}^{N}\alpha_i - \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_j\hat{y}_i\hat{y}_j \boxed{\vec{x}_i^T\vec{x}_j} \leftarrow K(x_i, x_j) \\ &\text{s.t. } 0 \leq \alpha_i \leq C \end{aligned}}$$

then recover $\vec{w}$ as

$$\boxed{\vec{w} = \sum_{i=1}^{N}\alpha_i\vec{x}_i}$$

2) Predict: on input $\vec{x}$,

$$h(\vec{x}) = \vec{w}^T\vec{x} = \left(\sum_{i=1}^{N}\alpha_i x_i\right)^T x$$

$$(B)\quad \boxed{h(\vec{x}) = \sum_{i=1}^{N}\alpha_i \boxed{\vec{x}_i^T\vec{x}}} \leftarrow K(x_i, x)$$

Kernel trick

- SVM training/prediction use inputs only through dot products
- What if we redefine dot product?

Kernel trick

$$K(\vec{x},\vec{z}) = \Phi(\vec{x})^T \Phi(\vec{z}), \qquad \Phi : \mathbb{R}^d \to \mathbb{R}^p$$

feature expansion

Any function of this form is a kernel

Why?

- Implement non-linear feature expansion implicitly

$$\begin{array}{l} x \longmapsto \Phi(x) \\ z \longmapsto \Phi(z) \end{array} \iff x^T z \longmapsto K(x,z)$$

May be much faster to compute than $\Phi(x)^T \Phi(z)$

- Non-linear models
- Faster computation
- Infinite dimensional expansions

Example 1:

$$k(\vec{x},\vec{z}) = \exp\left(-\gamma \|x-z\|^2\right)$$

There is an expansion $\Phi : \mathbb{R}^d \to \mathbb{R}^p$ s.t.
$K(\vec{x},\vec{z}) = \Phi(\vec{x})^T \Phi(\vec{z})$, but $p = \infty$. !!

Example 2: polynomial kernel

1) $K(x, z) = (x^T z)^k$

Claim: equivalent to expansion Φ that consists of all products of k original features

Proof: $(k=2, d=2)$

$$\Phi(x_1, x_2) = \begin{bmatrix} x_1^2 \\ x_2^2 \\ \sqrt{2} x_1 x_2 \end{bmatrix} \qquad \Phi(z_1, z_2) = \begin{bmatrix} z_1^2 \\ z_2^2 \\ \sqrt{2} z_1 z_2 \end{bmatrix}$$

$$\begin{aligned} (x^T z)^2 &= (x_1 z_1 + x_2 z_2)^2 \\ &= (x_1 z_1)^2 + 2 x_1 z_1 x_2 z_2 + (x_2 z_2)^2 \\ &= x_1^2 z_1^2 + (\sqrt{2} x_1 x_2)(\sqrt{2} z_1 z_2) + x_2^2 z_2^2 \\ &= \Phi(\vec{x})^T \Phi(\vec{z}) \end{aligned}$$

$\square$

Running time comparison

	$k=2$	general
$K(\vec{x}, \vec{z}) = (\vec{x}^T \vec{z})^2$	$O(d)$	$O(d)$
$K(\vec{x}, \vec{z}) = \Phi(\vec{x})^T \Phi(\vec{z})$	$O(d^2)$	$O(d^k)$

$$\Phi(\vec{x}) = \begin{bmatrix} x_1^2 \\ \vdots \\ x_d^2 \\ \sqrt{2} x_1 x_2 \\ \sqrt{2} x_1 x_3 \\ \vdots \\ \sqrt{2} x_{d-1} x_d \end{bmatrix}$$

More general polynomial kernel

$$K(\vec{x},\vec{z}) = (\vec{x}^T\vec{z} + 1)^k$$

$\equiv$ all products of up to k original features