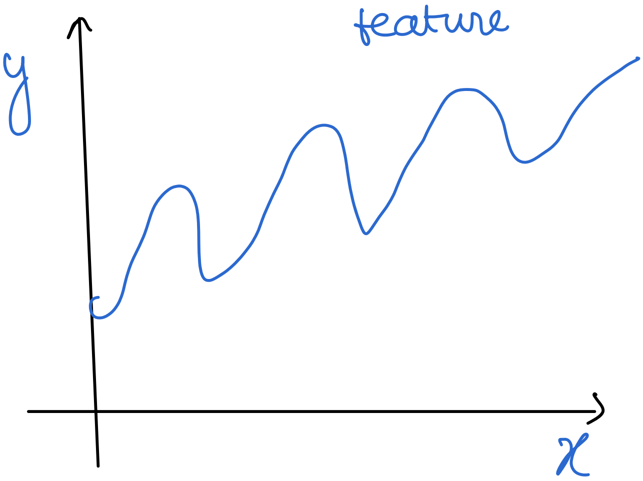
Kernel Methods

$$\underbrace{x}_{\text{attributes}} \rightarrow \phi(x) = \underbrace{\begin{bmatrix} 1 \\ x \\ x^2 \\ \vdots \\ x^4 \end{bmatrix}}_{\text{feature}}$$

1d-x mapped to higher dimensional space.
It can 4, 8 or ∞.



Similar for classification

Normally:

$$\theta^{(t+1)} = \theta^{(t)} + \alpha \sum_{i=1}^{n} \underbrace{\left(y^{(i)} - \theta^T x^{(i)}\right)}_{\text{Scalar}} . x^{(i)} \qquad \theta \in \mathbb{R}^d$$

with feature map

$$\theta^{(t+1)} = \theta^{(t)} + \alpha \sum_{i=1}^{n} \underbrace{\left(y^{(i)} - \theta^T \phi(x^{(i)})\right)}_{\text{Scalar}} \phi(x^{(i)}) \qquad \theta \in \mathbb{R}^p$$

$$\phi : \mathbb{R}^d \rightarrow \mathbb{R}^p$$

$$\phi(x) = \begin{bmatrix} 1 \\ x_1 \\ x_2 \\ \vdots \\ x_1^2 \\ x_1 x_2 \\ x_1 x_3 \\ \vdots \\ x_1^3 \\ x_1^2 x_2 \\ \vdots \end{bmatrix}$$

monomial terms of order ≤ 3

$p \approx O(d^3)$

$d^3 \rightarrow$ features

Normal	$d = 1000$	$O(d)$
Feature Map	$(1000)^3$	$O(d^3)$

Claim:

$$\theta^{(t)} = \sum_{i=1}^{n} \beta_i^{(t)} \phi(x^{(i)})$$

$\rightarrow$ every $\theta^{(t)}$ is linear comb^n of $\phi(x)$

evident from G.D. update rule

$$\theta^{(0)} = 0$$

$$\theta^{(1)} = \sum_{i=1}^{n} \underbrace{\alpha\, y^{(i)}}_{\beta_i^{(1)}} \phi(x^{(i)})$$

$$\theta^{(t+1)} = \theta^{(t)} + \alpha \sum_{i=1}^{n} \left(y^{(i)} - \theta^{(t)T} \phi(x^{(i)})\right) . \phi(x^{(i)})$$

$$= \sum_{i=1}^{n} \beta_i^{(t)} \phi(x^{(i)}) + \alpha \sum_{i=1}^{n} \left(y^{(i)} - \left(\sum_{j=1}^{n} \beta_j^{(t)} \phi(x^{(j)})\right)^T \phi(x^{(i)})\right) . \phi(x^{(i)})$$

$$= \sum_{i=1}^{n} \underbrace{\left[\beta_i^{(t)} + \alpha \sum \left(y^{(i)} - \left(\sum_{j=1}^{n} \beta_j \phi(x^{(j)})\right)^T \phi(x^{(i)})\right)\right]}_{\beta_i^{(t+1)}} \phi(x^{(i)})$$

$i \rightarrow i^{th}$ example

$\phi(x^{(j)})^T \phi(x^{(i)}) \rightarrow$ This can be precomputed as examples remain same.

$$\beta_i^{(t+1)} = \beta_i^{(t)} + \alpha \left(y^{(i)} - \sum_{j=1}^{n} \beta_j^{(t)} \phi(x^{(j)})^T \phi(x^{(i)})\right)$$

Two elements of the space

Kernel $\triangleq K : \mathcal{X} \cdot \mathcal{X} = \mathbb{R}$

$$K(x, z) = \langle \phi(x), \phi(z) \rangle$$
$$= \phi(x)^T \phi(z)$$

$x \in \mathcal{X}$, $x \in \mathbb{R}^d$ $\mathcal{X} = \mathbb{R}^d$

$$\phi(x) = \begin{bmatrix} 1 \\ x_1 \\ x_2 \\ \vdots \\ x_1^3 \\ \vdots \end{bmatrix}$$

$$K(x, z) = 1 + \langle x, z \rangle + \langle x, z \rangle^2 + \langle x, z \rangle^3$$
$$= \phi(x)^T \phi(z)$$

$O(d^3)$ reduced to $O(d)$

$$\phi : \mathcal{X} \to \mathbb{R}^p$$

LINEAR REGRESSION (Kernalized)

<1> Precompute:

$$K_{ij} = K(x^{(i)}, x^{(j)}) = \langle \phi(x^{(i)}), \phi(x^{(j)}) \rangle$$

($n^2 \rightarrow$ inner products)

$K \rightarrow$ stand for both kernel function & kernel matrix where kernel matrix is square symmetric matrix where dot product bet^n all egs. preevaluated.

<2> Loop:

$\forall\, i \in \{1 \ldots n\}$

$$\beta_i^{(t+1)} = \beta_i^{(t)} + \alpha\left(y^{(i)} - \sum_{j=1}^{n} \beta_j^{(t)} K_{ij}\right)$$

$$\beta^{(t+1)} = \beta^{(t)} + \alpha\left(\vec{y} - K\beta^{(t)}\right)$$

$\rightarrow$ vectorized form

$\beta^{(t)} \in \mathbb{R}^n$

Prediction

$$h_\theta(x) = \theta^T \phi(x)$$

$$= \sum_{i=1}^{n} \beta_i \phi(x^{(i)})^T \phi(x)$$

$$= \sum_{i=1}^{n} \beta_i K(x^{(i)}, x)$$

$x \rightarrow$ test example

$x^{(i)} \rightarrow$ all training examples

Observations

① Train: $\beta := \beta + \alpha(\vec{y} - K\beta)$

Test: $\hat{y} = \sum_{i=1}^{n} K(x^{(i)}, x) \cdot \beta_i$

$\phi(x)$ does not appear

② For Prediction

Need training example to be stored in memory

we give up $\phi(x)$

$$\begin{array}{cccc} \beta^{(1)} & \beta^{(0)} & x^{(i)} \in \mathbb{R}^d & \\ \begin{bmatrix} \\ \in \\ \\ \end{bmatrix} & \begin{bmatrix} \vdots \\ \\ \\ \end{bmatrix} & \begin{bmatrix} — x^{(1)T} — \\ \vdots \\ — x^{(n)T} — \end{bmatrix} & \begin{bmatrix} \vec{y} \\ \\ \\ \end{bmatrix} \end{array}$$

$\boxed{\theta \in \mathbb{R}^d}$ $\theta^{(0)}$

$\downarrow$

$\boxed{\quad\quad\quad}$ $\theta^{(1)}$ one comp per feature

$\downarrow$

$\boxed{\quad\quad\quad}$ $\theta^{(2)}$

$$\theta^{(t)} = \sum_{i=1}^{n} \beta_i^{(t)} x^{(i)}$$

$\beta \rightarrow 1$ per example (This is what allows us to scale to ∞ dimensions)

For logistic,

$$\beta^{(t+1)} = \beta^{(t)} + \alpha(\vec{y} - g(K\beta^{(t)}))$$

Kernel examples

$x \in \mathbb{R}^d$

1> $K(x,z) = \langle x, z \rangle^2$

$$\phi(x) = \begin{bmatrix} x_1 x_1 \\ x_1 x_2 \\ \vdots \\ x_d x_d \end{bmatrix}$$

First we come up with $K(x,z)$ then $\phi(x)$

2> $K(x,z) = (x^T z + c)^2$

$$\phi(x) = \begin{bmatrix} x_1^2 \\ x_1 x_2 \\ \vdots \\ \sqrt{2c}\, x_1 \\ \sqrt{2c}\, x_2 \end{bmatrix}$$

$$K(x,z) = \phi(x)^T \phi(z)$$

$K(x,z) \uparrow$ for similar (x,z)

$\downarrow$ for dissimilar (x,z)

$$K(x,z) = \phi(x)^T \phi(z)$$

$$K(x,z) = \exp\left(\frac{-\|x-z\|^2}{2\sigma^2}\right)$$ → ∞ dimensional kernel vector

↓

Popular kernel !!
Gaussian kernel

Necessary condⁿ for k to be a kernel

(1) k should symmetric

$$K(x,z) = K(z,x)$$

(2) k is P.S.D.

$$z^T K z = \sum_i \sum_j z_i k_{ij} z_j$$

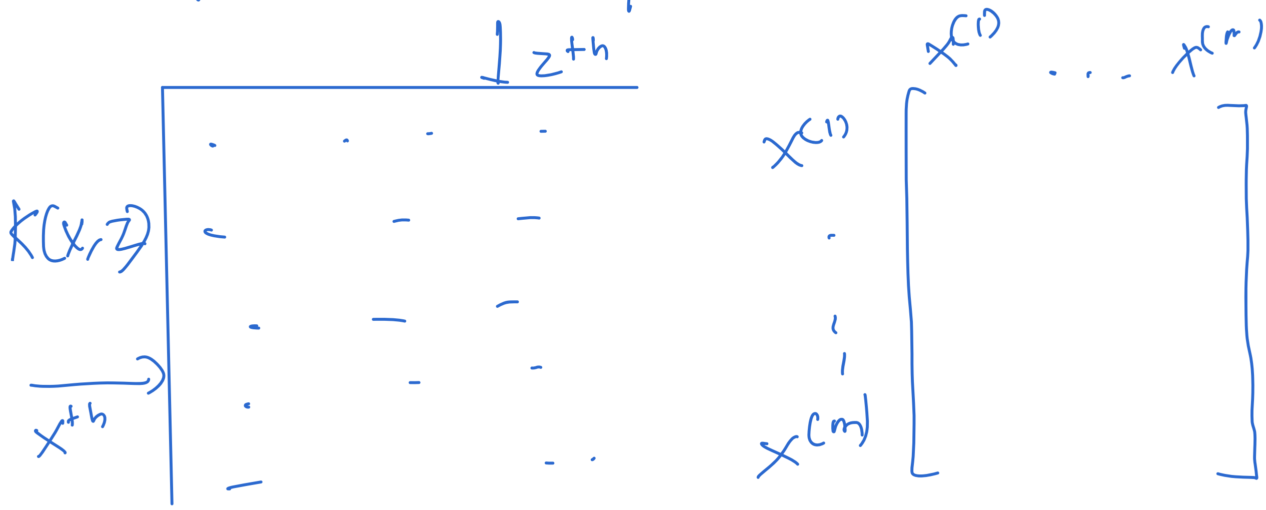
$$= \sum_k \left(\sum_i z_i \phi_k(x^{(i)}) \right)^2 \geq 0$$

Mercer Theorem

Let $K: \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ be a given function
For k to be kernel, it is necessary & sufficient
for any $\{x^{(1)} \ldots x^{(m)}\}$ the corresponding
matrix $K_{ij} = K(x^{(i)}, x^{(j)})$ is P.S.D.

vector $\longleftrightarrow$ functions

Matrices $\longleftrightarrow$ operator



① Construct ϕ

$$K(\cdot) = \phi^T \phi$$

O-R-

② Mercer's Theorem

$$\{x^{(1)} \dots x^{(m)}\}$$

$$K_{ij} = K(x^{(i)}, x^{(j)})$$

is P.S.D.

③ $\forall f$ $\iint f(x) K(x, x') f(x') \, dx \, dx' \geq 0$

SVM

$y^{(i)} \in \{+1, -$

Parameters $\rightarrow w, b$

$w \in \mathbb{R}^d \quad b \in \mathbb{R}$

$x \in \mathbb{R}^d$

Functional Margin

Margin: $y^{(i)}(w^T x^{(i)} + b) > 0 \rightarrow$ always

$w^T x + b > 0$ for $y = +1$

< 0 for $y = -1$

Desire: Margin large

Calculate smallest margin & choose the hyperplane which maximizes the smallest margin.

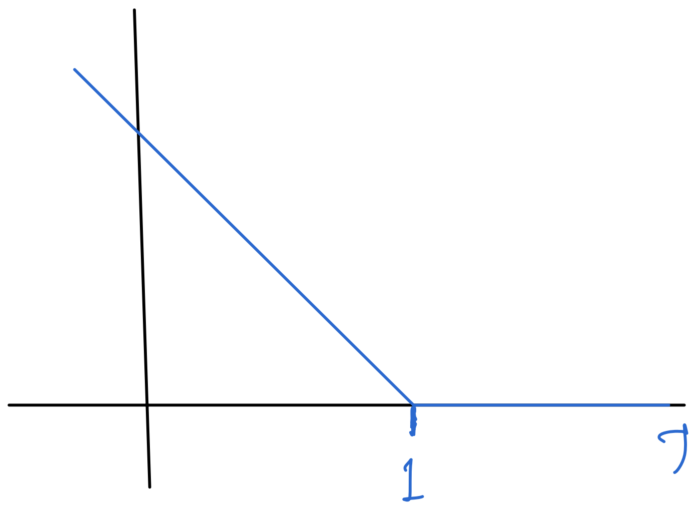
$w \rightarrow 2w$
$b \rightarrow 2b$ Then margin doubles

So we can game the system,

so use geometric margin

Margin $= \gamma^{(i)}$

$$\min_{w,b} \left(\sum_{i=1}^{n} \underbrace{\max\left[0,\ 1-y^{(i)}(w^T x^{(i)}+b)\right]}_{\text{Hinge loss/ SVM loss}} \right) + \underbrace{\frac{1}{C}\|w\|^2}_{\text{Penalizing}}$$



$$\min_{\xi, w, b} \quad \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n} \xi_i$$

$$\text{St.} \quad y^{(i)}(w^T x^{(i)}+b) \geq 1-\xi_i \quad \forall i \in \{1..n\}$$

$$\xi_i \geq 0, \quad i=1..n$$

Primal Convex Problem

Dual Convex Problem

$$\max = \sum_{i=1} \alpha_i - \frac{1}{2}\sum_{i,j} y^{(i)} y^{(j)} \alpha_i \alpha_j \langle x^{(i)}, x^{(j)} \rangle$$

$$0 \leq \alpha_i \leq C$$

$$\sum_{i=1}^{n} \alpha_i y^{(i)} = 0$$

We donot penalize b because then we donot give our algorithm to be close/ to the origin

Most α_i's will be 0, very few non-zero those set of examples → non zero → support vector, closest to margin

BAYESIAN METHODS