

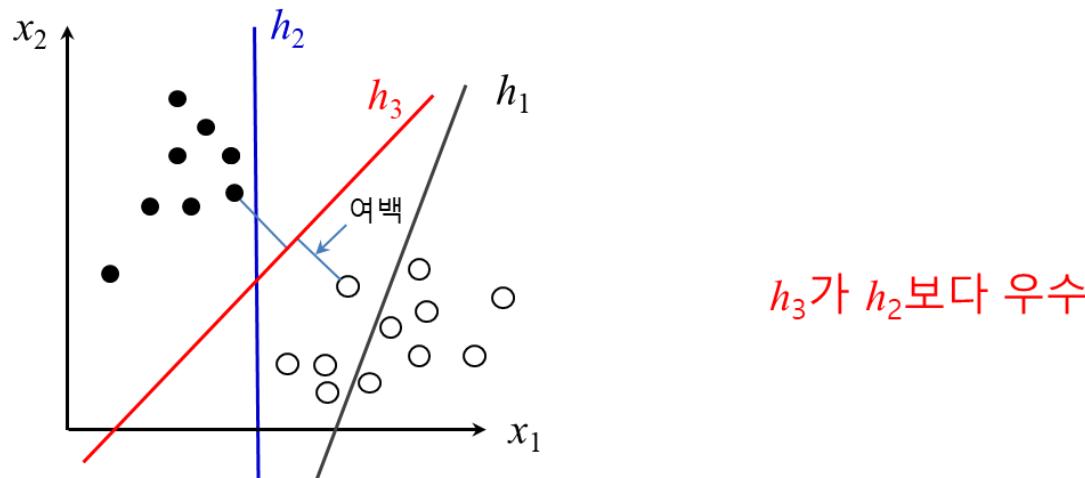
인공지능개론

기계학습

Support vector machines

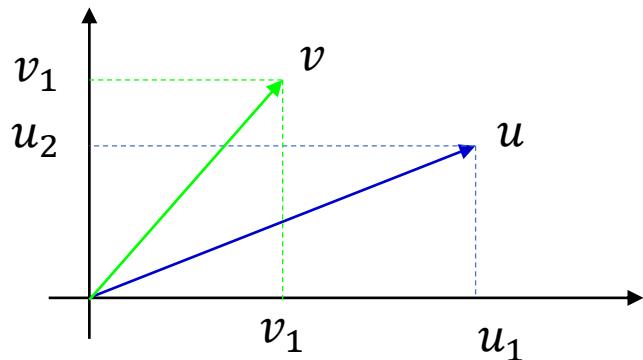
❖ Support Vector Machine (SVM)

- Vladimir Vapnik이 제안
- 분류 오차를 줄이면서 동시에 **여백**을 최대로 하는 결정 경계(decision boundary)를 찾는 **이진 분류기**(binary classifier)



- **여백**(margin)
 - 결정 경계와 가장 가까이에 있는 학습 데이터까지의 거리
- **서포트 벡터**(support vector)
 - 결정 경계로부터 가장 가까이에 있는 학습 데이터들

Vector inner product

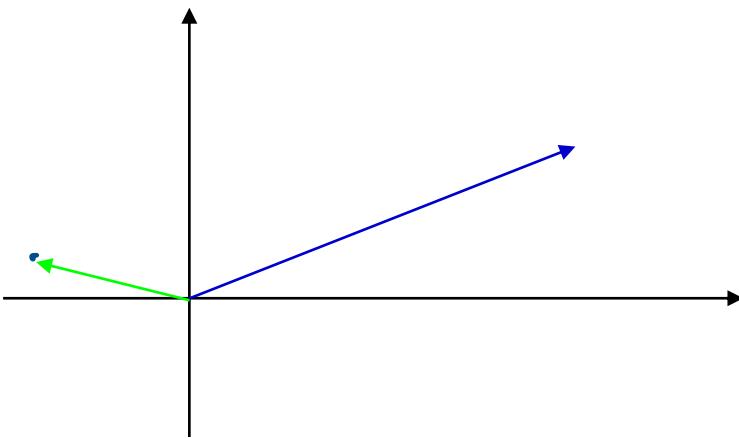


$$u = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}, \quad v = \begin{bmatrix} v_1 \\ v_2 \end{bmatrix}$$

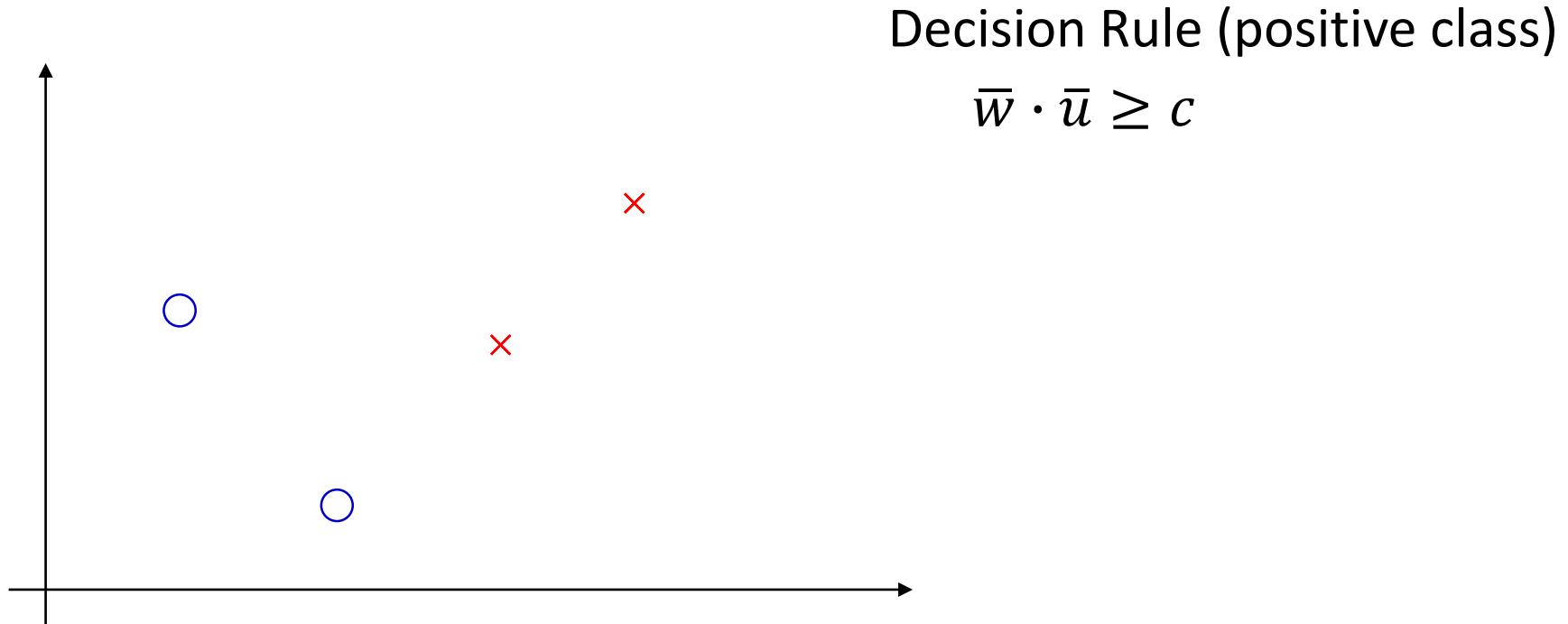
1. $u^T v$
2. $\|u\|$
3. Projection p

$$u^T v = p \cdot \|u\|$$

$$4. \frac{u}{\|u\|}$$



What's our goal?



Mathematical expression

$$\bar{w} \cdot \bar{x}_+ + b \geq 1$$

y_i such that $y_i = +1$ for positive class

$$\bar{w} \cdot \bar{x}_- + b \leq -1$$

$= -1$ for negative class



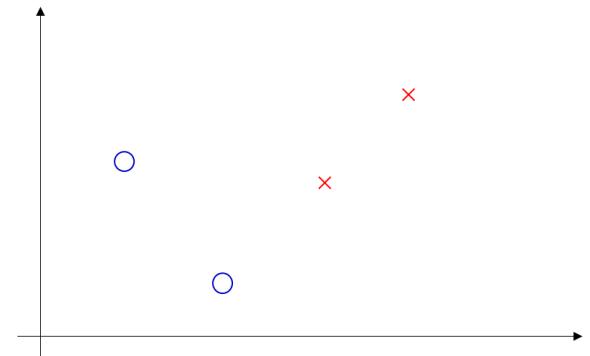
$$y_i(\bar{w} \cdot \bar{x}_+ + b) \geq 1$$

$$y_i(\bar{w} \cdot \bar{x}_- + b) \geq 1$$

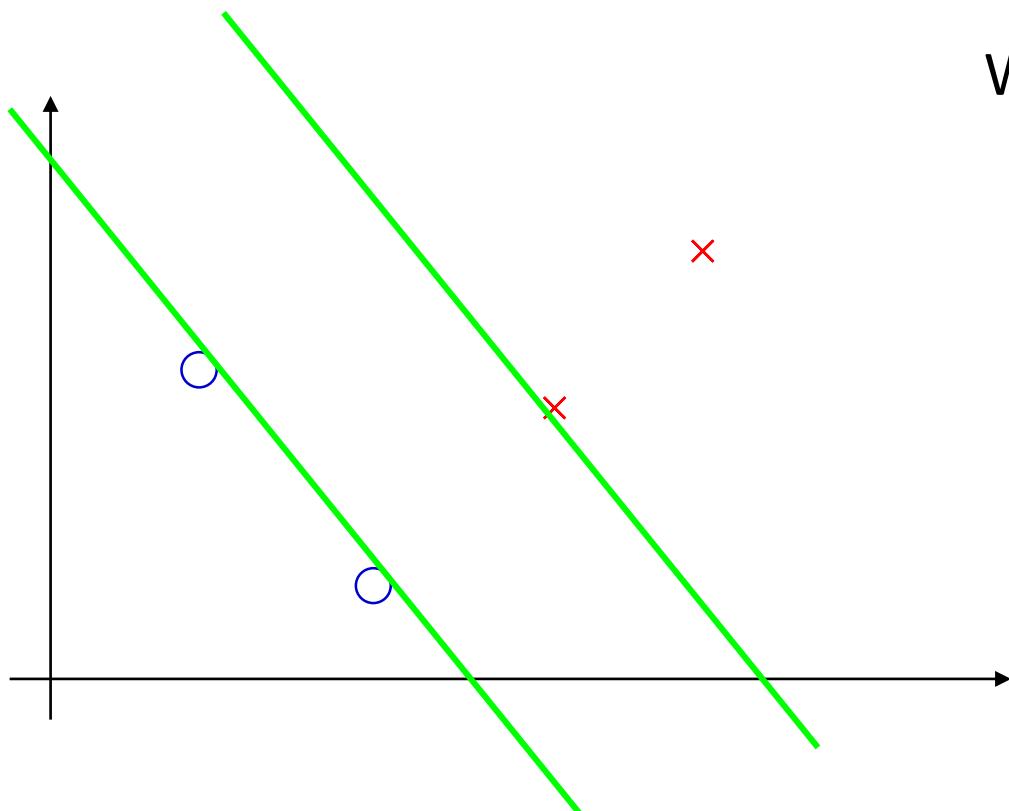


$$y_i(\bar{w} \cdot \bar{x}_i + b) - 1 \geq 0$$

$$y_i(\bar{w} \cdot \bar{x}_i + b) - 1 = 0$$



Optimal problem



$$\text{Width} = (\bar{x}_+ - \bar{x}_-) \cdot \frac{\bar{w}}{\|w\|}$$

We want maximize the width $\rightarrow \frac{1}{2} \|w\|^2$

Lagrangian function

$$L(w, b, \alpha) = \frac{1}{2} \|w\|^2 - \sum \alpha_i [y_i(\bar{w} \cdot \bar{x}_i + b) - 1]$$

$$\frac{\partial L}{\partial \bar{w}}$$

$$\frac{\partial L}{\partial b}$$

What is Lagrangian?

$$L(x, \lambda) = f(x) - \lambda g(x)$$

Inner product in SVM

Lagrangian

$$L = \frac{1}{2} \left(\sum \alpha_i y_i x_i \right) \cdot \left(\sum \alpha_i y_i x_i \right) - \left(\sum \alpha_i y_i x_i \right) \cdot \left(\sum \alpha_i y_i x_i \right) - \sum \alpha_i y_i b + \sum \alpha_i$$

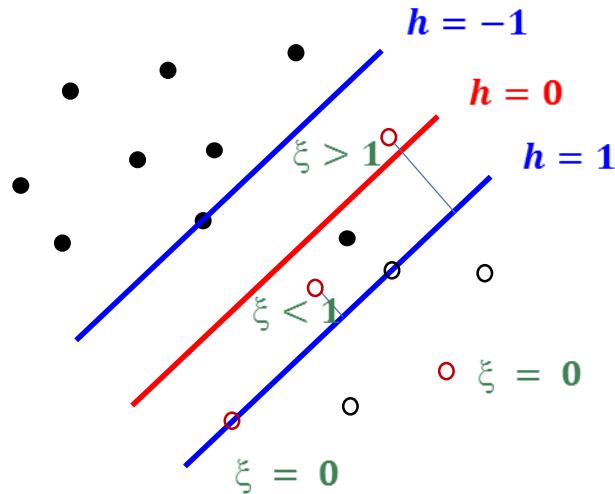
$$L = \sum \alpha_i - \frac{1}{2} \sum \sum \alpha_i \alpha_j y_i y_j x_i x_j$$

Decision Rule

$$\sum \alpha_i y_i x_i \cdot \bar{u} + b \geq 0$$

SVM using slack variable

❖ 선형 분리불가 문제의 SVM



- 슬랙변수(slack variable) ξ_i
 - 학습 데이터별로 하나씩 생성
 - SVM의 분할 초평면에서 서포트벡터보다 멀리 위치하면, $\xi_i = 0$
 - 이외의 경우, $\xi_i = |t_i - h(x_i)|$

$$t_i h(x_i) \geq 1 - \xi_i, \quad i = 1, \dots, N$$

SVM using slack variable

❖ 선형분리가 되지 않는 데이터에 대한 SVM

▪ 최적화 문제

- 슬랙 변수를 허용하는 제약조건

$$t_i h(\mathbf{x}_i) \geq 1 - \xi_i, \quad i = 1, \dots, N$$

- 슬랙 변수 값의 합을 최소화

Find \mathbf{w}, b which minimizes $J(\mathbf{w}) = \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^N \xi_i$

subject to $t_i h(\mathbf{x}_i) \geq 1 - \xi_i, \quad i = 1, \dots, N$

$\xi_i \geq 0, \quad i = 1, \dots, N$

$C > 0$

- 라그랑주 함수

$$L(\mathbf{w}, b, \xi, \alpha, \beta) = \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^N \xi_i + \sum_{i=1}^N \alpha_i (1 - t_i h(\mathbf{x}_i) - \xi_i) - \sum_{i=1}^N \beta_i \xi_i$$

SVM using slack variable

❖ 선형분리가 되지 않는 데이터에 대한 SVM

- 최적화 문제 - cont.
 - 라그랑주 함수

$$L(\mathbf{w}, b, \xi, \alpha, \beta) = \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^N \xi_i + \sum_{i=1}^N \alpha_i (1 - t_i h(\mathbf{x}_i) - \xi_i) - \sum_{i=1}^N \beta_i \xi_i$$

- 쌍대 문제(dual problem)

$$\tilde{L}(\alpha) = \sum_{i=1}^N \alpha_i - \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j t_i t_j \mathbf{x}_i \cdot \mathbf{x}_j$$

$$\sum_{i=1}^N \alpha_i t_i = 0$$

$$0 \leq \alpha_i \leq C \quad \text{상한(upper bound)에 대한 제약조건 추가}$$

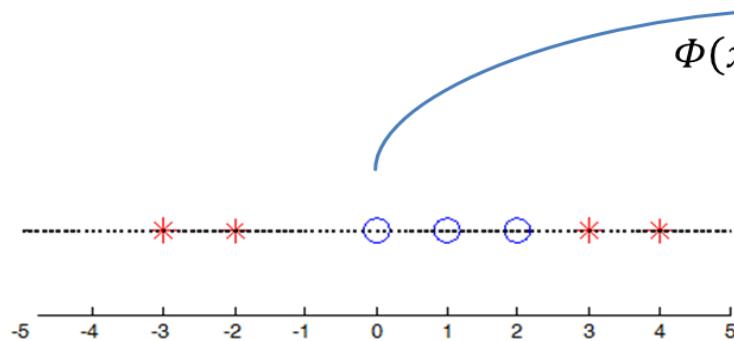
Kernel function

❖ 선형 SVM (linear SVM)

- 선형인 초평면으로 공간 분할
- 슬랙변수를 도입하더라도 한계

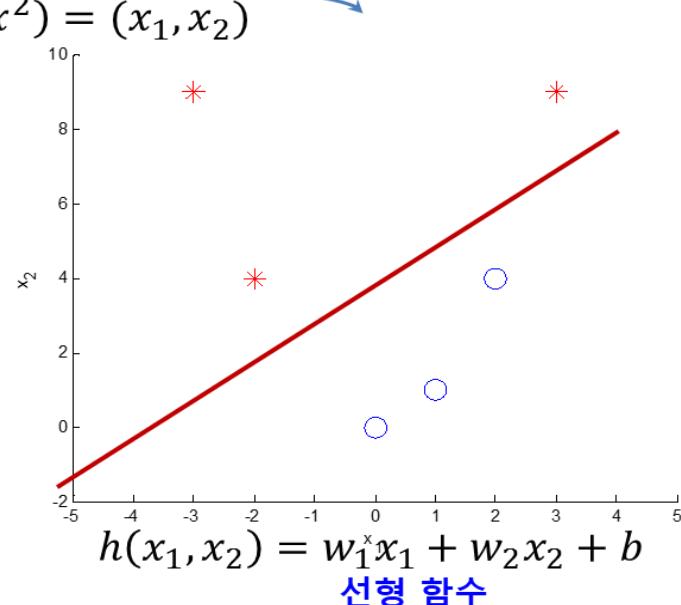
❖ 데이터의 고차원 사상 (high-dimensional mapping)

- 데이터를 고차원의 사상하면 선형 분리 가능



$$h(x_1, x_2) = w_1 x + w_2 x^2 + b$$

비선형 함수



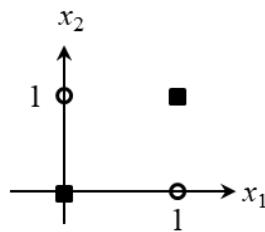
$$h(x_1, x_2) = w_1 x_1 + w_2 x_2 + b$$

선형 함수

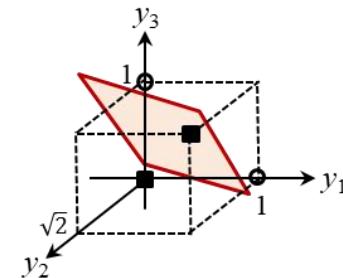
Kernel function

❖ 데이터의 고차원 사상 – cont.

- XOR 문제



$$\Phi(\mathbf{x}) = (x_1^2, \sqrt{2}x_1x_2, x_2^2) = (y_1, y_2, y_3)$$



❖ 고차원 변환의 문제점

- 차원의 저주(curse of dimensionality) 문제 발생
- 테스트 데이터에 대한 일반화(generalization) 능력 저하 가능
 - 여백(margin) 최대화를 통해 일반화 능력 유지
- 계산 비용 증가
 - 커널 트릭(kernel trick) 사용으로 해결

Kernel function

❖ SVM의 최적화 문제

▪ 선형 SVM

Find α which minimizes $\tilde{L}(\alpha) = \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j t_i t_j \boxed{\mathbf{x}_i \cdot \mathbf{x}_j} - \sum_{i=1}^N \alpha_i$

subject to $\sum_{i=1}^N \alpha_i t_i = 0, \quad i = 1, \dots, N$

$\alpha_i \geq 0, \quad i = 1, \dots, N$

$$h(\mathbf{x}) = \sum_{i=0}^N \alpha_i t_i \boxed{\mathbf{x}_i \cdot \mathbf{x}} + b$$

▪ 데이터의 고차원 변환

$$\mathbf{x}_i \rightarrow \Phi(\mathbf{x}_i)$$

▪ 비선형 SVM

$$\tilde{L}(\alpha) = -\frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j t_i t_j \boxed{\Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{x}_j)} + \sum_{i=1}^N \alpha_i$$

$$h(\mathbf{x}) = \sum_{i=1}^N \alpha_i \boxed{\Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{x})} + b$$

Kernel function

❖ 커널 트릭(kernel trick)

- $\Phi(x_i) \cdot \Phi(x_j)$ 를 고차원으로 변환하여 계산하지 않고, 원래 데이터에서 계산
- 고차원 변환없이 계산할 수 있는 커널 함수(kernel function) K 사용
 - $K(x_i, x_j) = \Phi(x_i) \cdot \Phi(x_j)$

❖ 커널 함수의 예

- $K(x, y) = (x^T y)^2$
 - $x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$ $y = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}$
 - $(x^T y)^2 = (x_1 y_1 + x_2 y_2)^2 = (x_1 y_1)^2 + 2(x_1 y_1)(x_2 y_2) + (x_2 y_2)^2$
 $= \begin{bmatrix} (x_1)^2 & \sqrt{2}x_1 x_2 & (x_2)^2 \end{bmatrix} \begin{bmatrix} (y_1)^2 & \sqrt{2}y_1 y_2 & (y_2)^2 \end{bmatrix}^T$
 $= \Phi(x) \cdot \Phi(y)$
 - $\Phi(x) = \begin{bmatrix} (x_1)^2 \\ \sqrt{2}x_1 x_2 \\ (x_2)^2 \end{bmatrix}$ $\Phi(y) = \begin{bmatrix} (y_1)^2 \\ \sqrt{2}y_1 y_2 \\ (y_2)^2 \end{bmatrix}$

Kernel function

- ❖ 대표적인 커널 함수(kernel function)

- 다항식 커널(polynomial kernel)

$$K(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i \cdot \mathbf{x}_j + 1)^p, \quad p \text{는 양의 정수}$$

- RBF(radial basis function) 커널

$$K(\mathbf{x}_i, \mathbf{x}_j) = e^{-\|\mathbf{x}_i - \mathbf{x}_j\|^2 / (2\sigma^2)}$$

- 쌍곡 탄젠트(hyperbolic tangent) 커널

$$K(\mathbf{x}_i, \mathbf{x}_j) = \tanh(\alpha \mathbf{x}_i \cdot \mathbf{x}_j + \beta)$$

Kernel function

❖ 커널 트릭을 사용할 때의 최적화 문제

$$\tilde{L}(\alpha) = -\frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j t_i t_j \boxed{\Phi(x_i) \cdot \Phi(x_j)} + \sum_{i=1}^N \alpha_i$$

$$h(x) = \sum_{i=1}^N a_i t_i \boxed{\Phi(x_i) \cdot \Phi(x)} + b$$

▪ 커널 함수 적용

$$\tilde{L}(\alpha) = -\frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j t_i t_j \boxed{K(x_i, x_j)} + \sum_{i=1}^N \alpha_i$$

$$h(x) = \sum_{i=1}^N a_i t_i \boxed{K(x_i, x)} + b$$