Machine Learning

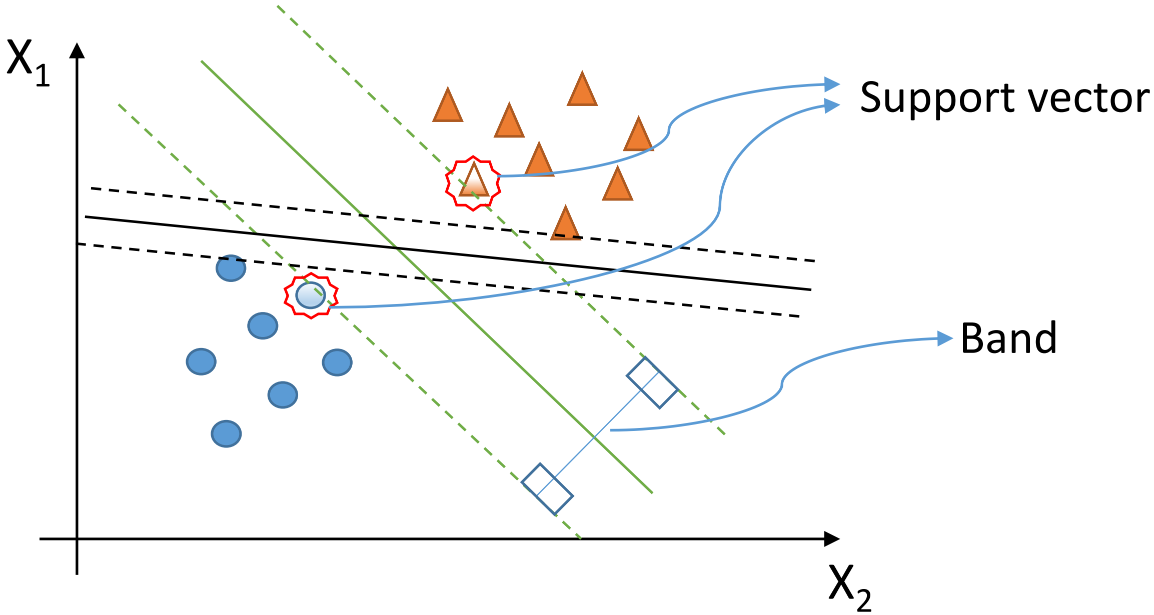
7장 Support vector machine (SVM)

고려대학교 통계학과
박유성

Contents

01 선형 SVM

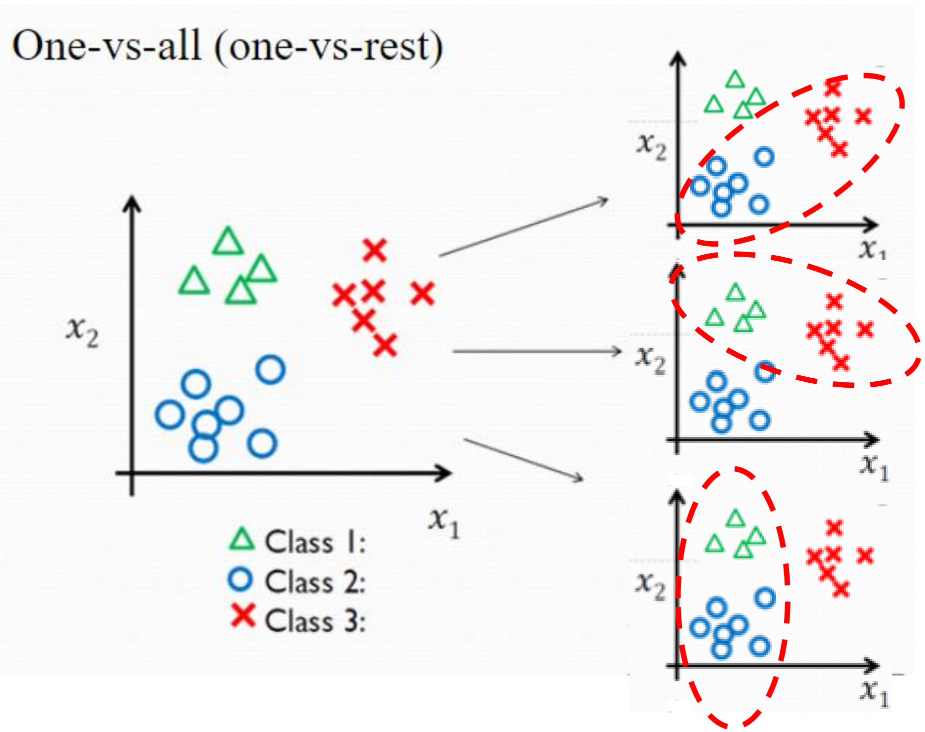
- X_1과 X_2에 따라 2개의 그룹으로 분류.

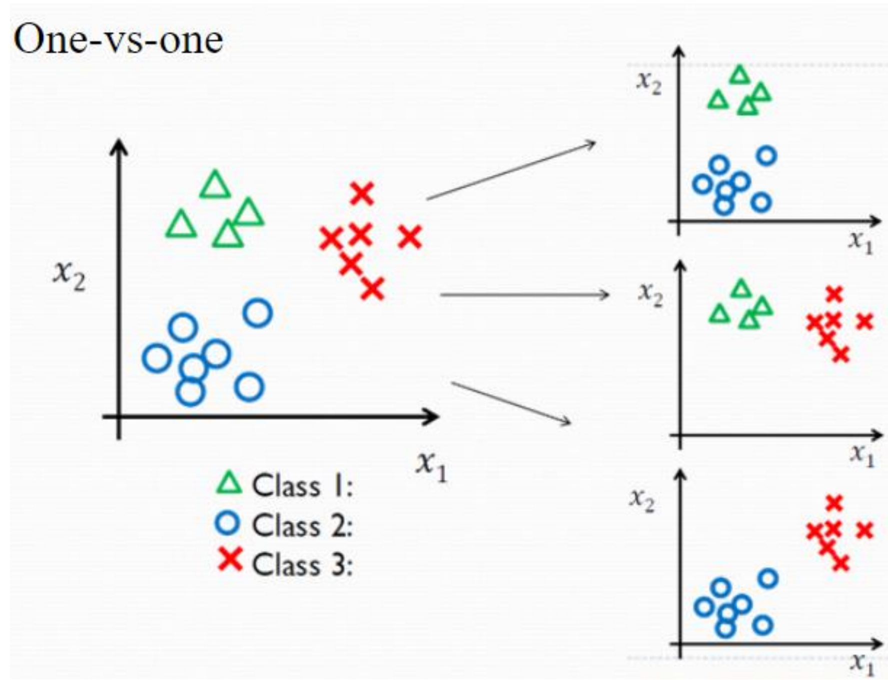


- 자료를 2개의 그룹으로 나누는 수많은 직선이 존재.
- 직선으로부터 점선까지의 거리(밴드)가 가장 큰 것이 합리적 분류선.
- 밴드를 구성하는 2개의 점선위의 자료(밴드를 만든 자료)→ 써포트 벡터.

2개 이상의 그룹이 있는 SVM

- 하나-나머지 방법(One-vs-Rest) 또는 하나-하나 방법(One-vs-One)





- 하나-나머지 방법은 이항 분류값이 가장 큰 값을 갖는 그룹으로 할당.

- 하나-하나 방법은 주어진 특성자료에 대해 가장 많이 할당된 그룹으로 할당.

  (투표방식): 파이썬 SVM의 그룹할당 방식.

Property of SVM (1)

- 두개의 그룹 y={-1,1} 이고, 직선 $f(X) = \beta_0 + \boldsymbol{\beta}^T \boldsymbol{X} = 0$ 일 때, 모든 관측치에 대해

$$\beta_0 + \boldsymbol{\beta}^T \boldsymbol{X}_i \geq 1 \qquad \text{만약 } y_i = 1$$

$$\beta_0 + \boldsymbol{\beta}^T \boldsymbol{X}_i \leq -1 \qquad \text{만약 } y_i = -1 \qquad (7.1)$$

- 위 식을 만족하는 β_0와 $\boldsymbol{\beta}$ 를 구함.

- Band의 상위 경계값=1 일때 특성 변수값 $\boldsymbol{X}_+$ 는 $\beta_0 + \boldsymbol{\beta}^T \boldsymbol{X}_+ = 1$ 을 만족.

- Band의 상위 경계값=-1 일때 특성 변수값 $\boldsymbol{X}_-$ 는 $\beta_0 + \boldsymbol{\beta}^T \boldsymbol{X}_- = -1$ 을 만족.

- 따라서 두 직선간의 거리는

$$\boldsymbol{\beta}^T (\boldsymbol{X}_+ - \boldsymbol{X}_-) = 2 \qquad (7.2)$$

- 두 그룹을 완전하게 구분하는 모든 직선은 (7.2)를 만족.→ 표준화 필요.

Property of SVM (2)

- 표준화 : $\| \beta \| = \sqrt{\sum_{j=1}^{d} \beta_j^2}$ 일때,

$$\frac{\beta^T}{\| \beta \|}(X_+ - X_-) = \frac{2}{\| \beta \|} \tag{7.3}$$

- 식의 좌측은 표준화된 밴드(Band)의 넓이(width).

- 넓이를 최대화 하는 것이 SVM의 목적 ➔ $\| \beta \|$ 를 최소화 하는 문제.

$y_i(\beta_0 + \beta^T x_i) \geq 1$ 인 조건하에서 $\| \beta \|$을 최소화 하는 β (7.4)

- 경마진분류(Hard-margin classification)라 함.
- 그러나 실제 자료에서는 두 그룹이 완전하게 구분되는 학습데이터 없음.
- 따라서 완화변수(Slack variable)도입 필요.

Soft-margin classification

- 완화변수 : $\zeta_i \geq 0,\ i = 1, 2, \cdots, n$ 을 도입하여 (7.4)를 변형.

  $y_i(\beta_0 + \boldsymbol{\beta}^T \boldsymbol{x}_i) \geq 1 - \zeta_i$ 인 조건하에서 $\|\boldsymbol{\beta}\|$을 최소화 하는 $\boldsymbol{\beta}$ (7.5)

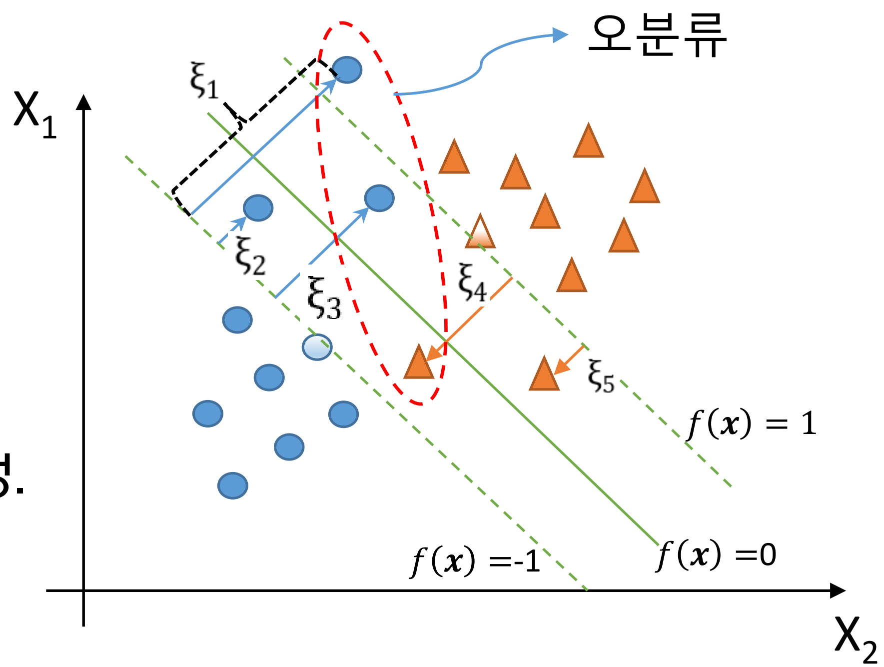
- 유연마진분류(Soft-margin classification)라 함.

- ζ_i 는 일종의 오차로 해석가능.

- 추정 문제는...

  →오차를 어느정도 허용한 상태

  $\|\boldsymbol{\beta}\|$ 을 최소화 하는 $\boldsymbol{\beta}$ 를 추정.



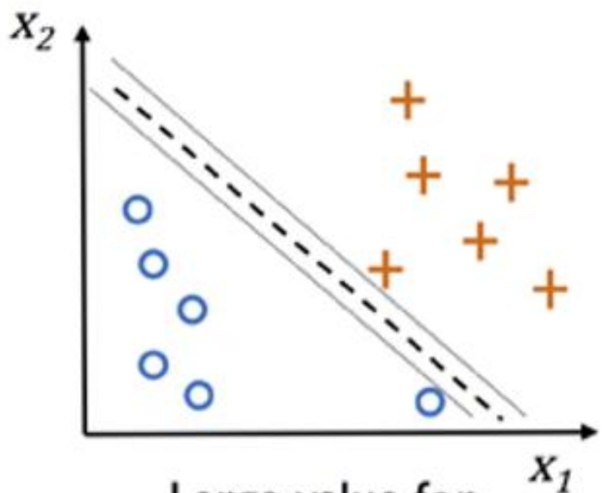
Hyperparameter

- 즉, $\zeta_i \geq 0$이고 C는 초모수($hyperparameter$)라 할때,

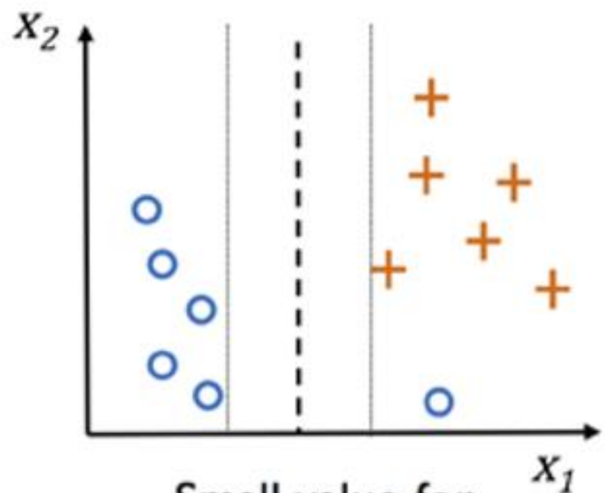
  $y_i(\beta_0 + \boldsymbol{\beta}^T \boldsymbol{x}_i) \geq 1 - \zeta_i$ 인 조건하에서 $\|\boldsymbol{\beta}\| + c\sum_{i=1}^{n}\zeta_i$ 을 최소화 하는 $\boldsymbol{\beta}$ (7.6)

- C가 크면 목적함수 커지고, 밴드의 넓이 좁아짐. ➔ 과대적합(Overfitting)

- C가 작으면 목적함수 작아지고, 밴드의 넓이 넓어짐. ➔ 편이발생(Bias)



Large value for parameter C



Small value for parameter C

- *C의 결정➔ Ch.09 : Cross validation에서 다룸.*

β_0와 $\boldsymbol{\beta}$의 추정 (1)

- 손실함수 : 조건이 있을 때의 최적화 방법이용.(라그랑지승수법–Ch.??)

$$L_p = \frac{1}{2} \parallel \boldsymbol{\beta} \parallel^2 + c\sum_{i=1}^{n} \zeta_i - \sum_{i=1}^{n} \alpha_i [y_i(\beta_0 + \boldsymbol{\beta}^T \boldsymbol{x}_i) - (1-\zeta_i)] - \sum_{i=1}^{n} \mu_i \zeta_i \qquad (7.7)$$

- 손실 함수를 $\boldsymbol{\beta}$, β_0, ζ_i에 대해 각각 편미분후 0으로 놓으면,

$$\boldsymbol{\beta} = \sum_{i=1}^{n} \alpha_i y_i \boldsymbol{x}_i \,,\; 0 = \sum_{i=1}^{n} \alpha_i y_i \,,\; \text{그리고} \;\; \alpha_i = c - \mu_i \qquad (7.8)$$

- 식(7.8)을 식(7.7)에 대입하면,

$$L_p = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j \boldsymbol{x}_i^T \boldsymbol{x}_j \qquad (7.9)$$

- 이때 Karush–Kuhn–Tucker 조건이 필요. 즉, 모든 $i = 1, 2, \cdots, n$에 대해

$$\alpha_i [y_i(\beta_0 + \boldsymbol{\beta}^T \boldsymbol{x}_i) - (1-\zeta_i)] = 0, \;\; \mu_i \zeta_i = 0, \;\; y_i(\beta_0 + \boldsymbol{\beta}^T \boldsymbol{x}_i) \geq (1-\zeta_i) \qquad (7.10)$$

를 충족 해야 함.

-계속

β_0와 β의 추정 (1)

- $\zeta_i > 0$ 인 경우

  (7.10)의 조건에 의해 $\mu_i = 0$가 되며 (7.8)식에 의해 $\alpha_i = c$ 가 됨.

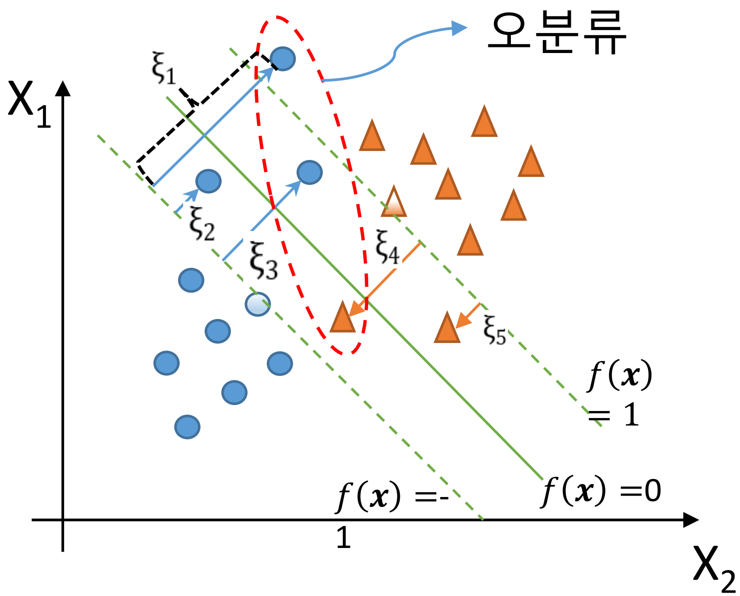
  → $\beta = \sum_{i=1}^{n} \alpha_i y_i \boldsymbol{x}_i$ 이므로 β 추정에 기여함.

- $\zeta_i = 0$ 인 경우

  (7.8)식에 의해 $0 \le \alpha_i \le c$ 가 됨.

  → $\alpha_i = 0$ 인 경우 β 추정에 기여 안함.

- 즉, 밴드 위에 위치한 자료, 밴드 안의 자료 그리고 잘못 분류 된 자료들이 β의 추정에 기여함. 이를 서포트 벡터라 함.



β_0와 β의 추정 (2)

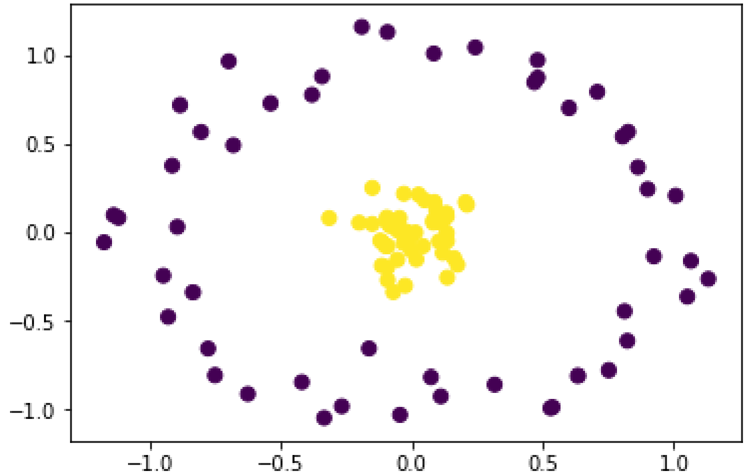
- β_0의 추정 $\hat{\beta}$ 추정 후 SVM의 마진선상 값($\beta_0+\beta^T x=1$ 인 x)을 대입.
- 안정적 추정을 위해 Support Vector 모든 값 대입 후 구한 β_0의 평균사용.
- 식(7.10) 조건하에서 식(7.8)의 해 ➔ Convex quadratic programming.

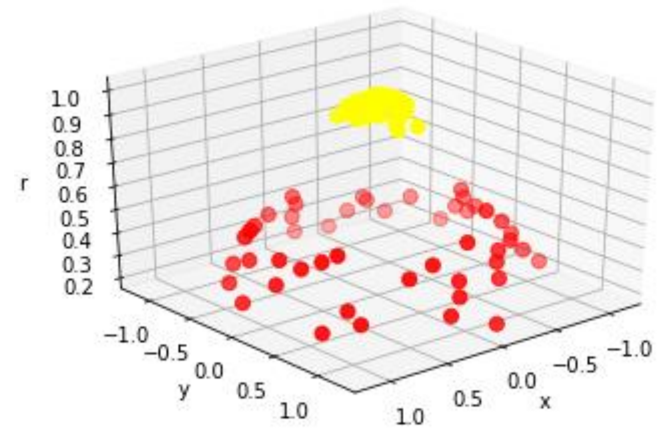
  ➔ Murray el.al(1981) 알고리즘사용.

- 결국, β_0 와 $\hat{\beta}$ 을 구하면 $\hat{f}(x)=\hat{\beta}_0+\hat{\beta}_1{}^T x$ 도출.
- 이때, $\hat{f}(x)$가 양이면 $y=1$, 음이면 $y=-1$ 로 분류 가능.

02 Kernel SVM

- 비선형 분류모형.





- 좌측의 평면에 표현된 자료는 선형 분류방법으로 분류 불가.
- 새로운 변수 z를 도입하여 (x_1, x_2, z)로 차원증가 시킴.[$z = \exp(-(x_1^2 + x_2^2))$]
- 2개의 그룹을 완전하게 나누는 선형평면을 도출.
- Basis expansion : $h_1(x_1, x_2) = x_1$, $h_2(x_1, x_2) = x_2$, $h_2(x_1, x_2) = \exp(-(x_1^2 + x_2^2))$
- (h_1, h_2, h_3)은 선형이지만 (x_1, x_2)은 비선형, 위 방법을 비선형 SVM이라함.

Kernel trick

- 특성함수의 생성 어려움 + 고차원 확장시 차원의 저주 문제 발생.

- 2차 다항커널 : 입력변수 x_1과 x_2이고 i번째 관측치와 j번째 관측치일때,

$$\begin{aligned} K(\boldsymbol{x}_i, \boldsymbol{x}_j) &= (1 + \boldsymbol{x}_i^T \boldsymbol{x}_j)^2 \\ &= (1 + x_{i,1}x_{j,1} + x_{i,2}x_{j,2})^2 \\ &= 1 + 2x_{i,1}x_{j,1} + 2x_{i,2}x_{j,2} + (x_{i,1}x_{j,1})^2 + (x_{i,2}x_{j,2})^2 + 2x_{i,1}x_{j,1}x_{i,2}x_{j,2} \end{aligned} \quad (7.11)$$

- 이때 다음과 같이 정의하면,

$$h_1(x_1, x_2) = 1,\ h_2(x_1, x_2) = \sqrt{2}x_1,\ h_3(x_1, x_2) = \sqrt{2}x_2,\ h_4(x_1, x_2) = x_1^2,\ h_5(x_1, x_2) = x_2^2,\ h_6(x_1, x_2) = \sqrt{2}x_1x_2$$

$$\boldsymbol{h}(x_1, x_2) = (h_1(x_1, x_2), h_2(x_1, x_2), \cdots, h_6(x_1, x_2))^T$$

- 식 (7.11)은 $K(\boldsymbol{x}_i, \boldsymbol{x}_j) = (1 + \boldsymbol{x}_i^T \boldsymbol{x}_j)^2 = \boldsymbol{h}(\boldsymbol{x}_i)^T \boldsymbol{h}(\boldsymbol{x}_j)$ 로 변형 가능.

- 특성함수를 정의하지 않고 커널 함수를 이용.

- 즉, $\hat{\beta}$이 $\boldsymbol{h}(\boldsymbol{x}_i)^T \boldsymbol{h}(\boldsymbol{x}_j)$의 형태이면, $K(\boldsymbol{x}_i, \boldsymbol{x}_j)$를 직접 이용하여 추정.

β_0와 β의 추정 – *by kernel trick*

- 특성변수 x로 부터 basis함수 $\boldsymbol{h}(\boldsymbol{x})$로 차원을 증대시키면 커널 SVM 목적함수.

$$L_k = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j \boldsymbol{h}(\boldsymbol{x}_i)^T \boldsymbol{h}(\boldsymbol{x}_j) \qquad (7.12)$$

- 선형 SVM 식 (7.11)은 $\widehat{f}(\boldsymbol{x}) = \widehat{\beta}_0 + \widehat{\boldsymbol{\beta}}^T \boldsymbol{x} = \widehat{\beta}_0 + \sum_{i=1}^{n} \widehat{\alpha}_i y_i \boldsymbol{x}_i^T \boldsymbol{x}$ 로 변형 가능.

- L_k 최소화한 모수 추정치를 $\widehat{\beta}_0^*$와 $\widehat{\boldsymbol{\beta}}^*$ 라 할 때 커널 SVM의 예측치

$$\widehat{f}(\boldsymbol{x}) = \widehat{\beta}_0^* + \sum_{i=1}^{n} \widehat{\alpha}_i^* y_i \boldsymbol{h}(\boldsymbol{x}_i)^T \boldsymbol{h}(\boldsymbol{x}) \qquad (7.13)$$

- 식(7.12)와 식(7.13) 모두 $\boldsymbol{h}(\boldsymbol{x}_i)^T \boldsymbol{h}(\boldsymbol{x}_j)$의 형태임.

- 식(7.12)에 $\boldsymbol{h}(\boldsymbol{x}_i)^T \boldsymbol{h}(\boldsymbol{x}_j)$ 대신 커널 함수 $K(\boldsymbol{x}_i, \boldsymbol{x})$를 대체하여 $\widehat{\beta}_0^*$와 $\widehat{\boldsymbol{\beta}}^*$ 를 추정.

- 식(7.13)도 $\boldsymbol{h}(\boldsymbol{x}_i)^T \boldsymbol{h}(\boldsymbol{x}_j)$를 이용하여 동일한 커널 SVM을 구함.

❖ 3장에서 커널 분포 함수 추정에 사용한 커널과 구분필요~~!!

03 Sklearn을 이용한 SVM (실습)

1. 선형 SVM.

   → 참고자료: Confusion Matrix

		Predicted class	
		P	N
Actual class	P	True positives (TP)	False negatives (FN)
	N	False positives (FP)	True negatives (TN)

$$accuracy = \frac{TP + TN}{TP + FP + FN + TN}$$

$$percision = \frac{TP}{TP + FP}$$

$$recall = \frac{TP}{TP + FN}, \text{ 그리고 } f1 = 2\frac{precision \times recall}{precision + recall}$$

2. 비선형 SVM(Kernel SVM).

3. 얼굴인식 (Face recognition)예제.

Q & A