

Chapter2. Kernel-based Learning

고려대학교 산업경영공학과

DMQA Lab.

임새린

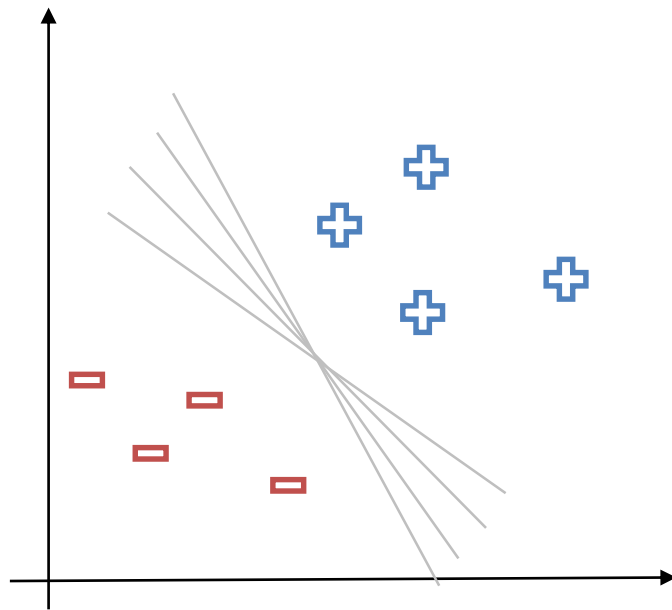
Support Vector Machine

Support Vector Machine

Methods

❖ Support Vector Machine

- 기계학습 알고리즘 중 하나로 탄탄한 이론적 배경, 손쉬운 적용, 그리고 준수한 성능을 가지고 있음
- '이진 분류 문제에서 어떤 decision boundary가 가장 좋은 decision boundary인가?'에 대한 하나의 답을 알고리즘으로 개발



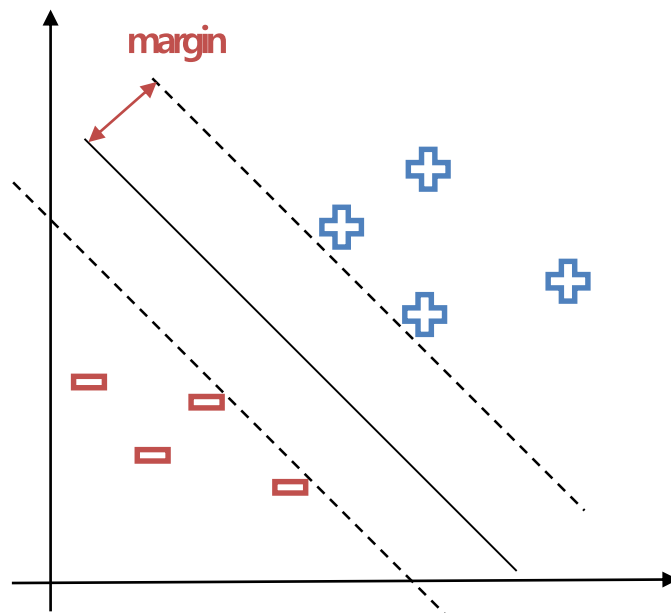
What is the best decision boundary?

Support Vector Machine

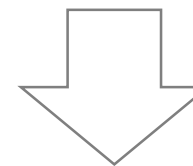
Methods

❖ Support Vector Machine

- 기계학습 알고리즘 중 하나로 탄탄한 이론적 배경, 손쉬운 적용, 그리고 준수한 성능을 가지고 있음
- '이진 분류 문제에서 어떤 decision boundary가 가장 좋은 decision boundary인가?'에 대한 하나의 답을 알고리즘으로 개발



What is the best decision boundary?



The decision boundary
that **maximize the margin**

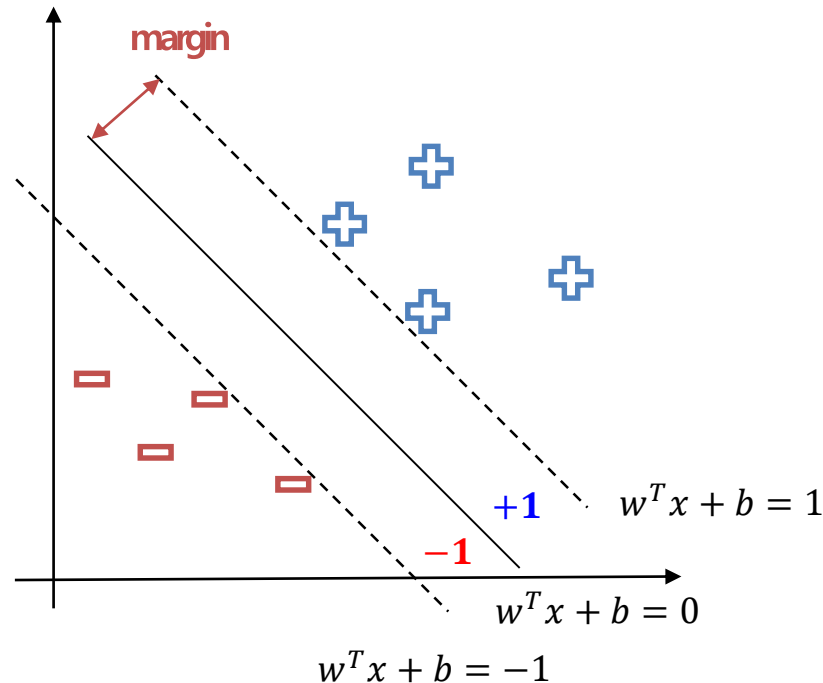
*Margin이란 분류 경계면으로부터 가장 가까운 양쪽 범주 객체와의 거리

Support Vector Machine

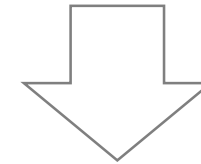
Methods

❖ Support Vector Machine

- 기계학습 알고리즘 중 하나로 탄탄한 이론적 배경, 손쉬운 적용, 그리고 준수한 성능을 가지고 있음
- '이진 분류 문제에서 어떤 decision boundary가 가장 좋은 decision boundary인가?'에 대한 하나의 답을 알고리즘으로 개발



What is the best decision boundary?



The decision boundary that **maximize the margin**

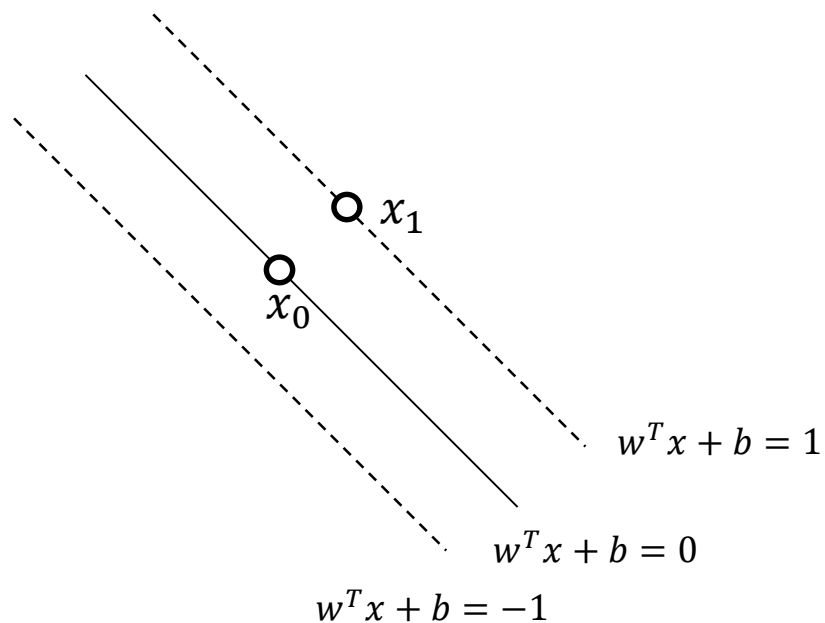
*Margin이란 분류 경계면으로부터 가장 가까운 양쪽 범주 객체와의 거리

Support Vector Machine

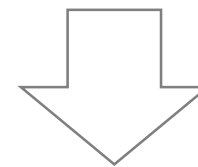
Methods

❖ How to get the margin?

- Hyperplane의 법선 벡터 w 를 활용하면 쉽게 유도 가능



How to get the margin?



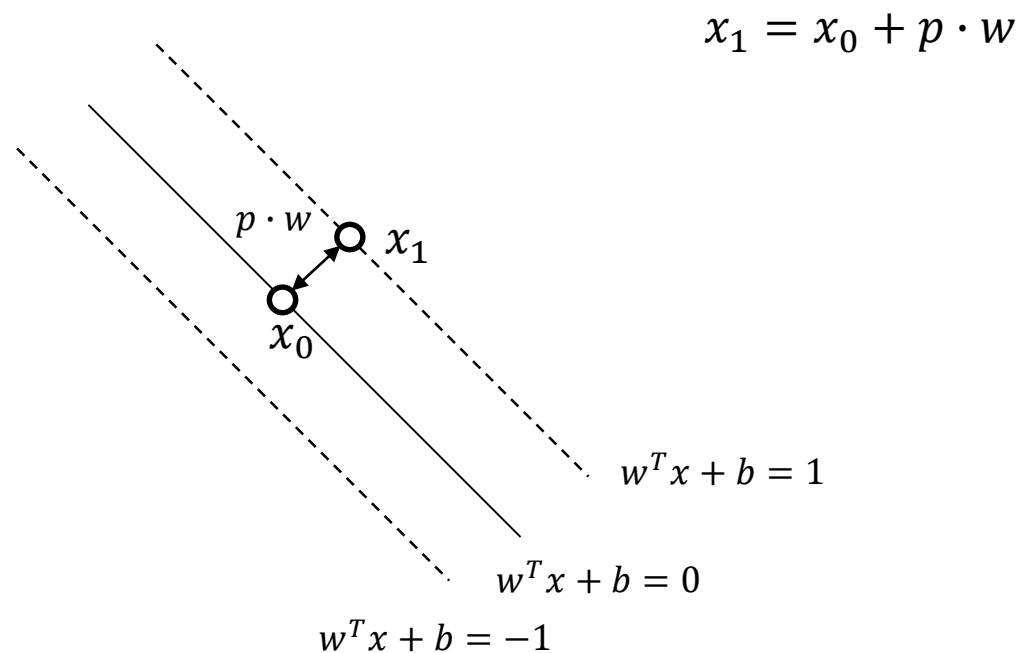
x_0 와 x_1 의 최단거리가 margin

Support Vector Machine

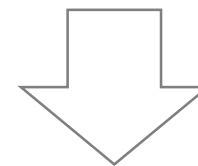
Methods

❖ How to get the margin?

- Hyperplane의 법선 벡터 w 를 활용하면 쉽게 유도 가능



How to get the margin?



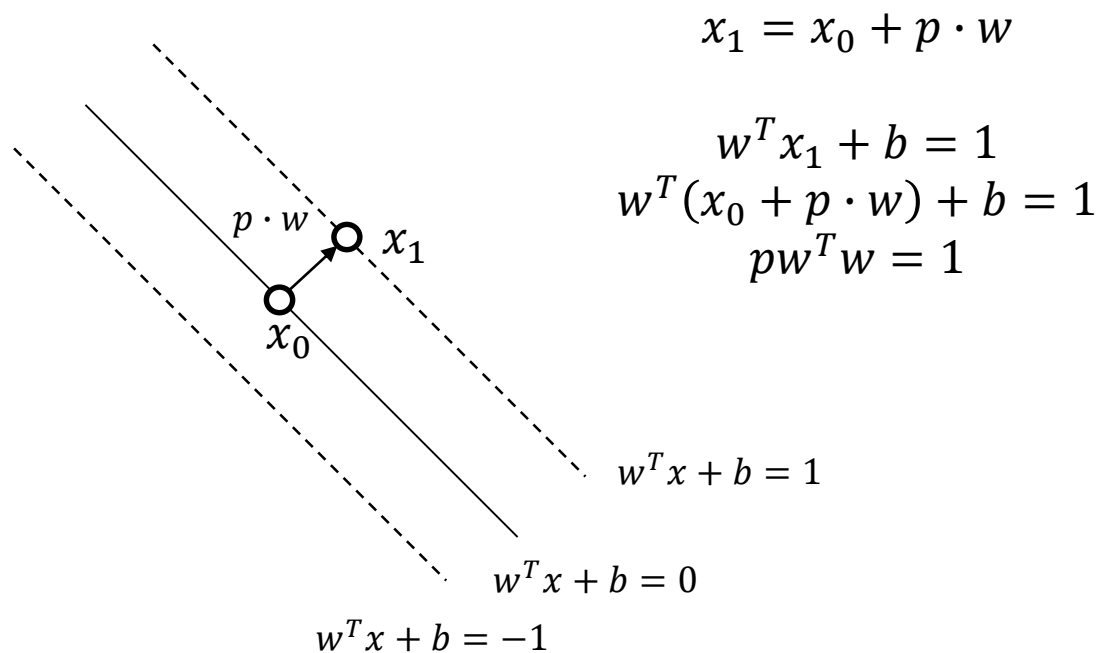
x_0 와 x_1 의 최단거리가 margin

Support Vector Machine

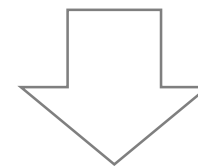
Methods

❖ How to get the margin?

- Hyperplane의 법선 벡터 w 를 활용하면 쉽게 유도 가능



How to get the margin?



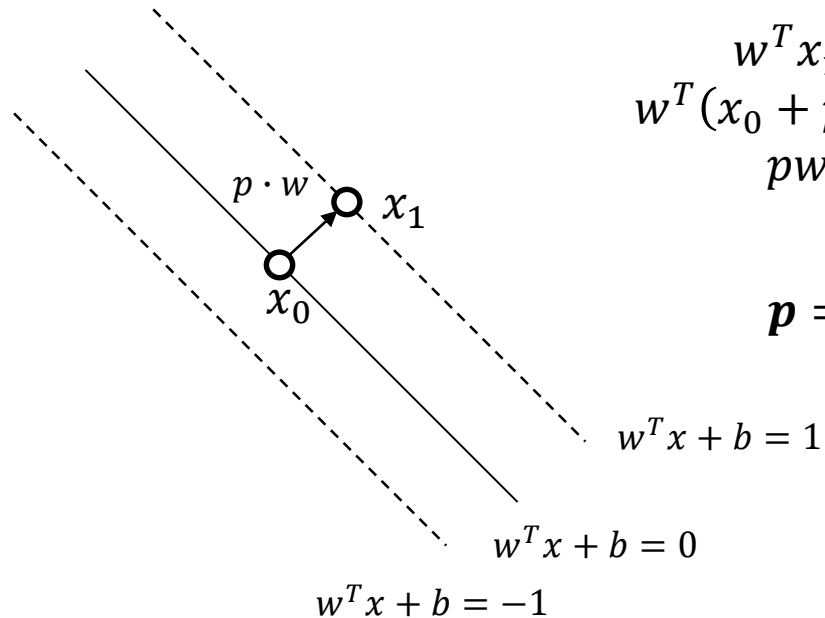
x_0 와 x_1 의 최단거리가 margin

Support Vector Machine

Methods

❖ How to get the margin?

- Hyperplane의 법선 벡터 w 를 활용하면 쉽게 유도 가능

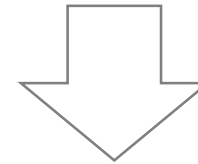


$$x_1 = x_0 + p \cdot w$$

$$\begin{aligned} w^T x_1 + b &= 1 \\ w^T (x_0 + p \cdot w) + b &= 1 \\ p w^T w &= 1 \end{aligned}$$

$$p = \frac{1}{\|w\|^2}$$

How to get the margin?



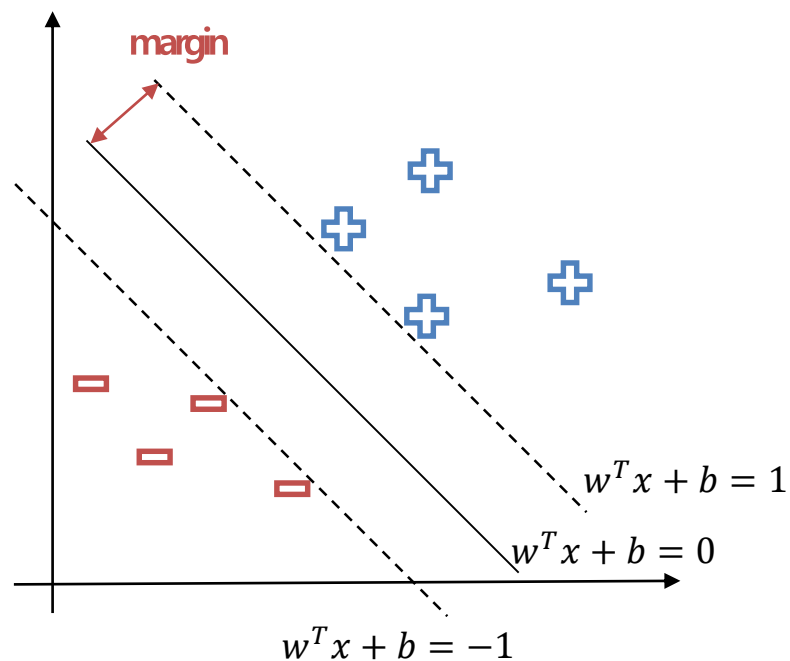
x_0 와 x_1 의 최단거리가 margin

Support Vector Machine

Methods

❖ Maximize margin

- 최종적으로 다음과 같은 최적화 문제를 푸는 것이 목표



$$\max \frac{1}{\|w\|^2} \leftrightarrow \min \frac{1}{2} \|w\|^2$$

$$s.t. y_i(w^T x_i + b) \geq 1, \text{ for all } i$$

$$y_i = \begin{cases} 1 & \text{if } w^T x_i + b \geq 1 \\ -1 & \text{o.w.} \end{cases}$$

Support Vector Machine

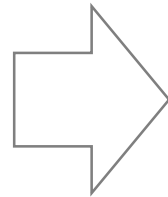
Methods

- ❖ Primal problem to Dual problem using Lagrange multiplier
 - 문제를 쉽게 해결하기 위해 Lagrange multiplier를 활용해 Dual problem으로 변경

Primal Problem

$$\min L_p(w) = \frac{1}{2} \|w\|^2$$

$$s.t. y_i(w^T x_i + b) \geq 1, \text{ for all } i$$



Lagrangian Problem

$$\min L_p(w, b, \alpha_i) = \frac{1}{2} \|w\|^2 - \sum_{i=1}^N \alpha_i (y_i (w^T x_i + b) - 1)$$

$$s.t. \alpha_i \geq 0, \text{ for all } i$$

KKT condition

$$\bullet \frac{\partial L_p}{\partial w} = 0 \rightarrow w = \sum_{i=1}^N \alpha_i y_i x_i \quad \bullet \frac{\partial L_p}{\partial b} = 0 \rightarrow \sum_{i=1}^N \alpha_i y_i = 0$$

Support Vector Machine

Methods

- ❖ Primal problem to Dual problem using Lagrange multiplier
 - 문제를 쉽게 해결하기 위해 Lagrange multiplier를 활용해 Dual problem으로 변경
 - α_i 에 대한 convex 함수이기 때문에 쉽게 최적해를 찾을 수 있음

Lagrangian Problem

$$\min L_p(w, b, \alpha_i) = \frac{1}{2} \|w\|^2 - \sum_{i=1}^N \alpha_i (y_i (w^T x_i + b) - 1)$$

$$s.t. \alpha_i \geq 0, \text{ for all } i$$

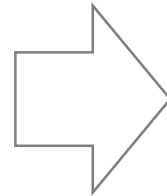
KKT condition

$$\bullet \frac{\partial L_p}{\partial w} = 0 \rightarrow w = \sum_{i=1}^N \alpha_i y_i x_i \quad \bullet \frac{\partial L_p}{\partial b} = 0 \rightarrow \sum_{i=1}^N \alpha_i y_i = 0$$

Dual Problem

$$\max L_D(\alpha_i) = \sum_{i=1}^N \alpha_i - \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j y_i y_j x_i^T x_j$$

$$s.t. \sum_{i=1}^N \alpha_i y_i = 0 \text{ and } \alpha_i \geq 0$$



Support Vector Machine

Methods

❖ Support Vector

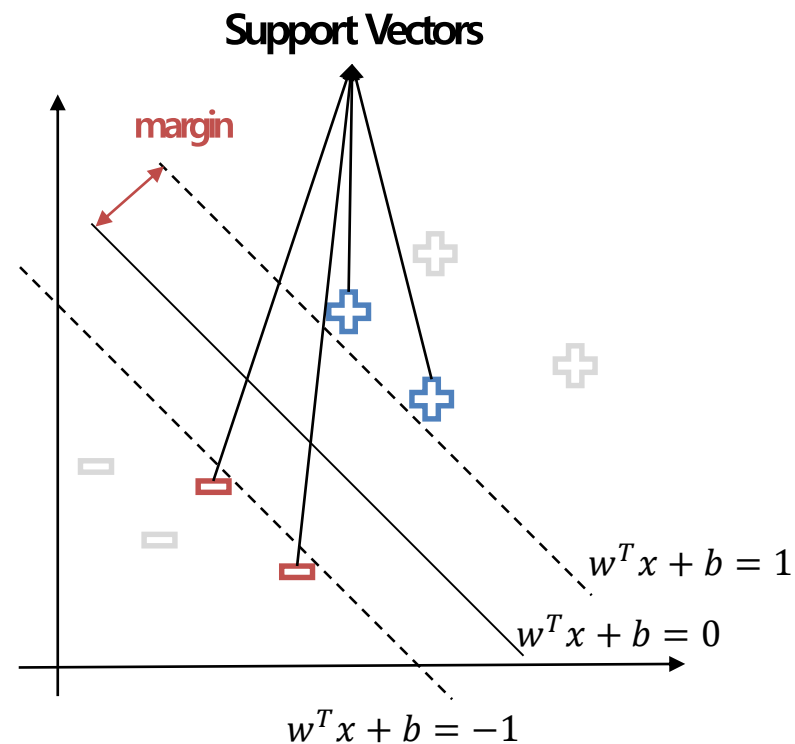
- KKT condition을 통해서 **margin** 위에 있는 관측치(support vector)만 있으면 문제를 해결이 가능함을 알 수 있음

KKT condition

$$\alpha_i(y_i(w^T x_i + b) - 1) = 0 \rightarrow \begin{cases} \alpha_i = 0 & \leftrightarrow y_i(w^T x_i + b) - 1 \neq 0 \\ \alpha_i \neq 0 & \leftrightarrow y_i(w^T x_i + b) - 1 = 0 \end{cases}$$



$$w = \sum_{i=1}^N \alpha_i y_i x_i = \sum_{i \in SV} \alpha_i y_i x_i$$

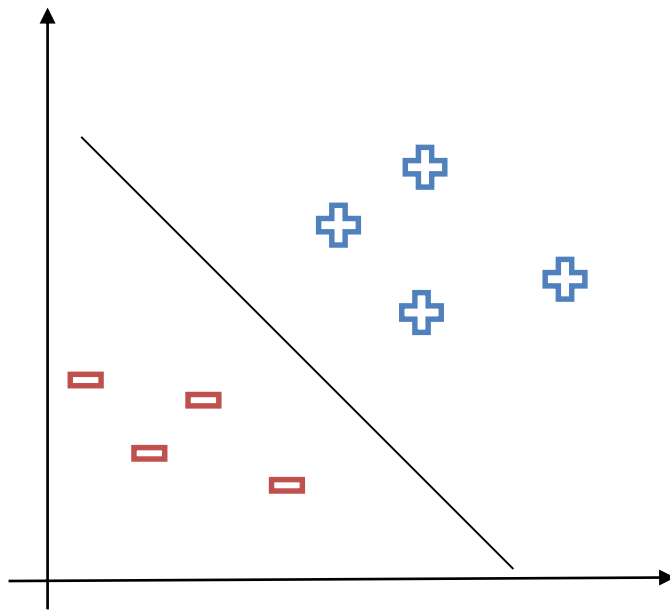


Support Vector Machine

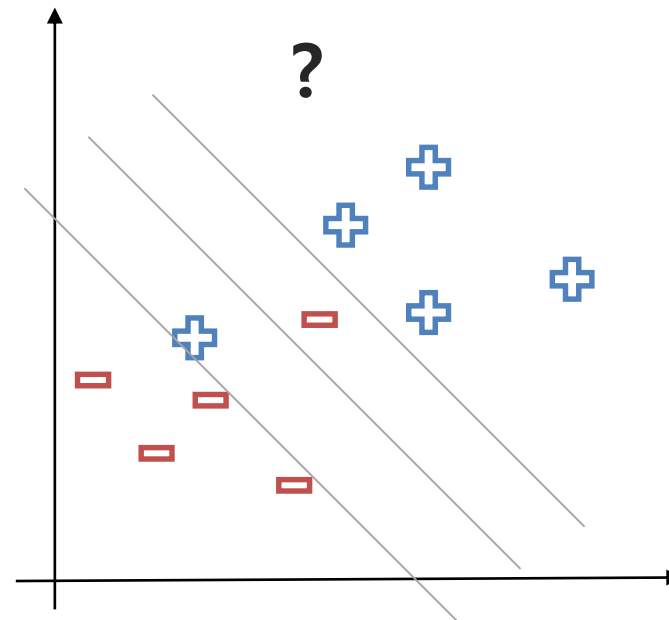
Methods

❖ Soft Margin SVM

- SVM은 데이터가 선형적으로 분리가 가능한 상황을 가정
- 하지만 현실에서 선형 분리가 가능한 데이터는 거의 존재하지 않음



Linearly separable



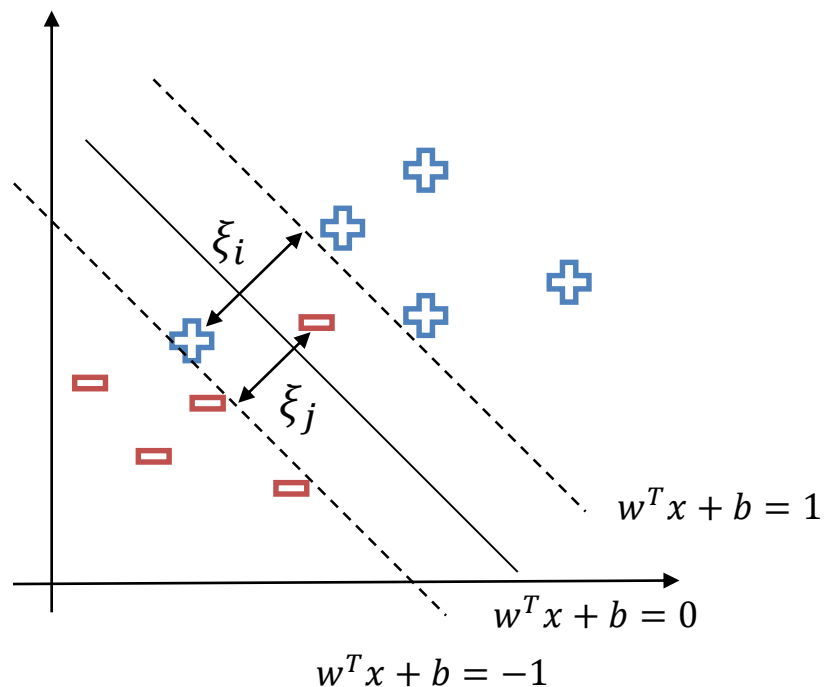
Linearly unseparable

Support Vector Machine

Methods

❖ Soft Margin SVM

- 패널티 개념을 도입하여 문제 해결
- 적절히 에러를 허용하며 margin을 최대화
- C 가 커지면 에러에 대해 엄격해지고 C 가 작아지면 에러에 대해 관대



$$\begin{aligned} \min L_p(w) &= \frac{1}{2} \|w\|^2 + C \sum_{i=1}^N \xi_i \\ \text{s.t. } y_i(w^T x_i + b) &\geq 1 - \xi_i, \quad \xi_i \geq 0 \quad \text{for all } i \end{aligned}$$

Support Vector Machine

Methods

❖ Soft Margin SVM

- Lagrangian problem으로 변환

Primal Problem

$$L_p(w) = \frac{1}{2} \|w\|^2 + C \sum_{i=1}^N \xi_i$$

$$s.t. y_i(w^T x_i + b) \geq 1 - \xi_i, \quad \xi_i \geq 0 \quad \text{for all } i$$



Lagrangian Problem

$$L_p(w, b, \alpha_i) = \frac{1}{2} \|w\|^2 + C \sum_{i=1}^N \xi_i - \sum_{i=1}^N \alpha_i (y_i (w^T x_i + b) - 1 + \xi_i) - \sum_{i=1}^N \mu_i \xi_i$$

$$s.t. \alpha_i \geq 0, \quad \mu_i \geq 0 \quad \text{for all } i$$

KKT condition

$$\begin{aligned} \bullet \quad \frac{\partial L_p}{\partial w} = 0 &\rightarrow w = \sum_{i=1}^N \alpha_i y_i x_i & \bullet \quad \frac{\partial L_p}{\partial b} = 0 &\rightarrow \sum_{i=1}^N \alpha_i y_i = 0 & \bullet \quad \frac{\partial L_p}{\partial \xi_i} = 0 &\rightarrow C - \alpha_i - \mu_i = 0 \end{aligned}$$

Support Vector Machine

Methods

❖ Soft Margin SVM

- KKT condition을 이용해 Dual problem으로 변환
- Dual problem은 convex 함수이기 때문에 쉽게 최적해를 찾을 수 있음

Lagrangian Problem

$$L_p(w, b, \alpha_i) = \frac{1}{2} \|w\|^2 + C \sum_{i=1}^N \xi_i - \sum_{i=1}^N \alpha_i (y_i (w^T x_i + b) - 1 + \xi_i) - \sum_{i=1}^N \mu_i \xi_i$$

$$s.t. \alpha_i \geq 0, \mu_i \geq 0 \text{ for all } i$$



Dual Problem

$$\max L_D(\alpha_i) = \sum_{i=1}^N \alpha_i - \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j y_i y_j x_i^T x_j$$
$$s.t. \sum_{i=1}^N \alpha_i y_i = 0 \text{ and } 0 \leq \alpha_i \leq C$$

Support Vector Machine

Methods

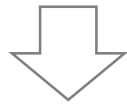
❖ Soft Margin SVM

- KKT condition을 이용해 Dual problem으로 변환
- Dual problem은 convex 함수이기 때문에 쉽게 최적해를 찾을 수 있음

KKT condition

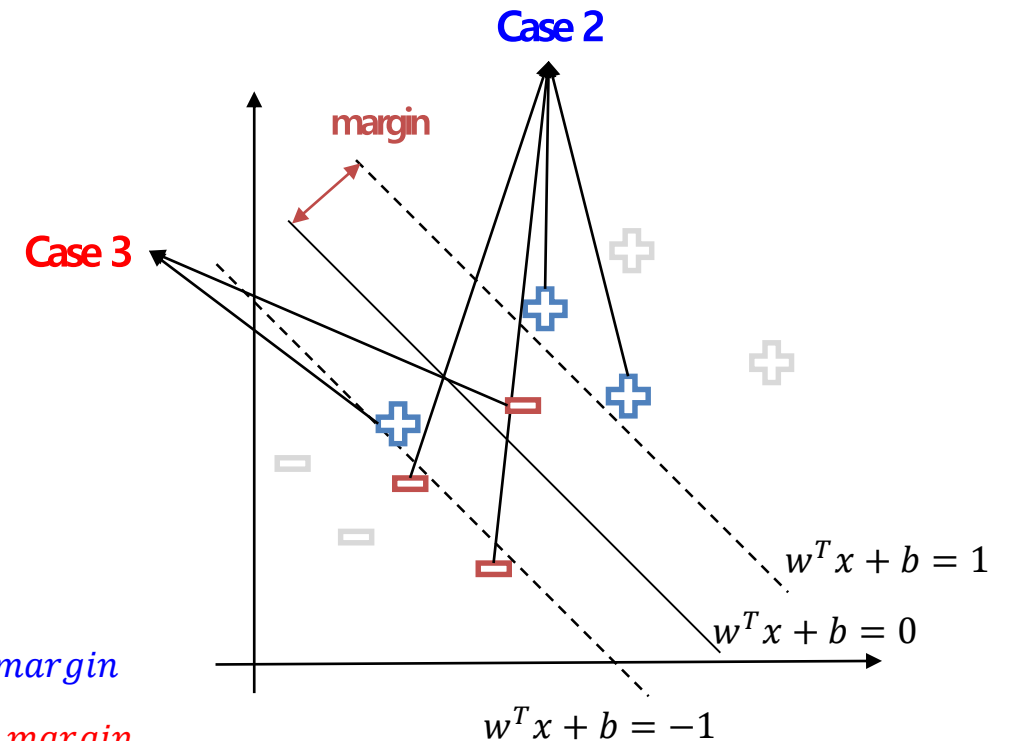
$$\alpha_i(y_i(w^T x_i + b) - 1 + \xi_i) = 0$$

$$C - \alpha_i - \mu_i = 0 \quad \& \quad \mu_i \xi_i = 0$$



$$w = \sum_{i=1}^N \alpha_i y_i x_i = \sum_{i \in SV} \alpha_i y_i x_i$$

- case 1: $\alpha_i = 0 \rightarrow$ non - support vectors
- case 2: $0 < \alpha_i < C \rightarrow$ support vectors on the margin
- case 3: $\alpha_i = C \rightarrow$ support vectors outside the margin

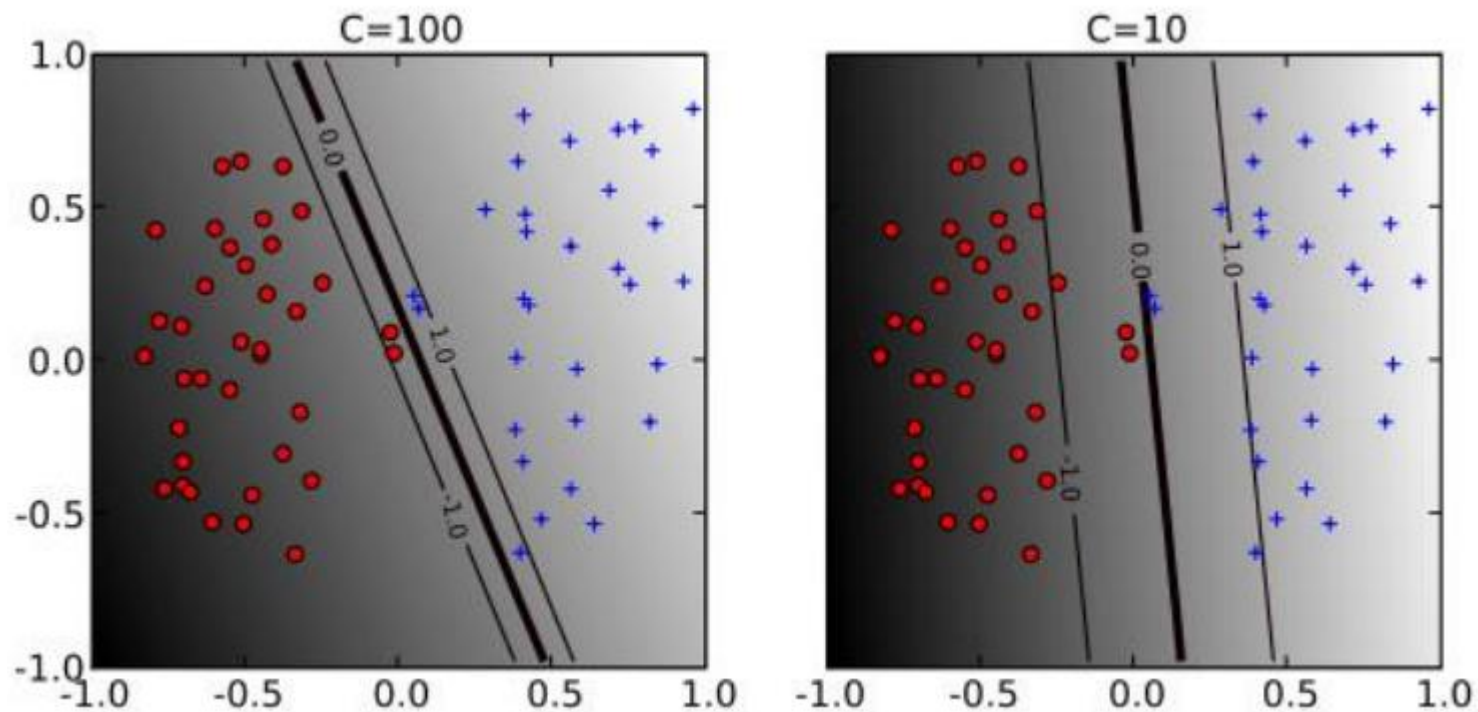


Support Vector Machine

Methods

❖ Soft Margin SVM

- C 가 크면 패널티를 거의 허용하지 않음 \rightarrow margin이 줄어들음
- C 가 작으면 패널티를 많이 허용 \rightarrow margin이 넓어짐

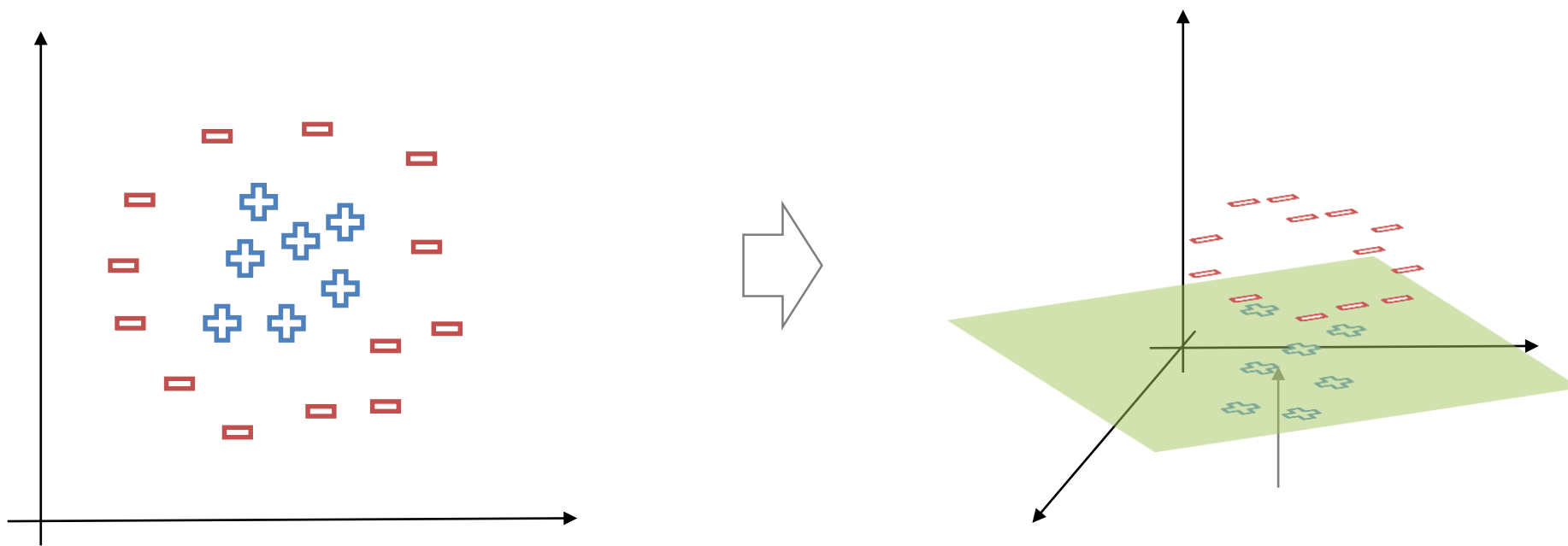


Support Vector Machine

Methods

❖ Non-linear decision boundary

- SVM은 기본적으로 선형 분류 경계선을 가정하기 때문에 non-linear한 상황에서는 좋은 성능을 낼 수 없음
- 고차원으로 매핑하는 함수 $\phi(x)$ 를 통해 관측치를 고차원으로 매핑하여 선형 분류



Support Vector Machine

Methods

❖ Non-linear decision boundary

- SVM은 기본적으로 선형 분류 경계선을 가정하기 때문에 non-linear한 상황에서는 좋은 성능을 낼 수 없음
- 고차원으로 매핑하는 함수 $\phi(x)$ 를 통해 관측치를 고차원으로 매핑하여 선형 분류

Primal Problem

$$L_p(w) = \frac{1}{2} \|w\|^2 + C \sum_{i=1}^N \xi_i$$

$$s.t. y_i(w^T \phi(x_i) + b) \geq 1 - \xi_i, \quad \xi_i \geq 0 \quad \text{for all } i$$



Lagrangian Problem

$$L_p(w, b, \alpha_i) = \frac{1}{2} \|w\|^2 + C \sum_{i=1}^N \xi_i - \sum_{i=1}^N \alpha_i (y_i(w^T \phi(x_i) + b) - 1 + \xi_i) - \sum_{i=1}^N \mu_i \xi_i$$

$$s.t. \alpha_i \geq 0, \quad \mu_i \geq 0 \quad \text{for all } i$$

KKT condition

$$\bullet \frac{\partial L_p}{\partial w} = 0 \rightarrow w = \sum_{i=1}^N \alpha_i y_i \phi(x_i) \quad \bullet \frac{\partial L_p}{\partial b} = 0 \rightarrow \sum_{i=1}^N \alpha_i y_i = 0 \quad \bullet \frac{\partial L_p}{\partial \xi_i} = 0 \rightarrow C - \alpha_i - \mu_i = 0$$

Support Vector Machine

Methods

❖ Non-linear decision boundary

- SVM은 기본적으로 선형 분류 경계선을 가정하기 때문에 non-linear한 상황에서는 좋은 성능을 낼 수 없음
- 고차원으로 매핑하는 함수 $\phi(x)$ 를 통해 관측치를 고차원으로 매핑하여 선형 분류

Lagrangian Problem

$$L_p(w, b, \alpha_i) = \frac{1}{2} \|w\|^2 + C \sum_{i=1}^N \xi_i - \sum_{i=1}^N \alpha_i (y_i (w^T \phi(x_i) + b) - 1 + \xi_i) - \sum_{i=1}^N \mu_i \xi_i$$

$$s.t. \alpha_i \geq 0, \mu_i \geq 0 \text{ for all } i$$



Dual Problem

$$\max L_D(\alpha_i) = \sum_{i=1}^N \alpha_i - \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j y_i y_j \phi(x_i)^T \phi(x_j) \\ s.t. \sum_{i=1}^N \alpha_i y_i = 0 \text{ and } 0 \leq \alpha_i \leq C$$

Support Vector Machine

Methods

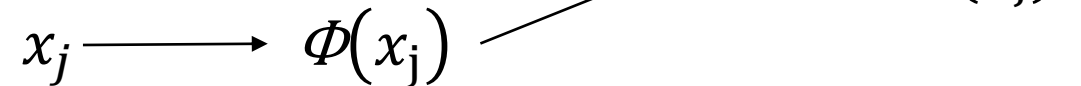
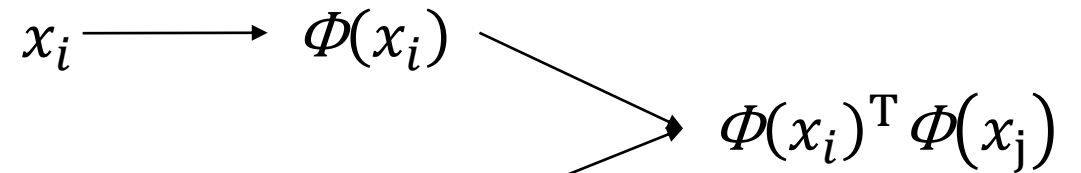
❖ Kernel trick for non-linear decision boundary

- 최적해를 찾기 위해서 우리가 필요한 값은 $\Phi(x_i)$ 가 아닌 $\Phi(x_i)^T \Phi(x_j)$
- Kernel function을 통해 더 효율적이며 현재까지 알려진 다양한 kernel function을 활용해 더 유연하게 문제 해결 가능

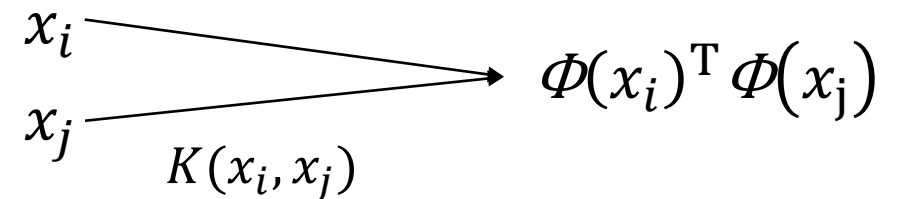
Dual Problem

$$\max L_D(\alpha_i) = \sum_{i=1}^N \alpha_i - \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j y_i y_j \Phi(x_i)^T \Phi(x_j)$$

$$s.t. \sum_{i=1}^N \alpha_i y_i = 0 \text{ and } 0 \leq \alpha_i \leq C$$



Kernel Trick



Support Vector Machine

Methods

❖ Kernel functions

- Polynomial kernel, Sigmoid kernel, Gaussian (RBF) kernel 등 많은 kernel 함수가 알려져 있음
- 그 중 차원을 무한으로 확장할 수 있는 Gaussian kernel을 가장 많이 활용

$$K(x, y) = \exp\left(-\frac{\|x - y\|^2}{2\sigma^2}\right), \quad \sigma \neq 0$$

Gaussian kernel

$$K(x, y) = (x \cdot y + c)^d, \quad c > 0$$

Polynomial kernel

$$K(x, y) = \tanh(a(x \cdot y) + b), \quad a, b \geq 0$$

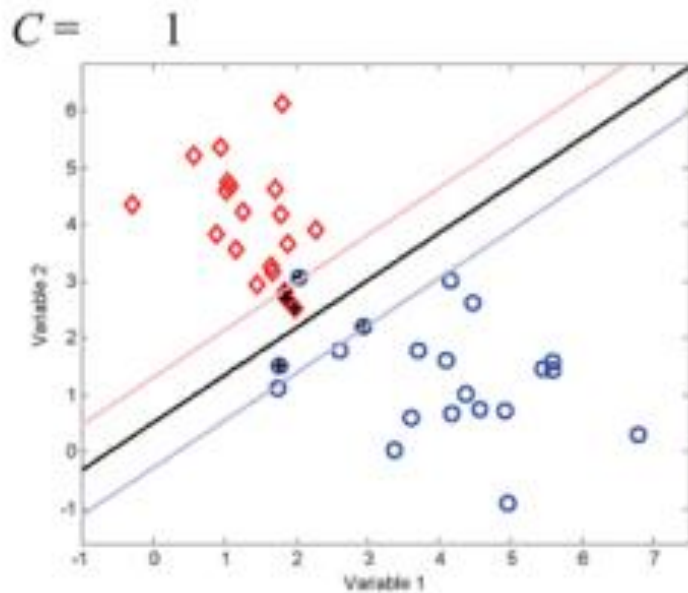
Sigmoid kernel

Support Vector Machine

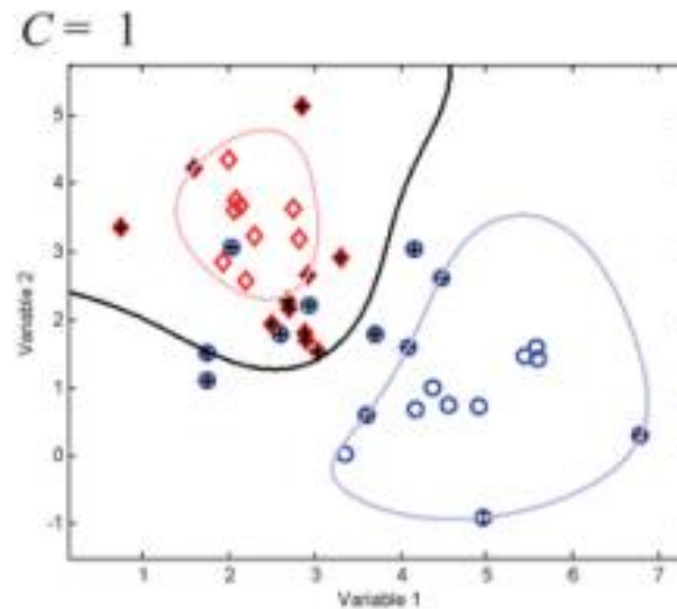
Methods

❖ Kernel functions

- Polynomial kernel, Sigmoid kernel, Gaussian (RBF) kernel 등 많은 kernel 함수가 알려져 있음
- 그 중 Gaussian kernel을 가장 많이 활용



Linear kernel



Gaussian kernel

감사합니다