

January 16, 2018



Inter-University Research Institute Corporation /
Research Organization of Information and Systems

National Institute of Informatics

Support Vector Machines

Data Mining 10 (データマイニング)

Mahito Sugiyama (杉山磨人)

Today's Outline

- Today's topic is **support vector machines** (SVMs)
 - A popular supervised classification method
- Perform binary classification by maximizing the margin
- Kernel trick for nonlinear classification

Classification Problem Setting

- Given a supervised dataset $D = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \dots, (\mathbf{x}_N, y_N)\}$, $\mathbf{x}_i \in \mathbb{R}^n$ (feature vector), $y_i \in C = \{-1, 1\}$ (label)

- Use a decision function (hyperplane) in the form of

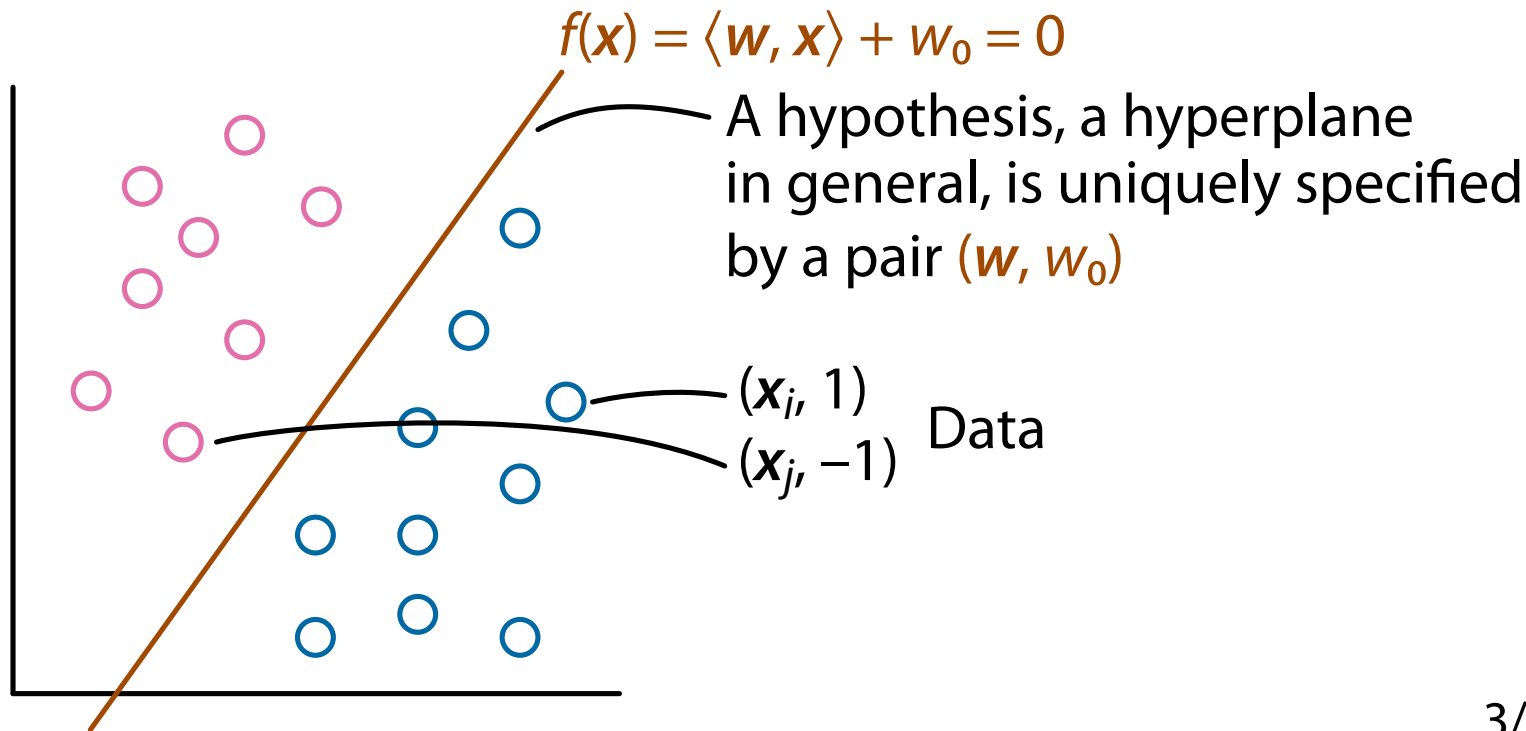
$$f(\mathbf{x}) = \langle \mathbf{w}, \mathbf{x} \rangle + w_0 = \sum_{j=1}^n w^j x^j + w_0$$

- A classifier $g(\mathbf{x})$ is given as

$$g(\mathbf{x}) = \begin{cases} 1 & \text{if } f(\mathbf{x}) > 0, \\ -1 & \text{if } f(\mathbf{x}) < 0 \end{cases}$$

- Goal: Find (\mathbf{w}, w_0) that correctly classifies the dataset

Classification by Hyperplane



Example: Perceptron

Algorithm 1: Perceptron

```
1 perceptron( $D$ )
2   Set small random values to  $\mathbf{w}$  and  $w_0$ ;           // initialization
3   foreach  $i \in \{1, 2, \dots, N\}$  do
4      $a \leftarrow \langle \mathbf{w}, \mathbf{x}_i \rangle + w_0$ 
5     if  $y_i \cdot a < 0$  then
6        $\mathbf{w} \leftarrow \mathbf{w} + y_i \mathbf{x}_i$ ;           // update the weight
7        $w_0 \leftarrow w_0 + y_i$ ;                       // update the bias
```

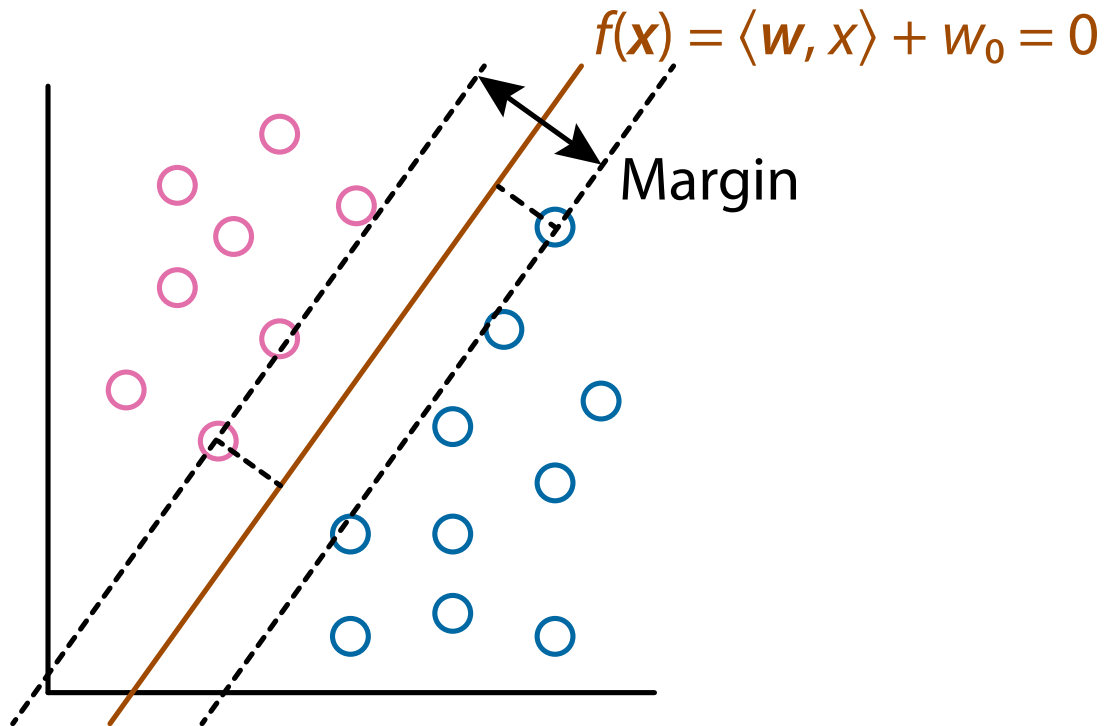
Correctness of Perceptron

- It is guaranteed that a perceptron always converges to a correct classifier
 - A correct classifier is a function f s.t.
 - $f(\mathbf{x}) > 0$ if $y = 1$,
 - $f(\mathbf{x}) < 0$ if $y = -1$
 - The convergence theorem
- Note: there are (infinitely) many functions that correctly classify two classes
 - A perceptron converges to one of them

Support Vector Machines (SVMs)

- A dataset D is **separable** by $f \iff y_i f(\mathbf{x}_i) > 0, \forall i \in \{1, 2, \dots, N\}$
- The **margin** is the distance from the classification hyperplane to the closest data point
- Support vector machines (SVMs) tries to find a hyperplane that **maximize** the margin

Margin



Formulation of SVMs

- The distance from a point \mathbf{x}_i to a hyperplane $f(\mathbf{x}) = \langle \mathbf{w}, \mathbf{x} \rangle + w_0$ is

$$\frac{|f(\mathbf{x}_i)|}{\|\mathbf{w}\|} = \frac{|\langle \mathbf{w}, \mathbf{x}_i \rangle + w_0|}{\|\mathbf{w}\|}$$

- Since $y_i f(\mathbf{x}_i) > 0$ should be satisfied, assume that there exists $M > 0$ such that $y_i f(\mathbf{x}_i) \geq M$ for all $i \in \{1, 2, \dots, N\}$
- The margin maximization problem can be written as

$$\max_{\mathbf{w}, w_0, M} \frac{M}{\|\mathbf{w}\|} \quad \text{subject to } y_i f(\mathbf{x}_i) \geq M, i \in \{1, 2, \dots, N\}$$

– $M = \min_{i \in \{1, 2, \dots, N\}} |\langle \mathbf{w}, \mathbf{x}_i \rangle + w_0|$

Hard Margin SVMs

- We can eliminate M and obtain

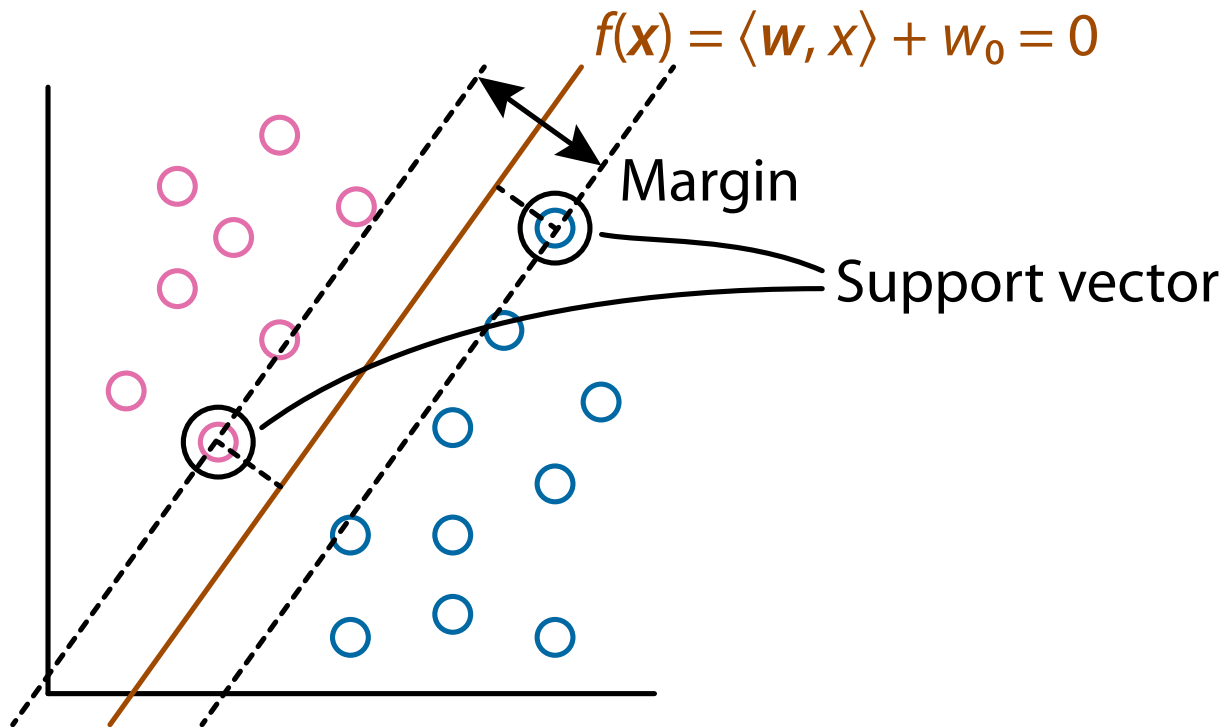
$$\max_{\mathbf{w}, w_0} \frac{1}{\|\mathbf{w}\|} \quad \text{subject to } y_i f(\mathbf{x}_i) \geq 1, i \in \{1, 2, \dots, N\}$$

- This is equivalent to

$$\min_{\mathbf{w}, w_0} \|\mathbf{w}\|^2 \quad \text{subject to } y_i f(\mathbf{x}_i) \geq 1, i \in \{1, 2, \dots, N\}$$

- The standard formulation of **hard margin SVMs**
- There are data points x_i satisfying $y_i f(\mathbf{x}_i) = 1$, called **support vectors**
- The solution does not change even data points that are not support vectors are removed

Margin



Soft Margin

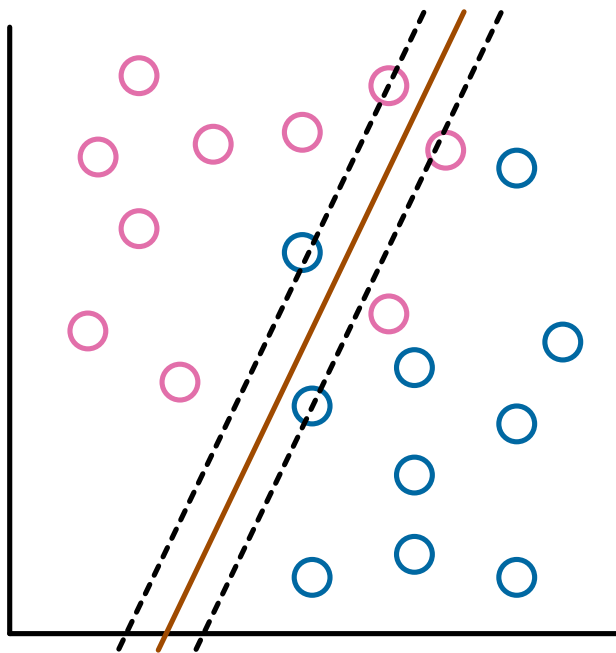
- Datasets are not often separable
- Extend SV classification to **soft margin** by relaxing $\langle \mathbf{w}, \mathbf{x} \rangle + w_0 \geq 1$
- Change the constraint $y_i f(\mathbf{x}_i) \geq 1$ using the **slack variable** ξ_i to $y_i f(\mathbf{x}_i) = y_i (\langle \mathbf{w}, \mathbf{x} \rangle + w_0) \geq 1 - \xi_i, \quad i \in \{1, 2, \dots, n\}$
- The formulation of **soft margin SVM** (C-SVM) is

$$\min_{\mathbf{w}, w_0, \xi} \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i \in \{1, 2, \dots, N\}} \xi_i \quad \text{s.t. } y_i f(\mathbf{x}_i) \geq 1 - \xi_i, \xi_i \geq 0, i \in \{1, 2, \dots, N\}$$

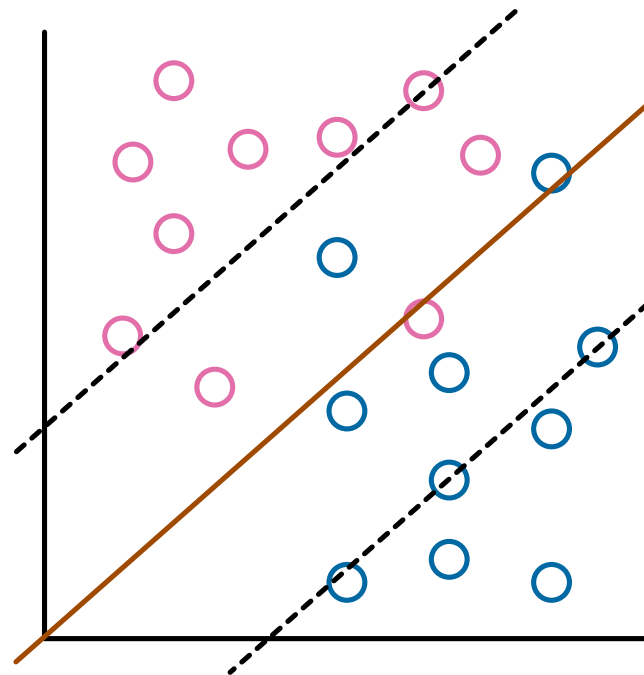
- C is called the **regularization parameter**

Soft Margin

C is large



C is small



Data Point Location

- $y_i f(\mathbf{x}_i) > 1$: \mathbf{x}_i is outside margin
 - These points do not affect to the classification hyperplane
- $y_i f(\mathbf{x}_i) = 1$: \mathbf{x}_i is on margin
- $y_i f(\mathbf{x}_i) < 1$: \mathbf{x}_i is inside margin
 - These points do not exist in hard margin
- Points on margin and inside margin are support vectors

Dual Problem (1/4)

- The formulation of C-SVM

$$\min_{\mathbf{w}, w_0, \boldsymbol{\xi}} \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i \in \{1, 2, \dots, N\}} \xi_i \quad \text{s.t. } y_i f(\mathbf{x}_i) \geq 1 - \xi_i, \xi_i \geq 0, i \in \{1, 2, \dots, N\}$$

is called the **primal problem**

- This is usually solved via the **dual problem**
- Make the **Lagrange function** using $\boldsymbol{\alpha} = (\alpha_1, \dots, \alpha_N)$, $\boldsymbol{\mu} = (\mu_1, \dots, \mu_N)$:

$$L(\mathbf{w}, w_0, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\mu}) = \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i \in [N]} \xi_i - \sum_{i \in [N]} \alpha_i (y_i f(\mathbf{x}_i) - 1 + \xi_i) - \sum_{i \in [N]} \mu_i \xi_i$$

$$- [N] = \{1, 2, \dots, N\}$$

Dual Problem (2/4)

- Let us consider

$$D(\boldsymbol{\alpha}, \boldsymbol{\mu}) = \min_{\mathbf{w}, w_0, \boldsymbol{\xi}} L(\mathbf{w}, w_0, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\mu})$$

and its maximization

$$\max_{\boldsymbol{\alpha} \geq 0, \boldsymbol{\mu} \geq 0} D(\boldsymbol{\alpha}, \boldsymbol{\mu}) = \max_{\boldsymbol{\alpha} \geq 0, \boldsymbol{\mu} \geq 0} \min_{\mathbf{w}, w_0, \boldsymbol{\xi}} L(\mathbf{w}, w_0, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\mu})$$

- The inside minimization is achieved when

$$\frac{\partial L}{\partial \mathbf{w}} = \mathbf{w} - \sum_{i \in [N]} \alpha_i y_i \mathbf{x}_i = 0, \quad \frac{\partial L}{\partial w_0} = - \sum_{i \in [N]} \alpha_i y_i = 0, \quad \frac{\partial L}{\partial \xi_i} = C - \alpha_i - \mu_i = 0$$

Dual Problem (3/4)

- Putting the three conditions to the Lagrange function to remove \mathbf{w} , w_0 , and $\boldsymbol{\xi}$, yielding

$$\begin{aligned} L &= \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i \in [N]} \xi_i - \sum_{i \in [N]} a_i (y_i f(\mathbf{x}_i) - 1 + \xi_i) - \sum_{i \in [N]} \mu_i \xi_i \\ &= \frac{1}{2} \|\mathbf{w}\|^2 - \sum_{i \in [N]} a_i y_i \langle \mathbf{w}, \mathbf{x}_i \rangle - w_0 \sum_{i \in [N]} a_i y_i + \sum_{i \in [N]} a_i + \sum_{i \in [N]} (C - a_i - \mu_i) \xi_i \\ &= -\frac{1}{2} \sum_{i, j \in [N]} a_i a_j y_i y_j \langle \mathbf{x}_i, \mathbf{x}_j \rangle + \sum_{i \in [N]} a_i \end{aligned}$$

Dual Problem (4/4)

- It can be proved that $\max_{\alpha \geq 0, \mu \geq 0} \min_{\mathbf{w}, w_0, \xi} L(\mathbf{w}, w_0, \xi, \alpha, \mu)$, that is, the **dual problem**

$$\max_{\alpha} -\frac{1}{2} \sum_{i,j \in [N]} a_i a_j y_i y_j \langle \mathbf{x}_i, \mathbf{x}_j \rangle + \sum_{i \in [N]} a_i \quad \text{s.t.} \quad \sum_{i \in [N]} a_i y_i = 0, \quad 0 \leq a_i \leq C, \quad i \in [N]$$

is equivalent to the **primal problem**

$$\min_{\mathbf{w}, w_0, \xi} \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i \in \{1, 2, \dots, N\}} \xi_i \quad \text{s.t.} \quad y_i f(\mathbf{x}_i) \geq 1 - \xi_i, \quad \xi_i \geq 0, \quad i \in [N]$$

KKT (Karush-Kuhn-Tucker) condition

- The necessary conditions for a solution to be optimal:

$$\frac{\partial L}{\partial \mathbf{w}} = \mathbf{w} - \sum_{i \in [N]} a_i y_i \mathbf{x}_i = 0, \quad \frac{\partial L}{\partial w_0} = - \sum_{i \in [N]} a_i y_i = 0, \quad \frac{\partial L}{\partial \xi_i} = C - a_i - \mu_i = 0$$

$$- (y_i f(\mathbf{x}_i) - 1 + \xi_i) \leq 0, \quad -\xi_i \leq 0,$$

$$a_i \geq 0, \quad \mu_i \geq 0,$$

$$a_i (y_i f(\mathbf{x}_i) - 1 - \xi_i) = 0, \quad \mu_i \xi_i = 0,$$

$$i \in [N]$$

Recovering Primal Variables

- Using these conditions, from the optimal α , we have

$$f(\mathbf{x}) = \sum_{i \in [N]} \alpha_i y_i \langle \mathbf{x}_i, \mathbf{x} \rangle + w_0,$$

$$w_0 = y_i - \sum_{j \in [N]} \alpha_j y_j \langle \mathbf{x}_j, \mathbf{x}_i \rangle, \quad \forall i \in \{i \in [N] \mid 0 < \alpha_i < C\}$$

- Since the second condition holds for all $i \in \{i \in [N] \mid 0 < \alpha_i < C\}$, one can take the average to avoid numerical errors

Data Point Location

- $y_i f(\mathbf{x}_i) > 1 \iff \alpha_i = 0$: \mathbf{x}_i is outside margin
 - These points do not affect to the classification hyperplane
- $y_i f(\mathbf{x}_i) = 1 \iff 0 < \alpha_i < C$: \mathbf{x}_i is on margin
- $y_i f(\mathbf{x}_i) < 1 \iff \alpha_i = C$: \mathbf{x}_i is inside margin
 - These points do not exist in hard margin
- Points on margin and inside margin are support vectors

How to Solve?

- The (dual) problem:

$$\max_{\mathbf{a}} -\frac{1}{2} \mathbf{a}^T Q \mathbf{a} + \mathbf{1}^T \mathbf{a} \quad \text{s.t. } \mathbf{y}^T \mathbf{a} = 0, 0 \leq \mathbf{a} \leq C \mathbf{1}$$

- $Q \in \mathbb{R}^{N \times N}$ is the matrix such that $q_{ij} = y_i y_j \langle \mathbf{x}_i, \mathbf{x}_j \rangle$
- Since analytical solution is not available, iterative approach for continuous optimization with constraints is needed
- One of standard methods is the **active set method**

Active Set Method

- Divide the set $[N]$ of indices into three sets:

$$O = \{i \in [N] \mid a_i = 0\}$$

$$M = \{i \in [N] \mid 0 < a_i < C\}$$

$$I = \{i \in [N] \mid a_i = C\}$$

- O and I are called **active sets**
- The problem can be solved w.r.t. $i \in M$, yielding

$$\begin{bmatrix} Q_M & \mathbf{y}_M \\ \mathbf{y}_M^T & 0 \end{bmatrix} \begin{bmatrix} \alpha_M \\ v \end{bmatrix} = -C \begin{bmatrix} Q_{M,I} & \mathbf{1} \\ \mathbf{1}^T & \mathbf{y}_I \end{bmatrix} + \begin{bmatrix} \mathbf{1} \\ 0 \end{bmatrix}$$

- This can be directly solved if Q_M is positive definite

Algorithm 2: Active Set Method

```
1 activeSetMethod( $D$ )
2   Initialize  $M, I, O$ 
3   while there exists  $i$  s.t.  $y_i f(\mathbf{x}_i) < 1, i \in O$  or  $y_i f(\mathbf{x}_i) > 1, i \in I$  do
4     Update  $M, I, O$ 
5     repeat
6        $\mathbf{a}_M^{\text{new}} \leftarrow$  the solution of the above equation
7        $\mathbf{d} \leftarrow \mathbf{a}_M^{\text{new}} - \mathbf{a}_M$ 
8        $\mathbf{a}_M \leftarrow \mathbf{a}_M + \eta \mathbf{d}$ ; // the maximum  $\eta$  satisfying  $\mathbf{a}_M \in [0, C]^{|M|}$ 
9       Move  $i \in M$  from  $M$  to  $I$  or  $O$  if  $a_i = C$  or  $a_i = 0$ 
10    until  $\mathbf{a}_M = \mathbf{a}_M^{\text{new}}$ ;
```


Extension to Nonlinear Classification

- To achieve nonlinear classification, convert each data point \mathbf{x} to some point $\varphi(\mathbf{x})$, and $f(\mathbf{x})$ becomes

$$f(\mathbf{x}) = \langle \mathbf{w}, \varphi(\mathbf{x}) \rangle + w_0$$

- The dual problem becomes

$$\max_{\mathbf{a}} -\frac{1}{2} \sum_{i,j \in [N]} a_i a_j y_i y_j \langle \varphi(\mathbf{x}_i), \varphi(\mathbf{x}_j) \rangle + \sum_{i \in [N]} a_i \quad \text{s.t.} \quad \sum_{i \in [N]} a_i y_i = 0, \quad 0 \leq a_i \leq C, \quad i \in [N]$$

- Only the dot product $\langle \varphi(\mathbf{x}_i), \varphi(\mathbf{x}_j) \rangle$ is used!
 - We do not even need to know $\varphi(\mathbf{x}_i)$ and $\varphi(\mathbf{x}_j)$
- **Kernel function:** $K(\mathbf{x}_i, \mathbf{x}_j) = \langle \varphi(\mathbf{x}_i), \varphi(\mathbf{x}_j) \rangle$

C-SVM with Kernel Trick

- Using the kernel function K , we have

$$\max_{\mathbf{a}} -\frac{1}{2} \sum_{i,j \in [N]} a_i a_j y_i y_j K(\mathbf{x}_i, \mathbf{x}_j) + \sum_{i \in [N]} a_i \quad \text{s.t.} \quad \sum_{i \in [N]} a_i y_i = 0, \quad 0 \leq a_i \leq C, i \in [N]$$

- The technique of using K is called **kernel trick**

Positive Definite Kernel

- A kernel $K : \Omega \times \Omega \rightarrow \mathbb{R}$ is a **positive definite kernel** if

(i) $K(x, y) = K(y, x)$

- (ii) For x_1, x_2, \dots, x_N , the $N \times N$ matrix

$$(K_{ij}) = \begin{bmatrix} K(x_1, x_1) & K(x_2, x_1) & \dots & K(x_N, x_1) \\ K(x_1, x_2) & K(x_2, x_2) & \dots & K(x_N, x_2) \\ \dots & \dots & \dots & \dots \\ K(x_1, x_N) & K(x_2, x_N) & \dots & K(x_N, x_N) \end{bmatrix}$$

is positive (semi-)definite, that is, $\sum_{i,j=1}^N c_i c_j K(x_i, x_j) \geq 0$
for any $c_1, c_2, \dots, c_N \in \mathbb{R}$

- $(K_{ij}) \in \mathbb{R}^{N \times N}$ is called the **Gram matrix**

Popular Positive Definite Kernels

- Linear Kernel

$$K(\mathbf{x}, \mathbf{y}) = \langle \mathbf{x}, \mathbf{y} \rangle$$

- Gaussian (RBF) kernel

$$K(\mathbf{x}, \mathbf{y}) = \exp\left(-\frac{1}{\sigma^2} \|\mathbf{x} - \mathbf{y}\|^2\right)$$

- Polynomial Kernel

$$K(\mathbf{x}, \mathbf{y}) = (\langle \mathbf{x}, \mathbf{y} \rangle + c)^d \quad c, d \in \mathbb{R}$$

Simple Kernels

- The all-ones kernel

$$K(\mathbf{x}, \mathbf{y}) = 1$$

- The delta (Dirac) kernel

$$K(\mathbf{x}, \mathbf{y}) = \begin{cases} 1 & \text{if } \mathbf{x} = \mathbf{y}, \\ 0 & \text{otherwise} \end{cases}$$

Closure Properties of Kernels

- For two kernels K_1 and K_2 , $K_1 + K_2$ is a kernel
- For two kernels K_1 and K_2 , the product $K_1 \cdot K_2$ is a kernel
- For a kernel K and a positive scalar $\lambda \in \mathbb{R}^+$, λK is a kernel
- For a kernel K on a set D , its zero-extension:

$$K_o(\mathbf{x}, \mathbf{y}) = \begin{cases} K(\mathbf{x}, \mathbf{y}) & \text{if } \mathbf{x}, \mathbf{y} \in D, \\ 0 & \text{otherwise} \end{cases}$$

is a kernel

Kernels on Structured Data

- Given objects X and Y , **decompose** them into substructures S and T
- The **R-convolution kernel** K_R by Haussler (1999) is given as

$$K_R(X, Y) = \sum_{s \in S, t \in T} K_{\text{base}}(s, t)$$

- K_{base} is an arbitrary base kernel, often the delta kernel
- For example, X is a graph and S is the set of all subgraphs

Summary

- SVM finds the “best” classification hyperplane
 - The **margin** is maximized
- Although the original SVM can perform only linear classification, it can be extended to nonlinear classification by using **kernels**
- Gaussian kernel + C-SVM can be the first choice