Support Vector Machines

COMP90051 Statistical Machine Learning

Semester 2, 2020

Qiuhong Ke

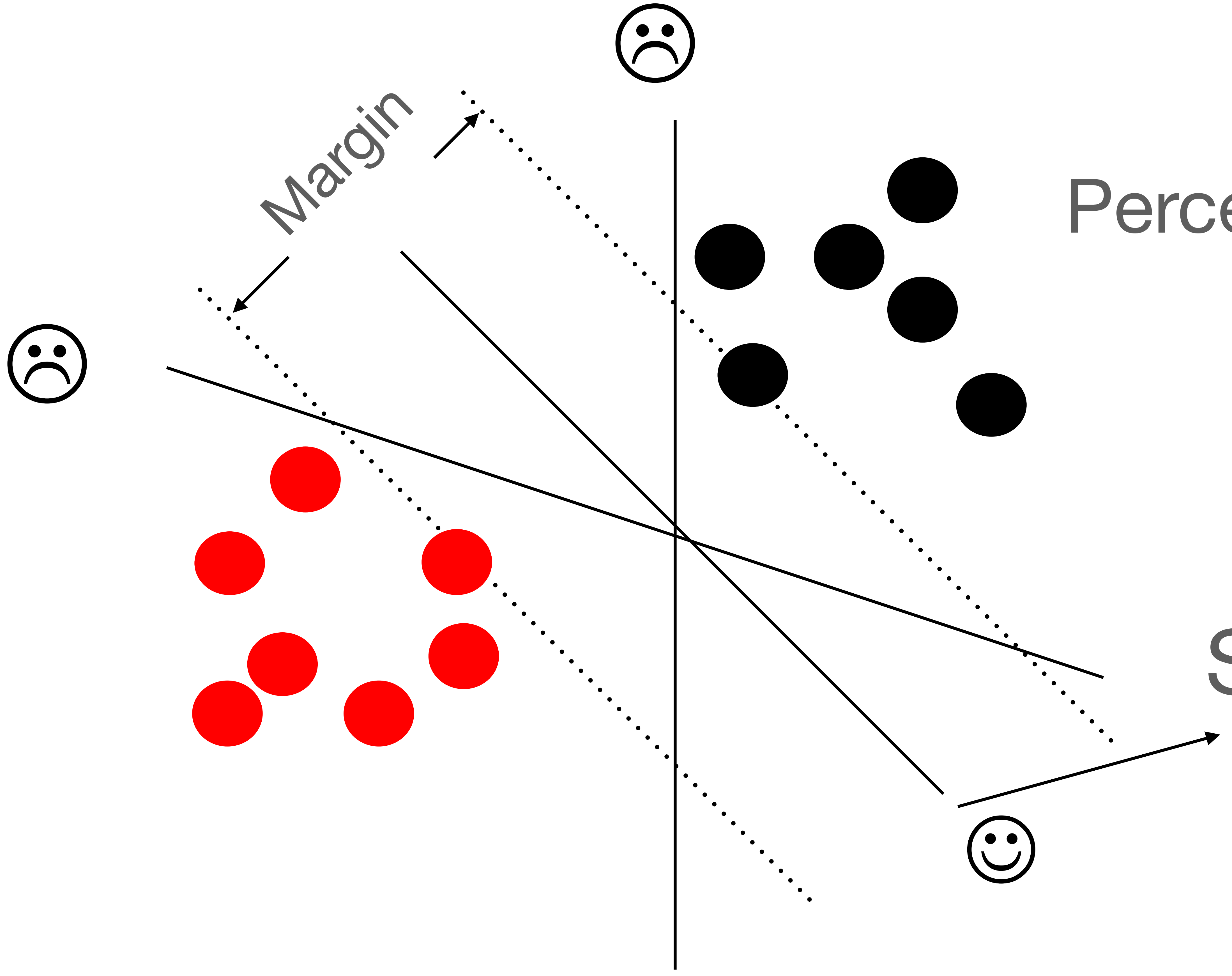
Before we start…

About me

- 2015.02-2018.04: PhD in UWA
- 2018.05-2019.12: Post-doc in MPII
- From 2020.01: Lecturer in UniMelb
- Research: Action recognition and prediction using machine learning
- Contact:
  - qiuhong.ke@unimelb.edu.com;
  - comp90051-2020s2-staff@lists.unimelb.edu.au

Binary Linear Classifier

SVM vs Perceptron



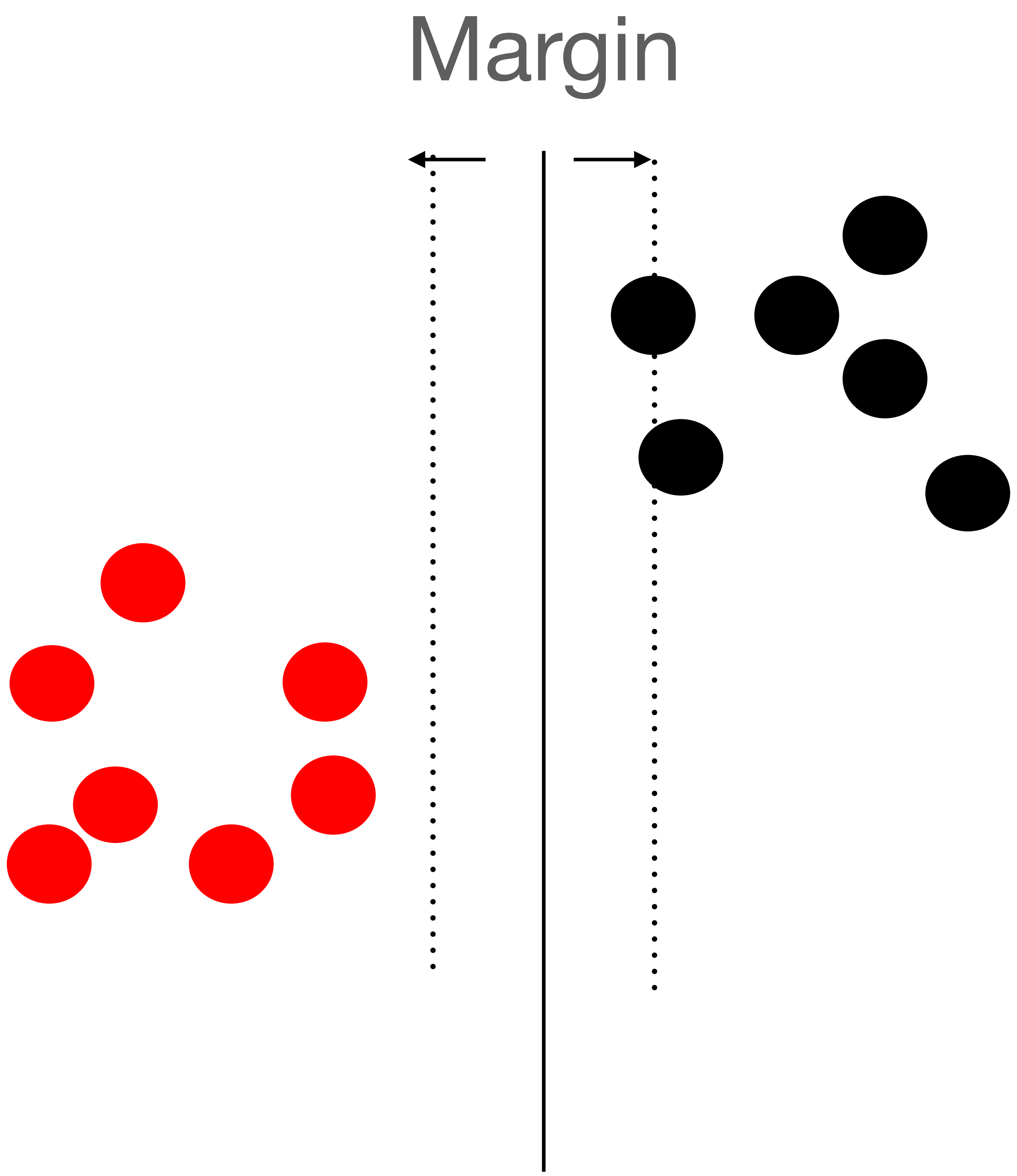
Perceptron :all boundaries are equally good as they separate classes perfectly

SVM: Maximum-margin Classifier

Margin: 2x minimum distance (boundary, data points)

Binary Linear Classifier

SVM vs Perceptron



Margin: 2x minimum distance (boundary, data points)

Outline

- Margin
- Lagrange Duality
- Soft-margin SVM
- Kernels

Linear classifier

$$f(x) = w^T x + b$$

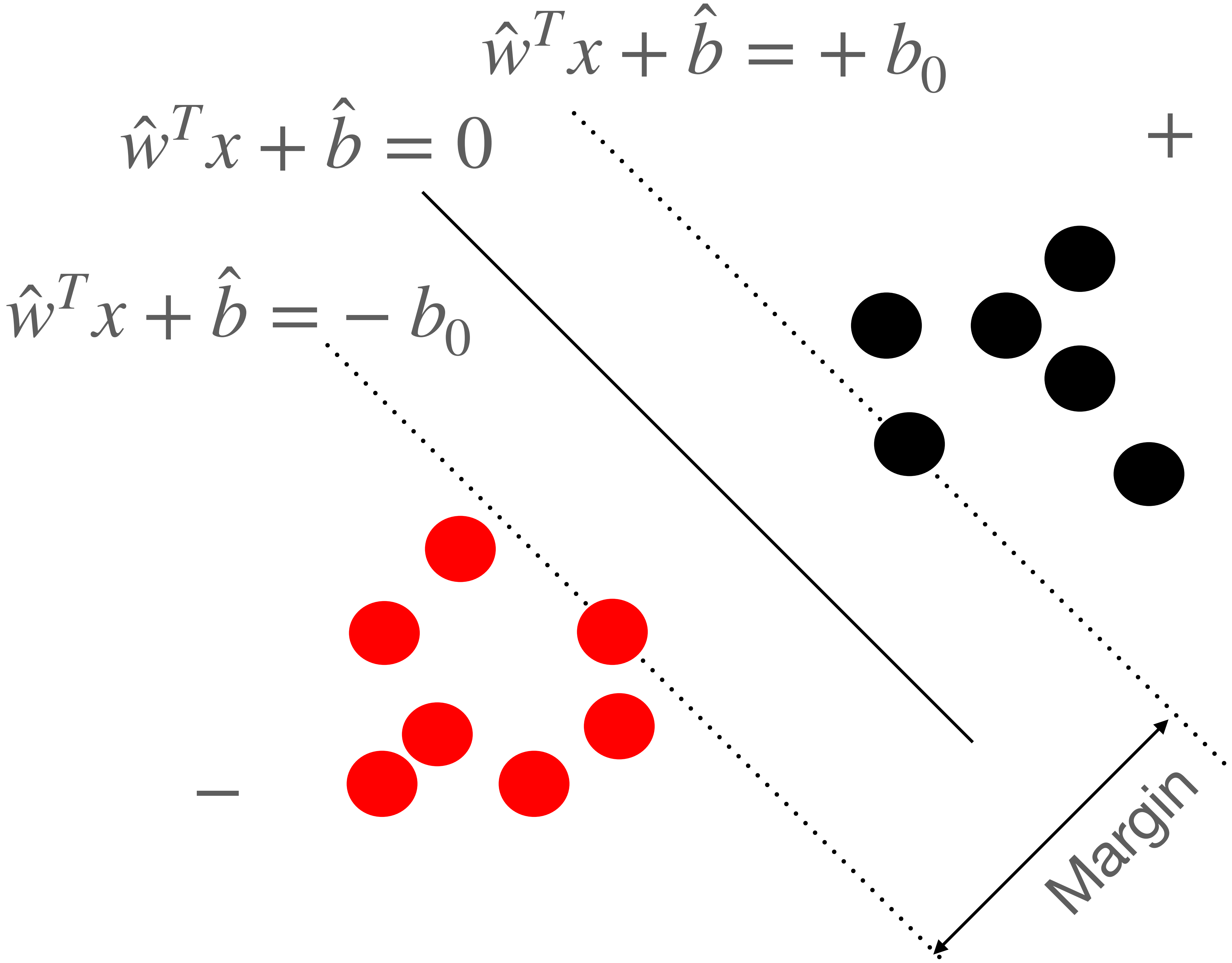
$x:$ Feature vector (column)

$w:$ Weight vector (column)

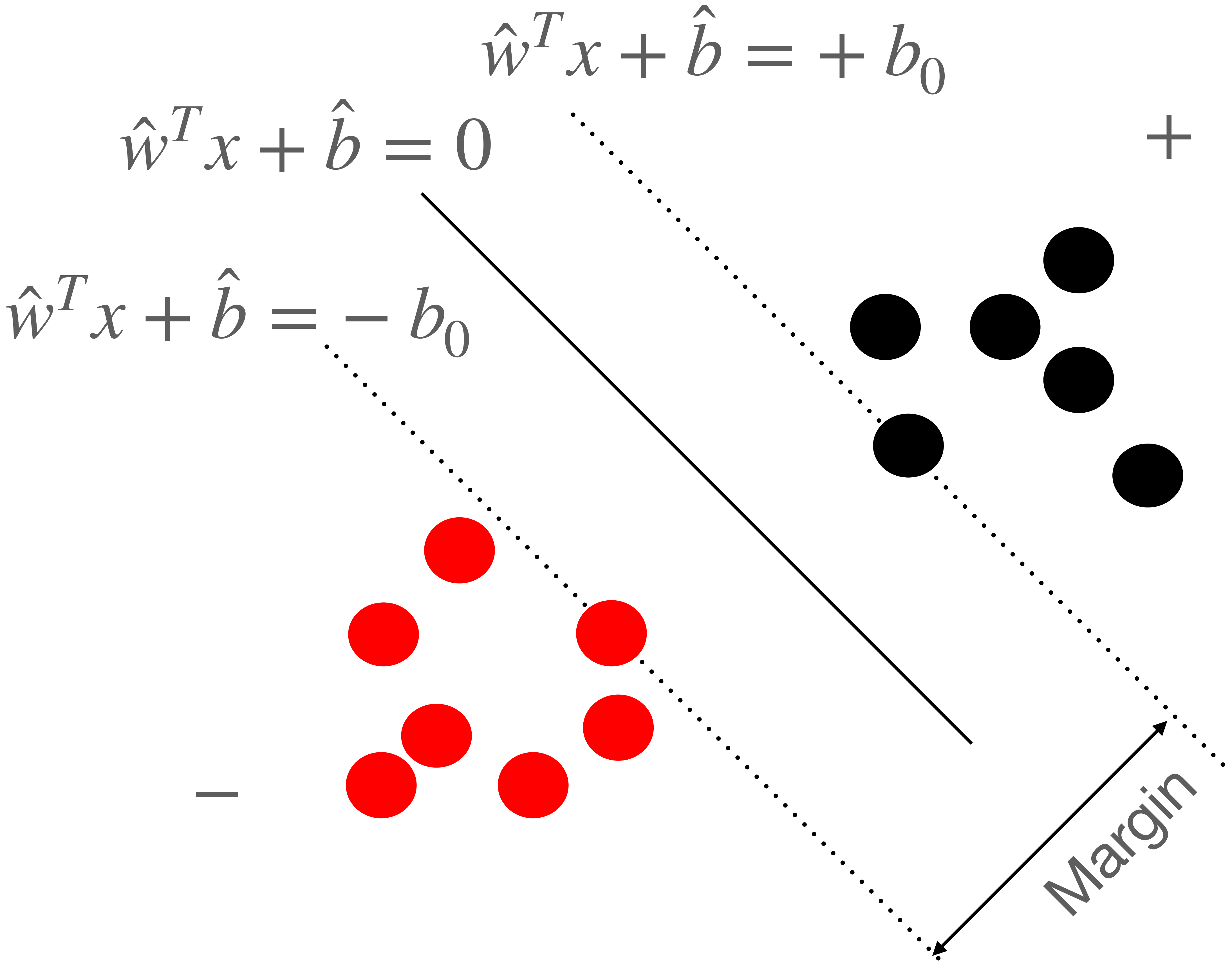
$T:$ Transpose

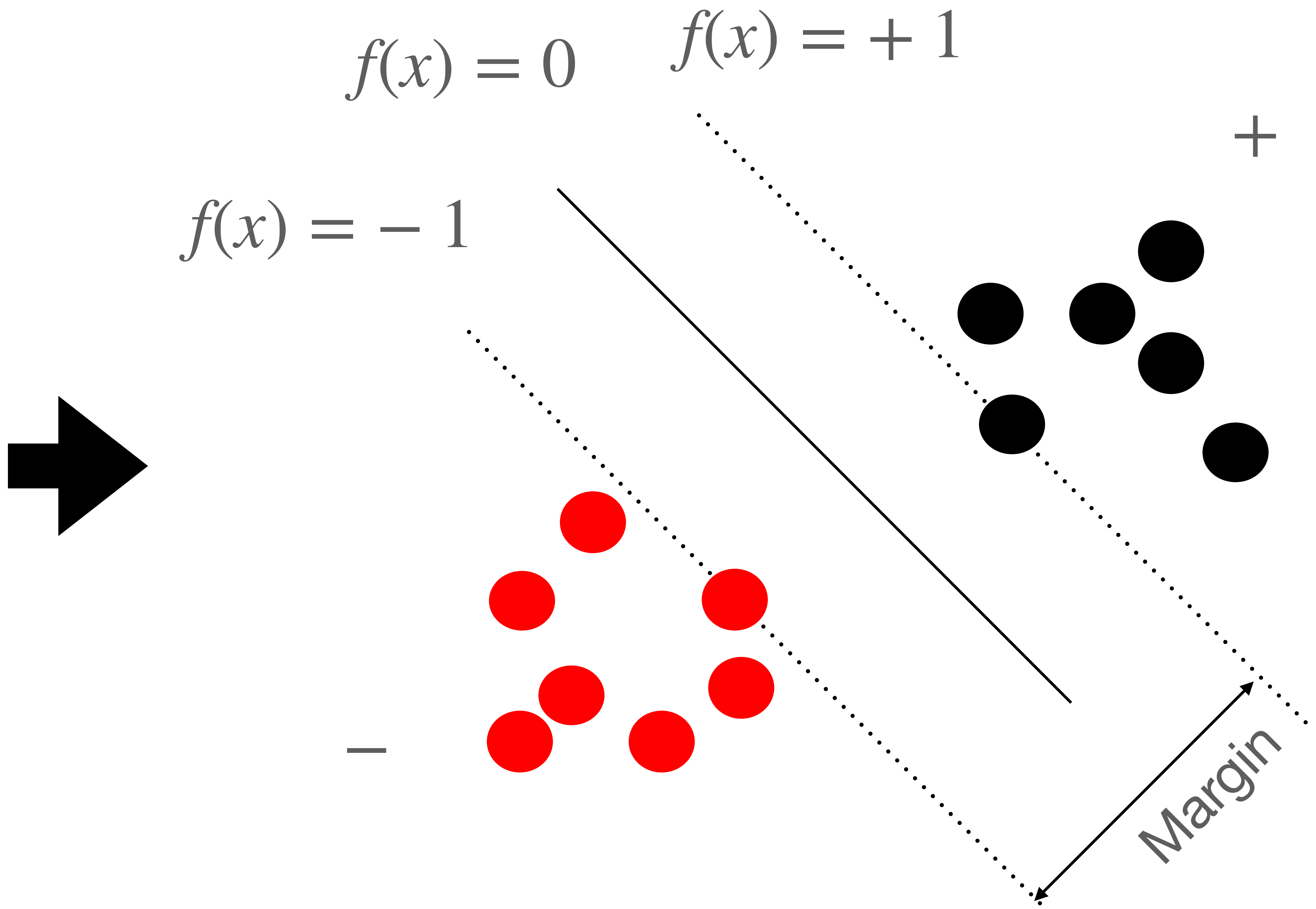
$b:$ Bias

$w^T x = \|w\| \|x\| \cos\theta$



$$f(x) = w^T x + b \quad w = \frac{\hat{w}}{b_0} \quad b = \frac{\hat{b}}{b_0}$$





Margin formula

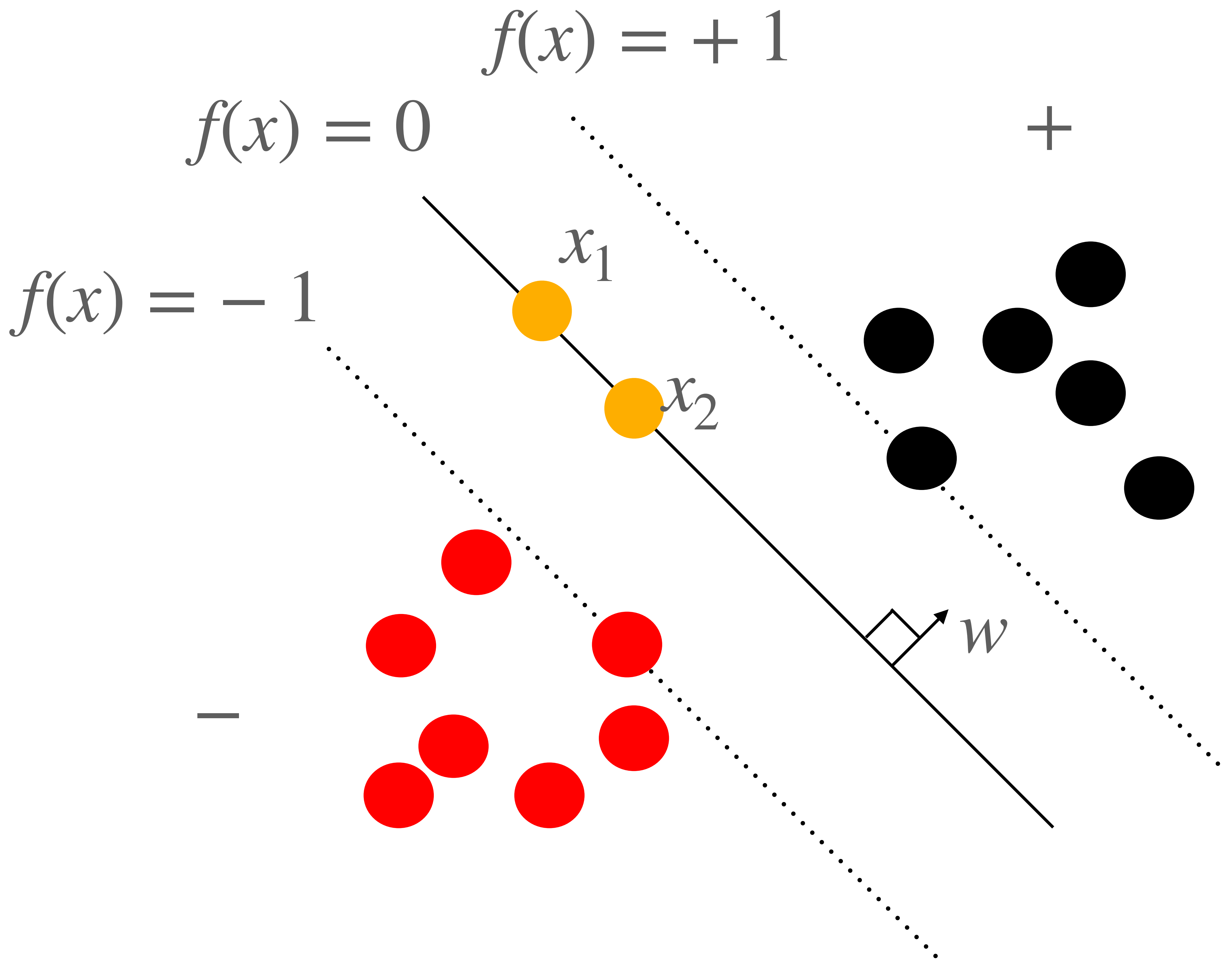
$$f(x) = w^T x + b$$

$$w^T x_1 + b = 0$$

$$w^T x_2 + b = 0$$

$$w^T (x_1 - x_2) = 0$$

$$\|w\| \|x_1 - x_2\| cos\theta = 0$$



Margin formula

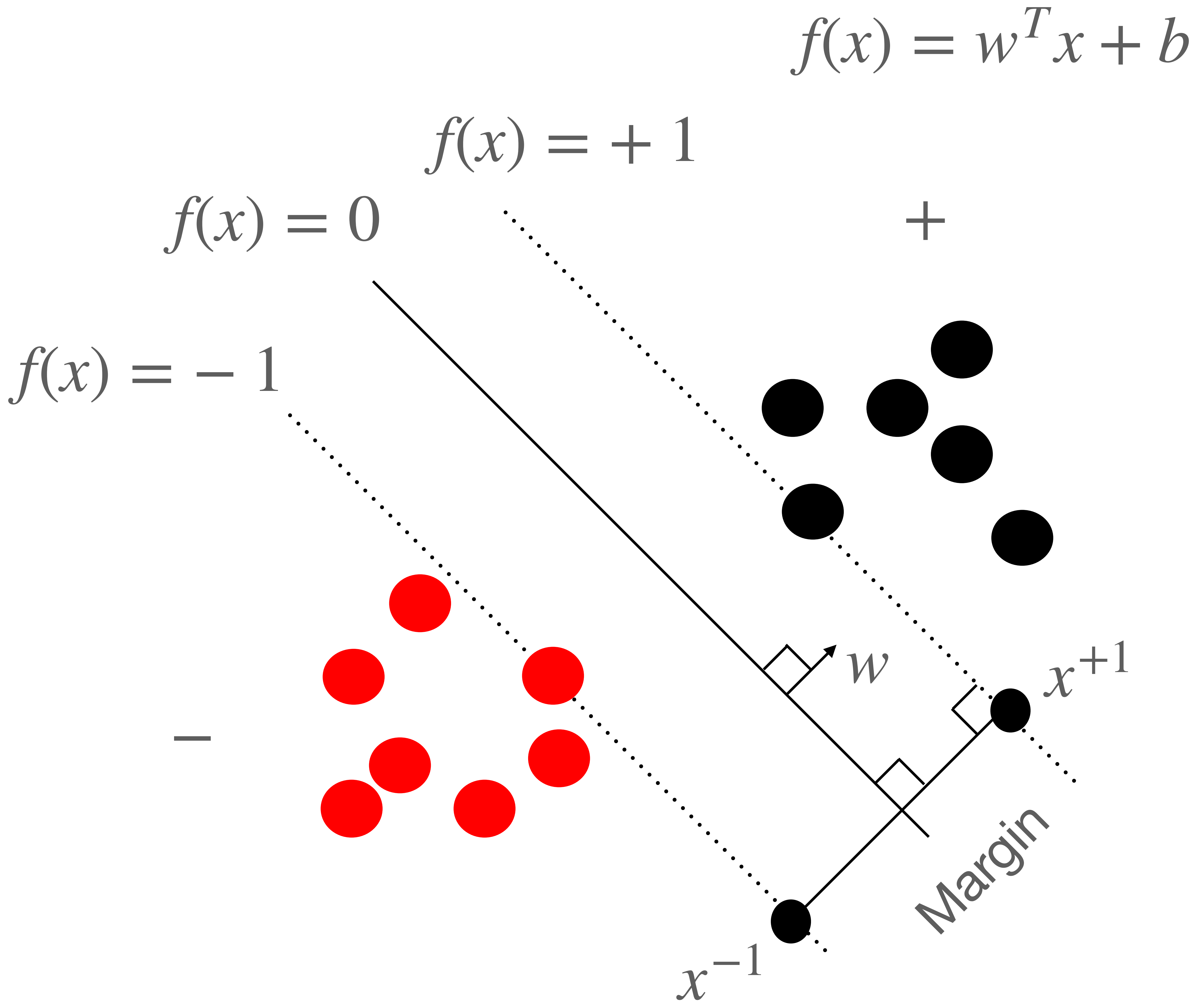
$$w^T x^{-1} + b = -1$$

$$w^T x^{+1} + b = 1$$

$$w^T (x^{+1} - x^{-1}) = 2$$

$$\|w\| \cdot \text{Margin} \cdot cos\theta = 2$$

$$\text{Margin} = \frac{2}{\|w\|}$$



SVM: Constrained optimisation problem

$$\min_{w} \frac{\|w\|^2}{2}$$

s.t

$1 - y^{(i)}(w^T x^{(i)} + b) \leq 0,\ i = 1, \cdots, n$ (data points)

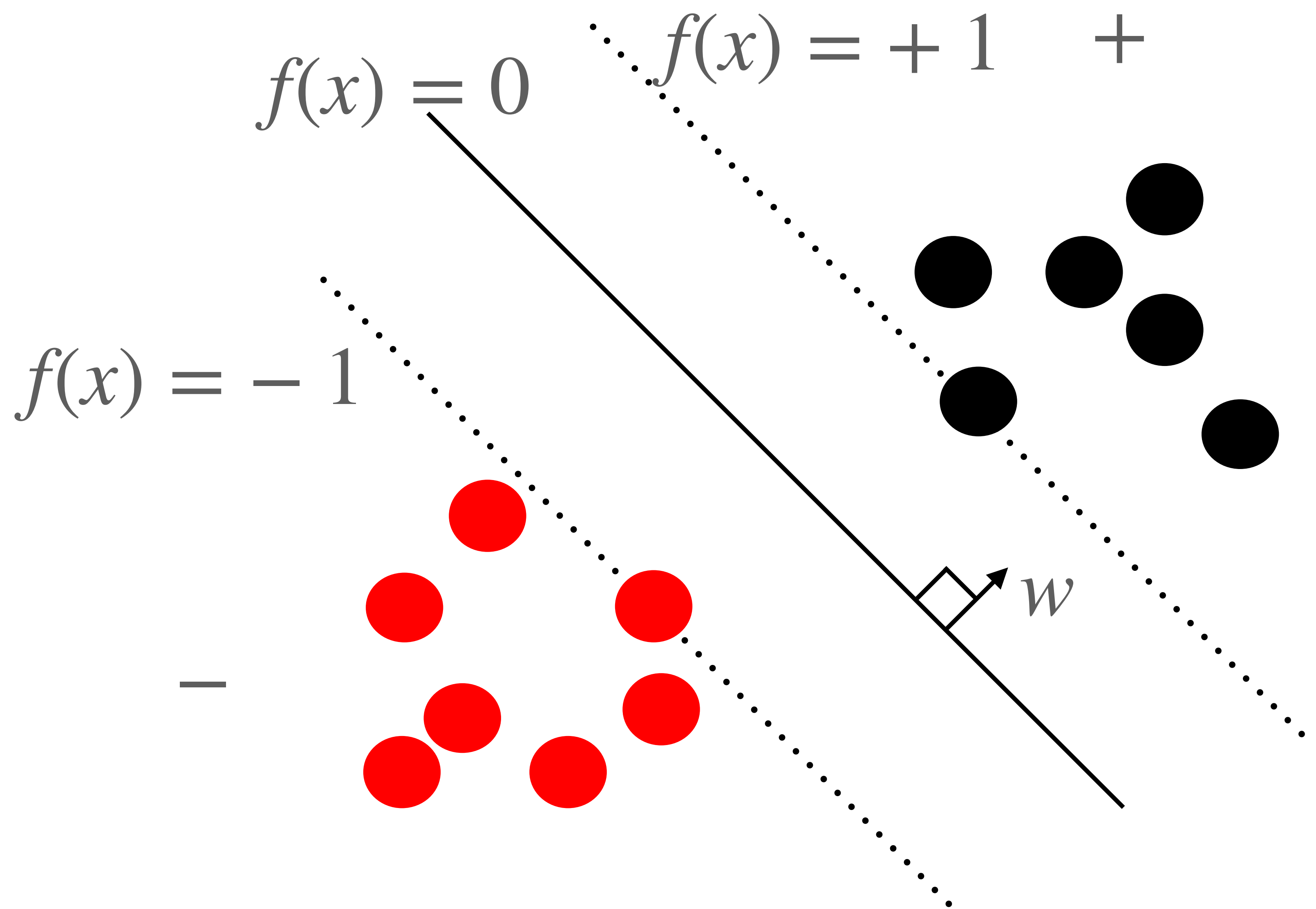
$$\max_{w} \frac{2}{\|w\|}$$

subject to

$if\ y^{(i)} = +1 : f(x^{(i)}) = w^T x^{(i)} + b \geq +1$

$if\ y^{(i)} = -1 : f(x^{(i)}) = w^T x^{(i)} + b \leq -1$

$(i = 1, \cdots, n$ data points)



Support Vector Machines

COMP90051 Statistical Machine Learning

Semester 2, 2020

Qiuhong Ke

Outline

- Margin
- Lagrange Duality
- Soft-margin SVM
- Kernels

Primal problem

$$\min_{w} \frac{\|w\|^2}{2}$$

s.t

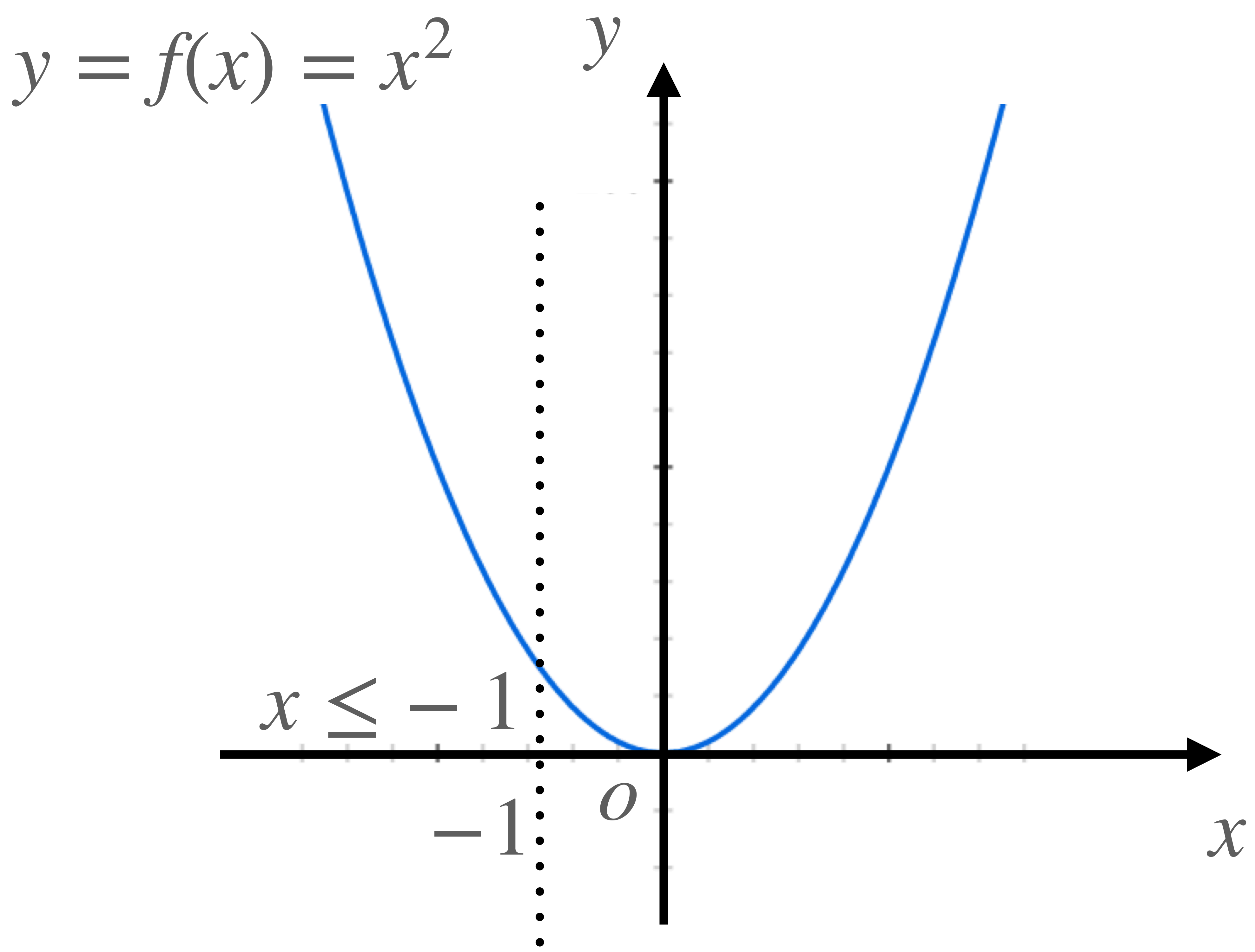
$1 - y^{(i)}(w^T x^{(i)} + b) \leq 0,\ i = 1,\cdots,n$ (data points)

Dual problem

What's the dual problem?

Why solving primal by solving dual problem?

Simple example



$$\min_{x} \; x^2$$

$$\text{s.t.} \quad x \leq -1$$

Primal problem

$$\min_{x} \; f(x)$$

s.t. $g(x) = x + 1 \leq 0$

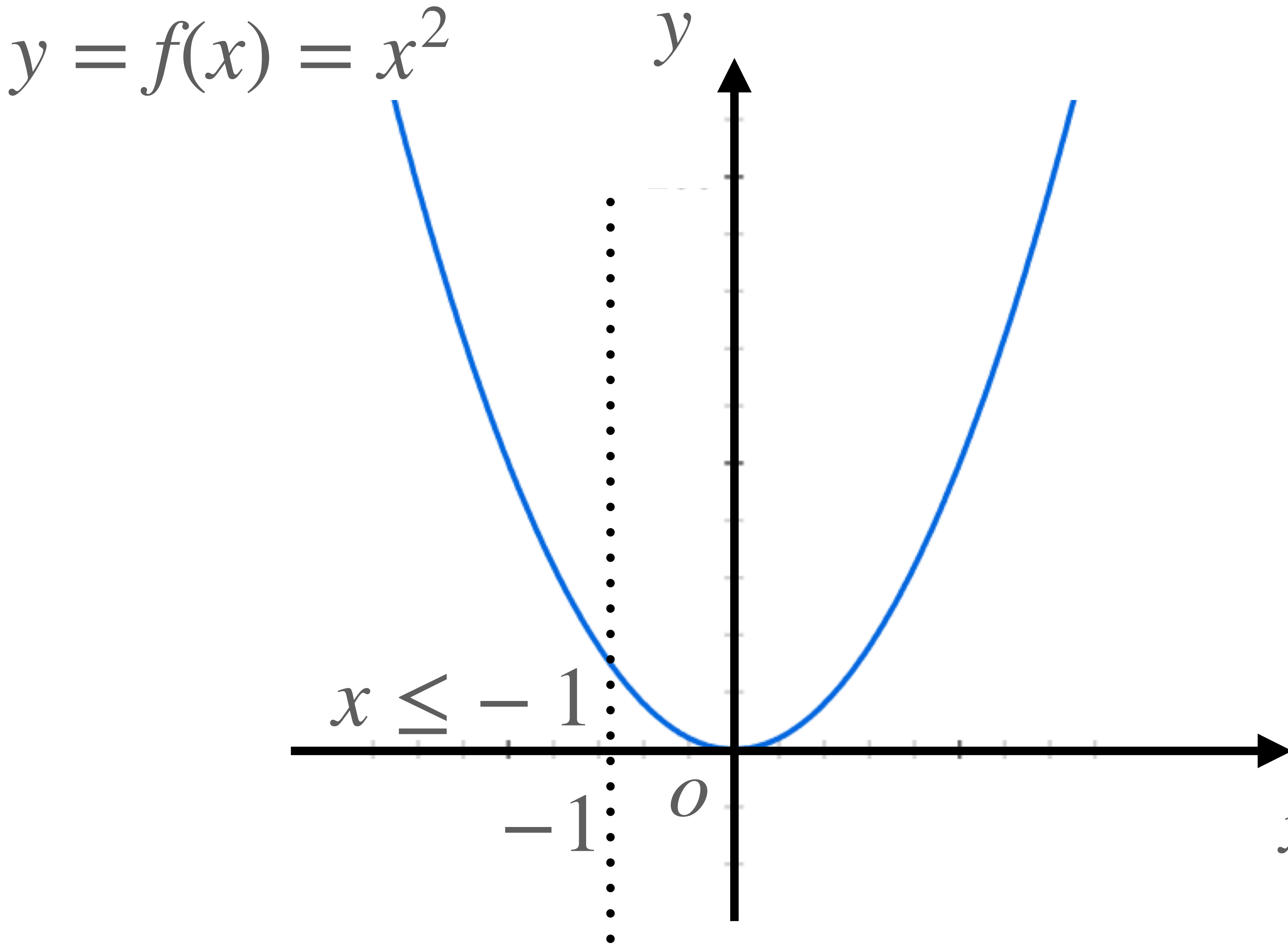
- Construct a function:

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

- Set $\lambda \geq 0$, calculate $\max_{\lambda} L(x, \lambda)$

$$g(x) > 0 : \max_{\lambda} L(x, \lambda) = \infty \text{ when } \lambda = \infty$$

$$g(x) \leq 0 : \max_{\lambda} L(x, \lambda) = f(x) \text{ when } \lambda = 0$$



$$\min_{x} \; x^2$$

s.t. $x \leq -1$

Primal problem

$$\min_{x} \ f(x)$$

$$\text{s.t.} \quad g(x) = x + 1 \leq 0$$

- Construct a function

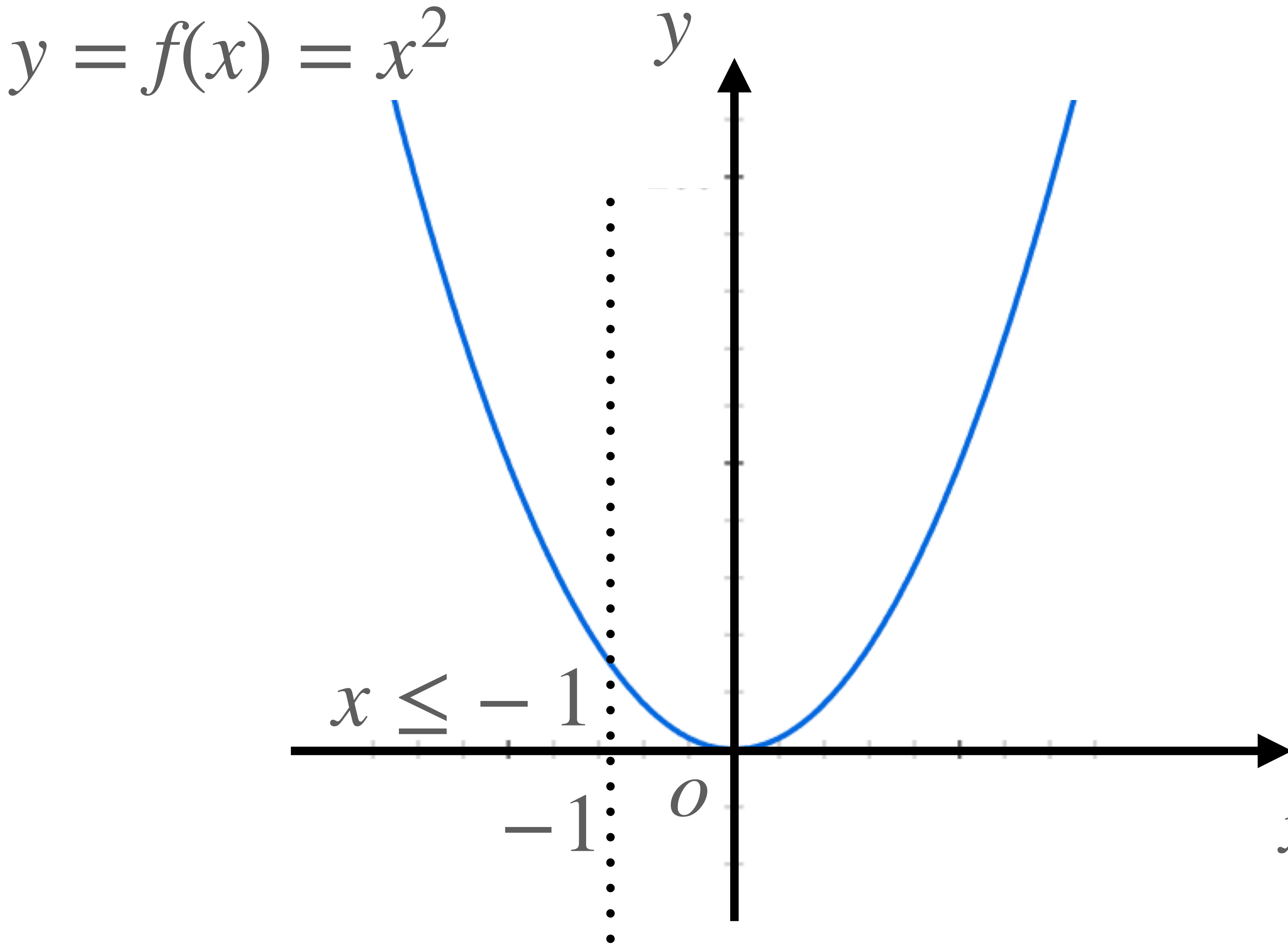
$$L(x, \lambda) = f(x) + \lambda g(x): \quad \text{Lagrangian function}$$

$$\lambda \geq 0: \quad \text{Lagrange multiplier}$$

- Primal function

$$\theta_p(x) = \max_{\lambda} L(x, \lambda) = f(x) \ if \ g(x) \leq 0$$

So: $$\min_{x} f(x) = \min_{x} \theta_p(x) = \min_{x} \max_{\lambda} L(x, \lambda)$$



$$\min_{x} \ x^2$$

$$\text{s.t.} \quad x \leq -1$$

From primal to dual problem

$$L(x,\lambda) = f(x) + \lambda g(x) \qquad \lambda \geq 0 \qquad g(x) \leq 0$$

- Primal problem:

$$\min_x f(x) = \min_x \max_\lambda \; L(x,\lambda)$$

- Dual problem:

$$\max_\lambda \min_x L(x,\lambda) = \max_\lambda \theta_d(\lambda)$$

Dual function: $\theta_d(\lambda) = \min_x \; L(x,\lambda)$

$$\theta_d(\lambda) = \min_x \; L(x,\lambda) \leq L(x,\lambda) = f(x) + \lambda g(x) \leq f(x)$$

From primal to dual problem

- Primal problem:

$$\min_x f(x) = \min_x \max L(x, \lambda)$$

- Dual problem:

$$\max_\lambda \min_x L(x, \lambda) = \max_\lambda \theta_d(\lambda)$$

Solutions:

x^* makes $f(x)$ minimum : $f(x^*) = p^*$

λ^* makes $\theta_d(\lambda)$ maximum : $\theta_d(\lambda^*) = d^*$

$$\theta_d(\lambda) = \min_x L(x, \lambda) \leq L(x, \lambda) = f(x) + \lambda g(x) \leq f(x)$$

From primal to dual problem

- Primal problem:

$$\min_x f(x) = \min_x \max_\lambda \; L(x, \lambda)$$

- Dual problem:

$$\max_\lambda \min_x L(x, \lambda) = \max_\lambda \theta_d(\lambda)$$

Solutions:

$$f(x^*) = p^* = \min_x f(x)$$

$$\theta_d(\lambda^*) = d^* = \max_\lambda \theta_d(\lambda)$$

$$\theta_d(\lambda) = \min_x \; L(x, \lambda) \le L(x, \lambda) = f(x) + \lambda g(x) \le f(x)$$

$$d^* = \theta_d(\lambda^*) = \min_x \; L(x, \lambda^*) \le L(x^*, \lambda^*) = f(x^*) + \lambda^* g(x^*) \le f(x^*) = p^*$$

Under some conditions: $d^* = p^*$?

From primal to dual problem

$$d^* = \theta_d(\lambda^*) = \min_x \; L(x, \lambda^*) \leq L(x^*, \lambda^*) = f(x^*) + \lambda^* g(x^*) \leq f(x^*) = p^*$$

$$if \min_x \; L(x, \lambda^*) == L(x^*, \lambda^*) \; and \; f(x^*) + \lambda^* g(x^*) == f(x^*)$$

$$d^* = p^*$$

KKT (Karush-Kuhn-Tucker) conditions :

$g(x) \leq 0$ (Primal feasibility)

$\lambda \geq 0$ (Dual feasibility)

$\lambda g(x) = 0$ (Complementary slackness)

$\frac{\partial L}{\partial x} = 0$ (Stationarity)

Dual problem of SVM

Primal problem $\min_{w} \dfrac{\|w\|^2}{2}$

s.t. $g_i(w, b) = 1 - y^{(i)}(w^T x^{(i)} + b) \le 0,\ i = 1, \cdots, n$ data points

(1) Lagrangian function:

$$L(w, b, \lambda) = \frac{\|w\|^2}{2} + \sum_{i=1}^{n} \lambda_i (1 - y^{(i)}(w^T x^{(i)} + b))$$

(2) dual function $\theta_d(\lambda) = \min_{w, b} L(w, b, \lambda)$:

$$\frac{\partial L}{\partial w_j} = 0 : \quad w_j = \sum_{i=1}^{n} \lambda_i y^{(i)} x_j^{(i)}$$

$$\frac{\partial L}{\partial b} = 0 : \quad \sum_{i=1}^{n} \lambda_i y^{(i)} = 0$$

Dual problem of SVM

Dual function

$$\theta_d(\lambda) = \sum_{i=1}^{n} \lambda_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{k=1}^{n} \lambda_i \lambda_k y^{(i)} y^{(k)} (x^{(i)})^T x^{(k)}$$

Dual problem

$$\max_{\lambda} \theta_d(\lambda)$$

s.t.

$\lambda_i \geq 0$ and $\sum_{i=1}^{n} \lambda_i g_i(x) = 0$

Support vectors

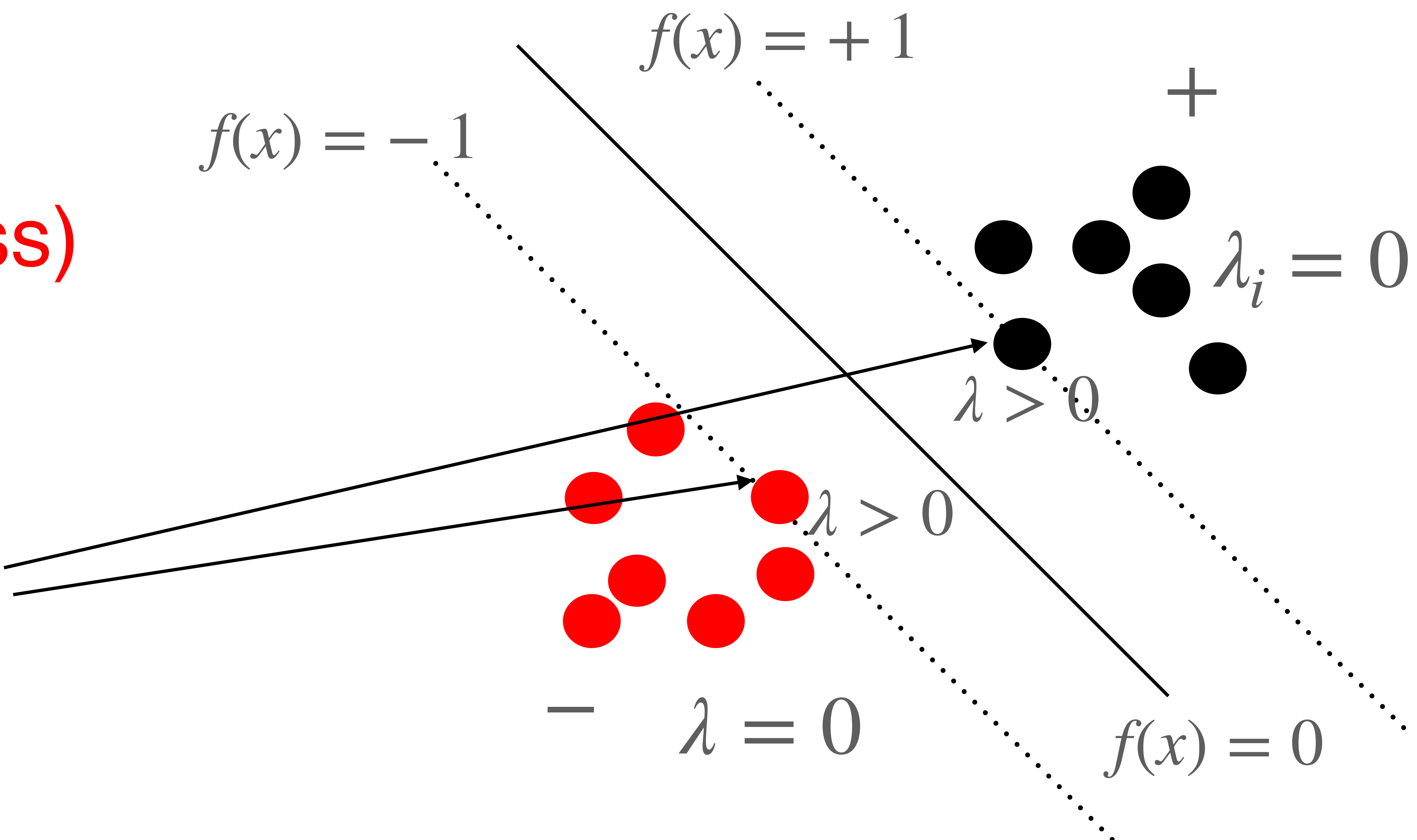
min $\frac{\|w\|^2}{2}$ s.t. $g_i(w,b) = 1 - y^{(i)}(w^T x^{(i)} + b) \le 0,\ i = 1,\cdots,n$ data points

Lagrangian function: $L(w,b,\lambda) = \frac{\|w\|^2}{2} + \sum_{i=1}^{n} \lambda_i (1 - y^{(i)}(w^T x^{(i)} + b))$

$\lambda_i \ge 0$ (Dual feasibility)

$\lambda_i g_i(w,b) = 0$ (Complementary slackness)

support vectors:



Primal vs Dual (Training)

- Primal problem: solve d+1 variables $(w_j\ and\ b)$ (d: dimension of weight vector w)

$$\min_{w} \frac{\|w\|^2}{2}$$

s.t.

$$g_i(w,b) = 1 - y^{(i)}(w^T x^{(i)} + b) \le 0,\ i = 1,\cdots,n$$ data points

- Dual problem: solve n variables (λ_i)

$$\max_{\lambda} \theta_d(\lambda) \quad \theta_d(\lambda) = \sum_{i=1}^{n} \lambda_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{k=1}^{n} \lambda_i \lambda_k y^{(i)} y^{(k)} (x^{(i)})^T x^{(k)}$$

s.t.

$\lambda_i \ge 0$ and $\sum_{i=1}^{n} \lambda_i g_i(x) = 0)$

If data size n is large, $(n \gg d)$ solving dual problem is slower than solving primal problem, and vice versa.

Primal vs Dual (Prediction)

- Primal form:

$$f(x) = w^T x + b$$

$$f(x) > 0 : \ positive\ \ class$$
$$f(x) < 0 : \ negative\ \ class$$

- Dual form:

$$w_j = \sum_{i=1}^{n} \lambda_i y^{(i)} x_j^{(i)}$$

$$f(x) = \sum_{i=1}^{n} \lambda_i y^{(i)} (x^{(i)})^T x + b$$

(b can be solved using support vectors: $f(x) = \pm 1$)

Why bother solving dual problem to solve primal problem

Training, solve:

$$\max_{\lambda} \theta_d(\lambda) \quad \theta_d(\lambda) = \sum_{i=1}^{n} \lambda_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{k=1}^{n} \lambda_i \lambda_k y^{(i)} y^{(k)} (x^{(i)})^T x^{(k)}$$

s.t.

$$\lambda_i \geq 0 \text{ and } \sum_{i=1}^{n} \lambda_i g_i(x) = 0)$$

Prediction: $f(x) = \sum_{i=1}^{n} \lambda_i y^{(i)} (x^{(i)})^T x + b$

- Use only support vectors for prediction: Efficient in prediction
- Inner product: Kernel trick can be used to efficiently handle non-linearly separable data

Support Vector Machines

COMP90051 Statistical Machine Learning

Semester 2, 2020

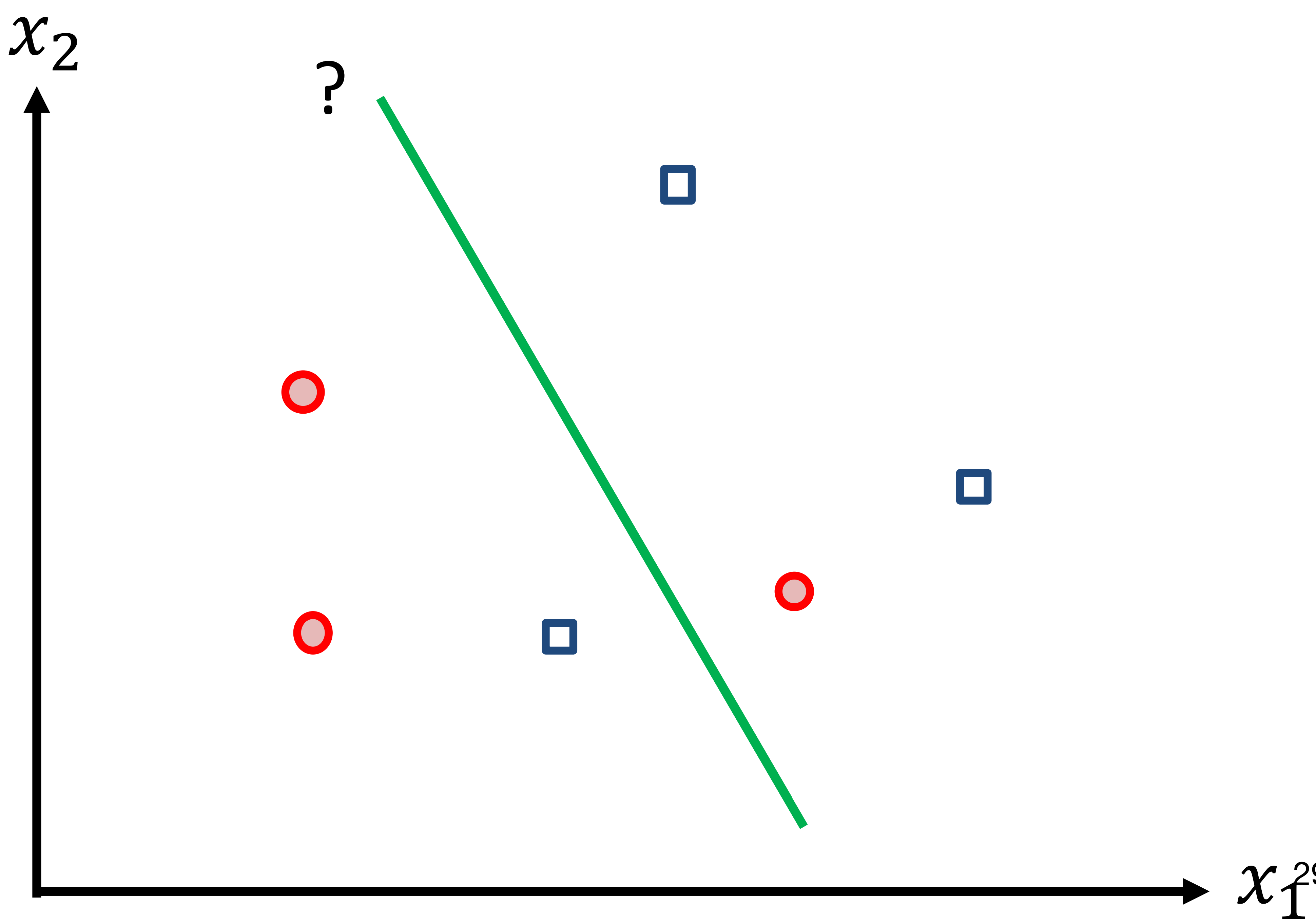
Qiuhong Ke

Outline

- Margin
- Lagrange Duality
- Soft-margin SVM
- Kernels

Data not linearly separable

- Hard-margin loss is too stringent (*hard*!)
- Real data is unlikely to be linearly separable
- If the data is not separable, hard-margin SVMs are in trouble

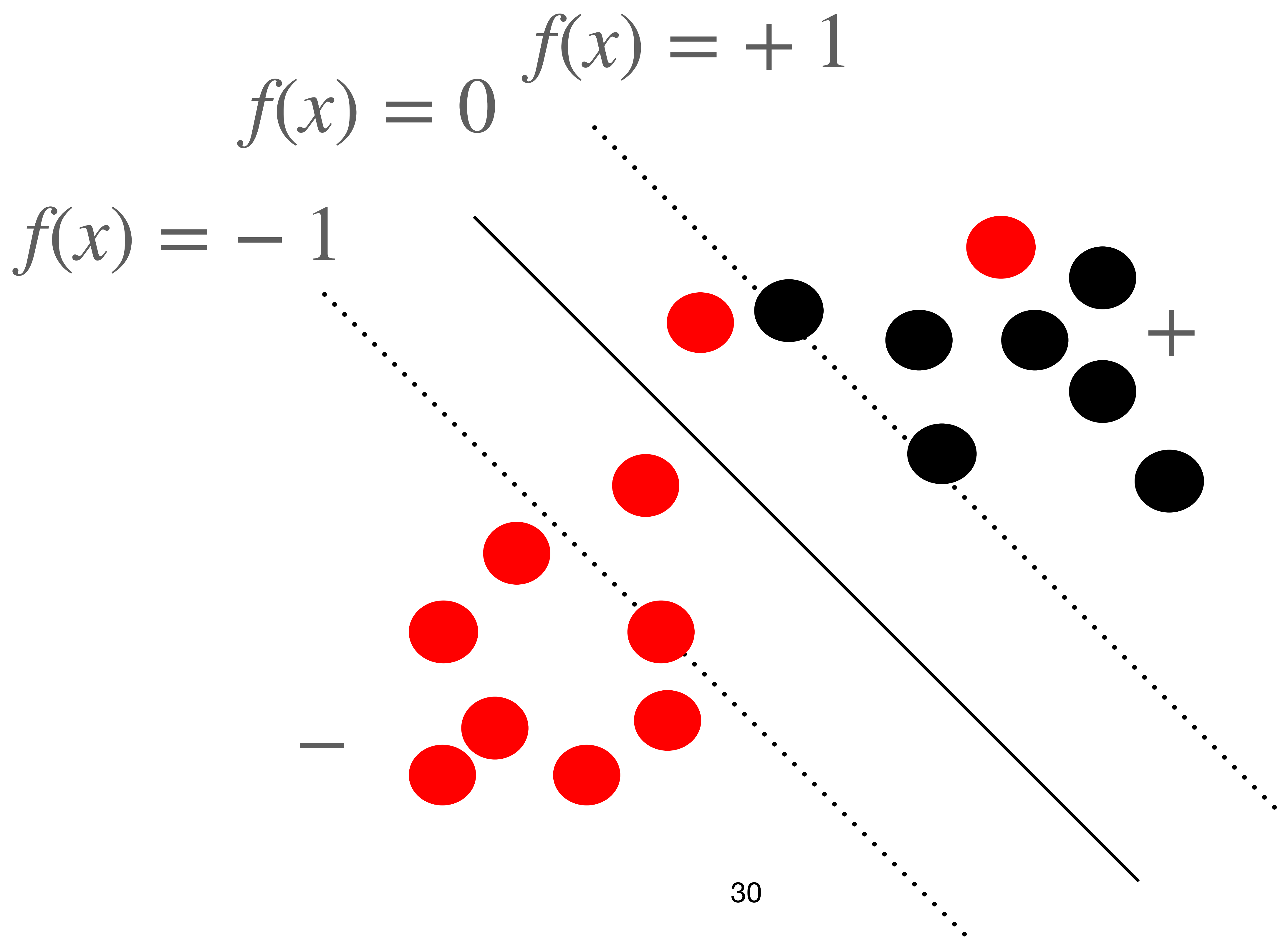


SVMs offer 3 approaches to address this problem:

1. *Relax the constraints (soft-margin)*
2. *Still use hard-margin SVM, but transform the data (kernel)*
3. *The combination of 1 and 2 ☺*

Soft-margin SVM: ‘soft' constraint

- Relax constraints to allow points to be inside the margin or even on the wrong side of the boundary



Objective of soft-margin SVM

$$\min_{w} \left(\frac{\|w\|^2}{2} + C\sum_{i=1}^{n} \xi_i\right) \quad \text{s.t.} \quad y^{(i)}(w^T x^{(i)} + b) \geq 1 - \xi_i\ , \quad \xi_i \geq 0 \quad (i = 1,\cdots,n \text{ data points})$$

Use slack variable to ‘soft’ constraint: allow violation of the constraint

$$\xi_i = \begin{cases} 0\ , \ y^{(i)}(w^T x^{(i)} + b) \geq 1, \\ 1 - y^{(i)}(w^T x^{(i)} + b)\ , \ otherwise \end{cases}$$

$$or\ \xi_i = \max(0, 1 - y^{(i)}(w^T x^{(i)} + b))$$ hinge loss

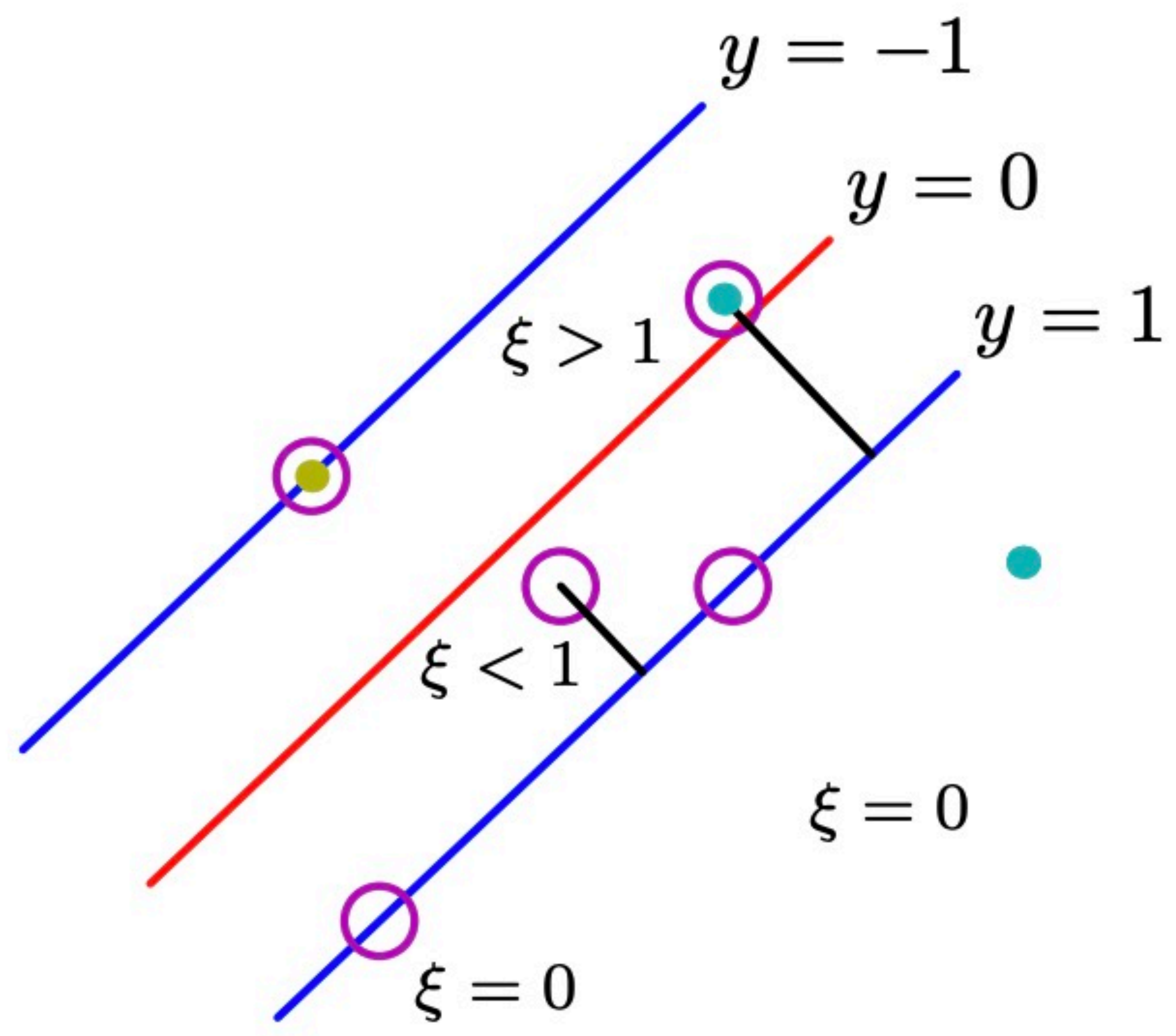


Figure 7.3 in Pattern Recognition and Machine Learning by Chris Bishop

Objective of soft-margin SVM

$$\min_{w} \left(\frac{\|w\|^2}{2} + C\sum_{i=1}^{n} \xi_i\right) \quad \text{s.t.} \quad y^{(i)}(w^T x^{(i)} + b) \geq 1 - \xi_i\ , \quad \xi_i \geq 0 \quad (i = 1, \cdots, n \text{ data points})$$

Slack penalty: $C > 0$

If C= 0: data is ignored ➡ Underfitting

If C= ∞: data has to be correctly classified ➡ Overfitting

KKT

$$L(w,b,\lambda,\beta,\xi) = \frac{\|w\|^2}{2} + C\sum_{i=1}^{n}\xi_i + \sum_{i=1}^{n}\lambda_i g_i(w,b,\xi) + \sum_{i=1}^{n}\beta_i(-\xi_i)$$

$$g_i(w,b,\xi) = 1 - \xi_i - y^{(i)}(w^T x^{(i)} + b) \le 0 \qquad -\xi_i \le 0$$

Primal feasibility: $g_i(w,b,\xi) \le 0 \quad -\xi_i \le 0$

Dual feasibility $\lambda_i \ge 0 \quad \beta_i \ge 0$

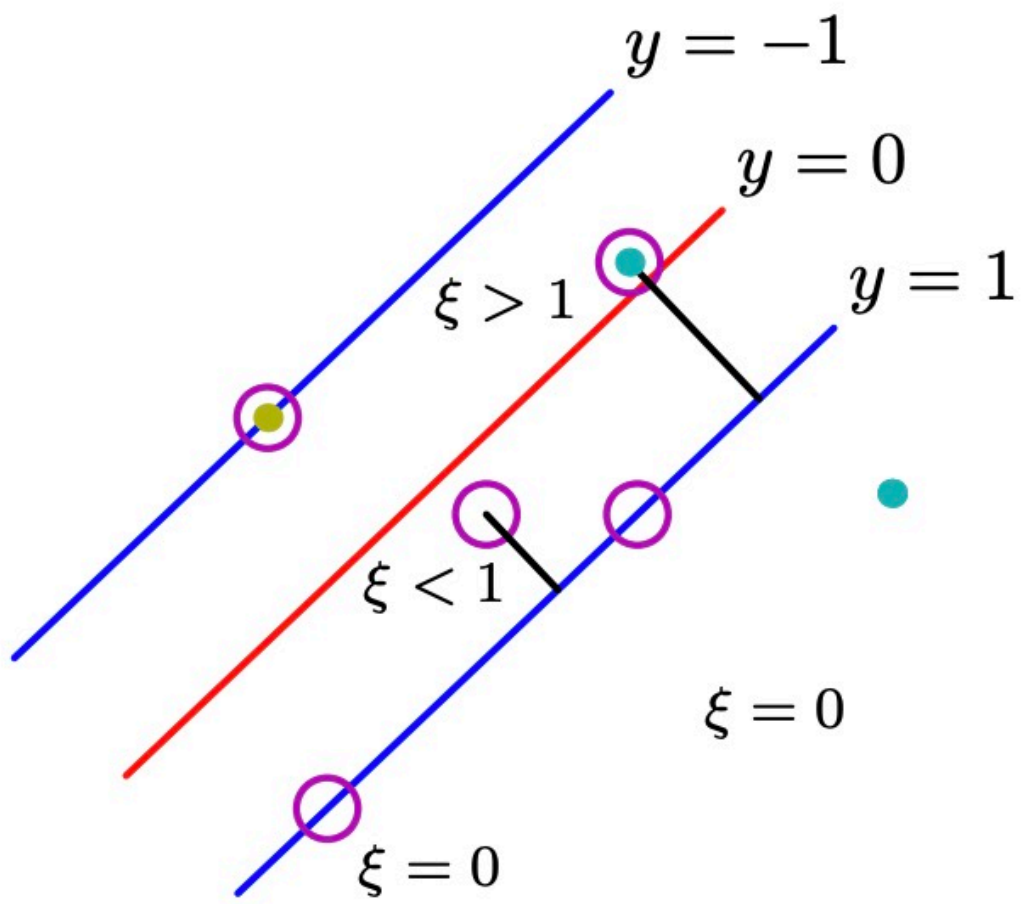
Complementary slackness $\lambda_i g_i(w,b,\xi) = 0 \quad \beta_i \xi_i = 0$

Stationarity

$$\frac{\partial L}{\partial w_j} = 0 : \; w_j = \sum_{i=1}^{n}\lambda_i y^{(i)} x_j^{(i)} \qquad \frac{\partial L}{\partial b} = 0 : \; \sum_{i=1}^{n}\lambda_i y^{(i)} = 0$$

$$\frac{\partial L}{\partial \xi_i} = 0 : \; C - \lambda_i - \beta_i = 0$$

KKT



Primal feasibility: $g_i(w,b,\xi) = 1-\xi_i - y^{(i)}(w^Tx^{(i)}+b) \le 0, \quad -\xi_i \le 0$

Dual feasibility $\lambda_i \ge 0 \quad \beta_i \ge 0$

Complementary slackness $\lambda_i g_i(w,b,\xi) = 0 \quad \beta_i \xi_i = 0$

$$C - \lambda_i - \beta_i = 0 : \quad 0 \le \lambda_i \le C$$

$$if\ \lambda_i = 0 : \quad \beta = C, \xi_i = 0 \quad y^{(i)}(w^Tx^{(i)}+b) \ge 1-\xi_i = 1$$

$$if\ \lambda_i = C : \ \beta_i = 0, -\xi_i \le 0 \quad y^{(i)}(w^Tx^{(i)}+b) = 1-\xi_i \le 1$$

$$if\ 0 < \lambda_i < C : \quad \xi_i = 0 \quad g_i(w,b,\xi) = 0 \quad y^{(i)}(w^Tx^{(i)}+b) = 1-\xi_i = 1$$

The point is a Support vector!

Support Vector Machines

COMP90051 Statistical Machine Learning

Semester 2, 2020

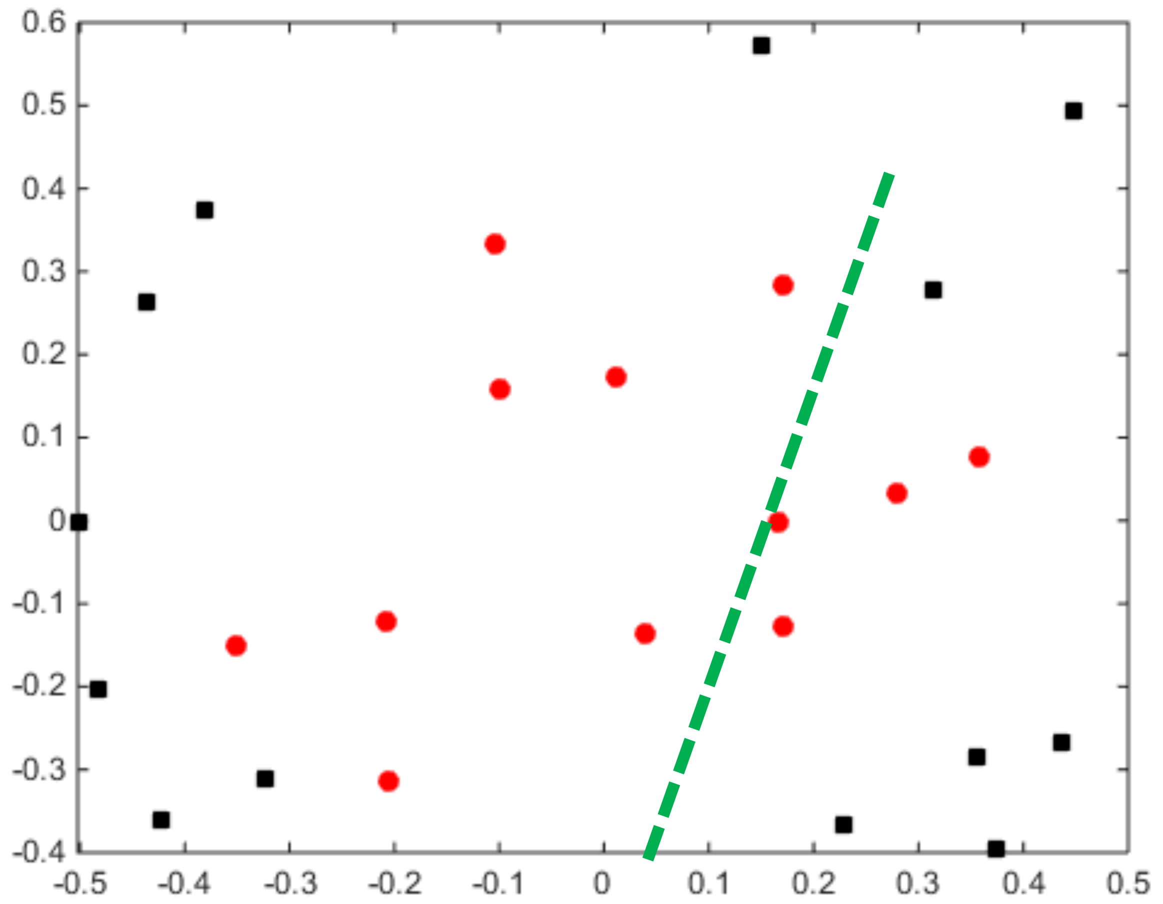
Qiuhong Ke

Outline

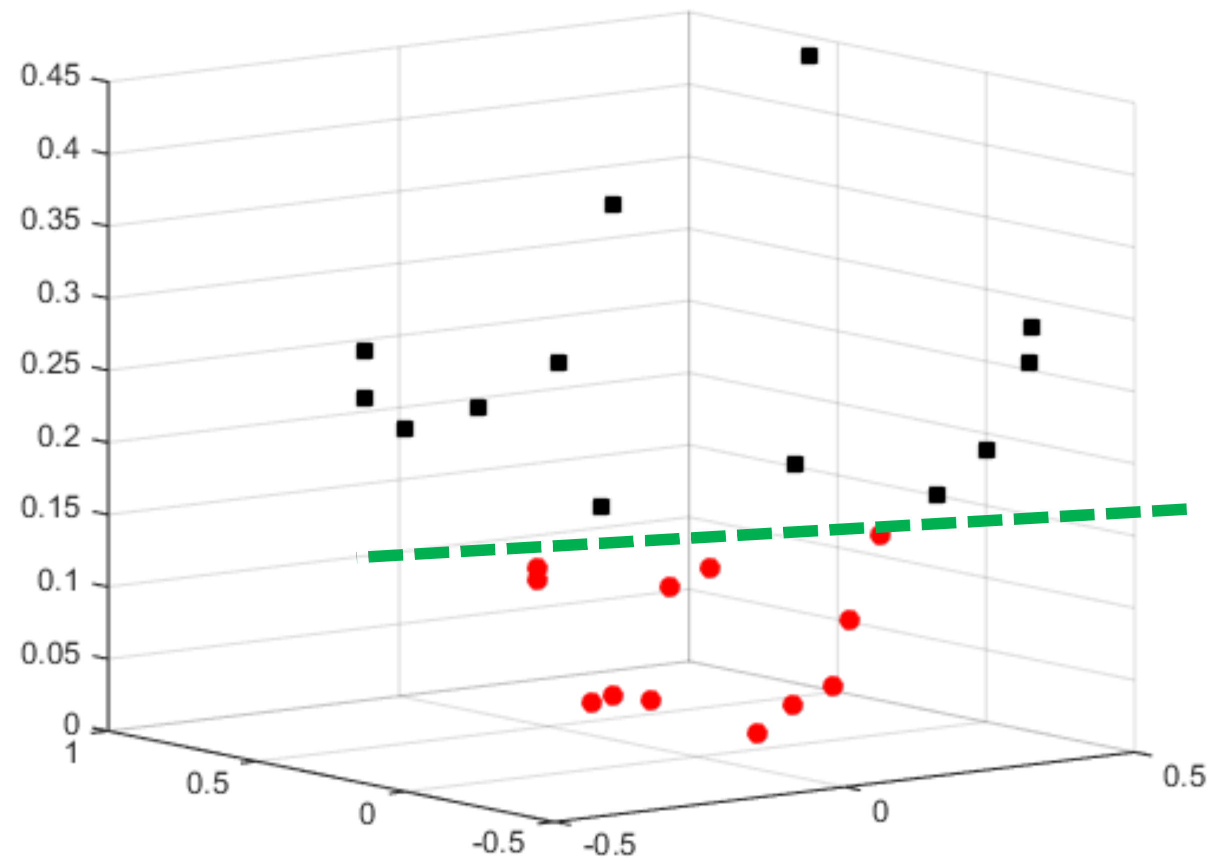
- Margin
- Lagrange Duality
- Soft-margin SVM
- Kernels

Non-linearly separable data

- Consider a binary classification problem
- Each example has features $[x_1, x_2]$
- Not linearly separable



- Now 'add' a feature $x_3 = x_1^2 + x_2^2$
- Each point is now $[x_1, x_2, x_1^2 + x_2^2]$
- Linearly separable!



Naïve workflow

- Choose/design a linear model
- Choose/design a high-dimensional transformation $\varphi(\boldsymbol{x})$
  - Hoping that after adding a lot of various features some of them will make the data linearly separable
- For each training example, and for each new instance compute $\varphi(\boldsymbol{x})$
- Train classifier/Do predictions

Hard-margin SVM in feature space

Training, solve:

$$\max_{\lambda} \theta_d(\lambda) \quad \theta_d(\lambda) = \sum_{i=1}^{n} \lambda_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{k=1}^{n} \lambda_i \lambda_k y^{(i)} y^{(k)} (x^{(i)})^T x^{(k)}$$

s.t.

$$\lambda_i \geq 0 \text{ and } \sum_{i=1}^{n} \lambda_i g_i(x) = 0$$

Prediction: $$f(x) = \sum_{i=1}^{n} \lambda_i y^{(i)} (x^{(i)})^T x + b$$

We just need the dot product!

Observation: Kernel representation

- Both parameter estimation and computing predictions depend on data only in a form of a dot product
  - In original space $\boldsymbol{u}'\boldsymbol{v} = \sum_{i=1}^{m} u_i v_i$
  - In transformed space $\varphi(\boldsymbol{u})'\varphi(\boldsymbol{v}) = \sum_{i=1}^{l} \varphi(\boldsymbol{u})_i \varphi(\boldsymbol{v})_i$

- Kernel is a function that can be expressed as a dot product in some feature space $K(\boldsymbol{u}, \boldsymbol{v}) = \varphi(\boldsymbol{u})'\varphi(\boldsymbol{v})$

Benefits:
- no need to find the mapping function.
- no need to do transformation.
- no need to do dot product.

Kernel as shortcut

- For *some* $\varphi(\boldsymbol{x})$'s, kernel is faster to compute directly than first mapping to feature space then taking dot product.
- For example, consider two vectors $\boldsymbol{u} = [u_1]$ and $\boldsymbol{v} = [v_1]$ and transformation $\varphi(\boldsymbol{x}) = [x_1^2, \sqrt{2c}x_1, c]$, some c
  - So $\varphi(\boldsymbol{u}) = \left[u_1^2, \sqrt{2c}u_1, c\right]'$ and $\varphi(\boldsymbol{v}) = \left[v_1^2, \sqrt{2c}v_1, c\right]'$
  - Then $\varphi(\boldsymbol{u})'\varphi(\boldsymbol{v}) = (u_1^2 v_1^2 + 2cu_1v_1 + c^2)$
- This can be alternatively computed directly as

$$\varphi(\boldsymbol{u})'\varphi(\boldsymbol{v}) = (u_1v_1 + c)^2$$

  - Here $K(\boldsymbol{u}, \boldsymbol{v}) = (u_1v_1 + c)^2$ is the corresponding kernel

Hard-margin SVM in feature space

Training: solve

$$\max_{\lambda} L(\lambda) \qquad L(\lambda) = \sum_{i=1}^{n} \lambda_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{k=1}^{n} \lambda_i \lambda_k y^{(i)} y^{(k)} \overset{K(x^{(i)}, x^{(k)})}{(\cancel{\varphi(x^{(i)}))^T \varphi(x^{(k)}})}$$

Making predictions:

$$f(x) = w^T x + b = \sum_{i=1}^{n} \lambda_i y^{(i)} \overset{K(x^{(i)}, x)}{\cancel{(\varphi(x^{(i)}))^T \varphi(x)}} + b$$

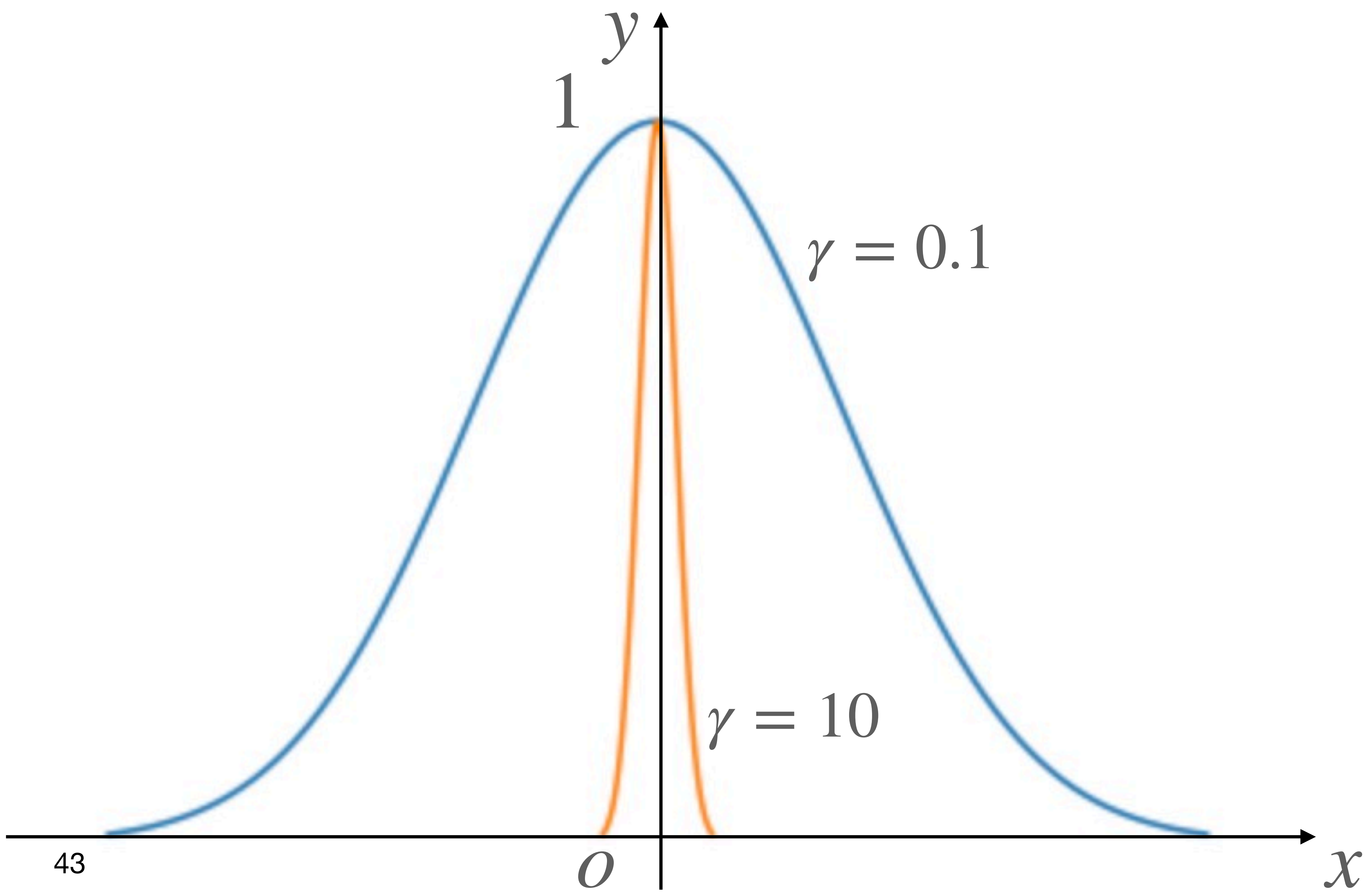
Radial Basis Function (RBF) kernel

$$K(u, v) = \exp(-\gamma \|u - v\|^2)$$

γ *is too small : underfitting*

γ *is too large : overfitting*

$$y = \exp(-\gamma x^2) = exp(-\frac{x^2}{2\sigma^2})$$



Identify new kernels

Mercer's theorem：

Consider a finite sequences of vectors $x_1, \cdots, x_n$

Construct n×n matrix A (Gram matrix) of pairwise values

$K(x_i, x_j)$ is a valid kernel if this matrix is positive semi-definite, and this holds for all possible sequences

$$A = \begin{bmatrix} K(x_1, x_1) & K(x_1, x_2) & \dots & K(x_1, x_n) \\ K(x_2, x_1) & K(x_2, x_2) & \dots & K(x_2, x_n) \\ \vdots & \vdots & \vdots & \vdots \\ K(x_n, x_1) & K(x_n, x_2) & \dots & K(x_n, x_n) \end{bmatrix}$$

Identify new kernels

Positive semi-definite matrix: a square symmetric matrix satisfies $v^T A v \geq 0$
$v \in \mathbb{R}^{n \times 1}$ any non-zero vector (column), $A \in \mathbb{R}^{n \times n}$, $A = A^T$

$$\mathrm{A} = \begin{bmatrix} K(x_1, x_1) & K(x_1, x_2) & \dots & K(x_1, x_n) \\ K(x_2, x_1) & K(x_2, x_2) & \dots & K(x_2, x_n) \\ \vdots & \vdots & \vdots & \vdots \\ K(x_n, x_1) & K(x_n, x_2) & \dots & K(x_n, x_n) \end{bmatrix}$$

Identify new kernels

Let K_1($\boldsymbol{u}$,$\boldsymbol{v}$), K_2($\boldsymbol{u}$,$\boldsymbol{v}$) be kernels, c>0 be a constant, and $f(\boldsymbol{x})$ be a real-valued function. Then each of the following is also a kernel:

1) $K(\boldsymbol{u},\boldsymbol{v})$= K_1($\boldsymbol{u}$,$\boldsymbol{v}$)+ K_2($\boldsymbol{u}$,$\boldsymbol{v}$)
2) $K(\boldsymbol{u},\boldsymbol{v})$=$c$ K_1($\boldsymbol{u}$,$\boldsymbol{v}$)
3) $K(\boldsymbol{u},\boldsymbol{v})$=$f(\boldsymbol{u})$ K_1($\boldsymbol{u}$,$\boldsymbol{v}$)$f(\boldsymbol{v})$

Summary

- What are the objective and constraints of hard-margin, soft-margin SVM
- What are KKT conditions?
- What are support vectors?
- What are Slack variables & slack penalty of soft-margin SVM?
- What is Kernel?
- How do parameters γ, C influence performance of SVM?
- How to identify new kernels?