

Learning Kernels -Tutorial

Part I: Introduction to Kernel Methods.

Corinna Cortes

Google Research

corinna@google.com

Mehryar Mohri

Courant Institute &

Google Research

mohri@cims.nyu.edu

Afshin Rostami

UC Berkeley

arostami@eecs.berkeley.edu

Outline

- Part I: Introduction to kernel methods.
- Part II: Learning kernel algorithms.
- Part III: Theoretical guarantees.
- Part IV: Software tools.

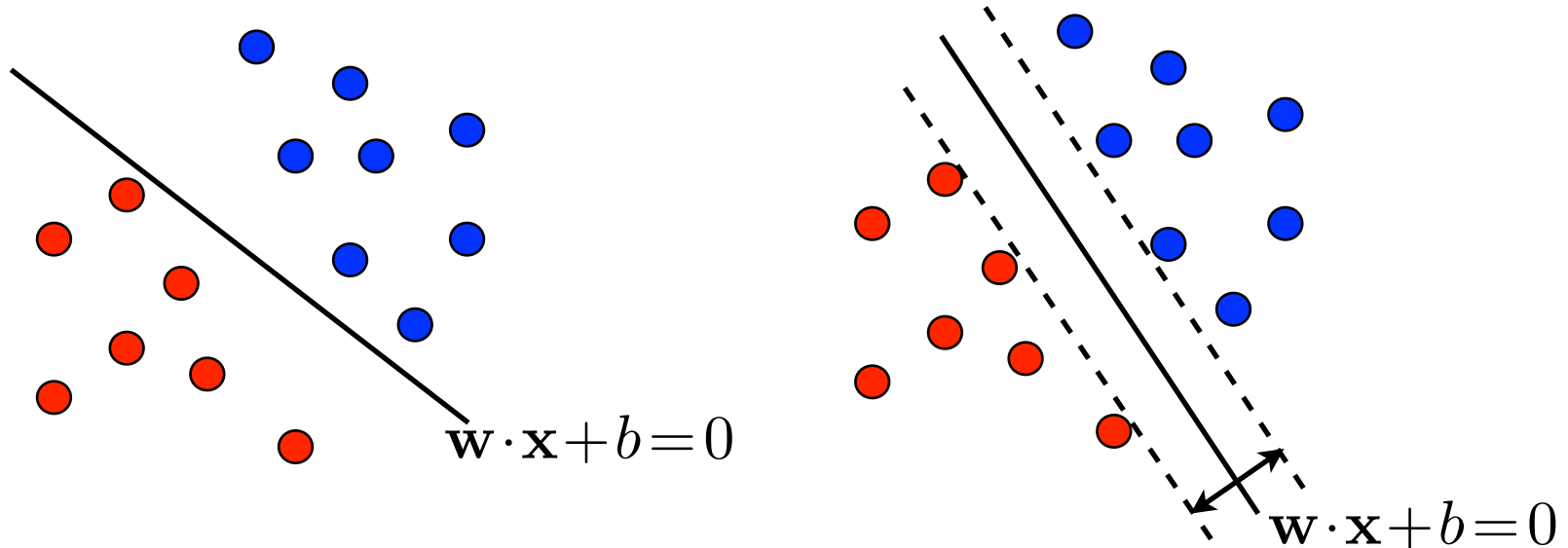
Binary Classification Problem

- **Training data:** sample drawn i.i.d. from set $X \subseteq \mathbb{R}^N$ according to some distribution D ,

$$S = ((x_1, y_1), \dots, (x_m, y_m)) \in X \times \{-1, +1\}.$$

- **Problem:** find hypothesis $h: X \mapsto \{-1, +1\}$ in H (classifier) with small generalization error $R_D(h)$.
- **Linear classification:**
 - Hypotheses based on hyperplanes.
 - Linear separation in high-dimensional space.

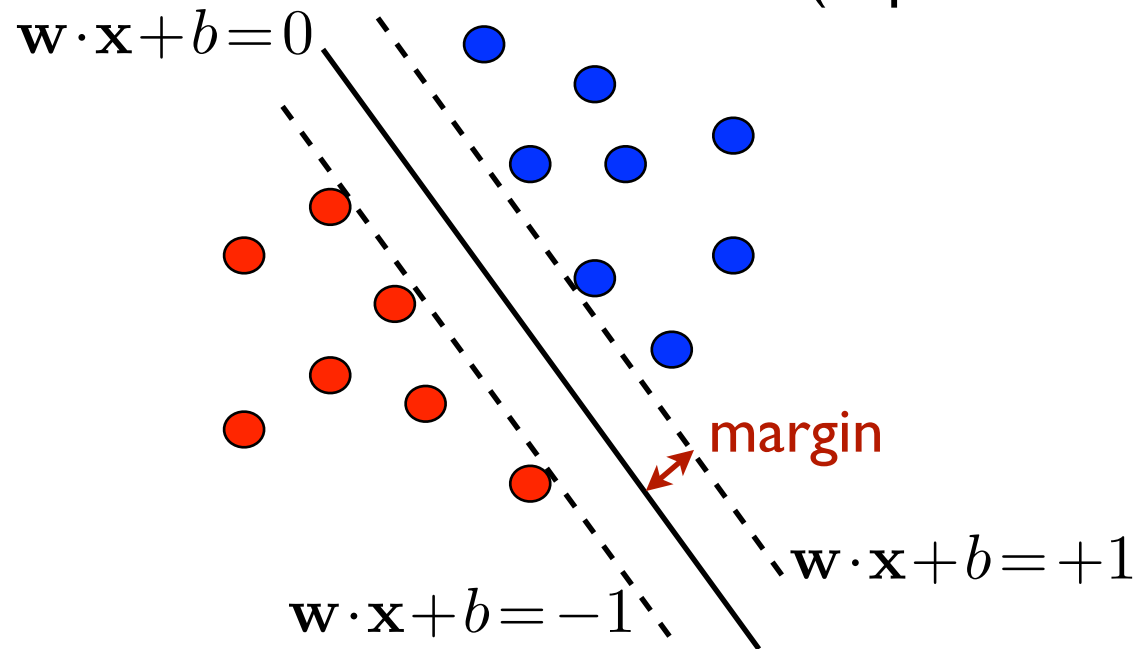
Linear Separation



■ **Classifiers:** $H = \{\mathbf{x} \mapsto \text{sgn}(\mathbf{w} \cdot \mathbf{x} + b) : \mathbf{w} \in \mathbb{R}^N, b \in \mathbb{R}\}.$

Optimal Hyperplane: Max. Margin

(Vapnik and Chervonenkis, 1964)



■ **Canonical hyperplane:** w and b chosen such that for closest points $|w \cdot x + b| = 1$.

■ **Margin:** $\rho = \min_{x \in S} \frac{|w \cdot x + b|}{\|w\|} = \frac{1}{\|w\|}$.

Optimization Problem

■ Constrained optimization:

$$\min_{\mathbf{w}, b} \quad \frac{1}{2} \|\mathbf{w}\|^2$$

subject to $y_i(\mathbf{w} \cdot \mathbf{x}_i + b) \geq 1, i \in [1, m].$

■ Properties:

- Convex optimization (strictly convex).
- Unique solution for linearly separable sample.

Support Vector Machines

(Cortes and Vapnik, 1995)

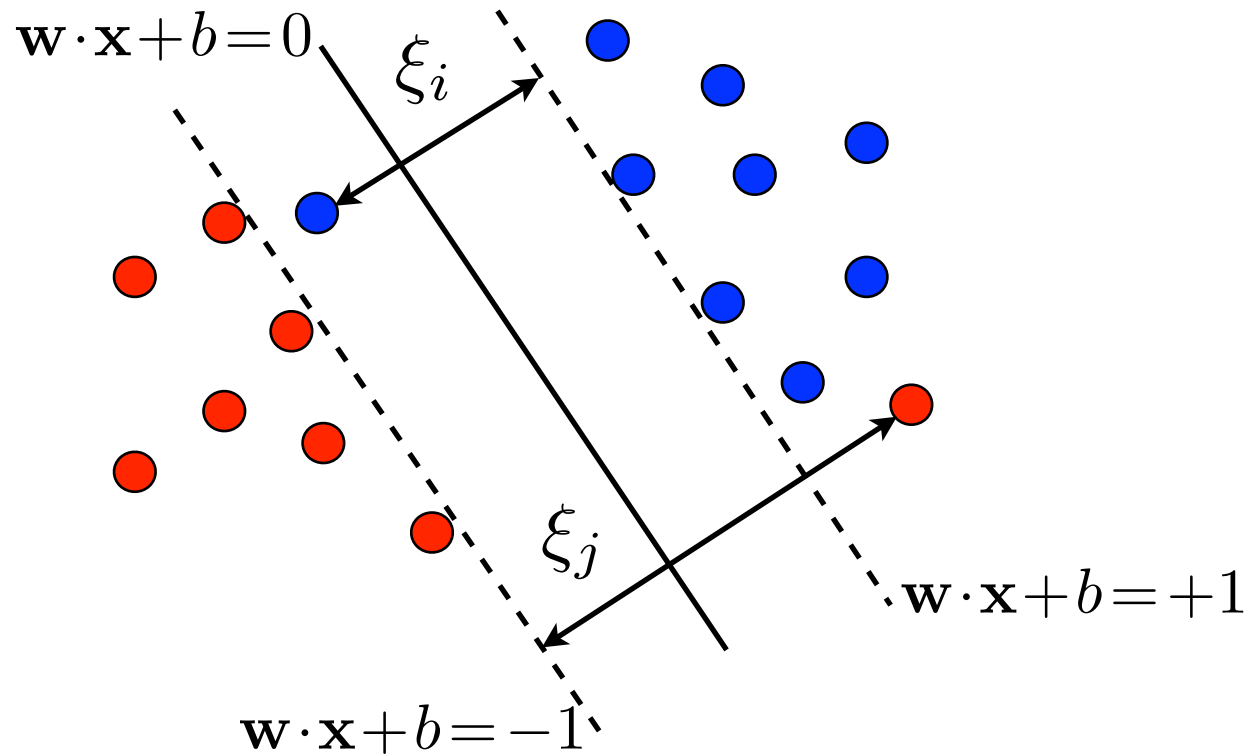
- **Problem:** data often not linearly separable in practice. For any hyperplane, there exists \mathbf{x}_i such that

$$y_i [\mathbf{w} \cdot \mathbf{x}_i + b] \not\geq 1.$$

- **Idea:** relax constraints using **slack variables** $\xi_i \geq 0$

$$y_i [\mathbf{w} \cdot \mathbf{x}_i + b] \geq 1 - \xi_i.$$

Soft-Margin Hyperplanes



- **Support vectors:** points along the margin or outliers.
- **Soft margin:** $\rho = 1/\|\mathbf{w}\|$.

Optimization Problem

(Cortes and Vapnik, 1995)

■ Constrained optimization:

$$\min_{\mathbf{w}, b, \xi} \quad \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^m \xi_i$$

subject to $y_i(\mathbf{w} \cdot \mathbf{x}_i + b) \geq 1 - \xi_i \wedge \xi_i \geq 0, i \in [1, m]$.

■ Properties:

- $C \geq 0$ trade-off parameter.
- Convex optimization (strictly convex).
- Unique solution.

Dual Optimization Problem

■ Constrained optimization:

$$\max_{\alpha} \sum_{i=1}^m \alpha_i - \frac{1}{2} \sum_{i,j=1}^m \alpha_i \alpha_j y_i y_j (\mathbf{x}_i \cdot \mathbf{x}_j)$$

$$\text{subject to: } 0 \leq \alpha_i \leq C \wedge \sum_{i=1}^m \alpha_i y_i = 0, i \in [1, m].$$

■ Solution:

$$h(x) = \text{sgn}\left(\sum_{i=1}^m \alpha_i y_i (\mathbf{x}_i \cdot \mathbf{x}) + b\right),$$

$$\text{with } b = y_i - \sum_{j=1}^m \alpha_j y_j (\mathbf{x}_j \cdot \mathbf{x}_i) \text{ for any } \mathbf{x}_i \text{ with } 0 < \alpha_i < C.$$

Kernel Methods

■ Idea:

- Define $K : X \times X \rightarrow \mathbb{R}$, called **kernel**, such that:

$$\Phi(x) \cdot \Phi(y) = K(x, y).$$

- K often interpreted as a similarity measure.

■ Benefits:

- Efficiency: K is often more efficient to compute than Φ and the dot product.
- Flexibility: K can be chosen arbitrarily so long as the existence of Φ is guaranteed (Mercer's condition).

Example - Polynomial Kernels

■ Definition:

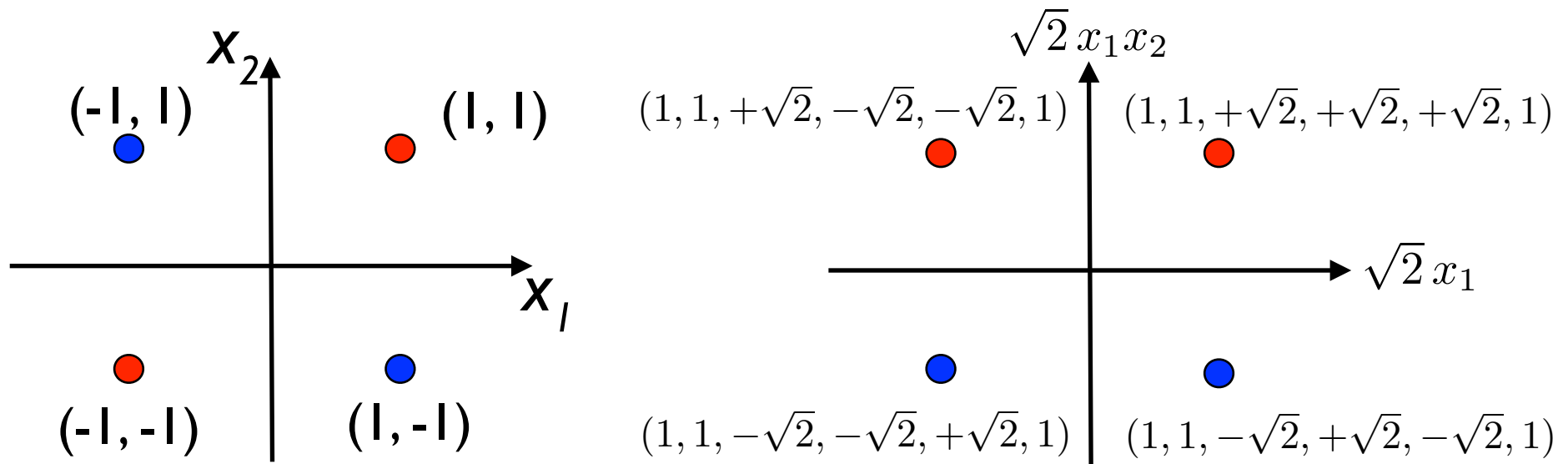
$$\forall x, y \in \mathbb{R}^N, \quad K(x, y) = (x \cdot y + c)^d, \quad c > 0.$$

■ Example: for $N=2$ and $d=2$,

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

XOR Problem

- Use second-degree polynomial kernel with $c = 1$:



Linearly non-separable

Linearly separable by
 $x_1 x_2 = 0$.

Other Standard PDS Kernels

■ Gaussian kernels:

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

■ Sigmoid Kernels:

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

Consequence: SVMs with PDS Kernels

(Boser, Guyon, and Vapnik, 1992)

■ Constrained optimization:

$$\max_{\alpha} \sum_{i=1}^m \alpha_i - \frac{1}{2} \sum_{i,j=1}^m \alpha_i \alpha_j y_i y_j K(x_i, x_j)$$

$$\text{subject to: } 0 \leq \alpha_i \leq C \wedge \sum_{i=1}^m \alpha_i y_i = 0, i \in [1, m].$$

■ Solution:

$$h(x) = \text{sgn}\left(\sum_{i=1}^m \alpha_i y_i K(x_i, x) + b\right),$$

$$\text{with } b = y_i - \sum_{j=1}^m \alpha_j y_j K(x_j, x_i) \text{ for any } x_i \text{ with } 0 < \alpha_i < C.$$

SVMs with PDS Kernels

■ Constrained optimization:

$$\begin{aligned} & \max_{\alpha} 2 \alpha^{\top} \mathbf{1} - \alpha^{\top} \mathbf{Y}^{\top} \mathbf{K} \mathbf{Y} \alpha \\ & \text{subject to: } \mathbf{0} \leq \alpha \leq \mathbf{C} \wedge \alpha^{\top} \mathbf{y} = 0. \end{aligned}$$

■ Solution:

$$h = \text{sgn}\left(\sum_{i=1}^m \alpha_i y_i K(x_i, \cdot) + b\right),$$

with $b = y_i - (\alpha \circ \mathbf{y})^{\top} \mathbf{K} \mathbf{e}_i$ for any x_i with $0 < \alpha_i < C$.

Regression Problem

- **Training data:** sample drawn i.i.d. from set X according to some distribution D ,

$$S = ((x_1, y_1), \dots, (x_m, y_m)) \in X \times Y,$$

with $Y \subseteq \mathbb{R}$ is a measurable subset.

- **Loss function:** $L: Y \times Y \rightarrow \mathbb{R}_+$ a measure of closeness, typically $L(y, y') = (y' - y)^2$ or $L(y, y') = |y' - y|^p$ for some $p \geq 1$.

- **Problem:** find hypothesis $h: X \rightarrow \mathbb{R}$ in H with small generalization error with respect to target f

$$R_D(h) = \mathbb{E}_{x \sim D} [L(h(x), f(x))] .$$

Kernel Ridge Regression

(Saunders et al., 1998)

■ Optimization problem:

$$\max_{\alpha \in \mathbb{R}^m} -\lambda \alpha^\top \alpha + 2\alpha^\top \mathbf{y} - \alpha^\top \mathbf{K} \alpha$$

or $\max_{\alpha \in \mathbb{R}^m} -\alpha^\top (\mathbf{K} + \lambda \mathbf{I}) \alpha + 2\alpha^\top \mathbf{y}.$

■ Solution:

$$h(x) = \sum_{i=1}^m \alpha_i K(x_i, x),$$

with $\alpha = (\mathbf{K} + \lambda \mathbf{I})^{-1} \mathbf{y}.$

Questions

- How should the user choose the kernel?
 - problem similar to that of selecting features for other learning algorithms.
 - poor choice → learning made very difficult.
 - good choice → even poor learners could succeed.
- The requirement from the user is thus critical.
 - can this requirement be lessened?
 - is a more automatic selection of features possible?