Erasmus School of Economics

Machine Learning

FEM31002

Support Vector Machines

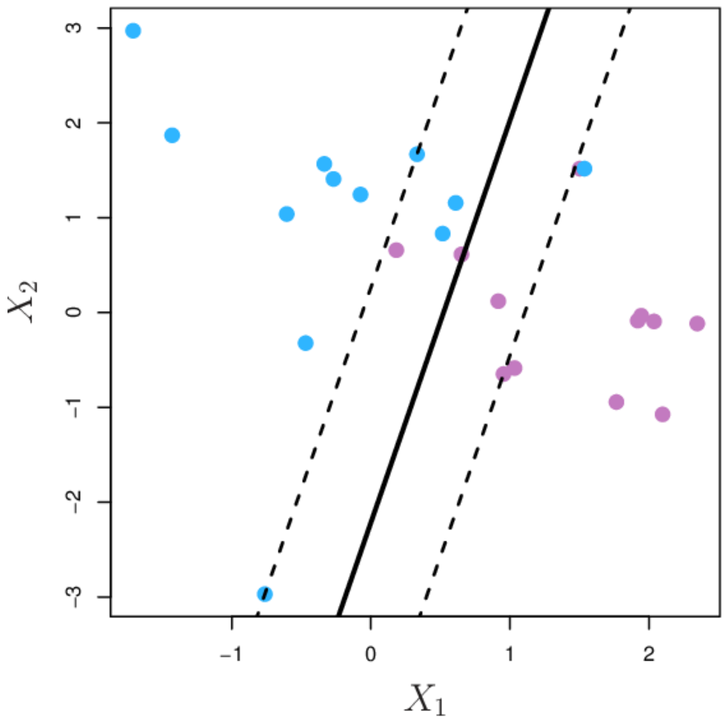
Part 2

Ilker Birbil

birbil@ese.eur.nl

ERASMUS UNIVERSITEIT ROTTERDAM

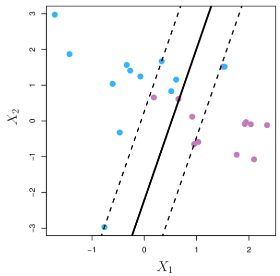
Not Perfectly Separable Data

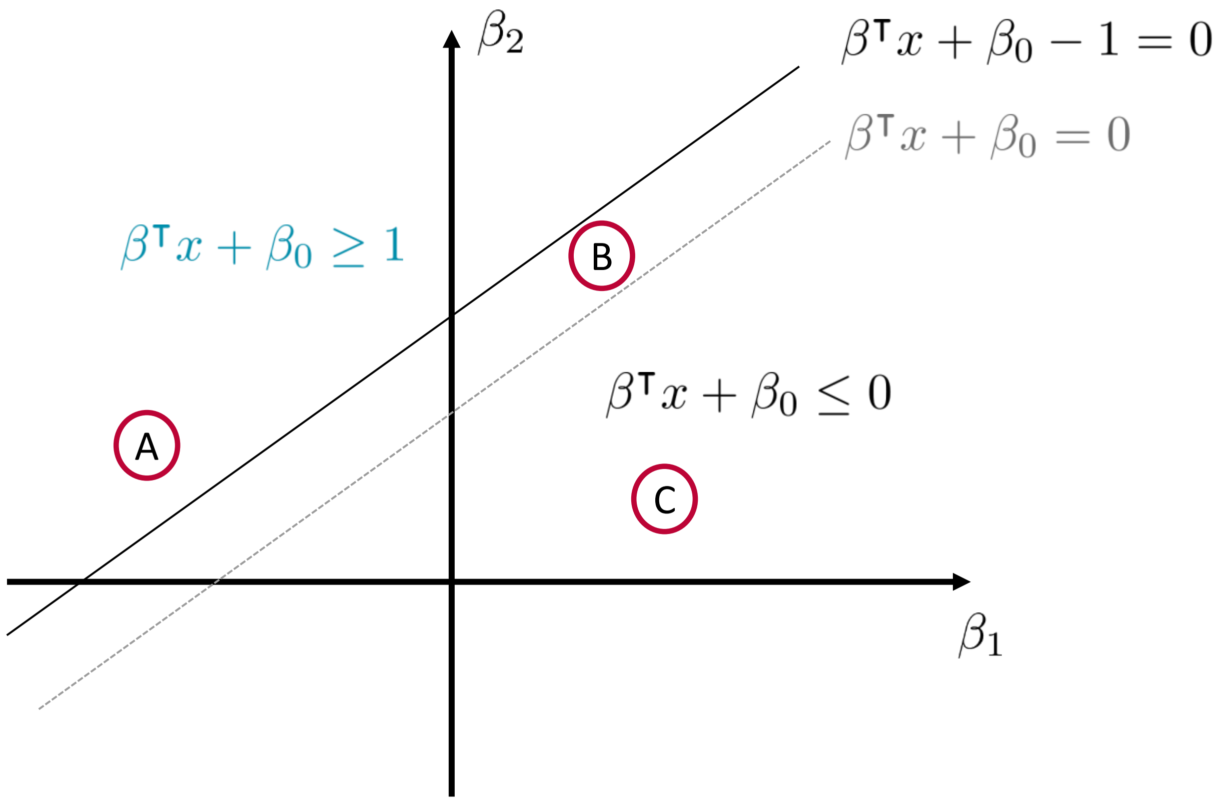


$$
\begin{array}{ll}
\text{minimize} & \frac{1}{2}\beta^\intercal \beta \\
\text{subject to} & y_i(\beta^\intercal x_i + \beta_0) \geq 1, \quad i = 1, 2, \ldots, n
\end{array}
$$

No feasible solution!

Not Perfectly Separable Data





$$(x_i, y_i)$$

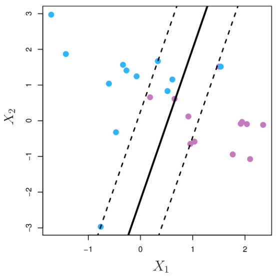
$$y_i = 1$$

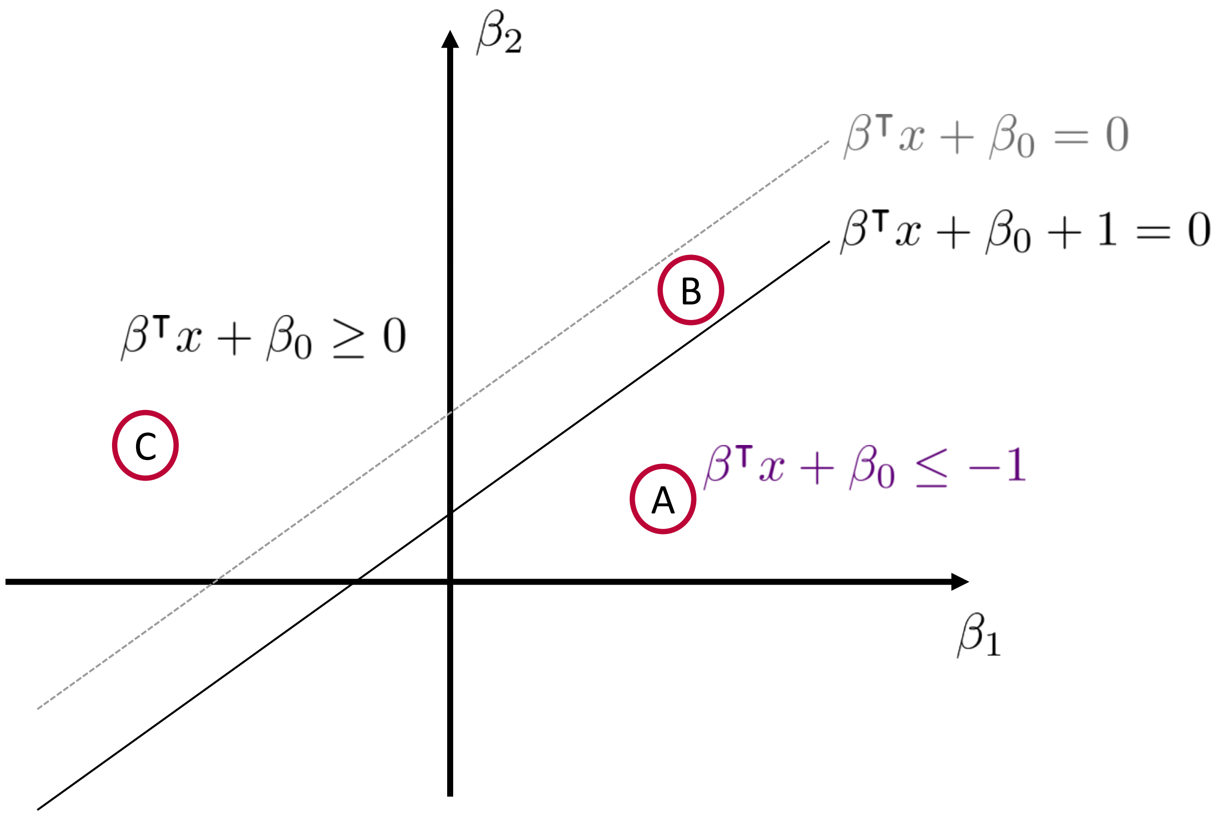
A $\quad y_i(\beta^\intercal x_i + \beta_0) \geq 1 - \varepsilon, \quad \varepsilon = 0$

B $\quad y_i(\beta^\intercal x_i + \beta_0) \geq 1 - \varepsilon, \quad 0 < \varepsilon < 1$

C $\quad y_i(\beta^\intercal x_i + \beta_0) \geq 1 - \varepsilon, \quad \varepsilon \geq 1$

Not Perfectly Separable Data





$$(x_i, y_i)$$

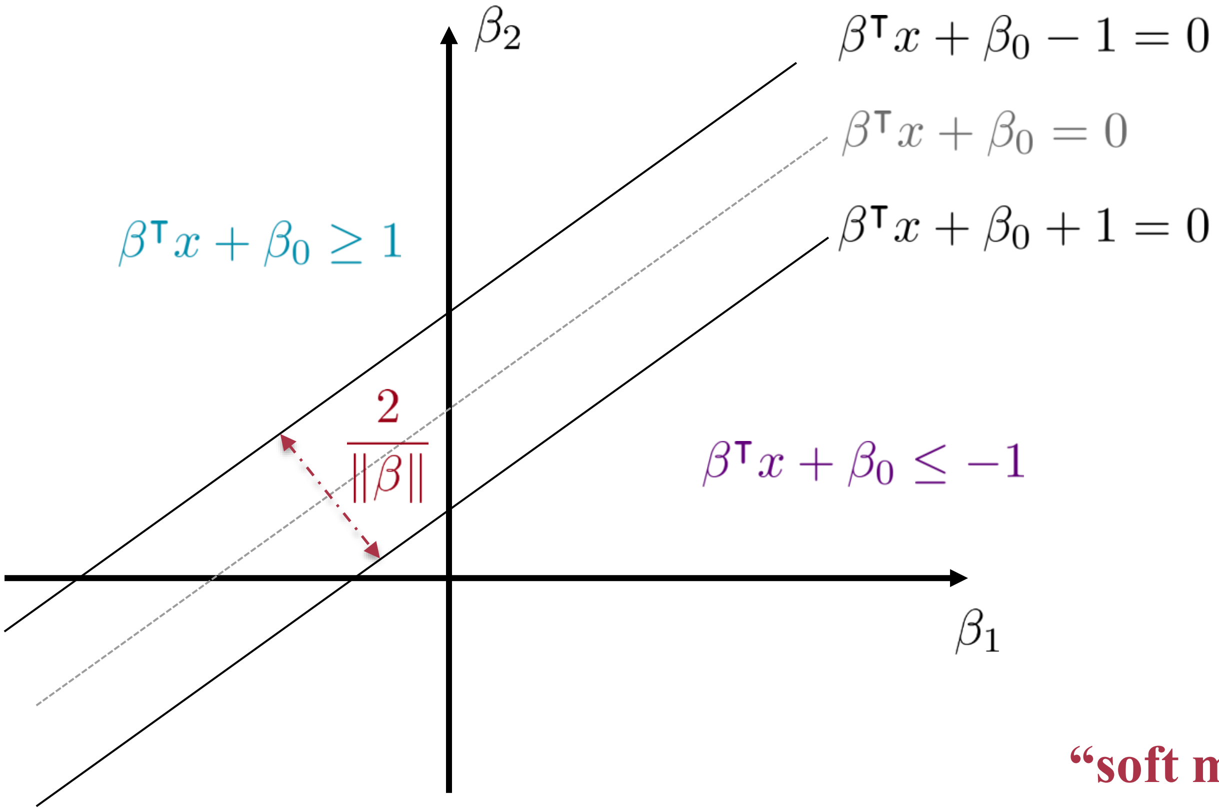
$$y_i = -1$$

A $\quad y_i(\beta^\intercal x_i + \beta_0) \geq 1 - \varepsilon, \quad \varepsilon = 0$

B $\quad y_i(\beta^\intercal x_i + \beta_0) \geq 1 - \varepsilon, \quad 0 < \varepsilon < 1$

C $\quad y_i(\beta^\intercal x_i + \beta_0) \geq 1 - \varepsilon, \quad \varepsilon \geq 1$

Not Perfectly Separable Data



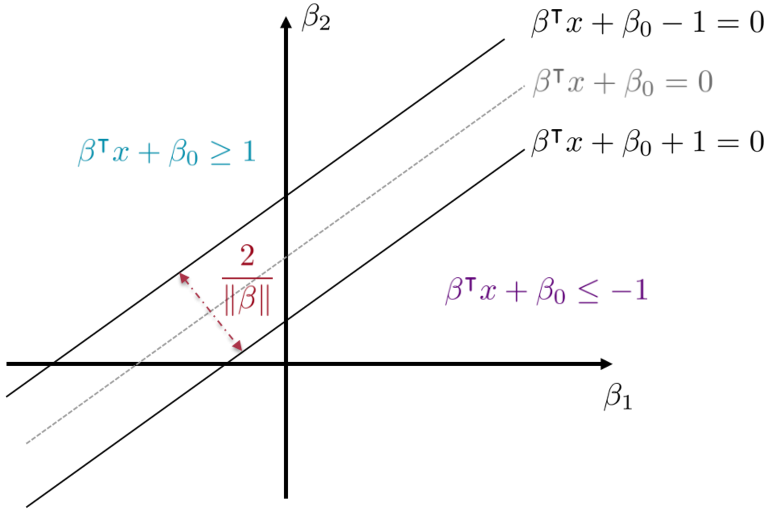
"soft margin"

$$y_i(\beta^\intercal x_i + \beta_0) \geq 1 - \varepsilon, \quad \varepsilon = 0$$

$$y_i(\beta^\intercal x_i + \beta_0) \geq 1 - \varepsilon, \quad 0 < \varepsilon < 1$$

$$y_i(\beta^\intercal x_i + \beta_0) \geq 1 - \varepsilon, \quad \varepsilon \geq 1$$

Not Perfectly Separable Data

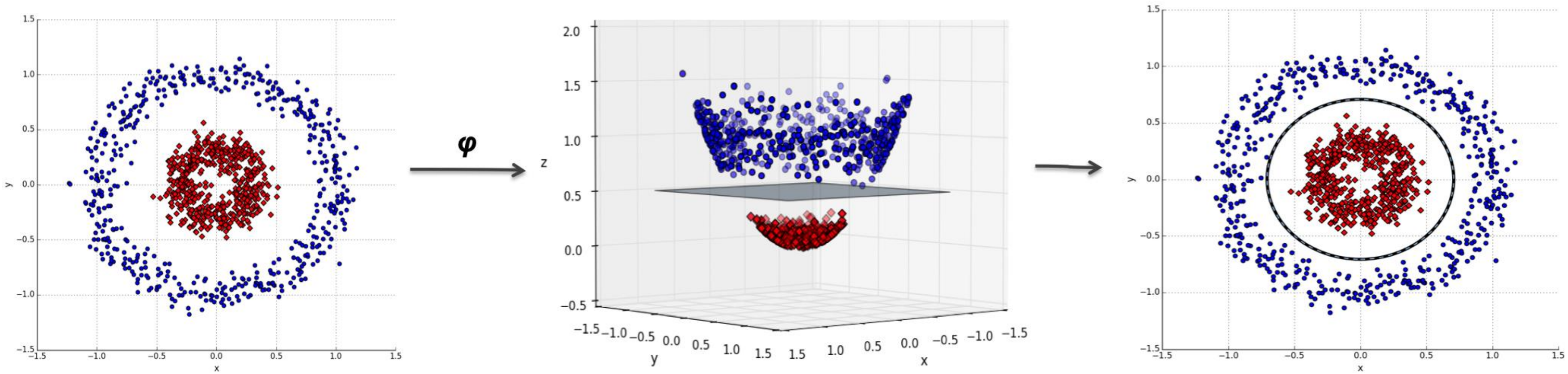


$$y_i(\beta^\intercal x_i + \beta_0) \geq 1 - \varepsilon, \quad \varepsilon = 0$$
$$y_i(\beta^\intercal x_i + \beta_0) \geq 1 - \varepsilon, \quad 0 < \varepsilon < 1$$
$$y_i(\beta^\intercal x_i + \beta_0) \geq 1 - \varepsilon, \quad \varepsilon \geq 1$$

$$\begin{array}{ll} \text{minimize} & \frac{1}{2}\beta^\intercal\beta + c\sum_{i=1}^{n} \varepsilon_i^d \\ \text{subject to} & y_i(\beta^\intercal x_i + \beta_0) \geq 1 - \varepsilon_i, \quad i = 1, \ldots, n \\ & \varepsilon_i \geq 0, \quad i = 1, \ldots, n \end{array}$$

Here c is a tuning parameter that we set by cross-validation.
(d = 1 and d = 2 are quite common in practice)

Path to Kernels



$$\varphi : \mathbb{R}^p \to \mathbb{R}^q$$

Does this make sense?

Path to Kernels

$$\begin{array}{ll} \text{minimize} & \frac{1}{2}\beta^\intercal \beta \\ \text{subject to} & y_i(\beta^\intercal x_i + \beta_0) \geq 1, \quad i = 1, 2, \dots, n \end{array}$$

Lagrange multipliers

Lagrangian Function

$$\mathcal{L}(\beta, \beta_0; \alpha) = \frac{1}{2}\beta^\intercal \beta - \sum_{i \in \mathcal{A}} \alpha_i \left(y_i(\beta^\intercal x_i + \beta_0) - 1\right)$$

set of active constraints

Optimality Conditions

$$\left\{\begin{array}{l} \nabla_\beta \mathcal{L}(\beta, \beta_0; \alpha) = \beta - \sum_{i \in \mathcal{A}} \alpha_i y_i x_i = 0 \implies \beta = \sum_{i \in \mathcal{A}} \alpha_i y_i x_i \\ \dfrac{\partial \mathcal{L}(\beta, \beta_0; \alpha)}{\partial \beta_0} = -\sum_{i \in \mathcal{A}} \alpha_i y_i = 0 \implies \sum_{i \in \mathcal{A}} \alpha_i y_i = 0 \end{array}\right.$$

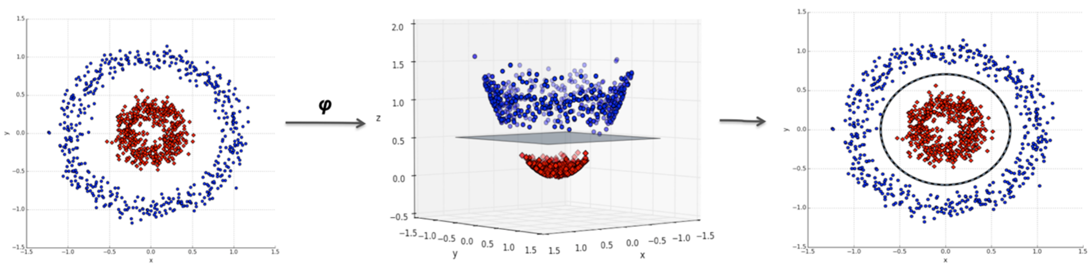
$$\begin{aligned} \mathcal{L}(\beta, \beta_0; \alpha) = \quad & \tfrac{1}{2}\left(\textstyle\sum_{i \in \mathcal{A}} \alpha_i y_i x_i\right)^\intercal \left(\textstyle\sum_{j \in \mathcal{A}} \alpha_j y_j x_j\right) - \left(\textstyle\sum_{i \in \mathcal{A}} \alpha_i y_i \left(\textstyle\sum_{j \in \mathcal{A}} \alpha_j y_j x_j\right)^\intercal x_i\right) \\ & - \textstyle\sum_{i \in \mathcal{A}} \alpha_i y_i \beta_0 + \textstyle\sum_{i \in \mathcal{A}} \alpha_i \\ = \quad & \textstyle\sum_{i \in \mathcal{A}} \alpha_i - \tfrac{1}{2} \textstyle\sum_{i \in \mathcal{A}} \textstyle\sum_{j \in \mathcal{A}} \alpha_i \alpha_j y_i y_j \boxed{x_i^\intercal x_j} \quad \textbf{!} \end{aligned}$$

Kernels

$$\mathcal{L}(\beta, \beta_0; \alpha) = \sum_{i \in \mathcal{A}} \alpha_i - \frac{1}{2} \sum_{i \in \mathcal{A}} \sum_{j \in \mathcal{A}} \alpha_i \alpha_j y_i y_j \boxed{x_i^\intercal x_j}^{*}$$

All we need is to calculate these inner products!

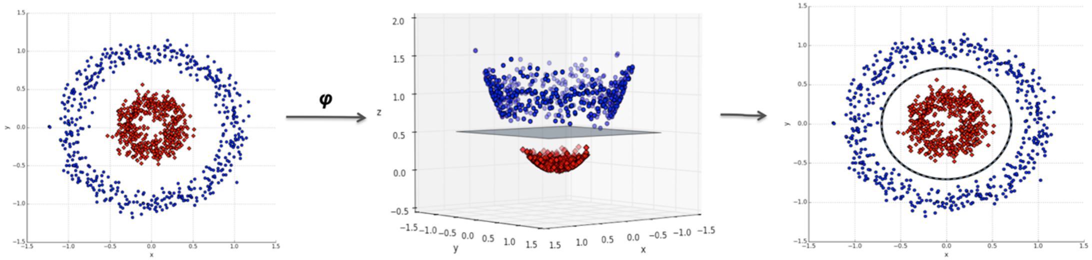
Idea: Replace inner products with kernels that measure the similarity of two observations in higher dimensions.



$$\varphi : \mathbb{R}^p \rightarrow \mathbb{R}^q$$

$$\boxed{K(x_i, x_j) = \varphi(x_i)^\intercal \varphi(x_j)}$$

Kernels



$$\varphi : \mathbb{R}^p \to \mathbb{R}^q$$

$$\boxed{K(x_i, x_j) = \varphi(x_i)^\intercal \varphi(x_j)}$$

Linear Kernel* $\quad K(x_i, x_j) = x_i^\intercal x_j$

Polynomial Kernel $\quad K(x_i, x_j) = (1 + x_i^\intercal x_j)^d$

Radial Kernel $\quad K(x_i, x_j) = e^{-\gamma \|x_i - x_j\|^2}, \ \gamma > 0$

Kernels

Polynomial Kernel $K(x_i, x_j) = (1 + x_i^\intercal x_j)^d$

$$x_i, x_j \in \mathbb{R}^2$$
$$d = 2$$

$$K(x_i, x_j) = (1 + x_i^\intercal x_j)^2 = \begin{array}{l} 1 + x_{i1}^2 x_{j1}^2 + x_{i2}^2 x_{j2}^2 + 2x_{i1}x_{j1} \\ +2x_{i2}x_{j2} + 2x_{i1}x_{i2}x_{j1}x_{j2} \end{array}$$

$$\varphi : \mathbb{R}^2 \to \mathbb{R}^6$$

$$\varphi(x_i) = (1, x_{i1}^2, x_{i2}^2, \sqrt{2}x_{i1}, \sqrt{2}x_{i2}, \sqrt{2}x_{i1}x_{i2})^\intercal$$

$$K(x_i, x_j) = \varphi(x_i)^\intercal \varphi(x_j) = (1 + x_i^\intercal x_j)^2$$

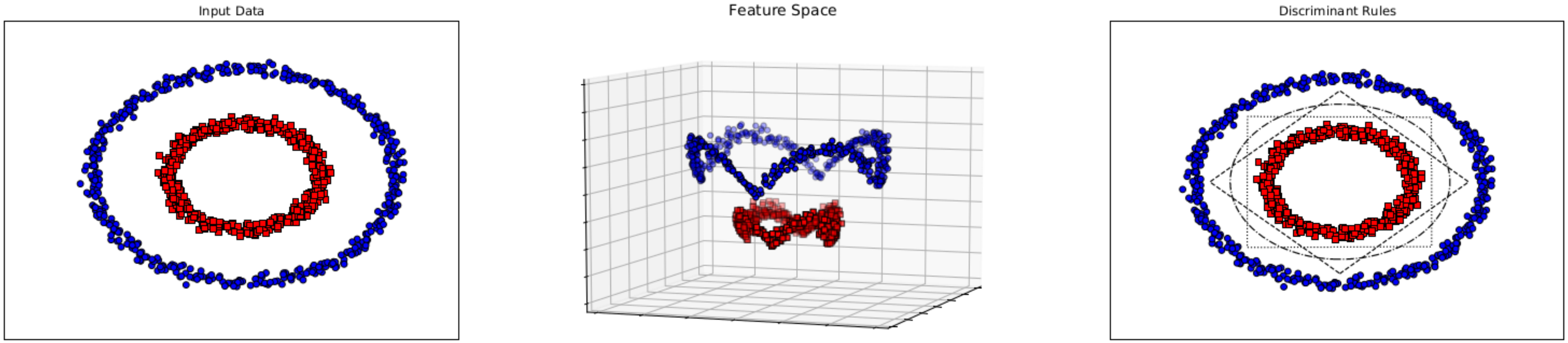
Simple Kernels

$$\varphi : \mathbb{R}^p \to \mathbb{R}^{p+1}$$

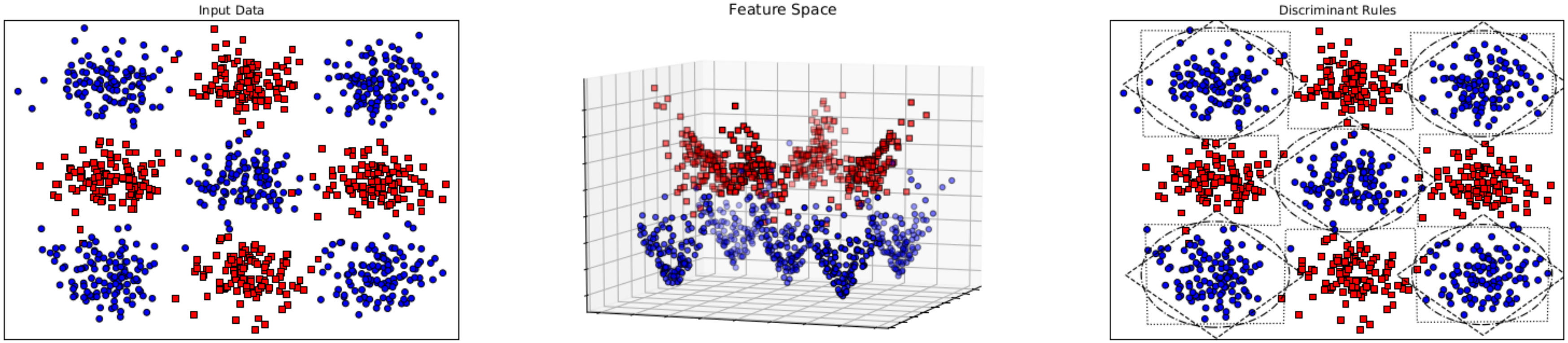
$$\varphi(x_i \mid a) = (x_{i1}, x_{i2}, \ldots, x_{ip}, \|x_i - a\|^2)^\intercal$$

anchor point
(*e.g.* sample mean)

$$K(x_i, x_j) = \varphi(x_i \mid a)^\intercal \varphi(x_j \mid a)$$



Simple Kernels



$$\varphi(x_i \mid a_i) = (x_{i1}, x_{i2}, \ldots, x_{ip}, \|x_i - a_i\|^2)^\intercal$$

$$a_i = \arg\min_{a \in \mathcal{A}} \{\|x_i - a\|^2\}$$

set of anchor points

$$K(x_i, x_j) = \varphi(x_i \mid a_i)^\intercal \varphi(x_j \mid a_j)$$

Simple Kernels

Table 2: Accuracies obtained with different methods on large datasets

Datasets	LIN	$\phi_{2,1}$	POL	RBF
Splice	85.29	86.02	85.61	89.93
Wilt	70.60	81.20	84.00	81.80
Guide1	95.62	96.10	**96.70**	96.62
Spambase	92.76	92.90	91.89	93.70
Phoneme	75.46	74.29	78.36	**87.55**
Magic	79.43	80.30	84.40	**87.71**
Adult	84.93	84.94	84.39	**85.06**

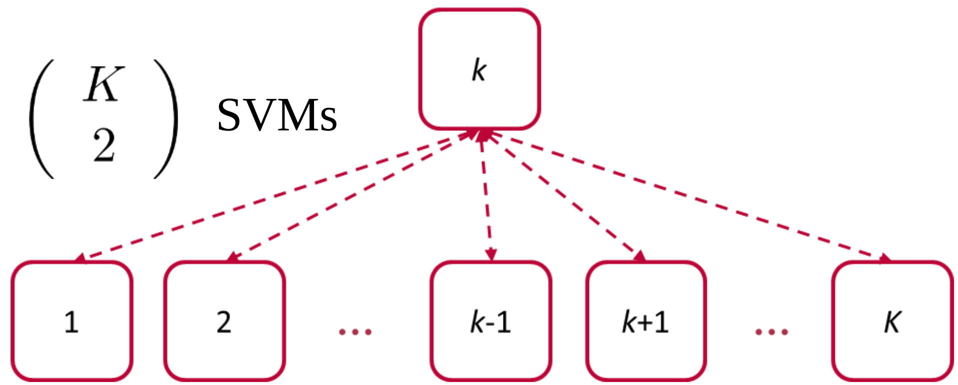
Table 3: Training times in seconds

Datasets/Kernels	LIN	$\phi_{2,1}$	POL	RBF
Splice	<1	<1	<1	<1
Wilt	<1	<1	<1	<1
Guide1	<1	<1	<1	<1
Spambase	<1	<1	<1	<1
Phoneme	<1	<1	169.88	<1
Magic	<1	<1	425.39	63.94
Adult	<1	1.21	89.20	151.36

Multi-class Classification

One-Versus-One (All-Pairs)

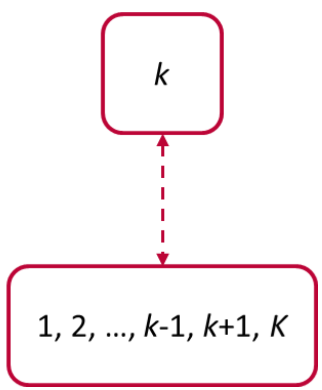
$\binom{K}{2}$ SVMs



x_0 is assigned with majority voting

One-Versus-All

K SVMs

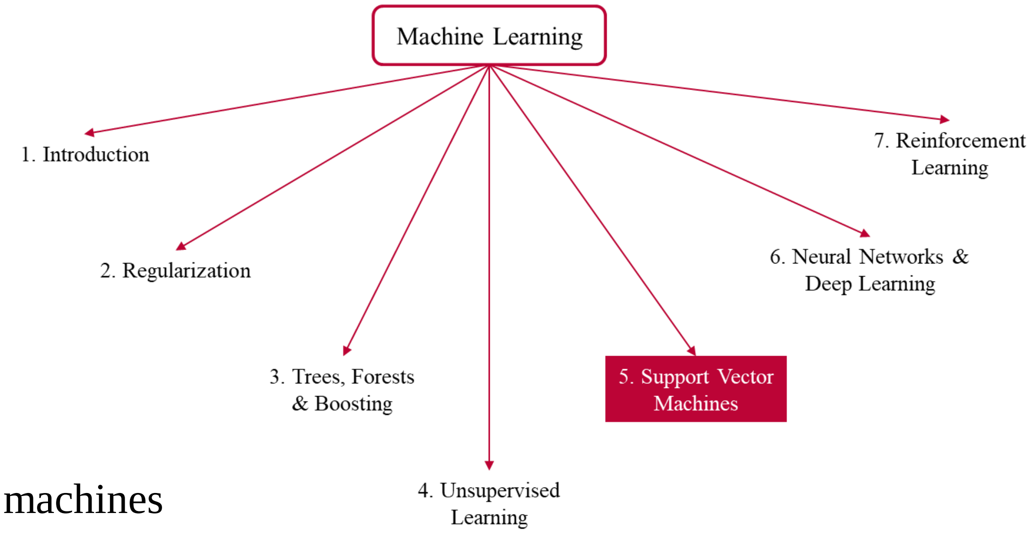


$$\beta_0^{(k)} + \beta^{(k)\intercal} x_0$$

$$\left.\begin{array}{c} \beta_0^{(1)} + \beta^{(1)\intercal} x_0 \\ \beta_0^{(2)} + \beta^{(2)\intercal} x_0 \\ \vdots \\ \beta_0^{(K)} + \beta^{(K)\intercal} x_0 \end{array}\right\}$$

x_0 is assigned to the class that attains the highest value

Outline



- Geometry of support vector machines
- Dual problem
- Path to kernels
- Simple kernels