Linear Support Vector Machine
-
Kernel Methods

B. Michel

Ecole Centrale de Nantes

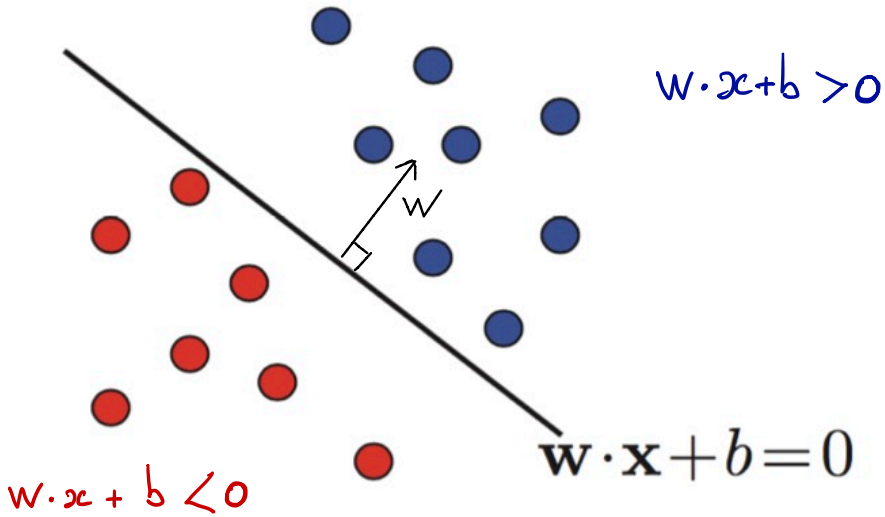
Introduction

- Support Vector Machine (SVM) are popular methods in Machine Learning.
- SVMs are particularly well suited for classification of complex but small- or medium-sized datasets.
- SVM algorithm is quite versatile: it also supports linear (and nonlinear) regression.
- SVM can be generalized to solve nonlinear problems: kernel methods.

Outline

Separating hyperplane

- A dataset $(x_i, y_i)_{i=1...n}$, where $y_i = \pm 1$ and $x_i \in \mathbb{R}^d$.
- The data set is linearly separable if we can find an hyperplane H of $\mathbb{R}^d$ that perfectly (and strictly) separates the two sets $\{i \,|\, y_i = 1\}$ and $\{i \,|\, y_i = -1\}$:



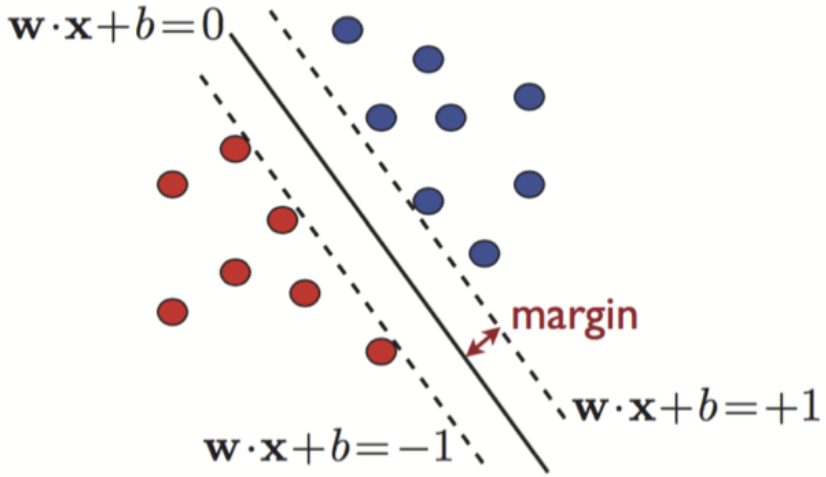
- $x \in H \Leftrightarrow \langle w, x \rangle + b = 0$, where $w \in \mathbb{R}^d$ and $b \in \mathbb{R}$.
- We say that H is a separating hyperplane.

Margin

- Note that for any $c > 0$:

$$\{x \,|\, \langle w, x\rangle + b = 0\} = \{x \,|\, (cw)'x + (cb) = 0\}.$$

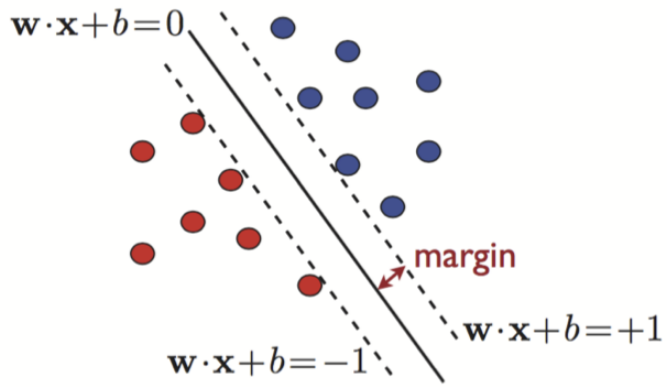
- Canonical Hyperplane: rescale the parameters w and b such that $\min_{i=1\ldots n} |\langle w, x_i\rangle + b| = 1$.



- The (geometric) margin is given by $\frac{1}{\|w\|}$.

- The data point is strictly linearly separable if
  - we can find $w \in \mathbb{R}^p$ and $b \in \mathbb{R}$ such that $\min_{i=1\ldots n} |\langle w, x\rangle + b| = 1$.
  - label y is given by $\text{sign}(\langle w, x\rangle + b)$ and thus $y_i(\langle w, x_i\rangle + b) \geq 1$, $i = 1 \ldots n$.

Large margin classification



- The (geometric) margin is given by $\frac{1}{\|w\|}$: to be maximized.
- Labels: $y(\langle w, x\rangle + b) \geq 1$: constraint
- **Optimization problem for large margin classification (Primal Problem):**

$$\min_{w\in\mathbb{R}^d, b\in\mathbb{R}} \quad \frac{1}{2}\|w\|_2^2$$

$$\text{subject to} \quad g_i(w,b) := 1 - y_i(\langle w, x_i\rangle + b) \leq 0,\ i = \ldots n$$

- QP problem: optimizing a quadratic function of several variables subject to linear constraints.

Duality in Optimization

- Given a constrained optimization problem (primal problem), by considering its **Lagrangian function**, it is possible to express a different but closely related problem, called its **dual problem**.
- The solution to the dual problem typically gives a lower bound to the solution of the primal problem.
- Under some conditions it can even have the same solutions as the primal problem and **SVM problem meet these conditions**.

Dual problem for SVM

After considering the Lagrangian for SVM, we find the **dual problem** for SVM is

$$\max_{\alpha \in \mathbb{R}^n} \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j \langle x_i, x_j \rangle \quad \text{s.t.} \quad \alpha_i \geq 0 \text{ and } \sum_{i=1}^{n} \alpha_i y_i = 0.$$

- Dual problem is also a **QP** problem.
- Formulation in terms of the **inner products** $\langle x_i, x_j \rangle$
- Only depends on the**support vectors** such that $\alpha_i > 0$.
- The solution α gives the **final classifier**

$$x \mapsto \text{sign}\left(\langle w, x \rangle + b\right) = \text{sign}\left(\sum_{i=1}^{n} \alpha_i y_i \langle x, x_i \rangle + b\right)$$

Dual problem vs Primal problem

Question: Why considering the dual problem when fitting SVM ?

Final classifier

$$x \mapsto \text{sign}\left(\langle w, x\rangle + b\right) = \text{sign}\left(\sum_{i=1}^{n} \alpha_i y_i \langle x, x_i\rangle + b\right)$$

- **Efficient computation of the final classifier.**
  - Solving the **primal** problem gives the optimal w, not the α_i's. If the dimension is **large** the computation of $\langle w, x\rangle$ can be **expensive**.
  - Solving the **dual** problem gives α, and $\sum_{i=1}^{n} \langle x, x_i\rangle$ can be very **efficiently** calculated if there are only **few** support vectors.

- **Alternative inner products.**
  Formulation in terms of the inner products $\langle x_i, x_j\rangle$ opens to door generalization to **kernel methods**.

From hard margin linear SVM ...

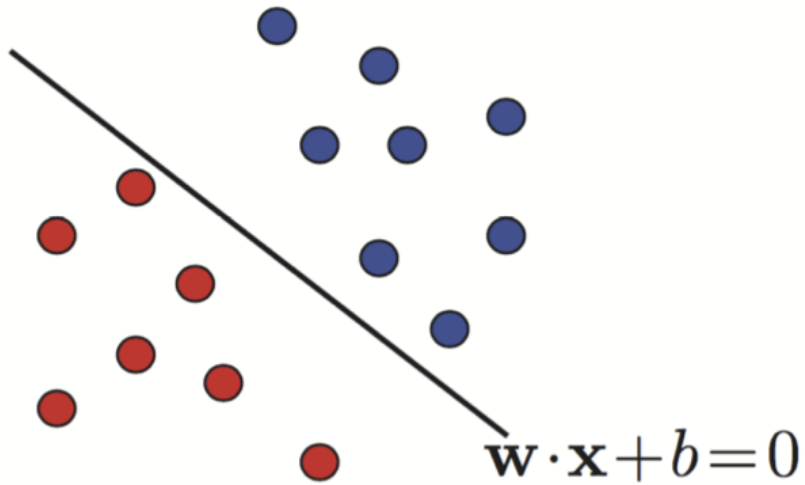
Question: What is the strong assumption we made until now ?

From hard margin linear SVM ...

Question: What is the strong assumption we made until now ?

The data point is separable if we can solve the optimization problem

$$\min_{w \in \mathbb{R}^d, b \in \mathbb{R}} \frac{1}{2}\|w\|^2 \quad \text{subject to} \quad y_i(\langle w, x_i \rangle + b) \geq 1, \; i = \dots n.$$



Datasets that are linearly separable are not really challenging ...

...to soft margin linear SVM

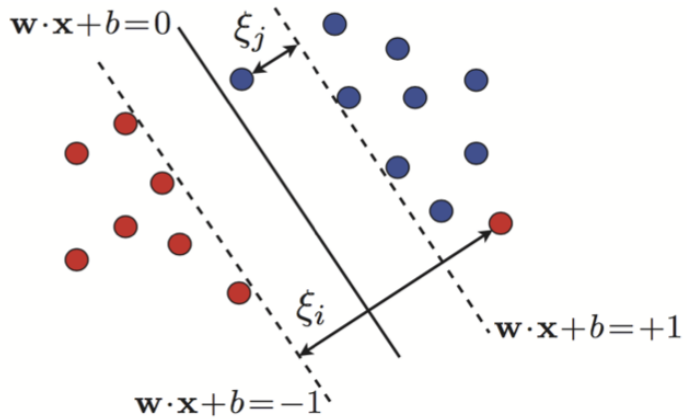
- Replace the constraints

$$y_i(\langle w, x_i \rangle + b) \geq 1$$

by the relaxed ones :

$$y_i(\langle w, x_i \rangle + b) \geq 1 - \xi_i$$

- **slack variables** : $\xi_1, \ldots, \xi_n \geq 0$.



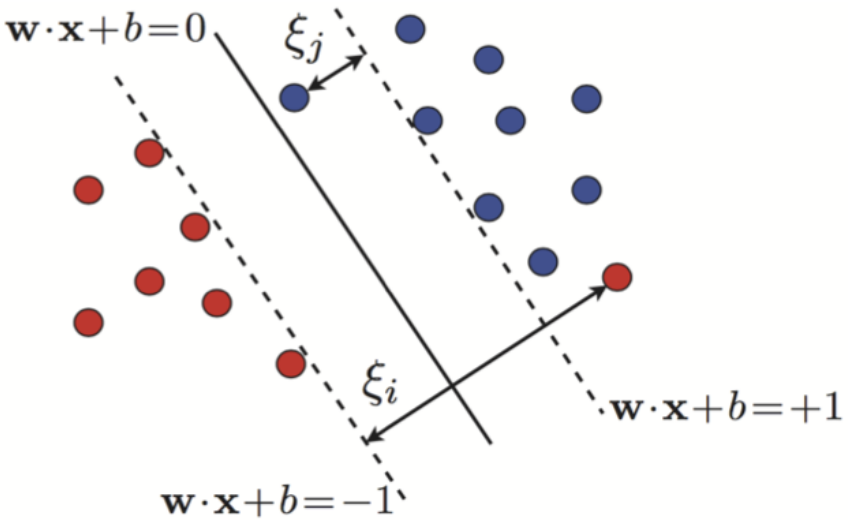
⚠ we also want to keep the slack variables **small**.

Soft margin linear SVM

Relaxed optimization problem :

$$\min_{w\in\mathbb{R}^d,\, b\in\mathbb{R},\, \xi\in\mathbb{R}^n} \quad \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}\xi_i$$

$$\text{subject to} \quad y_i(\langle w, x_i\rangle + b) \geq 1-\xi_i \quad \text{and} \quad \xi_i \geq 0,\ i = \ldots n$$



C makes a balance between the geometric **soft** margin and the amplitude of the slack variables.

Question: How can we choose C in practice ?

Dual formulation

- We find a similar dual formulation (with an additional constraint)

$$\max_{\alpha \in \mathbb{R}_+^n} \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i,j=1}^n \alpha_i \alpha_j y_i y_j \langle x_i, x_j \rangle$$

$$\text{subject to} \quad 0 \leq \alpha_i \leq C \quad \text{and} \quad \sum_{i=1}^n \alpha_i y_i = 0.$$

- QP problem.
- Formulation in terms of the inner products $\langle x_i, x_j \rangle$
- Only depends on the support vectors ($\alpha_i > 0$)
- The solution α gives the final classifier

$$x \mapsto \text{sign}\left(\langle w, x \rangle + b\right) = \text{sign}\left(\sum_{i=1}^n \alpha_i y_i \langle x, x_i \rangle + b\right)$$

Hinge loss formulation of soft margin linear SVM

- Let $f_{w,b}(x) = \langle w, x \rangle + b$.
- Hinge loss $\ell(u, v) = \max(0, 1 - uv)$.
- The primal problem

$$\min_{w \in \mathbb{R}^d,\, b \in \mathbb{R},\, \xi \in \mathbb{R}^n} \quad \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n} \xi_i \quad \text{s.t.} \quad y_i f_{w,b}(x_i) \geq 1 - \xi_i, , \xi_i \geq 0$$

  can be rewritten as

$$\min_{w \in \mathbb{R}^d,\, b \in \mathbb{R}} \quad \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n} \max\left(0, 1 - y_i f_{w,b}(x_i)\right)$$

$$= \min_{w \in \mathbb{R}^d,\ b \in \mathbb{R}} \quad \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n} \ell(y_i, f_{w,b}(x_i))$$

- Soft margin linear SVM corresponds to (regularized) Empirical Risk Minimization with hinge loss on the family of linear classifiers.

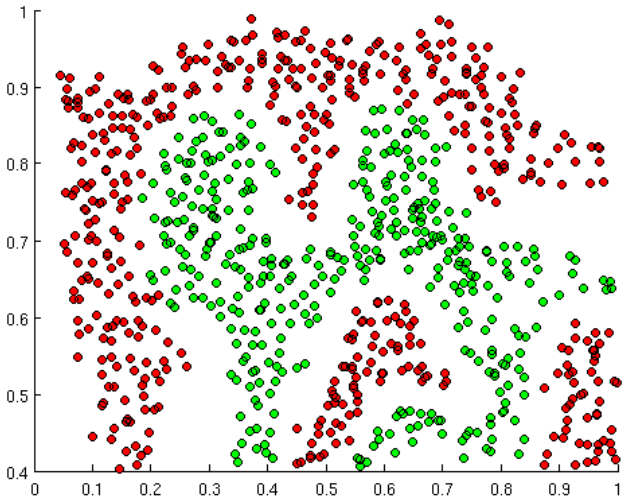
Exercice

Check that using the hinge loss makes sense for classification : compare $\ell(y_i, f_{w,b}(x_i))$ with $\mathbb{1}_{y_i \neq \text{sign}(f_{w,b}(x_i))}$ for different values of y_i and $f_{w,b}(x_i)$.

Outline

Limitations of linear svm

- Many datasets are not even close to being linearly separable:

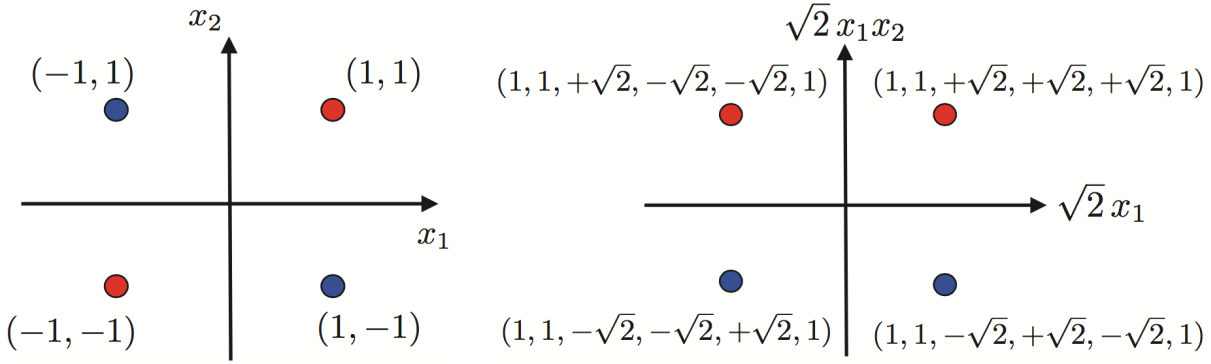


[source: openclassroom]

- One approach to handling nonlinear datasets is to add more features, such as polynomial features.
- Consider a feature map $\Phi : \mathbb{R}^d \mapsto \mathcal{F}$ high-dimensional (implicit) feature space that adds all these new features.

Polynomial mapping

The XOR (exclusive OR) operator:



[source: Mohri]

- cannot be linearly separated in $\mathbb{R}^2$
- solved by the so-called **polynomial mapping or order 2**
  $\Phi : \mathbb{R}^2 \mapsto \mathbb{R}^6$

$$\Phi(x_1, x_2) = (x_1^2, x_2^2, \sqrt{2}x_1x_2, \sqrt{2}x_1, \sqrt{2}x_2, 1)$$

  i.e. the images of the four points by Φ can be linearly separated in $\mathbb{R}^6$.

Polynomial mapping

- The mapping

$$\Phi(x_1, x_2) = (x_1^2, x_2^2, \sqrt{2}x_1x_2, \sqrt{2}x_1, \sqrt{2}x_2, 1)$$

  satisfies

$$\begin{aligned} \langle \Phi(x), \Phi(x') \rangle_{\mathbb{R}^6} &= (x_1x_1' + x_2x_2' + 1)^2 \\ &= (1 + \langle x, x' \rangle_{\mathbb{R}^2})^2 =: K(x, x') \\ &= \langle \Phi(x), \Phi(x') \rangle_{\mathbb{R}^6}. \end{aligned}$$

- More generally, the polynomial kernel of degree q is defined for $c > 0$ by

$$K(x, x') := (c + \langle x, x' \rangle)^q$$

  for some convenient polynomial mapping Φ (which can be determined).

- Note that computing Φ and then taking the inner product is more expensive that computing the kernel directly.

Kernels as pairwise comparisons

- An other situation where we define kernels is for studying complex objects on a space $\mathcal{X}$ which is not necessary endowed with a metric.
- For many settings, we know how to construct a **comparison function** K on $\mathcal{X}^2$ (e.g. images, words, texts, trees, graphs ...)
- Examples of kernels:
  - $K(x, x') = \exp(-cd(x, x'))$ where $c > 0$ and d is a pseudo distance on $\mathcal{X}$. When d is the norm on $\mathbb{R}^p$, K is the **Gaussian kernel**.
  - Kernels on string data with n-grams and suffix trees : compare the strings by means of the substrings they contain.
  - An example of kernel between graphs is the **random walk kernel**: performs random walks on two graphs simultaneously and counts the number of paths that were produced by both walks.
  - Motif kernels on genetic sequences.

Positive definite kernel

Definition

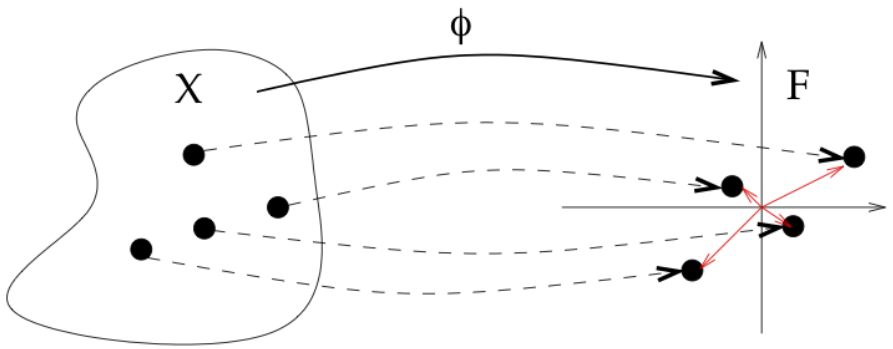
A function $K : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is said to be a positive definite (p.d.) kernel if

- *K is symmetric,*
- *for any $N \in \mathbb{N}$, $x_1, \ldots, x_N \in \mathcal{X}^N$, the similarity matrix $[K(x_i, x_j)]$ is definite positive: for any $a = (a_1, \ldots, a_N)' \in \mathbb{R}^N$, we have*

$$a'[K(x_i, x_j)]a = \sum_{i,j} a_i a_j K(x_i, x_j) \geq 0.$$

The matrix $[K(x_i, x_j)]_{i,j=1\ldots N}$ is the **Gram matrix** of $x_1, \ldots, x_N$.

Feature map and kernels

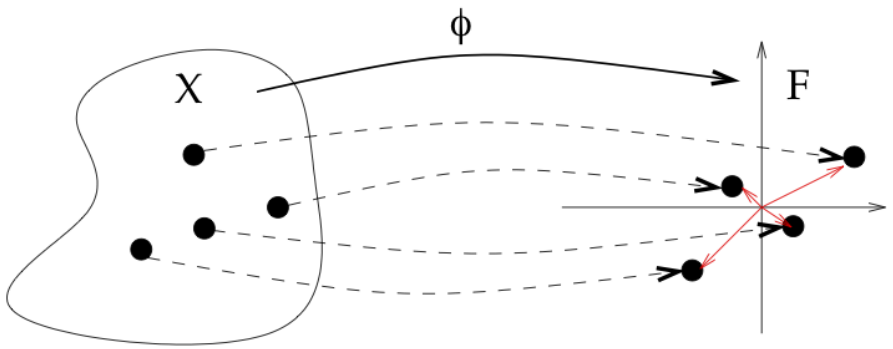


[J.P. Vert]

• In $\mathbb{R}^d$: let $\mathcal{X}$ be a set and $\Phi : \mathcal{X} \to \mathbb{R}^d$. Then the function $K : \mathcal{X}^2 \to \mathbb{R}$ defined as follows is p.d.:

$$\forall (x, x') \in \mathcal{X},\ K(x, x') = \langle \Phi(x), \Phi(x') \rangle_{\mathbb{R}^d}.$$

Feature map and kernels



[J.P. Vert]

- More generally :

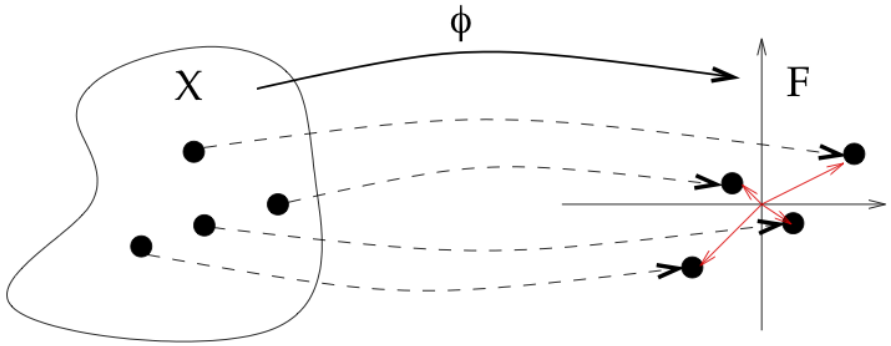
Theorem (Aronszajn, 1950)

K is a p.d. kernel on $\mathcal{X}$ **if and only if** *there exists an Hilbert space $(\mathcal{F}, \langle \cdot, \cdot \rangle_{\mathcal{F}})$ and a mapping $\Phi : \mathcal{X} \to \mathcal{F}$ such that*

$$K(x, x') = \langle \Phi(x), \Phi(x') \rangle_{\mathcal{F}}.$$

A **Hilbert space** is a **vector space** equipped with an **inner product** that induces a distance function for which the space is a **complete** metric space. Here, think of $\mathcal{F}$ as a "nice" functional space.

Feature map and kernels



[J.P. Vert]

- More generally :

Theorem (Aronszajn, 1950)

K is a p.d. kernel on $\mathcal{X}$ **if and only if** *there exists an Hilbert space $(\mathcal{F}, \langle \cdot, \cdot \rangle_{\mathcal{F}})$ and a mapping $\Phi : \mathcal{X} \to \mathcal{F}$ such that*

$$K(x, x') = \langle \Phi(x), \Phi(x') \rangle_{\mathcal{F}}.$$

$\mathcal{F}$ is call a reproducing kernel Hilbert space (RKHS). In short:

- $\forall x \in \mathcal{X}$, $K_x := K(\cdot, x) \in \mathcal{F}$ $\quad$ $x \to K_x$ is a possible mapping
- $\forall f \in \mathcal{F}$, $\langle f, K(\cdot, x) \rangle_{\mathcal{F}} = f(x)$ $\quad$ Reproducing property

Example of kernels on $\mathbb{R}^d$

- Linear kernel on $\mathbb{R}^d$:

$$K(x, y) = < x, y >_{\mathbb{R}^d}$$

- Polynomial kernel of degree q on $\mathbb{R}^d$:

$$K(x, y) = (1 + < x, y >_{\mathbb{R}^d})^q$$

Its mapping contains all the monomials of degree less than q of the coordinates of x.

- Gaussian kernel or Radial Basis Function (RBF) kernel $\mathbb{R}^d$:

$$K(x, y) = \exp(-c\|x - y\|^2)$$

Motivations for Kernel Methods on RKHS

- **Kernel trick:** Any algorithm defined on finite-dimensional vectors that can be expressed only in terms of pairwise inner products can be applied to (potentially) infinite-dimensional vectors in the feature space of a p.d. kernel by replacing each inner product evaluation by a kernel evaluation.

- **Representer Theorems:** Statistical learning problems can often be written as an optimization problem of the form

$$\min_{f \in \mathcal{F}} c(f(x_1), \ldots, f(x_n)) + \lambda \|f\|_{\mathcal{F}}^2 \tag{1}$$

where c measures the fit of f to a given problem and Ω is strictly increasing. The so-called **Representer Theorems** show that any solution of (1) on the RKHS associated to K admits a representation of the form:

$$f(x) = \sum_{i=1}^{n} \alpha_i K(x, x_i).$$

Outline

Kernel SVM Classifier: Primal problem

- Primal problem for linear SVM classifier (hinge loss ℓ):

$$\min_{w \in \mathbb{R}^d,\, b \in \mathbb{R}} \quad \frac{1}{2}\|w\|_2^2 + C \sum_{i=1}^{n} \ell\left(y_i, \langle x_i, w \rangle + b\right)$$

Label prediction is given by $x \mapsto \text{sign}\left(\langle x, w \rangle + b\right)$.

Kernel SVM Classifier: Primal problem

- Primal problem for linear SVM classifier (hinge loss ℓ):

$$\min_{w \in \mathbb{R}^d,\, b \in \mathbb{R}} \quad \frac{1}{2}\|w\|_2^2 + C \sum_{i=1}^{n} \ell\left(y_i, \langle x_i, w \rangle + b\right)$$

  Label prediction is given by $x \mapsto \text{sign}\left(\langle x, w \rangle + b\right)$.

- Let $\Phi : \mathcal{X} \to \mathcal{F}$ be a feature map associated to a p.d. kernel K on $\mathcal{X}$. Replacing x_i by $\Phi(x_i)$ in the primal gives the problem

$$\min_{w \in \mathcal{F},\, b \in \mathbb{R}} \quad \frac{1}{2}\|w\|_{\mathcal{F}}^2 + C \sum_{i=1}^{n} \ell\left(y_i, \langle \Phi(x_i), w \rangle_{\mathcal{F}} + b\right)$$

- Label prediction for $\Phi(x)$, and thus for x, is given by

$$x \mapsto \text{sign}\left(\langle \Phi(x), w \rangle_{\mathcal{F}} + b\right).$$

- For the Primal problem we need to know $\Phi(x) and \langle \cdot, \cdot \rangle_{\mathcal{F}}$.

Kernel Classifier: Dual problem

Dual problem for linear SVM classifier:

$$\max_{\alpha \in \mathbb{R}^n} \quad \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j \langle x_i, x_j \rangle$$

$$\text{subject to} \quad 0 \leq \alpha_i \leq C \quad \text{and} \quad \sum_{i=1}^{n} \alpha_i y_i = 0.$$

Dual label prediction is given by

$$\begin{aligned} x \quad \mapsto \quad & \text{sign}\left(\langle w, x \rangle + b\right) \\ & = \text{sign}\left(\langle x, \sum_{i=1}^{n} \alpha_i y_i x_i \rangle + b\right) \\ & = \text{sign}\left(\sum_{i=1}^{n} y_i \alpha_i \langle x, x_i \rangle + b\right) \end{aligned}$$

Kernel Classifier: Dual problem

Dual problem for kernel SVM classifier:

$$\max_{\alpha \in \mathbb{R}^n} \quad \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j \langle \Phi(x_i), \Phi(x_j) \rangle_{\mathcal{F}}$$

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

$$\text{subject to} \quad 0 \leq \alpha_i \leq C \quad \text{and} \quad \sum_{i=1}^{n} \alpha_i y_i = 0.$$

Label prediction for $\Phi(x)$, and thus for x, is given by

$$x \quad \mapsto \quad \text{sign} \left(\sum_{i=1}^{n} y_i \alpha_i \langle \Phi(x_i), \Phi(x) \rangle_{\mathcal{F}} + b \right)$$

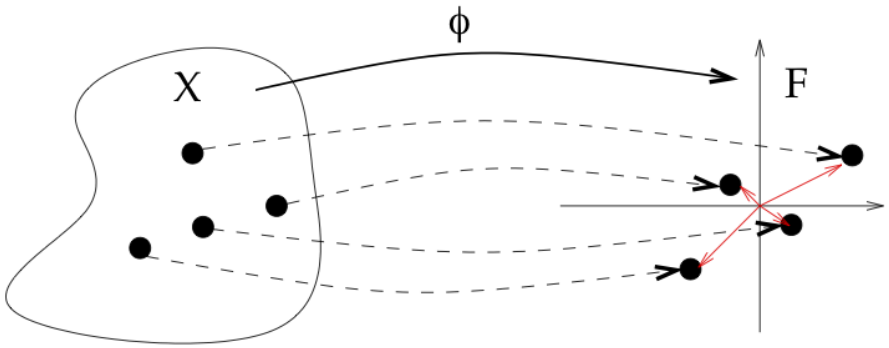
$$= \text{sign} \left(\sum_{i=1}^{n} y_i \alpha_i K(x_i, x) + b \right)$$

Pipeline of Kernel Classifier

- For a classification problem on $\mathcal{X}$ we consider a p.d. K.

Pipeline of Kernel Classifier

- For a classification problem on $\mathcal{X}$ we consider a p.d. K.
- For this kernel there exists a (non linear) mapping Φ into a RKHS $\mathcal{F}_K$. We consider the classification problem in this RKHS.
  - i.e. we push x on $\mathcal{F}$ by Φ
  - This feature map can be seen as an non linear transformation of the features ($\approx$ featuring)



[J.P. Vert]

Pipeline of Kernel Classifier

- For a classification problem on $\mathcal{X}$ we consider a p.d. K.
- For this kernel there exists a (non linear) mapping Φ into a RKHS $\mathcal{F}_K$. We consider the classification problem in this RKHS.
- We consider and solve the (QP) Dual Kernel SVM problem

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

$$\text{subject to} \quad 0 \leq \alpha_i \leq C \quad \text{and} \quad \sum_{i=1}^{n} \alpha_i y_i = 0.$$

Pipeline of Kernel Classifier

- For a classification problem on $\mathcal{X}$ we consider a p.d. K.
- For this kernel there exists a (non linear) mapping Φ into a RKHS $\mathcal{F}_K$. We consider the classification problem in this RKHS.
- We consider and solve the (QP) Dual Kernel SVM problem

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

$$\text{subject to} \quad 0 \leq \alpha_i \leq C \quad \text{and} \quad \sum_{i=1}^{n} \alpha_i y_i = 0.$$

- Kernel trick : the problem now only depends on the $K(x_i, x_j)'s$ (Gram Matrix).

Pipeline of Kernel Classifier

- For a classification problem on $\mathcal{X}$ we consider a p.d. K.
- For this kernel there exists a (non linear) mapping Φ into a RKHS $\mathcal{F}_K$. We consider the classification problem in this RKHS.
- We consider and solve the (QP) Dual Kernel SVM problem

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

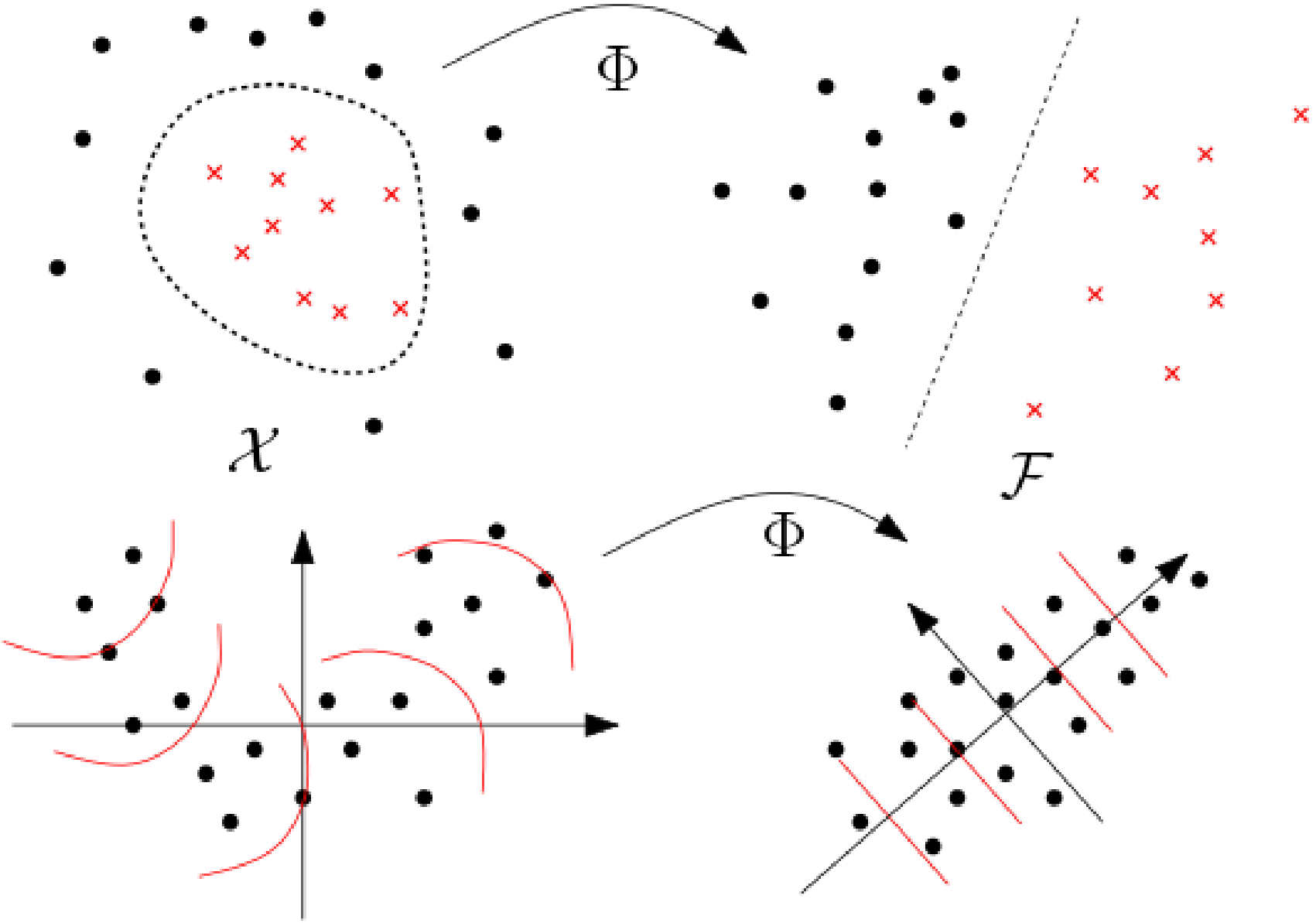
$$\text{subject to} \quad 0 \leq \alpha_i \leq C \quad \text{and} \quad \sum_{i=1}^{n} \alpha_i y_i = 0.$$

- Kernel trick : the problem now only depends on the $K(x_i, x_j)'s$ (Gram Matrix).
- Representation: the solution only depends on the support vectors:

$$x \mapsto \text{sign}\left(\sum_{i=1}^{n} \alpha_i y_i K(x, x_i) + b \right)$$

Outline

Kernel Methods: linear methods in RKHS



Kernel Ridge Regression

- Regression setting $y_i \in \mathbb{R}$ and $x_i \in \mathcal{X}$ (not necessary $\mathbb{R}^p$).
- Let $\Phi : \mathcal{X} \to \mathcal{F}$ be a feature map for the kernel K.
- Kernel ridge regression : find the optimal linear combination of features to predict Y :

$$\text{argmin}_{w \in \mathcal{F}} \sum_{i=1}^{n} (y_i - \langle w, \Phi(x_i) \rangle_{\mathcal{F}})^2 + \lambda \|w\|^2.$$

- By the representer theorem, the solution is a function of the form

$$\hat{w} = \sum_{i=1}^{n} \hat{\alpha}_i \Phi(x_i)$$

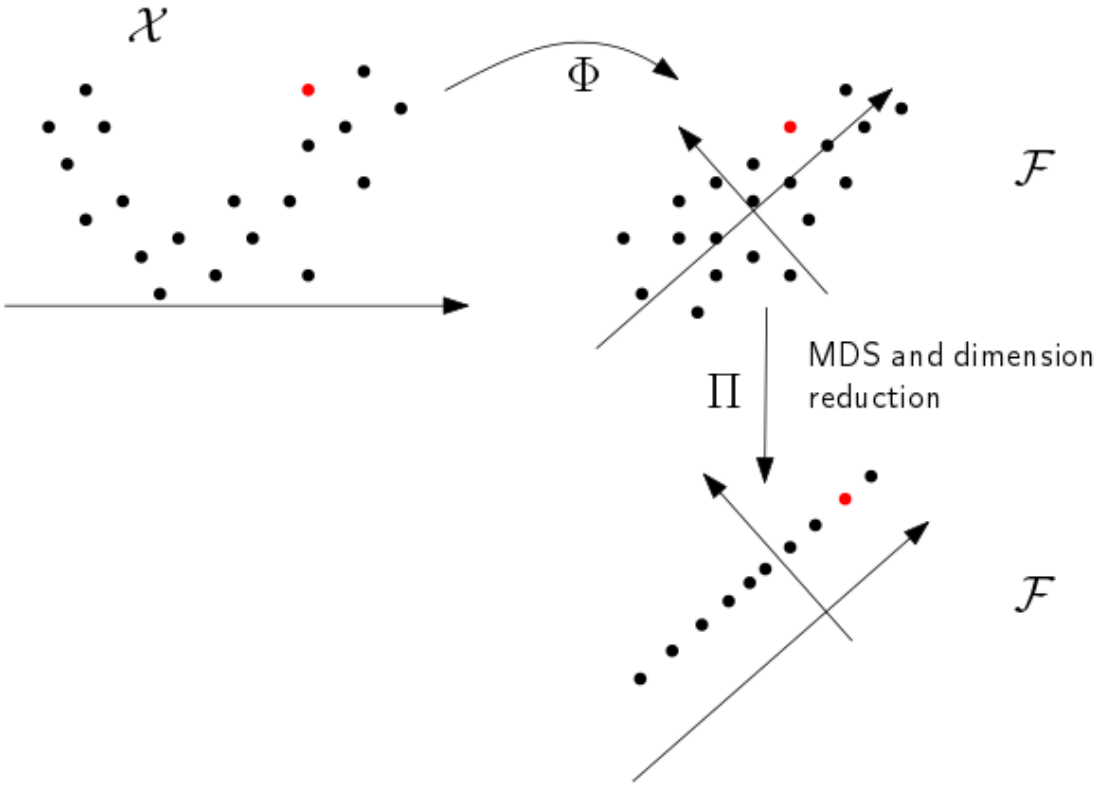
with $\hat{\alpha} = (\boldsymbol{K} + \lambda I_n)^{-1} y$ where $\boldsymbol{K}_{ij} = K(x_i, x_j) = \langle \Phi(x_i), \Phi(x_j) \rangle_{\mathcal{F}}$.

- Finally, the prediction at x is given by

$$\hat{y} = \langle \hat{w}, \Phi(x) \rangle_{\mathcal{F}} = \sum_{i=1}^{n} \alpha_i K(x_i, x).$$

Kernel PCA

- Multidimensional scaling of a matrix of pairwise distances is a dimension reduction method.
- It finds a configuration of points in $\mathbb{R}^d$ that matches as well as possible with the pairwise distances.
- Why not doing PCA in the feature space ? Because we can't compute projection of the (unknown) $\Phi(x_i)$.



- MDS : diagonalization of the Gram matrix in the feature space (= Kernel matrix)
- Kernel PCA = Kernel MDS

Take home message (1)

- Kernel methods provide complex non-linear decision functions.
- There exists “kernel versions" of any standard problem in statistical learning (regression, classification, clustering, dimension reduction ...)
- Representer Theorem: the solution lives in a subspace of dimension n, which can lead to efficient algorithms although the RKHS itself can be of infinite dimension.
- Solving a problem in the dual benefits from the kernel trick.
- We can use any p.d. kernel, but choosing the kernel is not always easy (combination of kernels ...).
- Well-founded methods: Vapnik-Tchervonenkis on RKHSpaces.

Take home message (2)

- Optimization : solutions of kernel method problems are given by Quadratic Programming solvers (nonlinear programming): interior point, active set, augmented Lagrangian, conjugate gradient, gradient projection ... (not presented in this lecture)
- Traditional kernel methods are computationally expensive when n is large because of computations on the $(n \times n)$ Gram matrix K. But many answers to this problem in the literature.