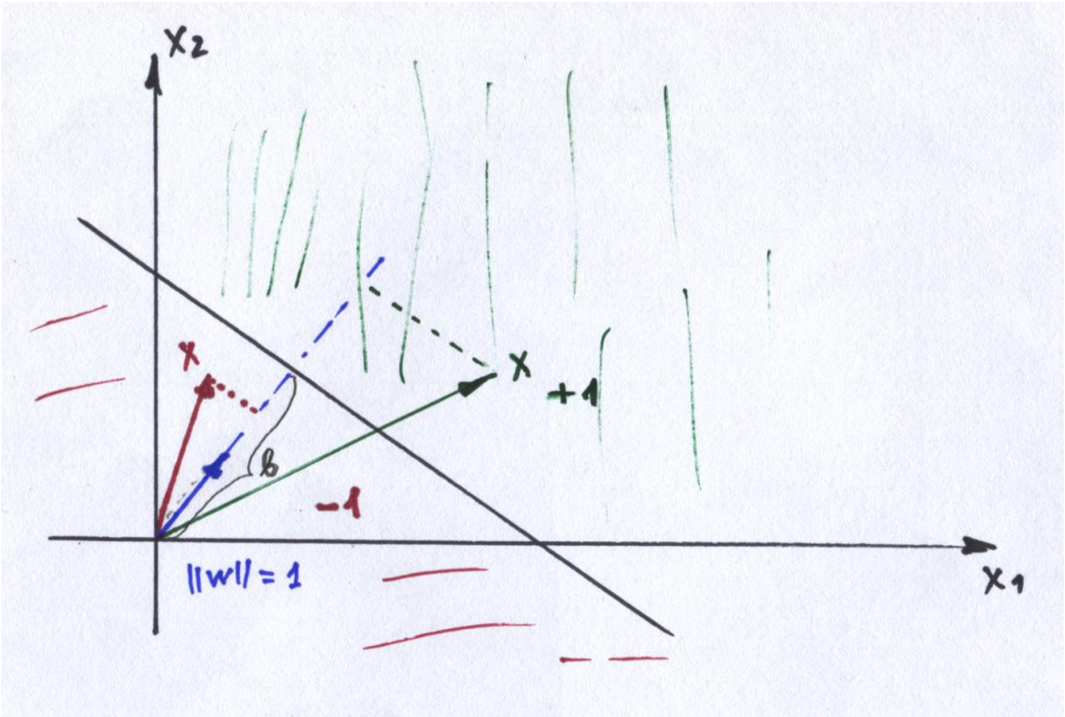
Machine Learning

Support Vector Machines

A building block for almost all – a mapping $f : \mathbb{R}^n \to \{+1, -1\}$,
a partitioning of the input space into half-spaces that correspond to classes.



Decision rule: $y = f(x) = \mathrm{sgn}(\langle x, w \rangle - b)$

w is the **normal** to the hyper plane $\langle x, w \rangle = b$

(Synonyms – Neuron model, Perceptron etc.)

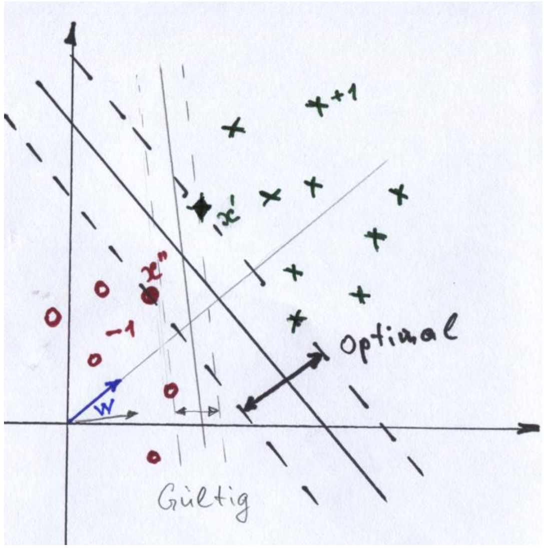
Two learning tasks

Let a training dataset $X = \big((x_i, y_i)\ldots\big)$ be given with

(i) data $x_i \in \mathbb{R}^n$ and (ii) classes $y_i \in \{-1, +1\}$

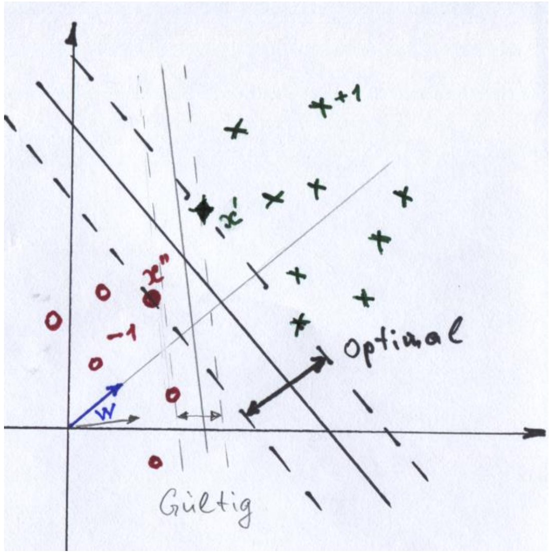
The goal is to find a hyper plane that separates the data (correctly)

$$y_i \cdot [\langle w, x_i \rangle + b] \geq 0 \quad \forall i$$



Now: The goal is to find a “corridor” (stripe) of **the maximal width** that separates the data (correctly).

Linear SVM



Remember that the solution is defined only up to a common scale
→ Use **canonical** (with respect to the learning data) form in order to avoid ambiguity:

$$\min_i |\langle w, x_i \rangle + b| = 1$$

The **margin**:

$$\langle w, x' \rangle + b = +1, \quad \langle w, x'' \rangle + b = -1$$
$$\langle w, x' - x'' \rangle = 2$$
$$\langle w/\|w\|, x' - x'' \rangle = 2/\|w\|$$

The optimization problem:

$$\|w\|^2 \rightarrow \min_{w,b}$$
$$\text{s.t.} \quad y_i \cdot [\langle w, x_i \rangle + b] \geq 1 \quad \forall i$$

Linear SVM

The Lagrangian of the problem:

$$L(w, b, \alpha) = \frac{1}{2}\|w\|^2 - \sum_i \alpha_i \cdot (y_i \cdot [\langle w, x_i \rangle + b] - 1) \to \max_{\alpha} \min_{w,b}$$

$$\alpha_i \geq 0 \quad \forall i$$

The meaning of the dual variables α :

a) $y_i \cdot [\langle w, x_i \rangle + b] - 1 < 0$ (a constraint is broken) → maximization wrt. α_i gives: $\alpha_i \to \infty,\ L(w, b, \alpha) \to \infty$ (surely not a minimum)

b) $y_i \cdot [\langle w, x_i \rangle + b] - 1 > 0$ → maximization wrt. α_i gives $\alpha_i = 0$ → no influence on the Lagrangian

c) $y_i \cdot [\langle w, x_i \rangle + b] - 1 = 0$ → α_i does not mater, the vector x_i is located "on the wall of the corridor" – **Support Vector**

Linear SVM

Lagrangian:

$$L(w, b, \alpha) = \frac{1}{2}\|w\|^2 - \sum_i \alpha_i \cdot (y_i \cdot [\langle w, x_i \rangle + b] - 1)$$

Derivatives:

$$\frac{\partial L}{\partial b} = \sum_i \alpha_i y_i = 0$$

$$\frac{\partial L}{\partial w} = w - \sum_i \alpha_i y_i x_i = 0$$

$$w = \sum_i \alpha_i y_i x_i$$

The solution is a **linear combination** of the data points.

Linear SVM

Substitute $w = \sum_i \alpha_i y_i x_i$ into the decision rule and obtain

$$f(x) = \operatorname{sgn}\big(\langle x, w\rangle + b\big) = \operatorname{sgn}\Big(\Big\langle x, \sum_i \alpha_i y_i x_i \Big\rangle + b\Big) =$$

$$\operatorname{sgn}\Big(\sum_i \alpha_i y_i \langle x, x_i\rangle + b\Big)$$

→ the vector w is not needed explicitly !!!

The decision rule can be expressed as a linear combination of **scalar products** with support vectors.

Only strictly positive α_i (i.e. those corresponding to the support vectors) are necessary for that.

Linear SVM

Substitute

$$\sum_i \alpha_i y_i = 0$$

$$w = \sum_i \alpha_i y_i x_i$$

into the Lagrangian

$$L(w, b, \alpha) = \frac{1}{2}\|w\|^2 - \sum_i \alpha_i \cdot (y_i \cdot [\langle w, x_i \rangle + b] - 1)$$

and obtain the **dual task**

$$\sum_i \alpha_i - \frac{1}{2}\sum_{ij} \alpha_i \alpha_j y_i y_j \langle x_i, x_j \rangle \to \max_{\alpha}$$

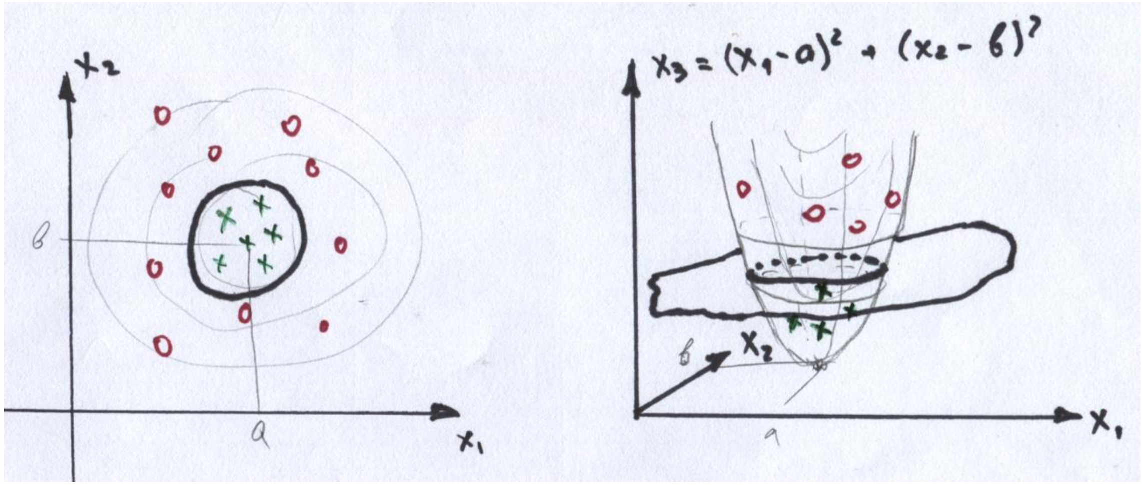
$$\text{s.t.} \quad \alpha_i \geq 0, \quad \sum_i \alpha_i y_i = 0$$

→ can also be expressed in terms of scalar products only, the data points x_i are not explicitly necessary.

Feature spaces

1. The input space $\mathcal{X}$ is mapped onto a feature space $\mathcal{H}$ by a non-linear transformation $\Phi : \mathcal{X} \rightarrow \mathcal{H}$
2. The data are separated (classified) by a linear decision rule in the feature space

Example: quadratic classifier $f(x) = \mathrm{sgn}(a \cdot x_1^2 + b \cdot x_1 x_2 + c \cdot x_2^2)$



The transformation is

$$\Phi : \mathbb{R}^2 \rightarrow \mathbb{R}^3$$
$$\Phi(x_1, x_2) = (x_1^2, \sqrt{2}x_1 x_2, x_2^2)$$

(the images $\Phi(\bar{x})$ are separable in the feature space)

Feature spaces

The images $\Phi(x)$ are not explicitly necessary in order to find the separating plane in the feature space, but their **scalar products**

$$\langle \Phi(x), \Phi(x') \rangle$$

For the example above:

$$\begin{aligned} \left\langle \Phi(x_1, x_2), \Phi(x_1', x_2') \right\rangle &= \left\langle (x_1^2, \sqrt{2} x_1 x_2, x_2^2), (x_1'^2, \sqrt{2} x_1' x_2', x_2'^2) \right\rangle = \\ & x_1^2 x_1'^2 + 2 x_1 x_2 x_1' x_2' + x_2^2 x_2'^2 = \\ & (x_1 x_1' + x_2 x_2')^2 = \langle x, x' \rangle^2 = \quad k(x, x') \end{aligned}$$

→ the scalar product can be computed in the input space, it is not necessary to map the data points onto the feature space explicitly.

Such functions $k(x, x')$ are called **Kernels**.

Kernels

Kernel is a function $k : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$ that computes scalar product in a feature space

$$k(x, x') = \left\langle \Phi(x), \Phi(x') \right\rangle$$

Neither the corresponding space $\mathcal{H}$ nor the mapping $\Phi : \mathcal{X} \rightarrow \mathcal{H}$ need to be specified thereby explicitly → "Black Box".

Alternative definition: if a function $k : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$ is a kernel, then there exists such a mapping $\Phi : \mathcal{X} \rightarrow \mathcal{H}$, that … The corresponding feature space $\mathcal{H}$ is called the **Hilbert space induced** by the kernel k .

Let a function $k : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$ be given. Is it a kernel?
→ Mercer's theorem.

Kernels

Let k_1 and k_2 be two kernels.

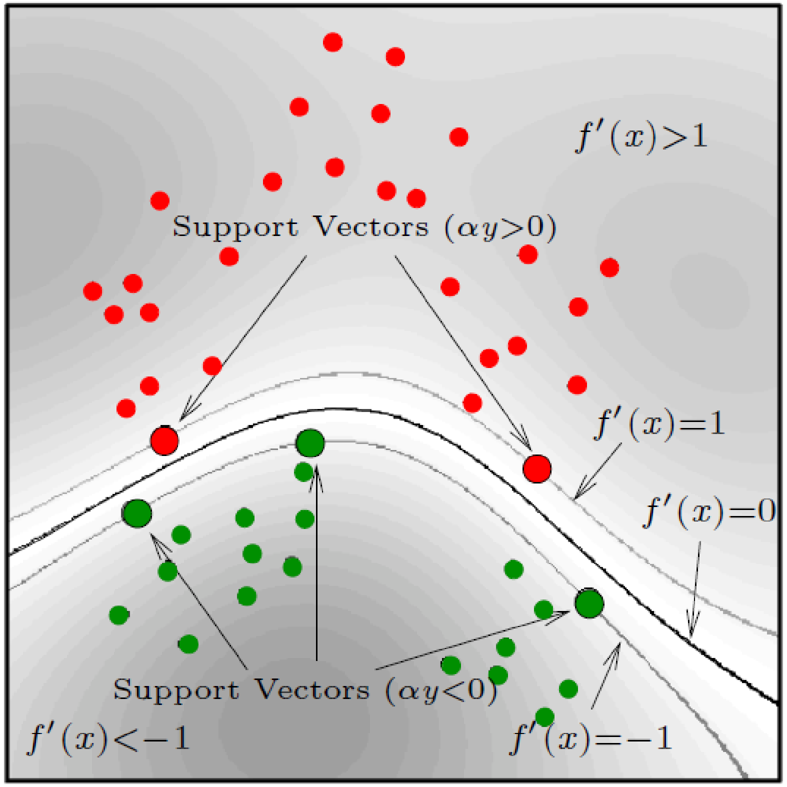
Than $\alpha k_1,\ k_1 + k_2,\ k_1 k_2$ are kernels as well
(there are also other possibilities to build kernels from kernels).

Popular Kernels:

- Polynomial: $k(x, x') = \big(\langle x, x' \rangle + c\big)^d$
- Sigmoid: $k(x, x') = \tanh\big(\kappa \langle x, x' \rangle + \Theta\big)$
- Gaussian: $k(x, x') = \exp\big(-\|x - x'\|^2 / (2\sigma^2)\big)$ (interesting : $\mathcal{H} = \mathbb{R}^\infty$)

An example

The decision rule with a Gaussian kernel $k(x, x') = \exp\left[-\frac{\|x - x'\|^2}{2\sigma^2}\right]$



$$f(x) = \operatorname{sgn}\left(f'(x)\right) = \operatorname{sgn}\left(\sum_i y_i \alpha_i \exp\left[-\frac{\|x - x_i\|^2}{2\sigma^2}\right]\right)$$

Conclusion

- SVM is a representative of **discriminative learning** – i.e. with all corresponding advantages (power) and drawbacks (overfitting) – remember e.g. the Gaussian kernel with $\mathcal{H} = \mathbb{R}^\infty$
- The building block – linear classifiers. All formalisms can be expressed in terms of **scalar products** – the data are not needed explicitly.
- **Feature spaces** – make non-linear decision rules in the input spaces possible.
- **Kernels** – scalar product in feature spaces, the latter need not be necessarily defined explicitly.

Literature (names):

- Bernhard Schölkopf, Alex Smola ...
- Nello Cristianini, John Shawe-Taylor ...