Métodos de kernel

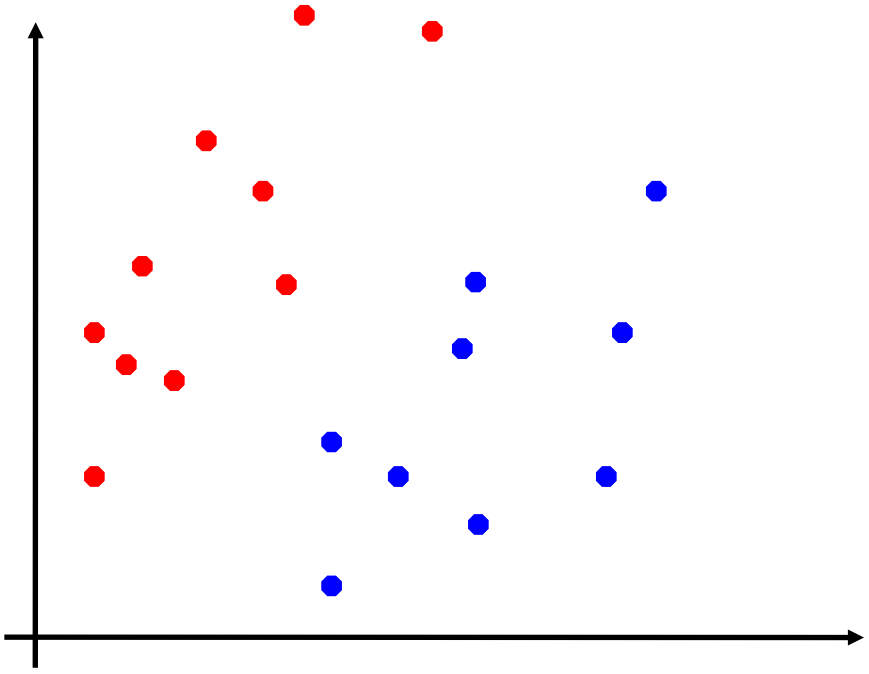
Resumen

- SVM - motivación
- SVM no separable
- Kernels
- Otros problemas
- Ejemplos

Muchas slides de Ronald Collopert

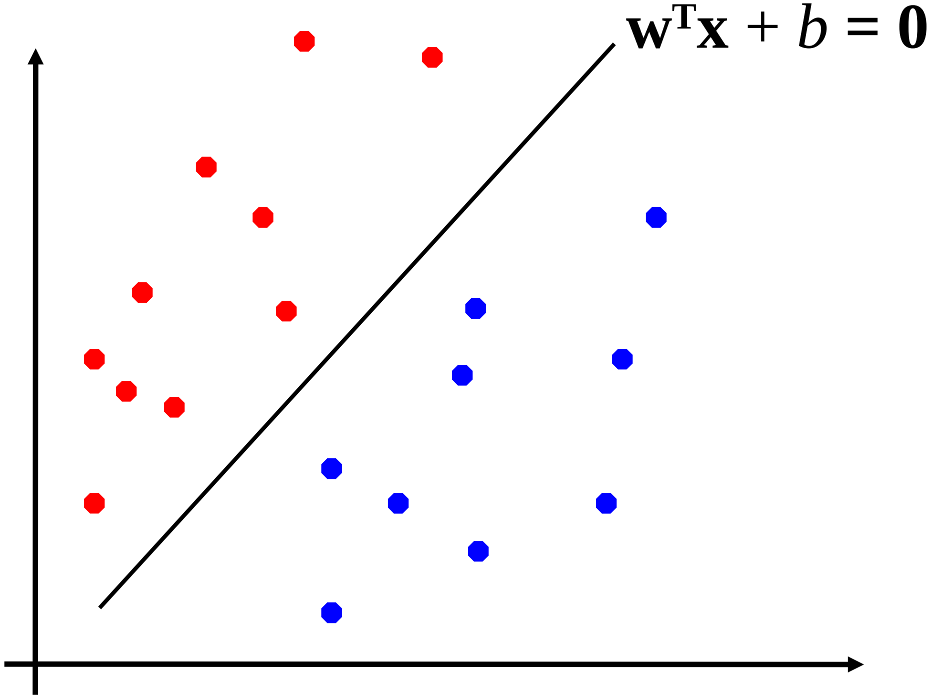
Back to Perceptron

Old method, linear solution



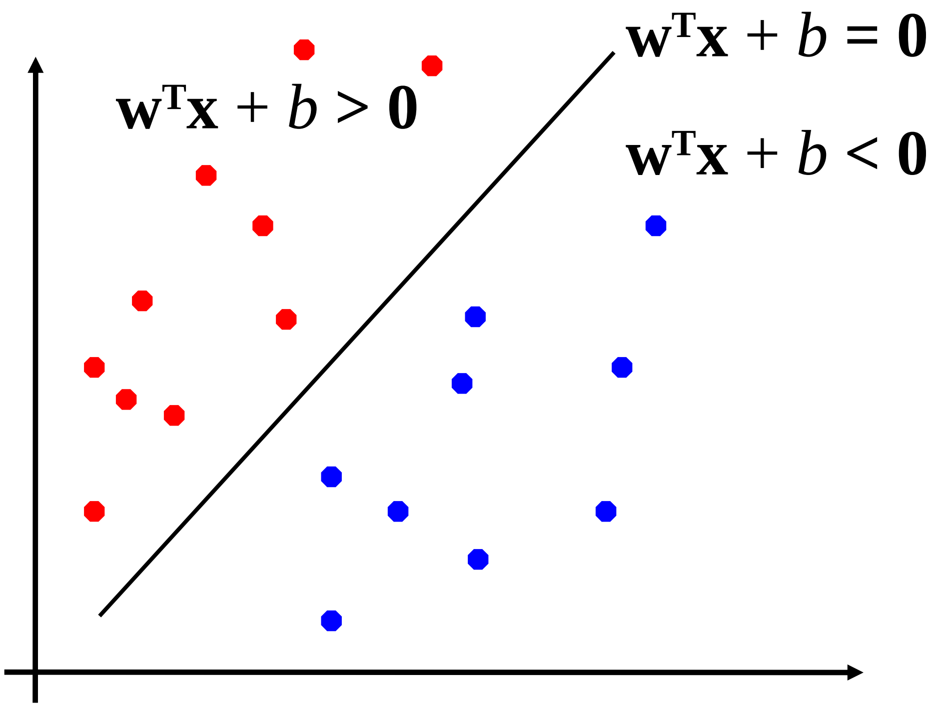
Back to Perceptron

Old method, linear solution



Back to Perceptron

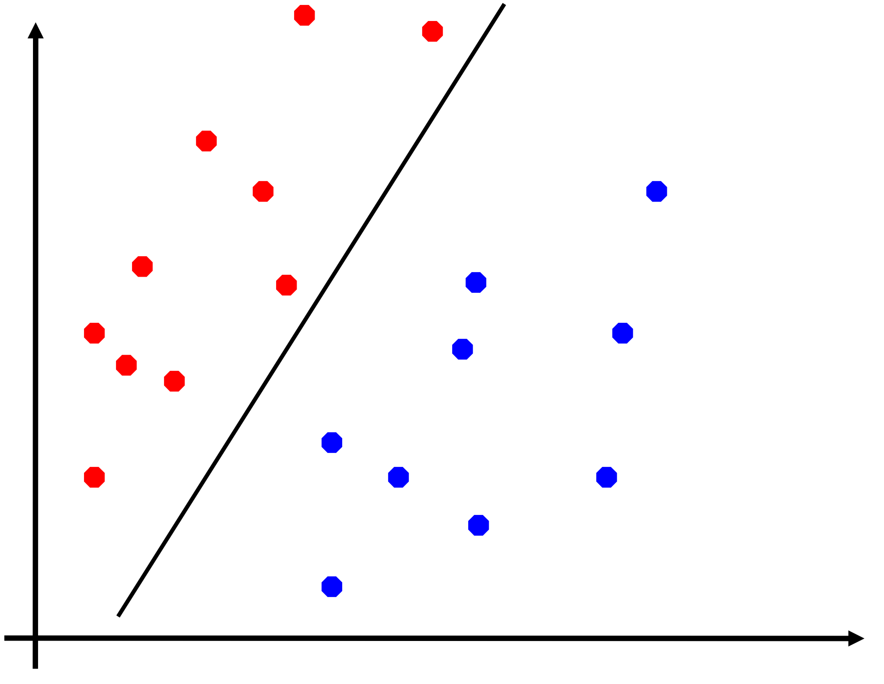
Old method, linear solution



$$f(\mathbf{x}) = \text{sign}(\mathbf{w}^T\mathbf{x} + b)$$

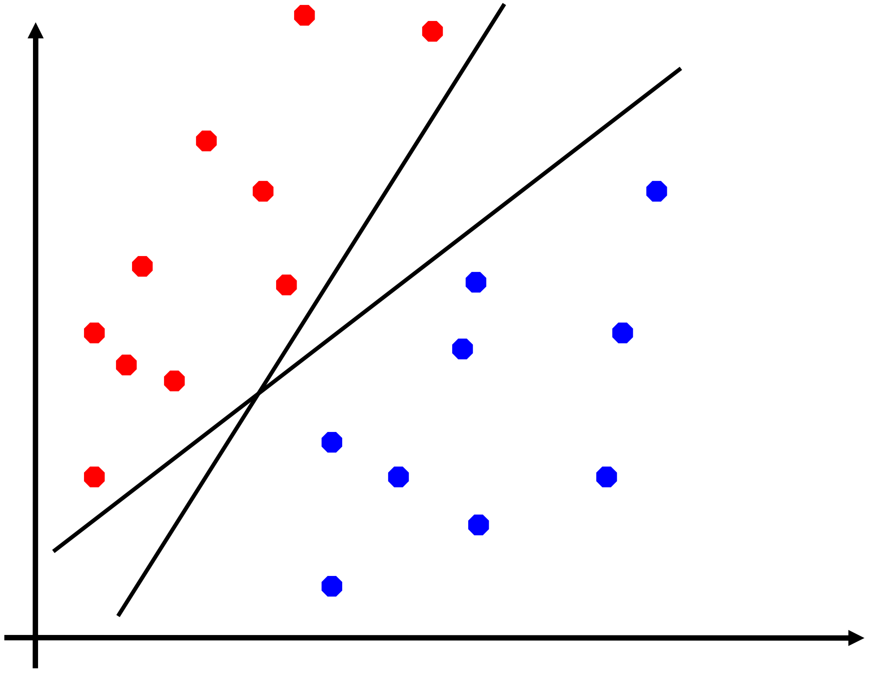
Linear Separators

Which of the linear separators is optimal?



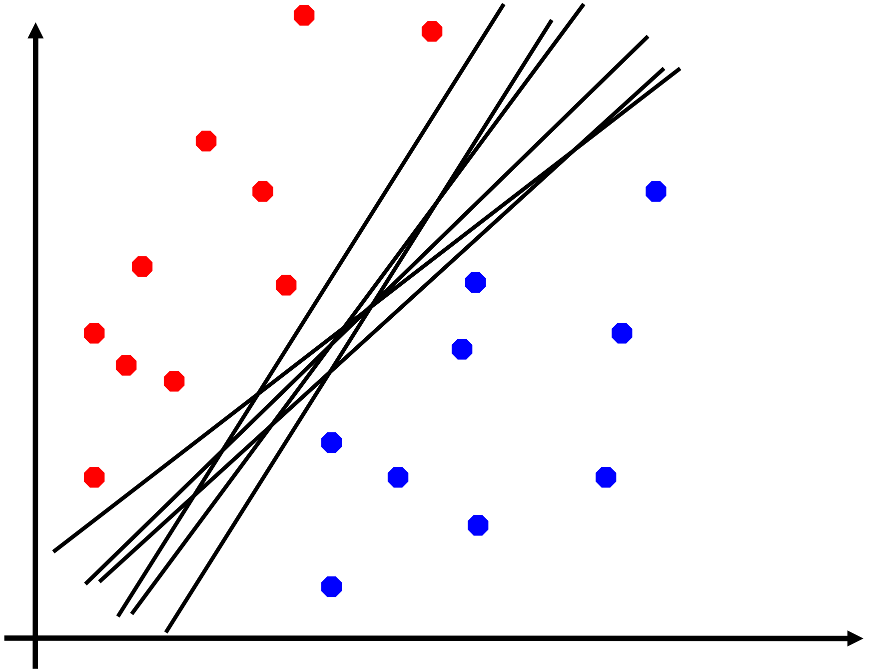
Linear Separators

Which of the linear separators is optimal?



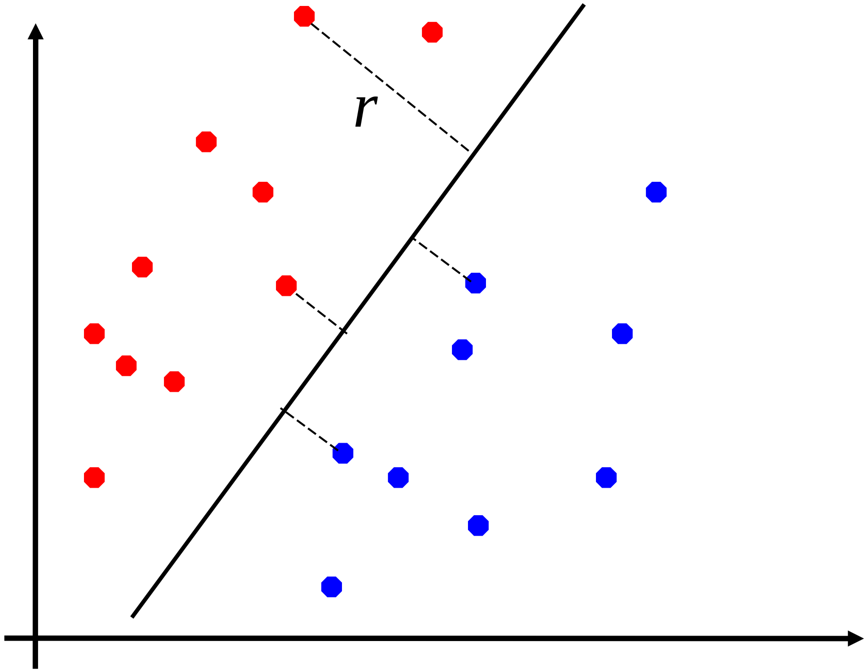
Linear Separators

Which of the linear separators is optimal?



Classification Margin

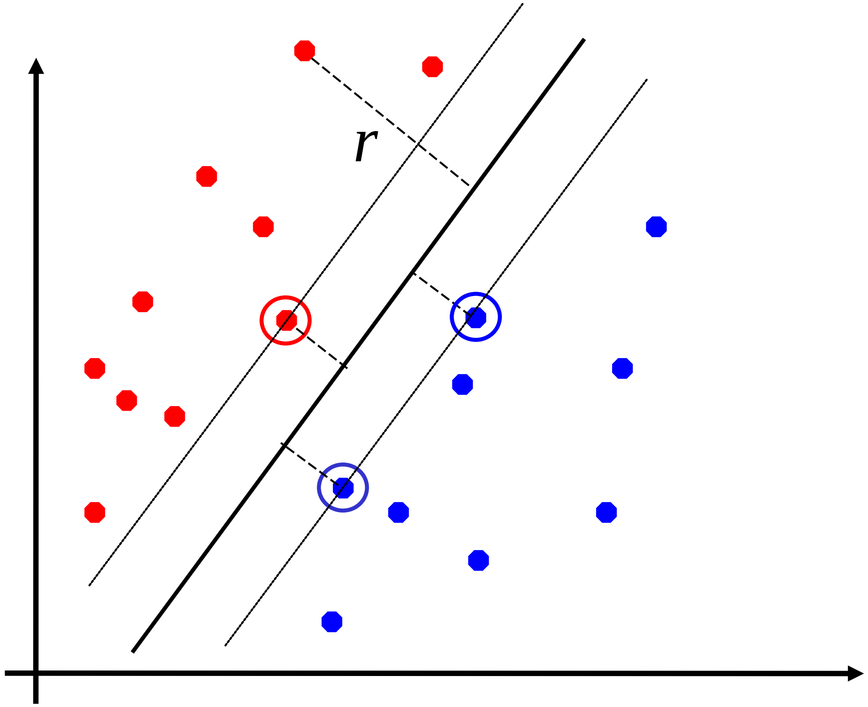
Distance from example $\mathbf{x}_i$ to the separator is $r = \dfrac{w^T x_i + b}{\|w\|}$



Classification Margin

Distance from example $\mathbf{x}_i$ to the separator is $r = \frac{w^T x_i + b}{\|w\|}$

Examples closest to the hyperplane are ***support vectors***.

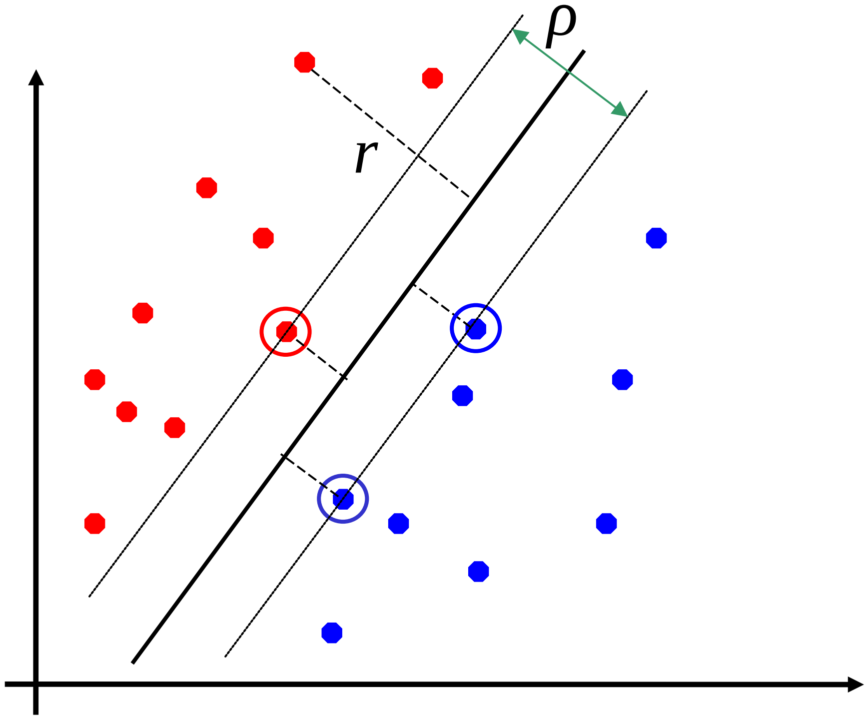


Classification Margin

Distance from example $\mathbf{x}_i$ to the separator is $r = \frac{w^T x_i + b}{\|w\|}$

Examples closest to the hyperplane are ***support vectors***.

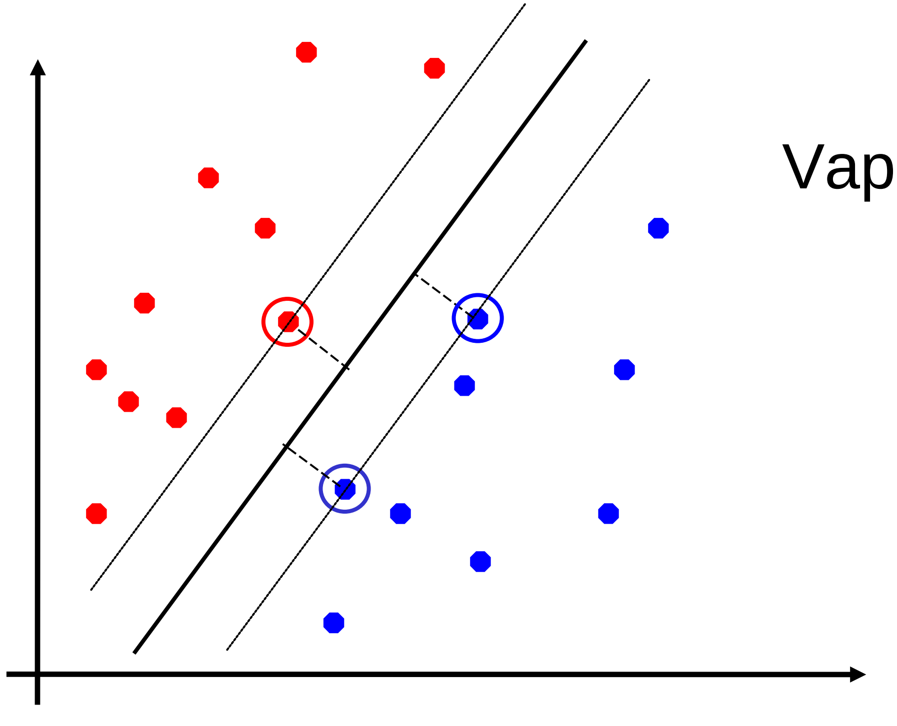
Margin ρ of the separator is the distance between support vectors.



Maximum Margin Classification

Maximizing the margin is good according to intuition and learning theory.

Implies that only support vectors matter; other training examples are ignorable.



Vapnik: Et< Ea + f(VC/m)

SVM formulation

- Training set:

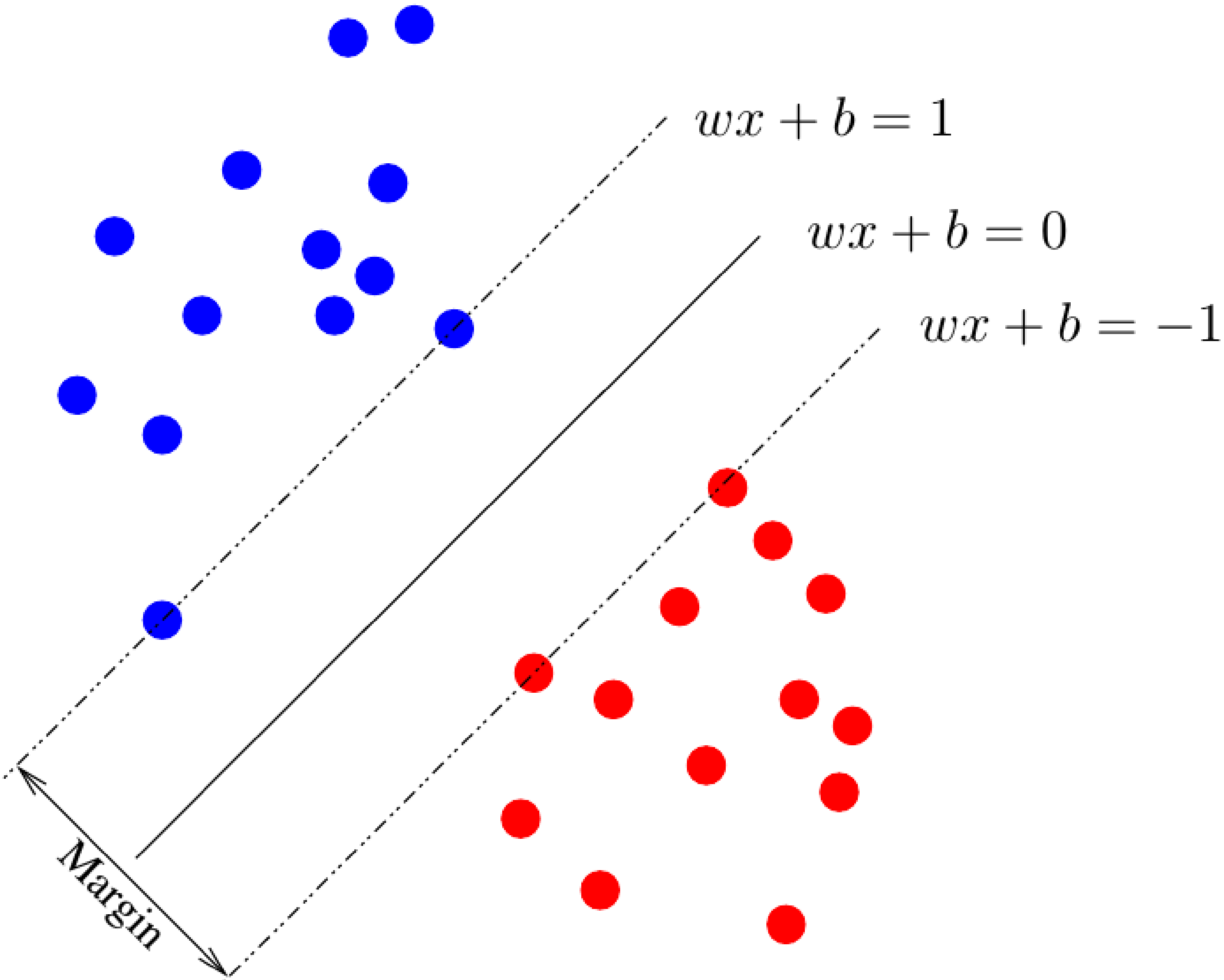
$$(x_t, y_t)_{t=1\dots T} \in \mathbb{R}^d \times \{-1, 1\}$$

- We would like to find *one* hyperplane

$$wx + b = 0 \quad (w \in \mathbb{R}^d,\ b \in \mathbb{R})$$

which separates the two classes and maximizes the margin.

SVM formulation



SVM formulation

- Margin to *maximize*:

$$\mathrm{dist}(wx + b = 1,\ wx + b = -1) = \frac{2}{\|w\|}$$

- We would like to minimize:

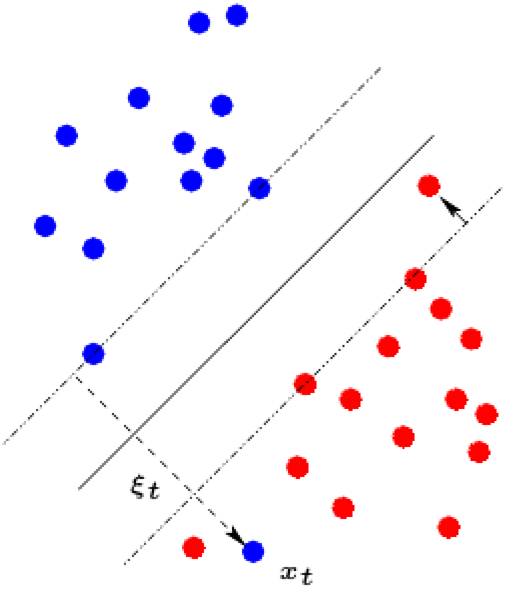
$$J(w,\ b) = \frac{\|w\|^2}{2}$$

Under the constraints:

$$y_t(wx_t + b) \geq 1 \quad \forall t$$

SVM formulation

This minimization problem does not have any solution if the two classes are not separable.



SVM formulation

- Relax the constraints: use a soft margin instead of a hard margin.
- We would like to minimize:

$$J(w, b, \xi) = \frac{\|w\|^2}{2} + C \sum_{t=1}^{T} \xi_t$$

Under the constraints:

$$y_t(wx_t + b) \geq 1 - \xi_t \quad \forall t$$

$$\xi_t \geq 0 \quad \forall t$$

SVM formulation

- We want to find u such that:

$$J(u) = \inf_{v \in U} J(v)$$

$$u \in U = \{v \in \mathbb{R}^n \;:\; \varphi_i(v) \leq 0 \quad \forall i\}$$

- Introduce the Lagrangian:

$$L(v,\, \mu) = J(v) + \sum_i \mu_i \, \varphi_i(v) \qquad (\mu_i \geq 0)$$

SVM formulation

- Theorem: If (u, λ) is a saddle point of the Lagrangian L, then (u, λ) is a solution of the constrained minimization problem.

- (u, λ) is a saddle point of the function L if u is a minimum for the function $v \mapsto L(v, \lambda)$ and λ is a maximum for the function $\mu \mapsto L(u, \mu)$.

SVM formulation

- Our Lagrangian:

$$L(w, b, \xi, \alpha, \mu) = J(w, b, \xi) + \sum_t \alpha_t[1 - \xi_t - y_t(wx_t + b)] - \sum_t \mu_t \xi_t$$

$$= \frac{\|w\|^2}{2} + C\sum_{t=1}^{T} \xi_t + \sum_t \alpha_t[1 - \xi_t - y_t(wx_t + b)] - \sum_t \mu_t \xi_t$$

$$(\alpha_t \geq 0 \quad \text{and} \quad \mu_t \geq 0)$$

- Look for (w, b, ξ) minimum of L:

$$\frac{\partial L}{\partial w} = 0 \quad \Leftrightarrow \quad w = \sum_t \alpha_t y_t \, x_t$$

$$\frac{\partial L}{\partial b} = 0 \quad \Leftrightarrow \quad \sum_t \alpha_t y_t = 0$$

$$\frac{\partial L}{\partial \xi} = 0 \quad \Leftrightarrow \quad C - \alpha_t - \mu_t = 0$$

SVM formulation

- Insert in the Lagrangian:

$$L = \sum_t \alpha_t - \frac{1}{2} \sum_{s,t} \alpha_s \alpha_t y_s y_t \, x_s x_t$$

$$0 \le \alpha_t \le C$$

$$\sum_t \alpha_t y_t = 0$$

$$w = \sum_t \alpha_t y_t \, x_t$$

- Look for $(\alpha, \, \mu)$ maximum of L:

$$\begin{aligned} \alpha_t [1 - \xi_t - y_t (w x_t + b)] &= 0 \\ \mu_t \xi_t &= 0 \end{aligned}$$

SVM formulation

- Finaly, we "just" have to minimize

$$\alpha \mapsto \frac{1}{2}\alpha^{\mathbf{T}} Q \alpha - \alpha^{\mathbf{T}} 1$$

where

$$Q_{ij} = y_i y_j \, x_i x_j$$

Under the constraints

$$0 \le \alpha_t \le C \quad \text{and} \quad \sum_t \alpha_t y_t = 0$$

- Then we obtain w and b with

$$w = \sum_t \alpha_t y_t \, x_t$$

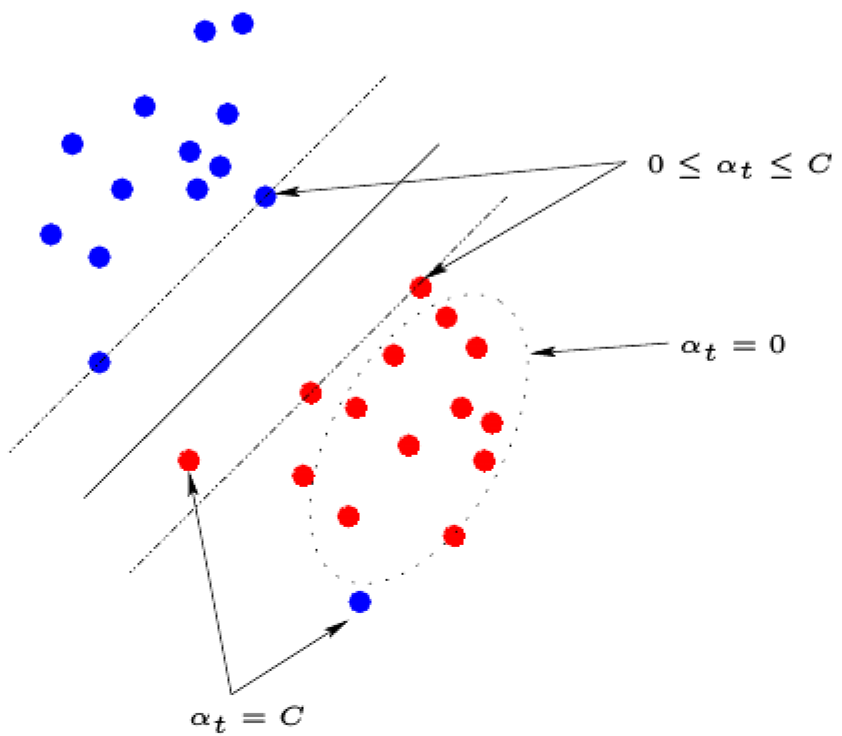
$$\alpha_t [1 - \xi_t - y_t(w x_t + b)] = 0$$

SVM formulation - end

- Note that the decision function could be rewritten as:

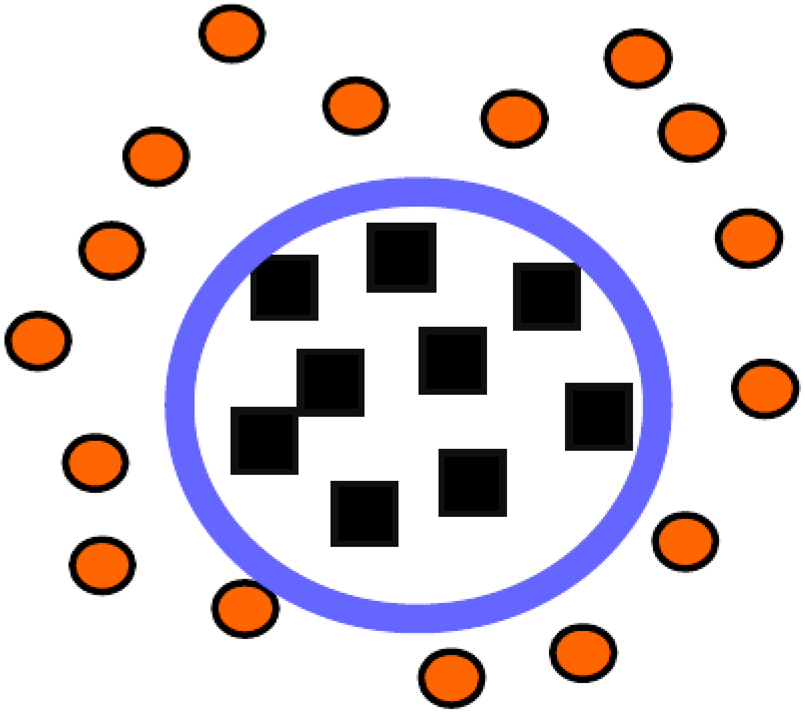
$$x \mapsto \sum_t \alpha_t y_t \, x_t x + b$$

- Training examples x_t with $\alpha_t \neq 0$ are support vectors.



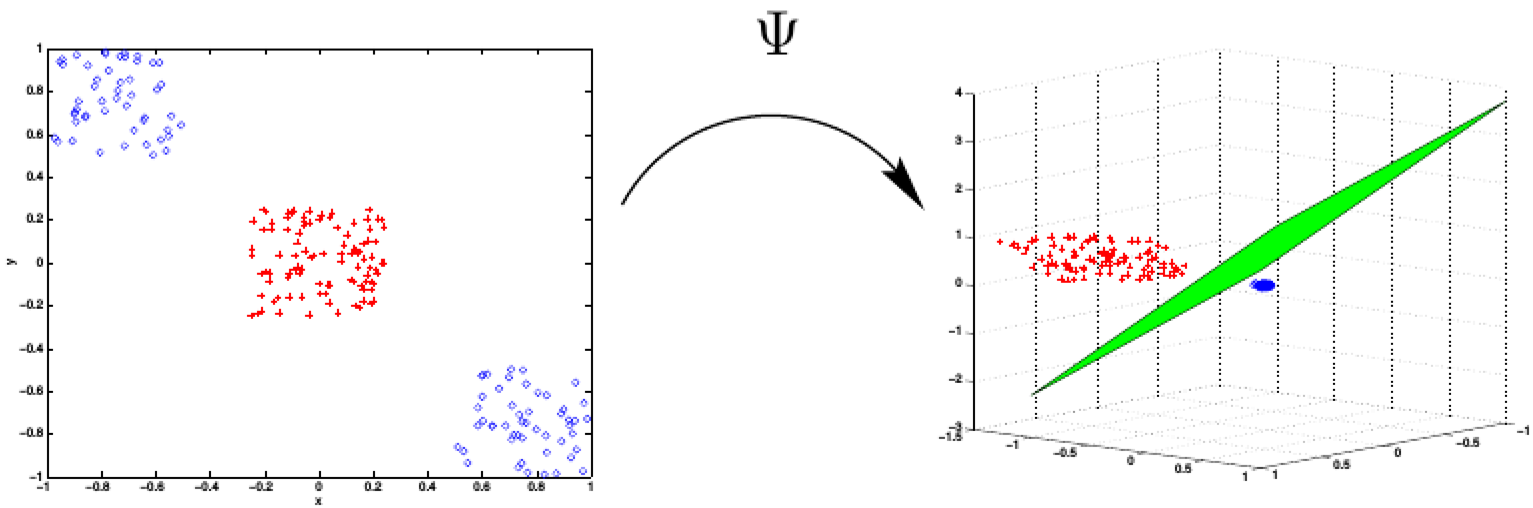
Kernels

What about this problem?



Kernels

- Project the data into a higher dimensional space: it should be easier to separate the two classes.
- Given a function $\Psi : \mathbb{R}^d \rightarrow F$, work with $\Psi(x_t)$ instead of working with x_t.



Kernels

- Note that we have only dot products $\Psi(x_s)\Psi(x_t)$ to compute.
- Unfortunately, it could be expensive in a high dimensional space.
- Use instead a kernel: a function $(x, z) \mapsto k(x, z)$ which represents a dot product in a "hidden" feature space.

$$k(x, z) = \Psi(x)\Psi(z)$$

- Example: instead of

$$\Psi(x) = \begin{pmatrix} x_1^2 \\ \sqrt{2}x_1x_2 \\ x_2^2 \end{pmatrix}$$

use

$$k(x, z) = (xz)^2$$

Kernels

- Polynomial:

$$k(x,\, z) = (u\, xz + v)^p \quad (u \in \mathbb{R},\ v \in \mathbb{R},\ p \in \mathbb{N}_+^*)$$

- Gaussian:

$$k(x,\, z) = \exp\left(-\frac{\|x - z\|^2}{2\sigma^2}\right) \quad (\sigma \in \mathbb{R}_+^*)$$

- ⚠ The function

$$k(x,\, z) = \tanh(u\, xz + v)$$

is not a kernel!

Kernels

- Any symmetric positive-definite kernel f(u,v) is a dot product in some space. Not matter what is the space.
- Kernel algebra → linear combinations of kernels are kernels
- Open door: kernels for non-vectorial objects

Using SVMs

- Choose a kernel $k()$.
- Minimize

$$\alpha \;\mapsto\; \frac{1}{2}\alpha^{\mathbf{T}} Q \alpha - \alpha^{\mathbf{T}} 1$$

where

$$Q_{ij} = y_i y_j \, k(x_i, x_j)$$

Under the constraints

$$0 \;\leq\; \alpha_t \;\leq\; C \quad \text{and} \quad \sum_t \alpha_t y_t = 0$$

- For $0 < \alpha_t < C$, compute b using

$$1 - y_t \left[\sum_s \alpha_s y_s \, k(x_s, x_t) + b \right] \;=\; 0$$

Using SVMs

- The decision function will be

$$x \;\mapsto\; \operatorname{sign}\left(\sum_t \alpha_t y_t \, k(x_t, x) + b\right)$$

Summary

- SVMs maximize the margin (*in the feature space*)
- Use the soft margin trick
- Project the data into a higher dimensional space for non-linear relations
- Kernels simplify the computation
- A Lagrangian method leads to a "nice" quadratic minimization problem under constraints.

In practice

- In order to tune the capacity, the kernel is the most important parameter to choose.
  - Polynomial kernel: increasing the degree will increase the capacity.
  - Gaussian kernel: increasing σ will decrease the capacity.
- Tune C, the trade-off between the margin and the errors.
  - For non-noisy data sets, C usually has not much influence.
  - Carefully choose C for noisy data sets: small values usually give better results.

Otros problemas con kernels

Other methods

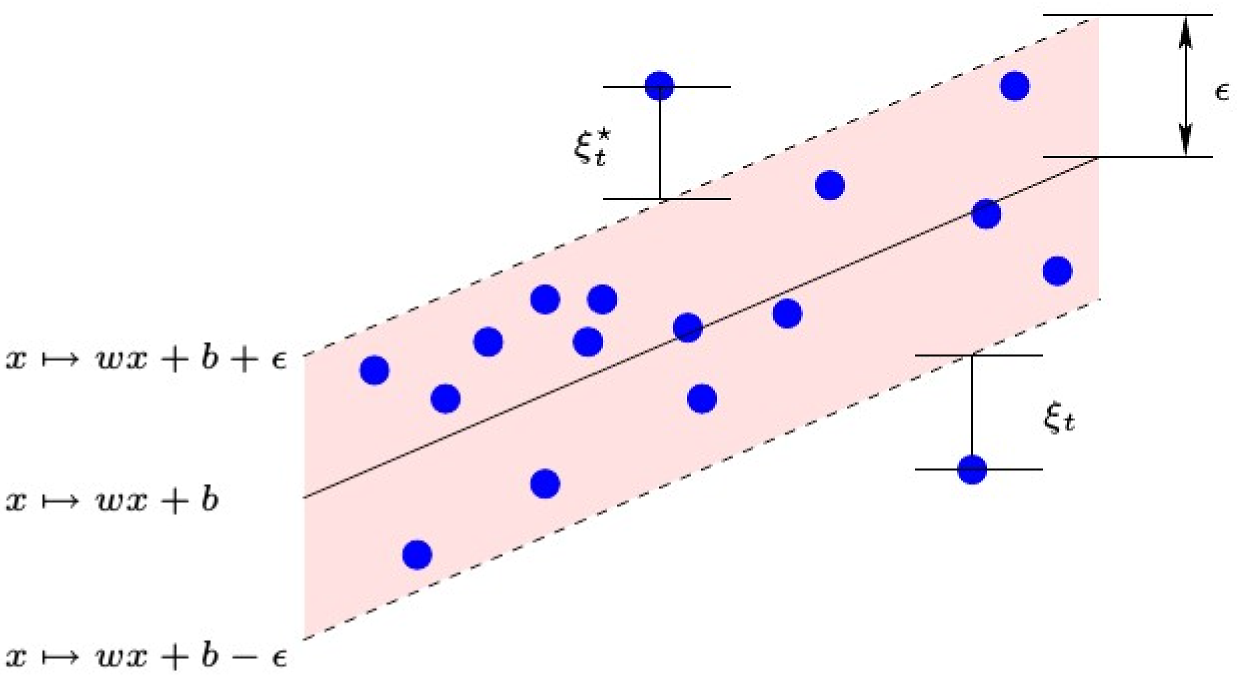
- Any Machine Learning method that only depends on inner products of the data can use kernels
- Lots of methods: kernel-pca, kernel regression, kernel-...

Multiclass classification

- Use ensembles: OVA, OVO. Ovo is more efficient
- There are some direct multiclass SVM formulations, not better than OVO.
- Lots of papers, diverse results

Regression

- We are looking for an hyperplane $x \mapsto wx + b$ such that...



Regression

- We would like to minimize
$$\frac{1}{2}\|w\|^2 + C\sum_t |wx_t + b - y_t|_\epsilon$$
where
$$|u|_\epsilon = \max(0, |u| - \epsilon)$$
- "Epsilon insensitive loss": we "ignore" errors lower than ϵ.
- Equivalent to minimize
$$\frac{1}{2}\|w\|^2 + C\sum_t (\xi_t + \xi_t^\star)$$
under the constraints
$$(wx_t + b) - y_t \leq \epsilon + \xi_t$$
$$y_t - (wx_t + b) \leq \epsilon + \xi_t^\star$$
$$\xi_t,\ \xi_t^\star \geq 0$$

Regression

- Non-linear regression via kernels
- A new parameter to set: the tube

Novelty detection

- Classical: use a density function, points below a threshold are outliers
- Two kernel versions

Novelty detection

- Tax & Duin: Find the minimal hypersphere that contains all the data, points outside are outliers

$$\min_{R,z,a} \quad R^2 + \frac{1}{m\nu}\sum_{i=1}^{m} z_i$$

$$s.t. \quad (x_i - a)^T (x_i - a) \le R^2 + z_i$$

$$z_i \ge 0 \quad i = 1,..,m$$

$$\longrightarrow$$

$$\min_{\alpha} \quad -\sum_{i=1}^{m} \alpha_i K(x_i, x_i) + \sum_{i,j=1}^{m} \alpha_i \alpha_j K(x_i, x_j)$$

$$s.t. \quad \sum_{i=1}^{m} \alpha_i = 1$$

$$\frac{1}{m\nu} \ge \alpha_i \ge 0 \quad i = 1,..,m$$

Outlier: $K(v,v) - 2\sum_{i=1}^{m} \alpha_i K(v, x_i) + \sum_{i,j=1}^{m} \alpha_i \alpha_j K(x_i, x_j) - R^2 \ge 0$

Novelty detection

- Scholkopf et al.: Only for Gaussian Kernel, find the hyperplane with max distance to the origin that left all points in one side.

$$\min_{\alpha} \quad \sum_{i,j=1}^{m} \alpha_i \alpha_j K(x_i, x_j)$$

$$s.t. \quad \sum_{i=1}^{m} \alpha_i = 1$$

$$\frac{1}{m\nu} \geq \alpha_i \geq 0 \quad i = 1,..,m$$

Outlier:

$$f(z) = sign\left(\sum_{j=1}^{m} \alpha_j K(x_j, v) - b \right)$$

Code

- Some examples in classification (R code)