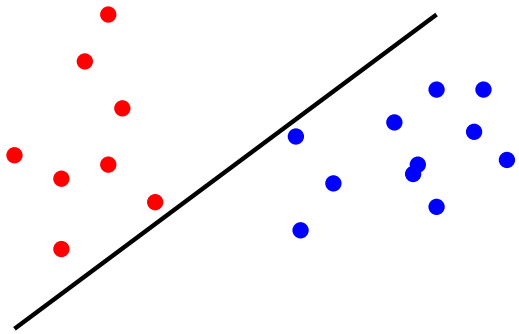
# Maximum Margin Classifiers

# MAXIMUM MARGIN IDEA

## Setting

Linear classification, two linearly separable classes.



## Recall Perceptron

- Selects *some* hyperplane between the two classes.
- Choice depends on initialization, step size etc.

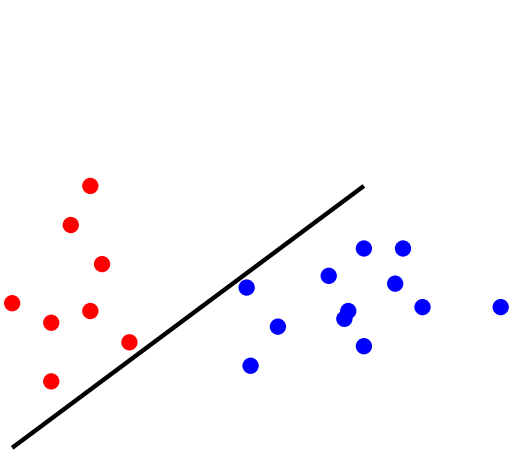
## Maximum margin idea

To achieve good generalization (low prediction error), place the hyperplane "in the middle" between the two classes.

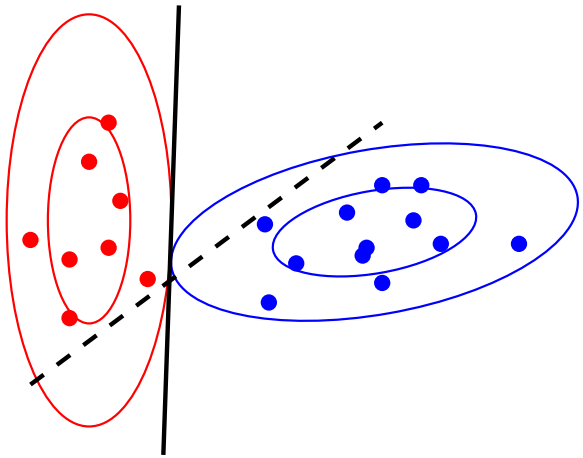
## More precisely

Choose plane such that distance to closest point in each class is maximal. This distance is called the *margin*.

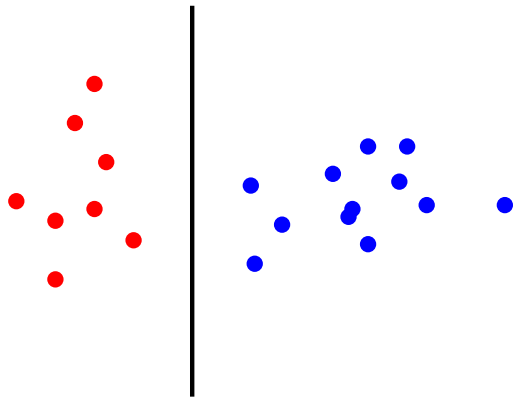
# Generalization Error



Possible Perceptron solution



Good generalization under a specific distribution (here: Gaussian)



Maximum margin solution

## Example: Gaussian data

- The ellipses represent lines of constant standard deviation (1 and 2 STD respectively).
- The 1 STD ellipse contains $\sim 65\%$ of the probability mass ($\sim 95\%$ for 2 STD; $\sim 99.7\%$ for 3 STD).

**Optimal generalization:** Classifier should cut off as little probability mass as possible from either distribution.

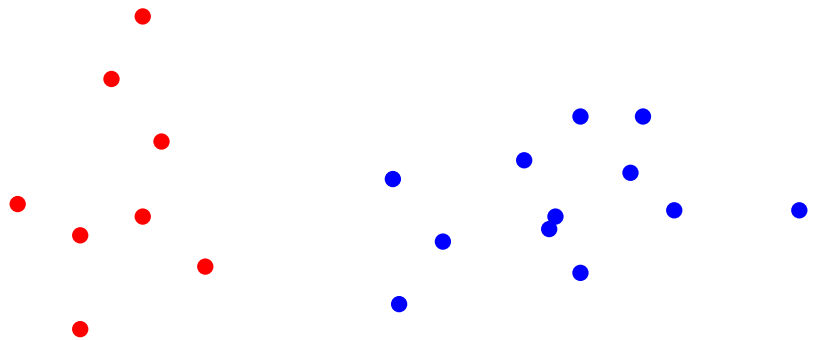
## Without distributional assumption: Max-margin classifier

- Philosophy: Without distribution assumptions, best guess is symmetric.
- In the Gaussian example, the max-margin solution would *not* be optimal.

# SUBSTITUTING CONVEX SETS

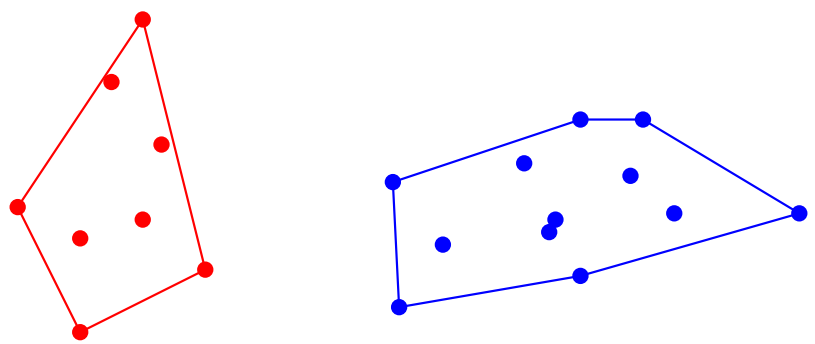
## Observation

Where a separating hyperplane may be placed depends on the "outer" points on the sets. Points in the center do not matter.



## In geometric terms

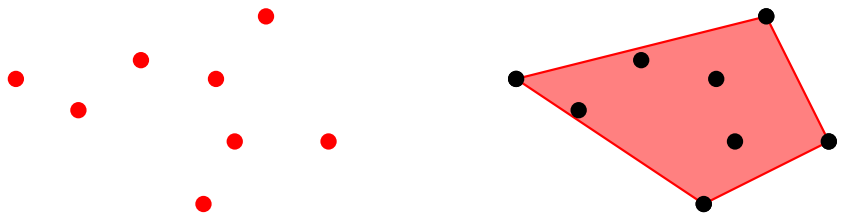
Substitute each class by the smalles convex set which contains all point in the class:



# SUBSTITUTING CONVEX SETS

## Definition

If $C$ is a set of points, the smallest convex set containing all points in $C$ is called the **convex hull** of $C$, denoted $\text{conv}(C)$.

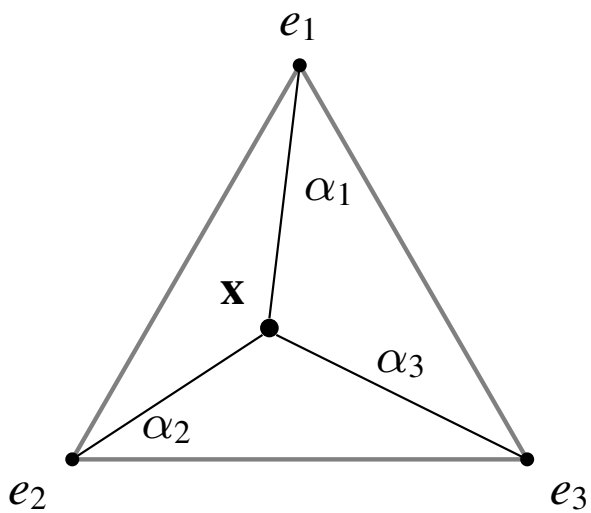


Corner points of the convex set are called **extreme points**.

## Barycentric coordinates

Every point $x$ in a convex set can be represented as a convex combination of the extreme points $\{e_1, \ldots, e_m\}$. There are weights $\alpha_1, \ldots, \alpha_m \in \mathbb{R}_+$ such that

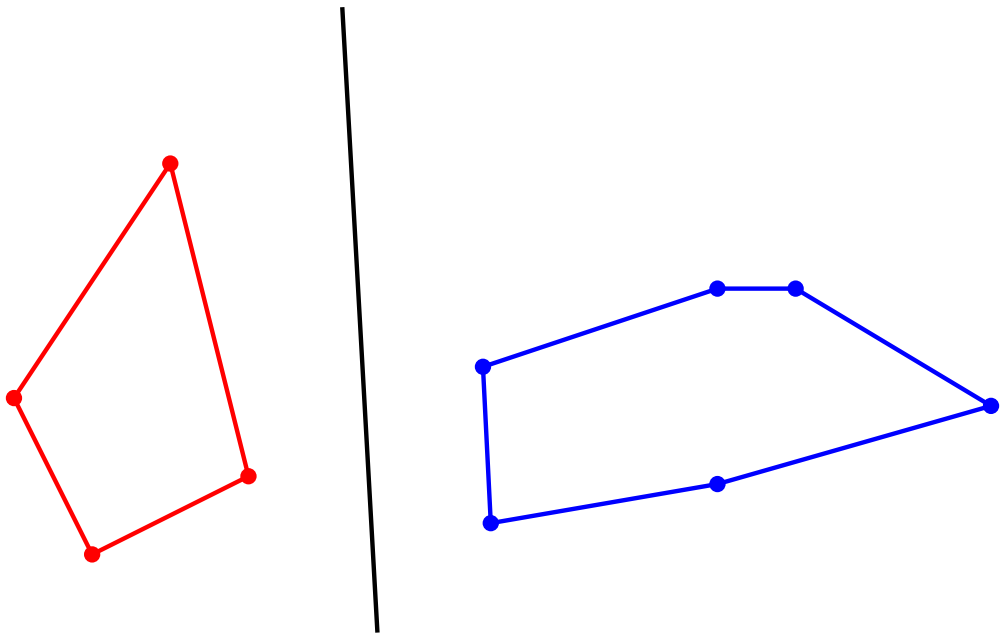
$$\mathbf{x} = \sum_{i=1}^{m} \alpha_i e_i \quad \text{and} \quad \sum_{i=1}^{m} \alpha_i = 1 \ .$$




The coefficients $\alpha_i$ are called **barycentric coordinates** of $x$.

## Key idea

A hyperplane separates two classes if and only if it separates their convex hull.



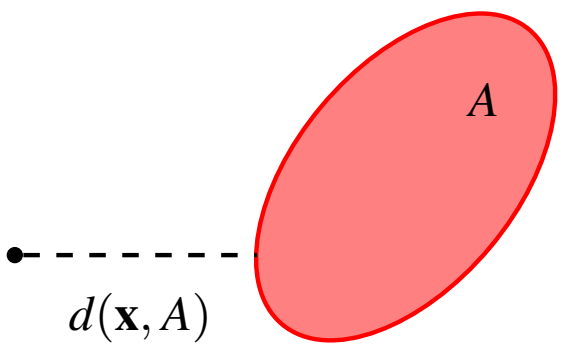
**Next:** We have to formalize what it means for a hyperplane to be "in the middle" between to classes.

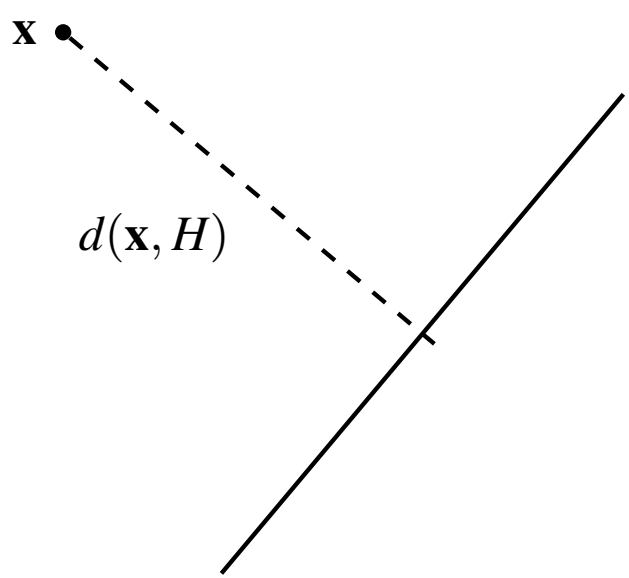
## Definition

The **distance** between a point $\mathbf{x}$ and a set $A$ the Euclidean distance between $x$ and the closest point in $A$:

$$d(\mathbf{x}, A) := \min_{\mathbf{y} \in A} \|\mathbf{x} - \mathbf{y}\|$$

In particular, if $A = H$ is a hyperplane, $d(\mathbf{x}, H) := \min_{\mathbf{y} \in H} \|\mathbf{x} - \mathbf{y}\|$.



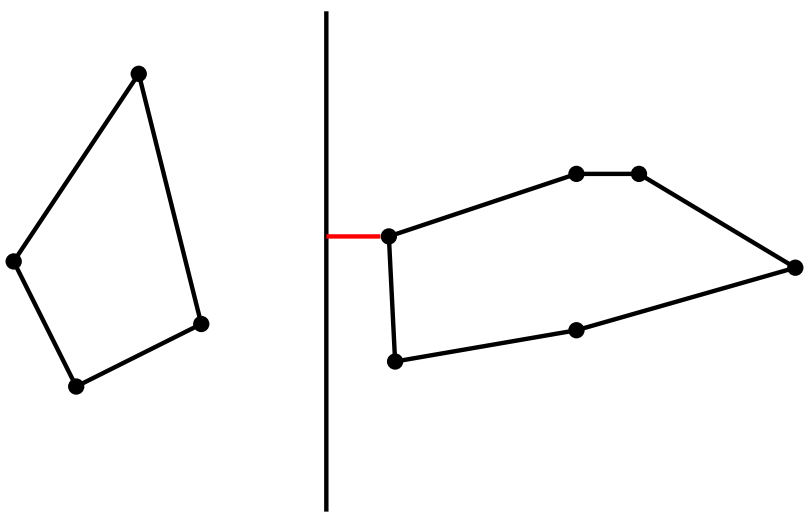
## Definition

The **margin** of a classifier hyperplane $H$ given two training classes $\mathcal{X}_1, \mathcal{X}_2$ is the shortest distance between the plane and any point in either set:

$$\text{margin} = \min_{x \in \mathcal{X}_1 \cup \mathcal{X}_2} d(x, H)$$

Equivalently: The shortest distance to either of the convex hulls.

$$\text{margin} = \min\{d(H, \text{conv}(\mathcal{X}_1)), d(H, \text{conv}(\mathcal{X}_2))\}$$



Idea in the following: $H$ is "in the middle" when margin maximal.

## Recall: Specifying affine plane

Normal vector $\mathbf{v}_H$.

$$\langle \mathbf{v}_H, \mathbf{x} \rangle - c \begin{cases} > 0 & \mathbf{x} \text{ on positive side} \\ < 0 & \mathbf{x} \text{ on negative side} \end{cases}$$

Scalar $c \in \mathbb{R}$ specifies shift (plane through origin if $c = 0$).

## Plane with margin

Demand

$$\langle \mathbf{v}_H, \mathbf{x} \rangle - c > 1 \text{ or } < -1$$

$\{-1, 1\}$ on the right works for any margin: Size of margin determined by $\|\mathbf{v}_H\|$. To increase margin, scale down $\mathbf{v}_H$.

## Classification

Concept of margin applies only to training, not to classification. Classification works as for any linear classifier. For a test point $\mathbf{x}$:

$$y = \text{sign}\left(\langle \mathbf{v}_H, \mathbf{x} \rangle - c\right)$$

## Finding the hyperplane

For $n$ training points $(\tilde{\mathbf{x}}_i, \tilde{y}_i)$ with labels $\tilde{y}_i \in \{-1, 1\}$, solve optimization problem:

$$\begin{aligned} \min_{\mathbf{v}_\mathrm{H}, c} \quad & \|\mathbf{v}_\mathrm{H}\| \\ \text{s.t.} \quad & \tilde{y}_i(\langle \mathbf{v}_\mathrm{H}, \tilde{\mathbf{x}}_i \rangle - c) \geq 1 \quad \text{for } i = 1, \ldots, n \end{aligned}$$

## Definition

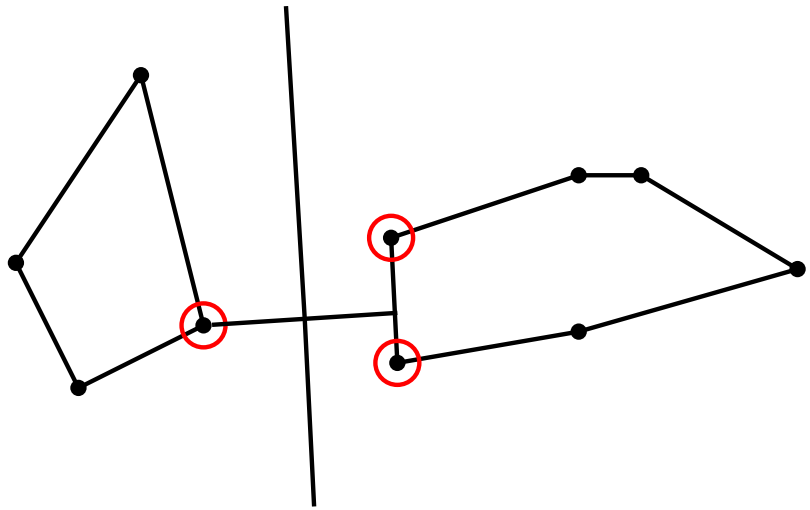
The classifier obtained by solving this optimization problem is called a **support vector machine**.

# Support Vectors

## Definition

Those extreme points of the convex hulls which are closest to the hyperplane are called the **support vectors**.

There are at least two support vectors, one in each class.



## Implications

- The maximum-margin criterion focuses all attention to the area closest to the decision surface.
- Small changes in the support vectors can result in significant changes of the classifier.
- In practice, the approach is combined with "slack variables" to permit overlapping classes. As a side effect, slack variables soften the impact of changes in the support vectors.

# Dual Optimization Problem

Solving the SVM opimization problem

$$\begin{aligned} \min_{\mathbf{v}_H, c} \quad & \|\mathbf{v}_H\| \\ \text{s.t.} \quad & \tilde{y}_i(\langle \mathbf{v}_H, \tilde{\mathbf{x}}_i \rangle - c) \geq 1 \quad \text{for } i = 1, \ldots, n \end{aligned}$$

is difficult, because the constraint is a function. It is possible to transform this problem into a problem which seems more complicated, but has simpler constraints:

$$\begin{aligned} \max_{\boldsymbol{\alpha} \in \mathbb{R}^n} \quad & W(\boldsymbol{\alpha}) := \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j \tilde{y}_i \tilde{y}_j \langle \tilde{\mathbf{x}}_i, \tilde{\mathbf{x}}_j \rangle \\ \text{s.t.} \quad & \sum_{i=1}^{n} \tilde{y}_i \alpha_i = 0 \\ & \alpha_i \geq 0 \quad \text{for } i = 1, \ldots, n \end{aligned}$$

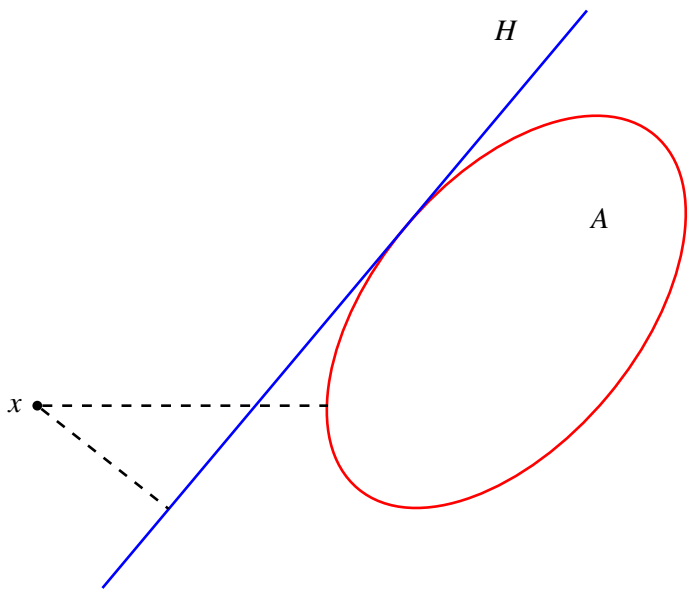
This is called the optimization problem **dual** to the minimization problem above. It is usually derived using Lagrange multipliers. We will use a more geometric argument.

## Sets and Planes

Many dual relations in convex optimization can be traced back to the following fact:

*The closest distance between a point $x$ and a convex set $A$ is the maximum over the distances between $x$ and all hyperplanes which separate $x$ and $A$.*

$$d(x, A) = \sup_{H \text{ separating}} d(x, H)$$



# Deriving the Dual Problem

## Idea

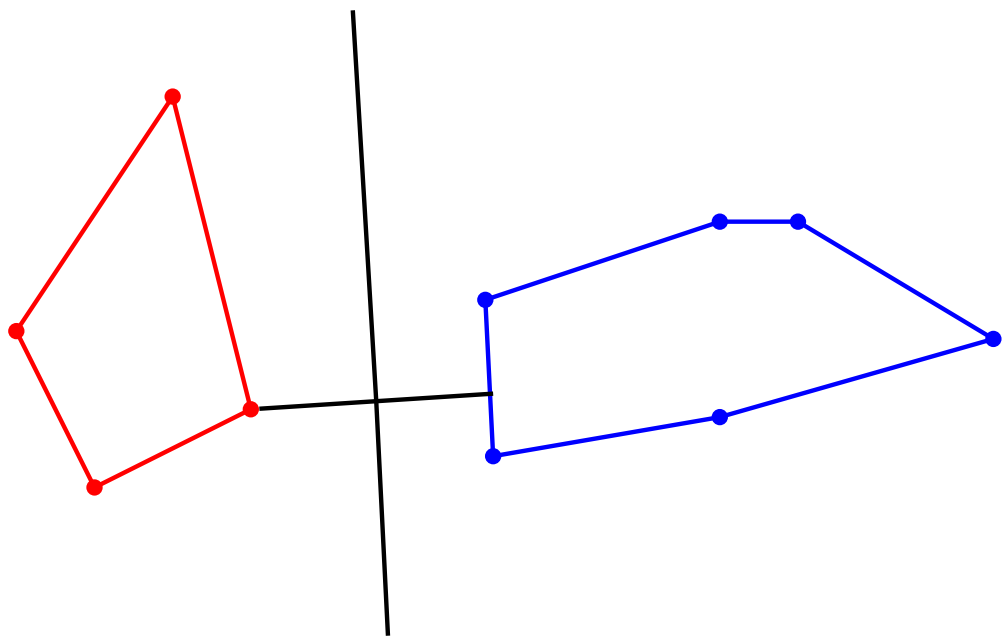
As a consequence of duality on previous slide, we can find the maximum-margin plane as follows:

1. Find shortest line connecting the convex hulls.
2. Place classifier orthogonal to line in the middle.

Convexity of sets ensures that this classifier has correct orientation.

## As optimization problem

$$\min_{\substack{\mathbf{u}\in\text{conv}(\mathcal{X}_1)\\ \mathbf{v}\in\text{conv}(\mathcal{X}_2)}} \quad \|\mathbf{u}-\mathbf{v}\|^2$$



## Dual optimization problem

$$\min_{\substack{\mathbf{u}\in\text{conv}(\mathcal{X}_1)\\ \mathbf{v}\in\text{conv}(\mathcal{X}_2)}} \quad \|\mathbf{u}-\mathbf{v}\|^2$$

The points $\mathbf{u}$ and $\mathbf{v}$ are in the convex hulls, and can be represented by barycentric coordinates:

$$\mathbf{u} = \sum_{i=1}^{n_1} \alpha_i \tilde{\mathbf{x}}_i \qquad \mathbf{v} = \sum_{i=n_1+1}^{n_1+n_2} \alpha_i \tilde{\mathbf{x}}_i \qquad \text{(where } n_1 = |\mathcal{X}_1|, n_2 = |\mathcal{X}_2|\text{)}$$

Substitute into minimization problem:

$$\begin{aligned} \min_{\alpha_1,\ldots,\alpha_n} \quad & \|\sum_{i\in C_1} \alpha_i \tilde{\mathbf{x}}_i - \sum_{i\in C_2} \alpha_i \tilde{\mathbf{x}}_i\|_2^2 \\ \text{s.t.} \quad & \sum_{i\in C_1} \alpha_i = \sum_{i\in C_2} \alpha_i = 1 \\ & \alpha_i \geq 0 \end{aligned}$$

## Dual problem

$$\begin{aligned} \| \sum_{i \in C_1} \alpha_i \tilde{\mathbf{x}}_i - \sum_{i \in C_2} \alpha_i \tilde{\mathbf{x}}_i \|_2^2 &= \| \sum_{i \in C_1} \tilde{y}_i \alpha_i \tilde{\mathbf{x}}_i + \sum_{i \in C_2} \tilde{y}_i \alpha_i \tilde{\mathbf{x}}_i \|_2^2 \\ &= \left\langle \sum_{i=1}^{n} \tilde{y}_i \alpha_i \tilde{\mathbf{x}}_i \, , \, \sum_{i=1}^{n} \tilde{y}_i \alpha_i \tilde{\mathbf{x}}_i \right\rangle = \sum_{i,j} \tilde{y}_i \tilde{y}_j \alpha_i \alpha_j \left\langle \tilde{\mathbf{x}}_i \middle| \tilde{\mathbf{x}}_j \right\rangle \end{aligned}$$

Note: Minimizing this term under the constraints is equivalent to *maximizing*

$$\sum_i \alpha_i - \frac{1}{2} \sum_{i,j} \tilde{y}_i \tilde{y}_j \alpha_i \alpha_j \left\langle \tilde{\mathbf{x}}_i \middle| \tilde{\mathbf{x}}_j \right\rangle$$

under the same constraints, since $\sum_i \alpha_i = 2$ is constant. That is just the dual problem defined four slides back.

# COMPUTING $c$

## Output of dual problem

$\mathbf{v}_{\mathrm{H}}^* = \sum_{i=1}^n \tilde{y}_i \alpha_i^* \tilde{\mathbf{x}}_i$. This vector describes a non-affine hyperplane.

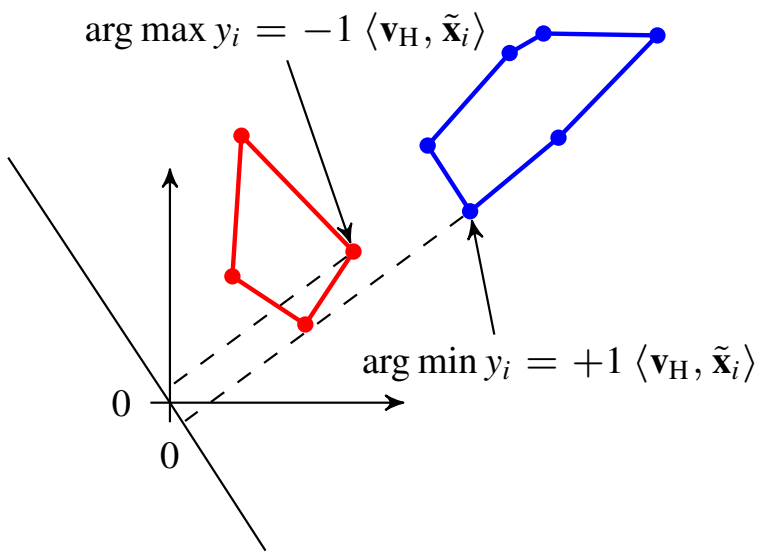
## Computing the offset

The offset $c$ is given by

$$c^* := -\frac{\max_{\tilde{y}_i=-1} \langle \mathbf{v}_{\mathrm{H}}^*, \tilde{\mathbf{x}}_i \rangle + \min_{\tilde{y}_i=+1} \langle \mathbf{v}_{\mathrm{H}}^*, \tilde{\mathbf{x}}_i \rangle}{2}$$

## Explanation

- The max and min are computed with respect to the $\mathbf{v}_{\mathrm{H}}$ plane *containing the origin*.
- That means the max and min determine a support vector in each class.
- We then compute the shift as the mean of the two distances.



## Output of dual optimization

- Optimal values $\alpha_i^*$ for the variables $\alpha_i$
- If $\tilde{\mathbf{x}}_i$ support vector: $\alpha_i > 0$, if not: $\alpha_i = 0$

## SVM Classifier

The classification function can be expressed in terms of the variables $\alpha_i$:

$$f(\mathbf{x}) = \operatorname{sgn}\left(\sum_{i=1}^{n} \tilde{y}_i \alpha_i^* \langle \tilde{\mathbf{x}}_i, \mathbf{x} \rangle - c^*\right)$$

Intuitively: To classify a data point, it is sufficient to know which side of each support vector it is on.