COMS 4771 Lecture 8

1. Support vector machines
2. Support vectors and the SVM dual
3. Kernelizing SVMs (maybe next time)
4. Soft-margin SVMs and surrogate losses (maybe next time)

Support vector machines

Linear classifiers

Setting: two linearly separable classes

Assume there is a linear classifier with **zero training error** on S: for some $\boldsymbol{w}_\star \in \mathbb{R}^d$ and $\theta_\star \in \mathbb{R}$,

$$y(\langle \boldsymbol{w}_\star, \boldsymbol{x} \rangle - \theta_\star) > 0, \quad \text{for all } (x, y) \in S.$$

Linear classifiers

Setting: two linearly separable classes

Assume there is a linear classifier with **zero training error** on S: for some $\boldsymbol{w}_\star \in \mathbb{R}^d$ and $\theta_\star \in \mathbb{R}$,

$$y(\langle \boldsymbol{w}_\star, \boldsymbol{x} \rangle - \theta_\star) > 0, \quad \text{for all } (x, y) \in S.$$

Linear programming

Solve linear feasibility problem: find $\boldsymbol{w} \in \mathbb{R}^d$ and $\theta \in \mathbb{R}$ such that

$$y(\langle \boldsymbol{w}, \boldsymbol{x} \rangle - \theta) > 0, \quad \text{for all } (x, y) \in S.$$

[$d + 1$ variables, $|S|$ constraints.] Can find *some* solution in polynomial time.

Linear classifiers

Setting: two linearly separable classes

Assume there is a linear classifier with **zero training error** on S: for some $\boldsymbol{w}_\star \in \mathbb{R}^d$ and $\theta_\star \in \mathbb{R}$,

$$y(\langle \boldsymbol{w}_\star, \boldsymbol{x} \rangle - \theta_\star) > 0, \quad \text{for all } (x, y) \in S.$$

Linear programming

Solve linear feasibility problem: find $\boldsymbol{w} \in \mathbb{R}^d$ and $\theta \in \mathbb{R}$ such that

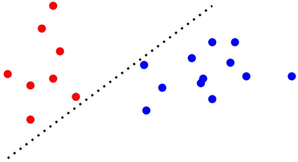
$$y(\langle \boldsymbol{w}, \boldsymbol{x} \rangle - \theta) > 0, \quad \text{for all } (x, y) \in S.$$

[$d + 1$ variables, $|S|$ constraints.] Can find *some* solution in polynomial time.

Perceptron algorithm

as if there is a margin

Assuming $\boxed{\gamma(S) > 0}$ finds *some* linear classifier that separates the classes in S *very quickly*.



Support vector machines (SVMs)

Motivation

- Ambiguity in what LP and Perceptron returns, even when $\gamma(S) > 0$.
- Returning an arbitrary linear classifier that separates the classes in S is not very *stable*.
- Unclear what to do (or if algorithms work) when S is not linearly separable, but you still hope to find a good linear separator with small classification error.

Support vector machines (SVMs)

Motivation

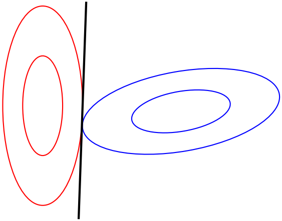
also include different data

- Ambiguity in what LP and Perceptron returns, even when $\gamma(S) > 0$.
- Returning an arbitrary linear classifier that separates the classes in S is not very *stable*. depends on input sequence, may not the maximum margin solution
- Unclear what to do (or if algorithms work) when S is not linearly separable, but you still hope to find a good linear separator with small classification error.

Support vector machines (Vapnik and Chervonenkis, 1963)

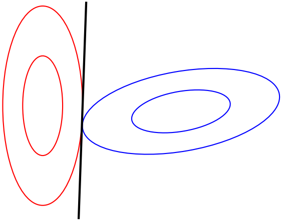
- Characterize a *stable* solution for linearly separable problems—the **maximum margin solution**.
- SVM specified as solution to a **convex optimization problem** that can be solved in polynomial time.
- Kernelizable via **convex duality**. (SVM gets its name from its dual form.)
- Slight alteration to optimization problem gives natural way to handle non-separable cases via **convex surrogate losses**.

Maximum margin solution

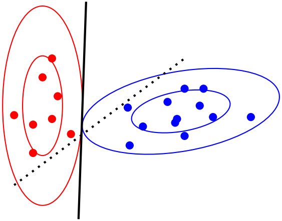


Best linear classifier on population

Maximum margin solution

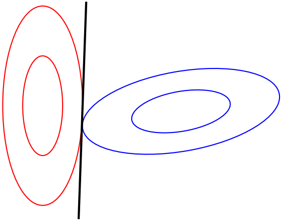


Best linear classifier on population

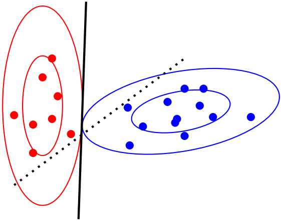


Possible Perceptron or LP solution on training data S

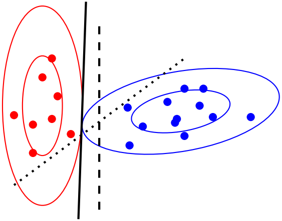
Maximum margin solution



Best linear classifier on population

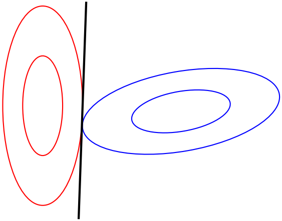


Possible Perceptron or LP solution on training data S

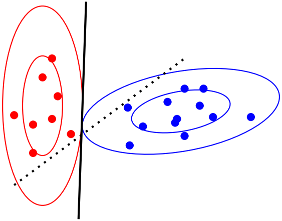


"Maximum margin" solution on training data S

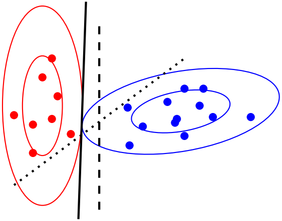
Maximum margin solution



Best linear classifier on population



Possible Perceptron or LP solution on training data S



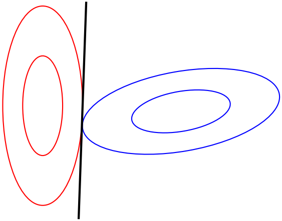
"Maximum margin" solution on training data S

Why use the "maximum margin" solution?

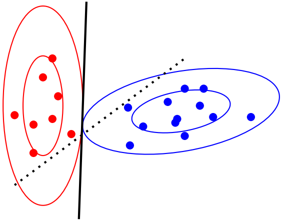
(i) Uniquely determined by S (except in degenerate cases), unlike LP's/Perceptron's.

(ii) It is a particular "learning bias"—i.e., an assumption about the problem—that seems to be commonly useful.

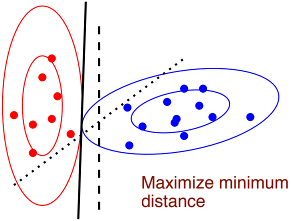
Maximum margin solution



Best linear classifier on population



Possible Perceptron or LP solution on training data S



"Maximum margin" solution on training data S

Why use the "maximum margin" solution?

(i) Uniquely determined by S (except in degenerate cases), unlike LP's/Perceptron's.

(ii) It is a particular "learning bias"—i.e., an assumption about the problem—that seems to be commonly useful.

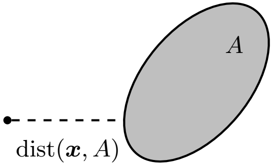
Next: mathematically characterize the "maximum margin" solution.

Distances to sets

Definition

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

$$\operatorname{dist}(\boldsymbol{x}, A) := \min_{\boldsymbol{x}' \in A} \|\boldsymbol{x} - \boldsymbol{x}'\|$$

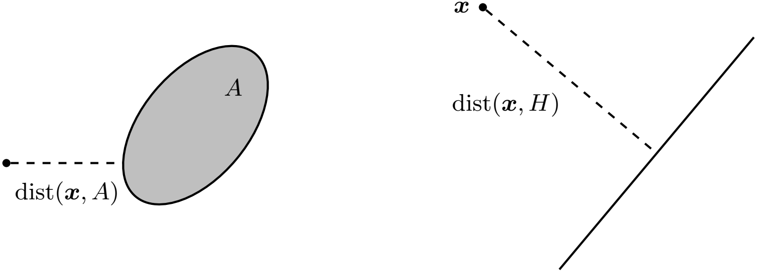


Distances to sets

Definition

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

$$\operatorname{dist}(\boldsymbol{x}, A) := \min_{\boldsymbol{x}' \in A} \|\boldsymbol{x} - \boldsymbol{x}'\|$$


A
$\operatorname{dist}(\boldsymbol{x}, A)$
$\boldsymbol{x}$
$\operatorname{dist}(\boldsymbol{x}, H)$

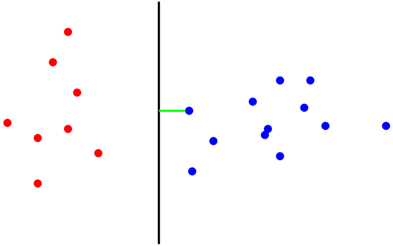
Margin

If $f_{\boldsymbol{w},\theta}$ linearly separates S, then we say its **margin on S** is

$$\min_{\boldsymbol{x} \in S_{\oplus} \cup S_{\ominus}} \operatorname{dist}(\boldsymbol{x}, H)$$

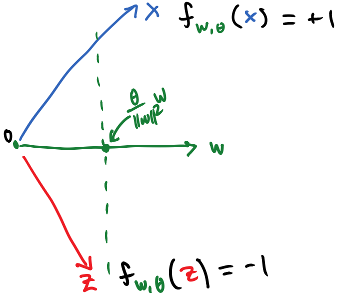
where

$$\begin{aligned} H &:= \text{affine hyperplane corresponding to } f_{\boldsymbol{w},\theta}, \\ S_{\oplus} &:= \{\boldsymbol{x} : (\boldsymbol{x}, +1) \in S\} \quad \text{(positive points)}, \\ S_{\ominus} &:= \{\boldsymbol{x} : (\boldsymbol{x}, -1) \in S\} \quad \text{(negative points)}. \end{aligned}$$



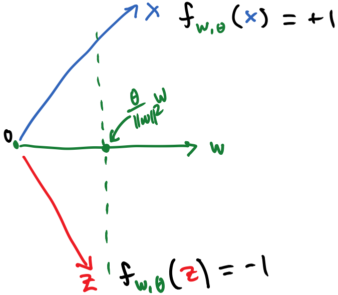
Distance to the boundary

Consider linear classifier $f_{\boldsymbol{w},\theta}$ (where $\boldsymbol{w} \in \mathbb{R}^d \setminus \{\mathbf{0}\}$ and $\theta \in \mathbb{R}$).



Distance to the boundary

Consider linear classifier $f_{\boldsymbol{w},\theta}$ (where $\boldsymbol{w} \in \mathbb{R}^d \setminus \{\boldsymbol{0}\}$ and $\theta \in \mathbb{R}$).



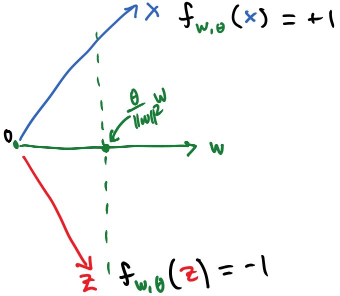
- Correct classification on $(\boldsymbol{x}, y)$:

$$f_{\boldsymbol{w},\theta}(\boldsymbol{x}) = y \quad \text{iff} \quad y(\langle \boldsymbol{w}, \boldsymbol{x} \rangle - \theta) > 0.$$

(Ignore boundary case.)

Distance to the boundary

Consider linear classifier $f_{\boldsymbol{w},\theta}$ (where $\boldsymbol{w} \in \mathbb{R}^d \setminus \{\boldsymbol{0}\}$ and $\theta \in \mathbb{R}$).



- Correct classification on $(\boldsymbol{x}, y)$:

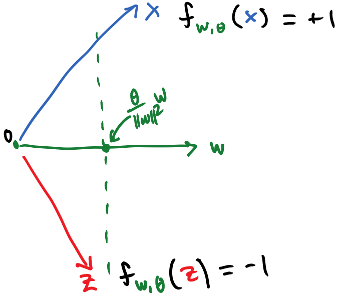
$$f_{\boldsymbol{w},\theta}(\boldsymbol{x}) = y \quad \text{iff} \quad y(\langle \boldsymbol{w}, \boldsymbol{x} \rangle - \theta) > 0.$$

(Ignore boundary case.)

- Projection of $\boldsymbol{x}$ onto $\operatorname{span}(\boldsymbol{w})$ is $\dfrac{\langle \boldsymbol{w}, \boldsymbol{x} \rangle}{\|\boldsymbol{w}\|_2^2}\boldsymbol{w}$.

Distance to the boundary

Consider linear classifier $f_{\boldsymbol{w},\theta}$ (where $\boldsymbol{w} \in \mathbb{R}^d \setminus \{\boldsymbol{0}\}$ and $\theta \in \mathbb{R}$).



- Correct classification on $(\boldsymbol{x}, y)$:

$$f_{\boldsymbol{w},\theta}(\boldsymbol{x}) = y \quad \text{iff} \quad y(\langle \boldsymbol{w}, \boldsymbol{x} \rangle - \theta) > 0.$$

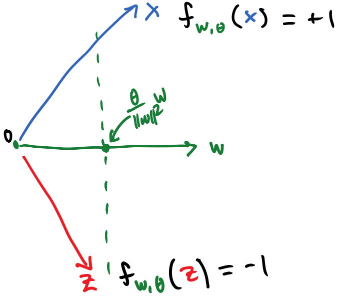
(Ignore boundary case.)

- Projection of $\boldsymbol{x}$ onto $\operatorname{span}(\boldsymbol{w})$ is $\dfrac{\langle \boldsymbol{w}, \boldsymbol{x} \rangle}{\|\boldsymbol{w}\|_2^2}\boldsymbol{w}$.
- Distance to affine hyperplane H is

$$\operatorname{dist}(\boldsymbol{x}, H) = \frac{|\langle \boldsymbol{w}, \boldsymbol{x} \rangle - \theta|}{\|\boldsymbol{w}\|_2}.$$

Distance to the boundary

Consider linear classifier $f_{\boldsymbol{w},\theta}$ (where $\boldsymbol{w} \in \mathbb{R}^d \setminus \{\boldsymbol{0}\}$ and $\theta \in \mathbb{R}$).



- Correct classification on $(\boldsymbol{x}, y)$:

$$f_{\boldsymbol{w},\theta}(\boldsymbol{x}) = y \quad \text{iff} \quad y(\langle \boldsymbol{w}, \boldsymbol{x} \rangle - \theta) > 0.$$

(Ignore boundary case.)

- Projection of $\boldsymbol{x}$ onto $\operatorname{span}(\boldsymbol{w})$ is $\frac{\langle \boldsymbol{w}, \boldsymbol{x} \rangle}{\|\boldsymbol{w}\|_2^2}\boldsymbol{w}$.
- Distance to affine hyperplane H is

$$\operatorname{dist}(\boldsymbol{x}, H) = \frac{|\langle \boldsymbol{w}, \boldsymbol{x} \rangle - \theta|}{\|\boldsymbol{w}\|_2}.$$

- If $f_{\boldsymbol{w},\theta}(\boldsymbol{x}) = y$, then

$$\operatorname{dist}(\boldsymbol{x}, H) = \frac{y(\langle \boldsymbol{w}, \boldsymbol{x} \rangle - \theta)}{\|\boldsymbol{w}\|_2}.$$

Maximizing distances to the boundary

Distance of a correctly classified point to the boundary

If $f_{\boldsymbol{w},\theta}(\boldsymbol{x}) = y$ (i.e., $y(\langle \boldsymbol{w}, \boldsymbol{x} \rangle - \theta) > 0$), then

$$\mathrm{dist}(\boldsymbol{x}, H) = \frac{y(\langle \boldsymbol{w}, \boldsymbol{x} \rangle - \theta)}{\|\boldsymbol{w}\|_2}.$$

Maximizing distances to the boundary

Distance of a correctly classified point to the boundary

If $f_{\boldsymbol{w},\theta}(\boldsymbol{x}) = y$ (i.e., $y(\langle \boldsymbol{w}, \boldsymbol{x} \rangle - \theta) > 0$), then

$$\mathrm{dist}(\boldsymbol{x}, H) = \frac{y(\langle \boldsymbol{w}, \boldsymbol{x} \rangle - \theta)}{\|\boldsymbol{w}\|_2}.$$

Maximizing the distances to the boundary

Therefore, find $(\boldsymbol{w}, \theta)$ such that

$$y(\langle \boldsymbol{w}, \boldsymbol{x} \rangle - \theta) > 0 \quad \text{for all } (\boldsymbol{x}, y) \in S$$

with $\|\boldsymbol{w}\|_2$ as small as possible.

Maximizing distances to the boundary

Distance of a correctly classified point to the boundary

If $f_{\boldsymbol{w},\theta}(\boldsymbol{x}) = y$ (i.e., $y(\langle \boldsymbol{w}, \boldsymbol{x} \rangle - \theta) > 0$), then

$$\boxed{\operatorname{dist}(\boldsymbol{x}, H) = \frac{y(\langle \boldsymbol{w}, \boldsymbol{x} \rangle - \theta)}{\|\boldsymbol{w}\|_2}.}$$

Maximizing the distances to the boundary

Therefore, find $(\boldsymbol{w}, \theta)$ such that

$$y(\langle \boldsymbol{w}, \boldsymbol{x} \rangle - \theta) > 0 \quad \text{for all } (\boldsymbol{x}, y) \in S$$

with $\|\boldsymbol{w}\|_2$ as small as possible.

Note: If we have *any* $(\boldsymbol{w}, \theta)$ such that

$$y(\langle \boldsymbol{w}, \boldsymbol{x} \rangle - \theta) > 0 \quad \text{for all } (\boldsymbol{x}, y) \in S,$$

then can rescale $(\boldsymbol{w}, \theta)$ so that

$$y(\langle \boldsymbol{w}, \boldsymbol{x} \rangle - \theta) \geq 1 \quad \text{for all } (\boldsymbol{x}, y) \in S.$$

how to scale?

Maximum margin linear classifier

The solution $(\hat{\boldsymbol{w}}, \hat{\theta})$ to the following optimization problem:

$$\begin{aligned} \min_{\boldsymbol{w} \in \mathbb{R}^d, \theta \in \mathbb{R}} \quad & \frac{1}{2}\|\boldsymbol{w}\|_2^2 \\ \text{s.t.} \quad & y(\langle \boldsymbol{w}, \boldsymbol{x} \rangle - \theta) \geq 1 \qquad \text{for all } (\boldsymbol{x}, y) \in S \end{aligned}$$

gives the **linear classifier with the maximum margin on S**.
(The $1/2$ and squaring is for mathematical convenience and now the convention.)

Maximum margin linear classifier

The solution $(\hat{\boldsymbol{w}}, \hat{\theta})$ to the following optimization problem:

$$\begin{aligned} \min_{\boldsymbol{w} \in \mathbb{R}^d, \theta \in \mathbb{R}} \quad & \frac{1}{2} \|\boldsymbol{w}\|_2^2 \\ \text{s.t.} \quad & y(\langle \boldsymbol{w}, \boldsymbol{x} \rangle - \theta) \geq 1 \qquad \text{for all } (\boldsymbol{x}, y) \in S \end{aligned}$$

gives the **linear classifier with the maximum margin on S**.
(The $1/2$ and squaring is for mathematical convenience and now the convention.)

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

Maximum margin linear classifier

The solution $(\hat{\boldsymbol{w}}, \hat{\theta})$ to the following optimization problem:

$$\begin{aligned} \min_{\boldsymbol{w}\in\mathbb{R}^d,\theta\in\mathbb{R}} \quad & \frac{1}{2}\|\boldsymbol{w}\|_2^2 \\ \text{s.t.} \quad & y(\langle \boldsymbol{w}, \boldsymbol{x}\rangle - \theta) \boxed{\geq 1} \quad \text{for all } (\boldsymbol{x}, y) \in S \end{aligned}$$

gives the **linear classifier with the maximum margin on S**.
(The $1/2$ and squaring is for mathematical convenience and now the convention.)

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

The optimization problem is a **convex optimization problem** that can be solved in polynomial time. (Actual algorithm to come later.)

Maximum margin linear classifier

The solution $(\hat{\boldsymbol{w}}, \hat{\theta})$ to the following optimization problem:

$$\begin{aligned} \min_{\boldsymbol{w} \in \mathbb{R}^d, \theta \in \mathbb{R}} \quad & \frac{1}{2}\|\boldsymbol{w}\|_2^2 \\ \text{s.t.} \quad & y(\langle \boldsymbol{w}, \boldsymbol{x} \rangle - \theta) \geq 1 \qquad \text{for all } (\boldsymbol{x}, y) \in S \end{aligned}$$

gives the **linear classifier with the maximum margin on S**.
(The $1/2$ and squaring is for mathematical convenience and now the convention.)

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

The optimization problem is a **convex optimization problem** that can be solved in polynomial time. (Actual algorithm to come later.)

If there is a solution (i.e., the problem is separable), then the solution is *unique*.
(Compare to LP's and Perceptron's lack of determinism from S.)

Support vectors and the SVM dual

Motivation

- Explain how the SVM solution is *entirely determined by certain data points called the "support vectors"*

  In other words: can throw away all data except the support vectors, re-solve SVM optimization problem, and get the same solution.

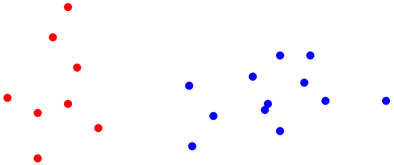
- Illustrate power of *convex duality*—a useful analytic (& algorithmic) tool.

  (Linear programming duality is a special case.)

Substituting convex sets

Observation

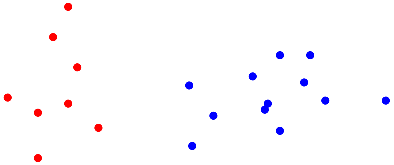
Where a separating hyperplane may be placed depends on the "outer" points on the sets. Points in the interior do not matter.



Substituting convex sets

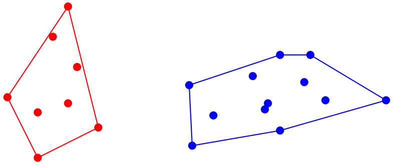
Observation

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



In geometric terms

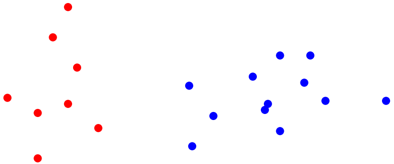
For each class, replace all points from the class with *the smallest* **convex set** *which contains all these points*:



Substituting convex sets

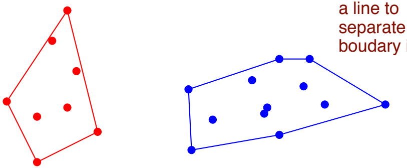
Observation

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



In geometric terms

For each class, replace all points from the class with *the smallest* **convex set** *which contains all these points*:



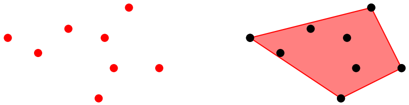
a line to separate boudary is ok

A set C is **convex** if it contains the line segments between all pairs of points in C.

Substituting convex sets

Definition

If $A \subset \mathbb{R}^d$ is a set of points, the smallest convex set containing all points in A is called the **convex hull** of A, denoted $\operatorname{conv}(A)$.

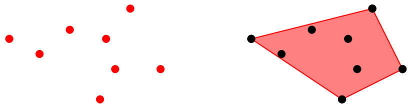


"Corner points" of the convex set are called **extreme points**.

Substituting convex sets

Definition

If $A \subset \mathbb{R}^d$ is a set of points, the smallest convex set containing all points in A is called the **convex hull** of A, denoted $\mathrm{conv}(A)$.



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

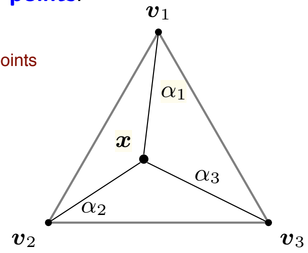
Barycentric coordinates

If convex set C has extreme points $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_m$, corner points

then can write any $\boldsymbol{x} \in C$ as

$$\boldsymbol{x} = \sum_{i=1}^{m} \alpha_i \boldsymbol{v}_i \quad \text{where } \alpha_i \geq 0,\ \textstyle\sum_{i=1}^{m} \alpha_i = 1.$$

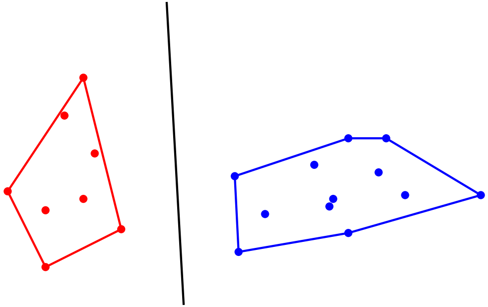
Called the **barycentric coordinates** of $\boldsymbol{x}$.



Convex hulls and classification

Key idea

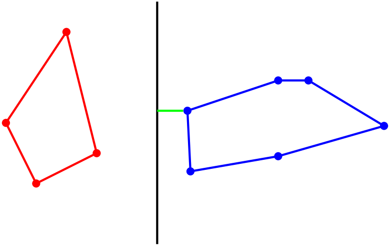
A hyperplane separates two classes if and only if it separates their convex hulls.



Equivalent definition of margin

If $f_{\boldsymbol{w},\theta}$ linearly separates S, then we say its **margin on S** is

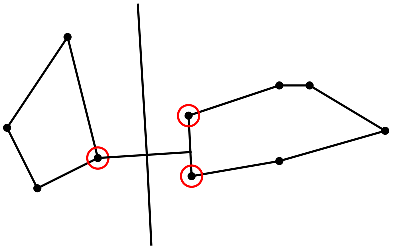
$$\boxed{\min\{\operatorname{dist}(H, \operatorname{conv}(S_{\oplus})), \operatorname{dist}(H, \operatorname{conv}(S_{\ominus}))\}.}$$



Distance between two sets A and B: $\boxed{\operatorname{dist}(A, B) := \min_{\boldsymbol{x} \in A} \operatorname{dist}(\boldsymbol{x}, B).}$

Support vectors

Extreme points of the convex hulls closest to the hyperplane are the **support vectors**.



Implications

- SVM optimization problem focuses attention to the area closest to the decision surface.
- SVM classifier entirely determined by support vectors.
  (Much like mistake examples in Perceptron—hope for kernelization?)

The dual of the SVM optimization problem

Questions

- Can we determine which points are the support vectors?
- Can we kernelize SVM like we did Perceptron?

The dual of the SVM optimization problem

Questions

- Can we determine which points are the support vectors?
- Can we kernelize SVM like we did Perceptron?

These questions are answered by looking at a corresponding *dual optimization problem*, which we will derive from geometric principles.

The dual of the SVM optimization problem

Questions

- Can we determine which points are the support vectors?
- Can we kernelize SVM like we did Perceptron?

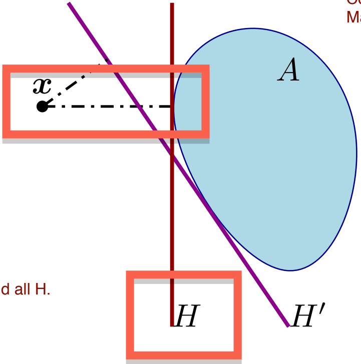
These questions are answered by looking at a corresponding *dual optimization problem*, which we will derive from geometric principles.

(Can also derive the dual problem using Lagrangian duality.)

Convex duality

Fact: The closest distance between a point $\boldsymbol{x}$ and a convex set A is the maximum over the distances between $\boldsymbol{x}$ and all (affine) hyperplanes that separate $\boldsymbol{x}$ and A.

$$\operatorname{dist}(\boldsymbol{x}, A) = \max_{\substack{H \text{ separating} \\ \boldsymbol{x} \text{ from } A}} \operatorname{dist}(\boldsymbol{x}, H)$$



Convert Distance between convex to Margin!!!

The maximum between between x and all H.

Deriving the dual problem

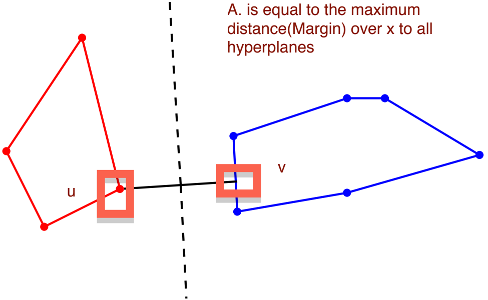
Main idea

Equivalent definition of max-margin affine hyperplane:

1. Find shortest line segment connecting the convex hulls.
2. Place affine hyperplane orthogonal to line segment at midpoint.

The dual problem

$$\min_{\substack{\boldsymbol{u}\in\mathrm{conv}(S_\oplus),\\ \boldsymbol{v}\in\mathrm{conv}(S_\ominus)}} \frac{1}{2}\|\boldsymbol{u}-\boldsymbol{v}\|_2^2$$



The closest distance between x with set A. is equal to the maximum distance(Margin) over x to all hyperplanes

Dual problem in barycentric coordinates

Dual problem

$$\min_{\substack{\boldsymbol{u}\in\mathrm{conv}(S_\oplus),\\ \boldsymbol{v}\in\mathrm{conv}(S_\ominus)}} \frac{1}{2}\|\boldsymbol{u}-\boldsymbol{v}\|_2^2.$$

Dual problem in barycentric coordinates

Dual problem

$$\min_{\substack{\boldsymbol{u}\in\mathrm{conv}(S_\oplus),\\ \boldsymbol{v}\in\mathrm{conv}(S_\ominus)}} \frac{1}{2}\|\boldsymbol{u}-\boldsymbol{v}\|_2^2.$$

New parameterization

Express u and v in barycentric coordinates.
Let $S_\oplus = \{\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(2)}, \ldots, \boldsymbol{x}^{(m)}\}$, $S_\ominus = \{\boldsymbol{x}^{(m+1)}, \boldsymbol{x}^{(m+2)}, \ldots, \boldsymbol{x}^{(n)}\}$,

$$\boldsymbol{u} = \sum_{i=1}^{m} \alpha_i \boldsymbol{x}^{(i)}, \qquad \boldsymbol{v} = \sum_{i=m+1}^{n} \alpha_i \boldsymbol{x}^{(i)}.$$

Dual problem in barycentric coordinates

Dual problem

$$\min_{\substack{\boldsymbol{u}\in\mathrm{conv}(S_\oplus),\\ \boldsymbol{v}\in\mathrm{conv}(S_\ominus)}} \frac{1}{2}\|\boldsymbol{u}-\boldsymbol{v}\|_2^2.$$

New parameterization

Express $\boldsymbol{u}$ and $\boldsymbol{v}$ in barycentric coordinates.
Let $S_\oplus = \{\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(2)}, \ldots, \boldsymbol{x}^{(m)}\}$, $S_\ominus = \{\boldsymbol{x}^{(m+1)}, \boldsymbol{x}^{(m+2)}, \ldots, \boldsymbol{x}^{(n)}\}$,

some ai may equal to 0.

$$\boldsymbol{u} = \sum_{i=1}^{m} \alpha_i \boldsymbol{x}^{(i)}, \qquad \boldsymbol{v} = \sum_{i=m+1}^{n} \alpha_i \boldsymbol{x}^{(i)}.$$

Dual problem in terms of $\alpha_1, \ldots, \alpha_n$ (called "dual variables"):

$$\min_{\alpha_1,\ldots,\alpha_n} \quad \frac{1}{2}\left\|\sum_{i=1}^{m} \alpha_i \boldsymbol{x}^{(i)} - \sum_{i=m+1}^{n} \alpha_i \boldsymbol{x}^{(i)}\right\|_2^2$$

$$\text{s.t.} \quad \sum_{i=1}^{m} \alpha_i = \sum_{i=m+1}^{n} \alpha_i = 1, \quad \alpha_i \geq 0.$$

Dual problem in barycentric coordinates

Dual problem

$$\min_{\substack{\boldsymbol{u}\in\operatorname{conv}(S_\oplus),\\ \boldsymbol{v}\in\operatorname{conv}(S_\ominus)}} \frac{1}{2}\|\boldsymbol{u}-\boldsymbol{v}\|_2^2.$$

New parameterization

Express u and v in barycentric coordinates.
Let $S_\oplus = \{\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(2)}, \dots, \boldsymbol{x}^{(m)}\}$, $S_\ominus = \{\boldsymbol{x}^{(m+1)}, \boldsymbol{x}^{(m+2)}, \dots, \boldsymbol{x}^{(n)}\}$,

$$\boldsymbol{u} = \sum_{i=1}^{m} \alpha_i \boldsymbol{x}^{(i)}, \qquad \boldsymbol{v} = \sum_{i=m+1}^{n} \alpha_i \boldsymbol{x}^{(i)}.$$

Dual problem in terms of $\alpha_1, \dots, \alpha_n$ (called "dual variables"):

$$\begin{aligned} \min_{\alpha_1,\dots,\alpha_n} \quad & \frac{1}{2}\left\|\sum_{i=1}^{m} \alpha_i y^{(i)} \boldsymbol{x}^{(i)} + \sum_{i=m+1}^{n} \alpha_i y^{(i)} \boldsymbol{x}^{(i)}\right\|_2^2 \\ \text{s.t.} \quad & \sum_{i=1}^{m} \alpha_i = \sum_{i=m+1}^{n} \alpha_i = 1, \quad \alpha_i \geq 0. \end{aligned}$$

Dual problem in barycentric coordinates

Dual problem

$$\min_{\substack{\boldsymbol{u}\in\mathrm{conv}(S_\oplus),\\ \boldsymbol{v}\in\mathrm{conv}(S_\ominus)}} \frac{1}{2}\|\boldsymbol{u}-\boldsymbol{v}\|_2^2.$$

New parameterization

Express $\boldsymbol{u}$ and $\boldsymbol{v}$ in barycentric coordinates.
Let $S_\oplus = \{\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(2)}, \dots, \boldsymbol{x}^{(m)}\}$, $S_\ominus = \{\boldsymbol{x}^{(m+1)}, \boldsymbol{x}^{(m+2)}, \dots, \boldsymbol{x}^{(n)}\}$,

$$\boldsymbol{u} = \sum_{i=1}^{m} \alpha_i \boldsymbol{x}^{(i)}, \qquad \boldsymbol{v} = \sum_{i=m+1}^{n} \alpha_i \boldsymbol{x}^{(i)}.$$

Dual problem in terms of $\alpha_1, \dots, \alpha_n$ (called "dual variables"):

$$\begin{aligned} \min_{\alpha_1,\dots,\alpha_n} \quad & \frac{1}{2}\left\|\sum_{i=1}^{n} \alpha \boxed{y^{(i)}\boldsymbol{x}}^{(i)}\right\|_2^2 \\ \text{s.t.} \quad & \sum_{i=1}^{m} \alpha_i = \sum_{i=m+1}^{n} \alpha_i = 1, \quad \alpha_i \geq 0. \end{aligned}$$

negative
great!!!

Dual problem in usual form

Expanding the square...

$$\begin{aligned}\frac{1}{2}\left\|\sum_{i=1}^{n}\alpha_i y^{(i)}\boldsymbol{x}^{(i)}\right\|_2^2 &= \frac{1}{2}\left\langle\sum_{i=1}^{n}\alpha_i y^{(i)}\boldsymbol{x}^{(i)}, \sum_{i=1}^{n}\alpha_i y^{(i)}\boldsymbol{x}^{(i)}\right\rangle \\ &= \frac{1}{2}\sum_{i,j=1}^{n}\alpha_i\alpha_j y^{(i)}y^{(j)}\langle\boldsymbol{x}^{(i)},\boldsymbol{x}^{(j)}\rangle.\end{aligned}$$

Dual problem in usual form

Expanding the square...

$$
\begin{aligned}
\frac{1}{2}\left\|\sum_{i=1}^{n}\alpha_i y^{(i)}\boldsymbol{x}^{(i)}\right\|_2^2 &= \frac{1}{2}\left\langle \sum_{i=1}^{n}\alpha_i y^{(i)}\boldsymbol{x}^{(i)}, \sum_{i=1}^{n}\alpha_i y^{(i)}\boldsymbol{x}^{(i)} \right\rangle \\
&= \frac{1}{2}\sum_{i,j=1}^{n}\alpha_i\alpha_j y^{(i)} y^{(j)} \boxed{\langle \boldsymbol{x}^{(i)}, \boldsymbol{x}^{(j)}\rangle .}
\end{aligned}
$$

Usual form of the dual problem

$$
\begin{aligned}
\min_{\alpha_1,\ldots,\alpha_n} \quad & \frac{1}{2}\sum_{i,j=1}^{n}\alpha_i\alpha_j y^{(i)} y^{(j)} \langle \boldsymbol{x}^{(i)}, \boldsymbol{x}^{(j)}\rangle \\
\text{s.t.} \quad & \sum_{i=1}^{m}\alpha_i = \sum_{i=m+1}^{n}\alpha_i = 1, \quad \alpha_i \geq 0.
\end{aligned}
$$

Dual problem in usual form

Expanding the square. . .

$$\begin{aligned}\frac{1}{2}\left\|\sum_{i=1}^{n}\alpha_i y^{(i)}\boldsymbol{x}^{(i)}\right\|_2^2 &= \frac{1}{2}\left\langle\sum_{i=1}^{n}\alpha_i y^{(i)}\boldsymbol{x}^{(i)}, \sum_{i=1}^{n}\alpha_i y^{(i)}\boldsymbol{x}^{(i)}\right\rangle \\ &= \frac{1}{2}\sum_{i,j=1}^{n}\alpha_i\alpha_j y^{(i)}y^{(j)}\langle\boldsymbol{x}^{(i)}, \boldsymbol{x}^{(j)}\rangle.\end{aligned}$$

Usual form of the dual problem

$$\begin{aligned}\max_{\alpha_1,\dots,\alpha_n} \quad & \sum_{i=1}^{n}\alpha_i - \frac{1}{2}\sum_{i,j=1}^{n}\alpha_i\alpha_j y^{(i)}y^{(j)}\langle\boldsymbol{x}^{(i)}, \boldsymbol{x}^{(j)}\rangle \\ \text{s.t.} \quad & \sum_{i=1}^{m}\alpha_i y^{(i)} = 0, \quad \alpha_i \geq 0.\end{aligned}$$

(Don't worry about getting to this form.)

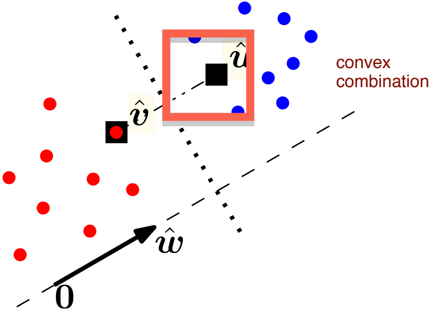
Using the dual solution

Getting a linear classifier from the dual solution

Let $\hat{\alpha}_1, \ldots, \hat{\alpha}_n$ be the solution to the dual problem. Then

$$\boxed{\begin{aligned}\hat{\boldsymbol{w}} &:= \hat{\boldsymbol{u}} - \hat{\boldsymbol{v}} = \sum_{i=1}^{n} \hat{\alpha}_i y^{(i)} \boldsymbol{x}^{(i)}; \\ \hat{\theta} &:= \frac{\min_{\boldsymbol{x} \in S_\oplus} \langle \hat{\boldsymbol{w}}, \boldsymbol{x} \rangle + \max_{\boldsymbol{x} \in S_\ominus} \langle \hat{\boldsymbol{w}}, \boldsymbol{x} \rangle}{2}.\end{aligned}}$$

(We want $\hat{\theta} := b\|\hat{\boldsymbol{w}}\|_2^2$ such that $b\hat{\boldsymbol{w}}$ is the midpoint between $\hat{\boldsymbol{u}}$ and $\hat{\boldsymbol{v}}$.)



Using the dual solution

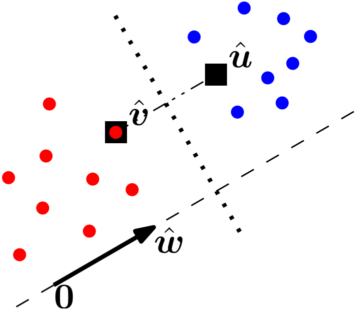
Getting a linear classifier from the dual solution

Let $\hat{\alpha}_1, \dots, \hat{\alpha}_n$ be the solution to the dual problem. Then

$$\hat{\boldsymbol{w}} := \hat{\boldsymbol{u}} - \hat{\boldsymbol{v}} = \sum_{i=1}^{n} \hat{\alpha}_i y^{(i)} \boldsymbol{x}^{(i)};$$

$$\hat{\theta} := \frac{\min_{\boldsymbol{x} \in S_\oplus} \langle \hat{\boldsymbol{w}}, \boldsymbol{x} \rangle + \max_{\boldsymbol{x} \in S_\ominus} \langle \hat{\boldsymbol{w}}, \boldsymbol{x} \rangle}{2}.$$

(We want $\hat{\theta} := b\|\hat{\boldsymbol{w}}\|_2^2$ such that $b\hat{\boldsymbol{w}}$ is the midpoint between $\hat{\boldsymbol{u}}$ and $\hat{\boldsymbol{v}}$.)



$\hat{\boldsymbol{w}}$ depends on $(\boldsymbol{x}^{(i)}, y^{(i)})$ with $\hat{\alpha}_i > 0$, i.e., the **support vectors**.

Using the dual solution

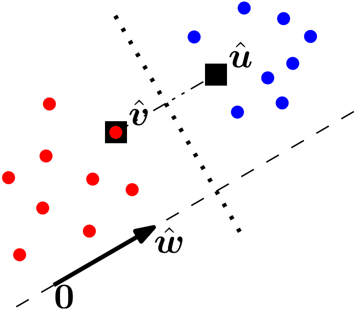
Getting a linear classifier from the dual solution

Let $\hat{\alpha}_1, \dots, \hat{\alpha}_n$ be the solution to the dual problem. Then

$$\boxed{\hat{\boldsymbol{w}} := \hat{\boldsymbol{u}} - \hat{\boldsymbol{v}} = \sum_{i=1}^{n} \hat{\alpha}_i y^{(i)} \boldsymbol{x}^{(i)};}$$

$$\hat{\theta} := \frac{\min_{\boldsymbol{x} \in S_\oplus} \langle \hat{\boldsymbol{w}}, \boldsymbol{x} \rangle + \max_{\boldsymbol{x} \in S_\ominus} \langle \hat{\boldsymbol{w}}, \boldsymbol{x} \rangle}{2}.$$

(We want $\hat{\theta} := b\|\hat{\boldsymbol{w}}\|_2^2$ such that $b\hat{\boldsymbol{w}}$ is the midpoint between $\hat{\boldsymbol{u}}$ and $\hat{\boldsymbol{v}}$.)



$\hat{\boldsymbol{w}}$ depends on $(\boldsymbol{x}^{(i)}, y^{(i)})$ with $\hat{\alpha}_i > 0$, i.e., the **support vectors**.

These support vector examples have

$$\boxed{y^{(i)} \left(\langle \hat{\boldsymbol{w}}, \boldsymbol{x}^{(i)} \rangle - \hat{\theta} \right) = \frac{1}{2} \|\hat{\boldsymbol{w}}\|_2^2.}$$

Non-support vectors $(\boldsymbol{x}^{(j)}, y^{(j)})$ have

$$y^{(j)} \left(\langle \hat{\boldsymbol{w}}, \boldsymbol{x}^{(j)} \rangle - \hat{\theta} \right) > \frac{1}{2} \|\hat{\boldsymbol{w}}\|_2^2.$$

Solving the primal SVM problem

Technically, to solve the primal SVM problem, need

$$y^{(i)}\Big(\langle \boldsymbol{w}, \boldsymbol{x}^{(i)}\rangle - \theta\Big) \geq 1 \quad \text{for } i = 1, 2, \ldots, n,$$

but we have

$$y^{(i)}\Big(\langle \boldsymbol{w}, \boldsymbol{x}^{(i)}\rangle - \theta\Big) \geq \frac{1}{2}\|\boldsymbol{w}\|_2^2 \quad \text{for } i = 1, 2, \ldots, n.$$

Solving the primal SVM problem

Technically, to solve the primal SVM problem, need

$$y^{(i)}\Big(\langle \boldsymbol{w}, \boldsymbol{x}^{(i)}\rangle - \theta\Big) \geq 1 \quad \text{for } i = 1, 2, \ldots, n,$$

but we have

$$y^{(i)}\Big(\langle \boldsymbol{w}, \boldsymbol{x}^{(i)}\rangle - \theta\Big) \geq \frac{1}{2}\|\boldsymbol{w}\|_2^2 \quad \text{for } i = 1, 2, \ldots, n.$$

Solution: Rescale $\hat{\boldsymbol{w}}$ and $\hat{\theta}$ by dividing by $\frac{1}{2}\|\hat{\boldsymbol{w}}\|_2^2$.

Kernelizing SVMs

Kernelizing SVMs (Boser, Guyon, and Vapnik, 1992)

SVM dual problem

$$\max_{\alpha_1,\ldots,\alpha_n} \quad \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y^{(i)} y^{(j)} \langle \boldsymbol{x}^{(i)}, \boldsymbol{x}^{(j)} \rangle$$

$$\text{s.t.} \quad \sum_{i=1}^{m} \alpha_i y^{(i)} = 0, \quad \alpha_i \geq 0.$$

SVM dual solution

$$\hat{\boldsymbol{w}} := \sum_{i=1}^{n} \hat{\alpha}_i y^{(i)} \boldsymbol{x}^{(i)};$$

$$\hat{\theta} := \frac{\min_{\boldsymbol{x} \in S_\oplus} \langle \hat{\boldsymbol{w}}, \boldsymbol{x} \rangle + \max_{\boldsymbol{x} \in S_\ominus} \langle \hat{\boldsymbol{w}}, \boldsymbol{x} \rangle}{2}.$$

Kernelizing SVMs (Boser, Guyon, and Vapnik, 1992)

SVM dual problem

$$\begin{aligned} \max_{\alpha_1,\ldots,\alpha_n} \quad & \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y^{(i)} y^{(j)} K(\boldsymbol{x}^{(i)}, \boldsymbol{x}^{(j)}) \\ \text{s.t.} \quad & \sum_{i=1}^{m} \alpha_i y^{(i)} = 0, \quad \alpha_i \geq 0. \end{aligned}$$

SVM dual solution

$$\langle \hat{\boldsymbol{w}}, \cdot \rangle := \sum_{i=1}^{n} \hat{\alpha}_i y^{(i)} K(\boldsymbol{x}^{(i)}, \cdot);$$

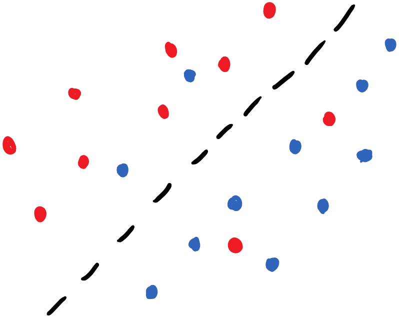
$$\hat{\theta} := \frac{\min_{\boldsymbol{x} \in S_\oplus} K(\hat{\boldsymbol{w}}, \boldsymbol{x}) + \max_{\boldsymbol{x} \in S_\ominus} K(\hat{\boldsymbol{w}}, \boldsymbol{x})}{2}.$$

Just need to keep around the **support vectors** (i.e., examples where $\hat{\alpha}_i > 0$).

Soft-margin SVMs and surrogate losses

Non-separable cases

Non-separable cases:
No linear classifier has zero training error on S.



But if non-separability is only due to a handful of points, can we still find a good linear classifier?

Soft-margin SVMs (Cortes and Vapnik, 1995)

When $S = ((\boldsymbol{x}^{(1)}, y^{(1)}), \dots, (\boldsymbol{x}^{(n)}, y^{(n)}))$ is not linearly separable, the (primal) SVM optimization problem

$$\begin{aligned} \min_{\boldsymbol{w} \in \mathbb{R}^d, \theta \in \mathbb{R}} \quad & \frac{1}{2} \|\boldsymbol{w}\|_2^2 \\ \text{s.t.} \quad & y^{(i)} \Big(\langle \boldsymbol{w}, \boldsymbol{x}^{(i)} \rangle - \theta \Big) \geq 1 \qquad \text{for } i = 1, 2, \dots, n \end{aligned}$$

has no solution.

Soft-margin SVMs (Cortes and Vapnik, 1995)

When $S = ((\boldsymbol{x}^{(1)}, y^{(1)}), \ldots, (\boldsymbol{x}^{(n)}, y^{(n)}))$ is not linearly separable, the (primal) SVM optimization problem

$$\begin{aligned} \min_{\boldsymbol{w} \in \mathbb{R}^d, \theta \in \mathbb{R}} \quad & \frac{1}{2}\|\boldsymbol{w}\|_2^2 \\ \text{s.t.} \quad & y^{(i)}\Big(\langle \boldsymbol{w}, \boldsymbol{x}^{(i)} \rangle - \theta\Big) \geq 1 \qquad \text{for } i = 1, 2, \ldots, n \end{aligned}$$

has no solution.

Introduce **slack variables** $\xi_1, \xi_2, \ldots, \xi_n \geq 0$, and a trade-off parameter $C > 0$:

$$\begin{aligned} \min_{\boldsymbol{w} \in \mathbb{R}^d, \theta \in \mathbb{R}, \boldsymbol{\xi} \in \mathbb{R}^n} \quad & \frac{1}{2}\|\boldsymbol{w}\|_2^2 + C \sum_{i=1}^{n} \xi_i \\ \text{s.t.} \quad & y^{(i)}\Big(\langle \boldsymbol{w}, \boldsymbol{x}^{(i)} \rangle - \theta\Big) \geq 1 - \xi_i \qquad \text{for } i = 1, 2, \ldots, n \\ & \xi_i \geq 0 \qquad \text{for } i = 1, 2, \ldots, n \end{aligned}$$

which is **always feasible**—"soft margin" SVM.

Soft-margin SVMs (Cortes and Vapnik, 1995)

Winner of 2008 ACM Paris Kanellakis Award:

> *For "their revolutionary development of a highly effective algorithm known as Support Vector Machines (SVM), a set of related supervised learning methods used for data classification and regression", which is "one of the most frequently used algorithms in machine learning, and is used in medical diagnosis, weather forecasting, and intrusion detection among many other practical applications".*

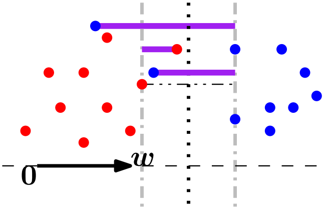
Other winners include: public key cryptography, Lempel-Ziv compression, Splay Trees, interior point method for linear programming, . . .
and AdaBoost (discussed later in the course).

Slack interpretation

$$\min_{\boldsymbol{w}\in\mathbb{R}^d,\theta\in\mathbb{R},\boldsymbol{\xi}\in\mathbb{R}^n} \quad \frac{1}{2}\|\boldsymbol{w}\|_2^2 + C\sum_{i=1}^{n}\xi_i$$

$$\text{s.t.} \quad y^{(i)}\Big(\langle\boldsymbol{w},\boldsymbol{x}^{(i)}\rangle - \theta\Big) \geq 1 - \xi_i \qquad \text{for } i = 1,2,\ldots,n$$

$$\xi_i \geq 0 \qquad \text{for } i = 1,2,\ldots,n$$

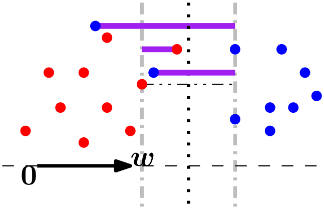


Slack interpretation

$$\min_{\boldsymbol{w}\in\mathbb{R}^d,\theta\in\mathbb{R},\boldsymbol{\xi}\in\mathbb{R}^n} \quad \frac{1}{2}\|\boldsymbol{w}\|_2^2 + C\sum_{i=1}^{n}\xi_i$$

$$\text{s.t.} \quad y^{(i)}\Big(\langle \boldsymbol{w}, \boldsymbol{x}^{(i)}\rangle - \theta\Big) \geq 1 - \xi_i \qquad \text{for } i = 1, 2, \ldots, n$$

$$\xi_i \geq 0 \qquad \text{for } i = 1, 2, \ldots, n$$



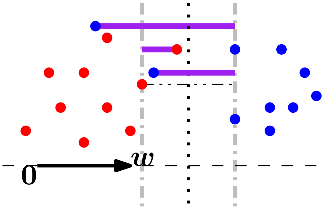
- For a given $(\boldsymbol{w}, \theta)$, $\xi_i/\|\boldsymbol{w}\|_2$ measures distance that $\boldsymbol{x}^{(i)}$ must be moved so that $y^{(i)}\langle \boldsymbol{w}, \boldsymbol{x}^{(i)} - \theta\rangle \geq 1$.

Slack interpretation

$$\min_{\boldsymbol{w}\in\mathbb{R}^d,\theta\in\mathbb{R},\boldsymbol{\xi}\in\mathbb{R}^n} \quad \frac{1}{2}\|\boldsymbol{w}\|_2^2 + C\sum_{i=1}^{n}\xi_i$$

$$\text{s.t.} \quad y^{(i)}\Big(\langle \boldsymbol{w}, \boldsymbol{x}^{(i)}\rangle - \theta\Big) \geq 1 - \xi_i \qquad \text{for } i = 1, 2, \ldots, n$$

$$\xi_i \geq 0 \qquad \text{for } i = 1, 2, \ldots, n$$



- For a given $(\boldsymbol{w}, \theta)$, $\xi_i/\|\boldsymbol{w}\|_2$ measures distance that $\boldsymbol{x}^{(i)}$ must be moved so that $y^{(i)}\langle \boldsymbol{w}, \boldsymbol{x}^{(i)} - \theta\rangle \geq 1$.
- C controls trade-off between slack penalties and size of margin (which is $1/\|\boldsymbol{w}\|_2$).

A different interpretation of slack

Constraints with non-negative slack variables: (using $\lambda = 1/(nC)$)

$$\begin{aligned} \min_{\boldsymbol{w}\in\mathbb{R}^d,\theta\in\mathbb{R},\boldsymbol{\xi}\in\mathbb{R}^n} \quad & \frac{\lambda}{2}\|\boldsymbol{w}\|_2^2 + \frac{1}{n}\sum_{i=1}^{n}\xi_i \\ \text{s.t.} \quad & y^{(i)}\left(\langle\boldsymbol{w},\boldsymbol{x}^{(i)}\rangle - \theta\right) \geq 1 - \xi_i \qquad \text{for } i = 1,2,\ldots,n \\ & \xi_i \geq 0 \qquad \text{for } i = 1,2,\ldots,n \end{aligned}$$

A DIFFERENT INTERPRETATION OF SLACK

Constraints with non-negative slack variables: (using $\lambda = 1/(nC)$)

$$\begin{aligned} \min_{\boldsymbol{w}\in\mathbb{R}^d,\theta\in\mathbb{R},\boldsymbol{\xi}\in\mathbb{R}^n} \quad & \frac{\lambda}{2}\|\boldsymbol{w}\|_2^2 + \frac{1}{n}\sum_{i=1}^{n}\xi_i \\ \text{s.t.} \quad & y^{(i)}\Big(\langle \boldsymbol{w}, \boldsymbol{x}^{(i)}\rangle - \theta\Big) \geq 1 - \xi_i \qquad \text{for } i = 1, 2, \ldots, n \\ & \xi_i \geq 0 \qquad \text{for } i = 1, 2, \ldots, n \end{aligned}$$

Equivalent "hinge loss" form:

$$\min_{\boldsymbol{w}\in\mathbb{R}^d,\theta\in\mathbb{R}} \quad \frac{\lambda}{2}\|\boldsymbol{w}\|_2^2 + \frac{1}{n}\sum_{i=1}^{n}\Big[1 - y^{(i)}\Big(\langle \boldsymbol{w}, \boldsymbol{x}^{(i)}\rangle - \theta\Big)\Big]_+$$

Notation: $[a]_+ := \max\{0, a\}$.

A different interpretation of slack

Constraints with non-negative slack variables: (using $\lambda = 1/(nC)$)

$$\begin{aligned} \min_{\boldsymbol{w}\in\mathbb{R}^d, \theta\in\mathbb{R}, \boldsymbol{\xi}\in\mathbb{R}^n} \quad & \frac{\lambda}{2}\|\boldsymbol{w}\|_2^2 + \frac{1}{n}\sum_{i=1}^{n}\xi_i \\ \text{s.t.} \quad & y^{(i)}\Big(\langle \boldsymbol{w}, \boldsymbol{x}^{(i)}\rangle - \theta\Big) \geq 1 - \xi_i \qquad \text{for } i = 1, 2, \dots, n \\ & \xi_i \geq 0 \qquad \text{for } i = 1, 2, \dots, n \end{aligned}$$

Equivalent "hinge loss" form:

$$\min_{\boldsymbol{w}\in\mathbb{R}^d, \theta\in\mathbb{R}} \quad \frac{\lambda}{2}\|\boldsymbol{w}\|_2^2 + \frac{1}{n}\sum_{i=1}^{n}\Big[1 - y^{(i)}\Big(\langle \boldsymbol{w}, \boldsymbol{x}^{(i)}\rangle - \theta\Big)\Big]_+$$

Notation: $[a]_+ := \max\{0, a\}$.

The **hinge loss** of a linear classifier $f_{\boldsymbol{w},\theta}$ on an example $(\boldsymbol{x}, y)$ is defined to be

$$\mathrm{HL}(\boldsymbol{w}, \theta; \boldsymbol{x}, y) := \Big[1 - y(\langle \boldsymbol{w}, \boldsymbol{x}\rangle - \theta)\Big]_+.$$

A different interpretation of slack

Constraints with non-negative slack variables: (using $\lambda = 1/(nC)$)

$$\begin{aligned} \min_{\boldsymbol{w}\in\mathbb{R}^d,\theta\in\mathbb{R},\boldsymbol{\xi}\in\mathbb{R}^n} \quad & \frac{\lambda}{2}\|\boldsymbol{w}\|_2^2 + \frac{1}{n}\sum_{i=1}^{n}\xi_i \\ \text{s.t.} \quad & y^{(i)}\Big(\langle \boldsymbol{w}, \boldsymbol{x}^{(i)}\rangle - \theta\Big) \geq 1 - \xi_i \qquad \text{for } i = 1, 2, \ldots, n \\ & \xi_i \geq 0 \qquad \text{for } i = 1, 2, \ldots, n \end{aligned}$$

Equivalent "hinge loss" form:

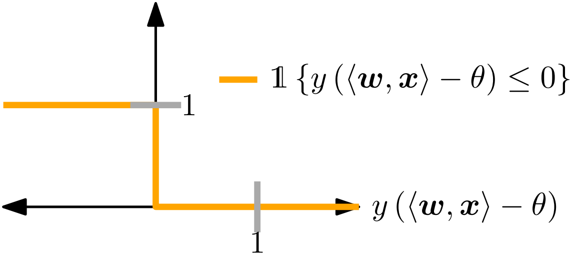
$$\min_{\boldsymbol{w}\in\mathbb{R}^d,\theta\in\mathbb{R}} \quad \frac{\lambda}{2}\|\boldsymbol{w}\|_2^2 + \frac{1}{n}\sum_{i=1}^{n}\mathrm{HL}(\boldsymbol{w},\theta;\boldsymbol{x}^{(i)},y^{(i)})$$

Notation: $[a]_+ := \max\{0, a\}$.

The **hinge loss** of a linear classifier $f_{\boldsymbol{w},\theta}$ on an example $(\boldsymbol{x}, y)$ is defined to be

$$\mathrm{HL}(\boldsymbol{w},\theta;\boldsymbol{x},y) := \Big[1 - y(\langle \boldsymbol{w}, \boldsymbol{x}\rangle - \theta)\Big]_+.$$

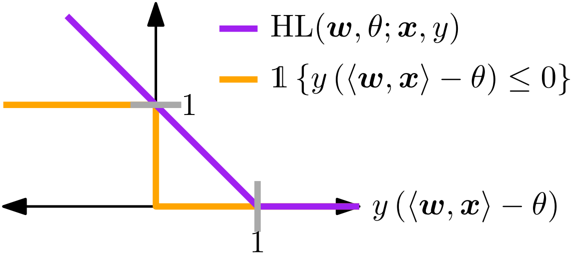
Zero-one loss vs. hinge loss



Zero-one loss: count if $f_{\boldsymbol{w},\theta}(\boldsymbol{x}) \neq y$.

$$\mathbb{1}\{y(\langle \boldsymbol{w}, \boldsymbol{x}\rangle - \theta) \leq 0\} \;\leq\; \Big[1 - y(\langle \boldsymbol{w}, \boldsymbol{x}\rangle - \theta)\Big]_+ \;=\; \mathrm{HL}(\boldsymbol{w}, \theta; \boldsymbol{x}, y).$$

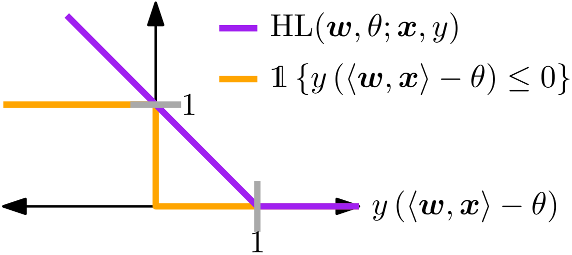
Zero-one loss vs. hinge loss



Hinge loss: an upper-bound on zero-one loss.

$$\mathbb{1}\{y(\langle \boldsymbol{w}, \boldsymbol{x} \rangle - \theta) \leq 0\} \ \leq \ \Big[1 - y(\langle \boldsymbol{w}, \boldsymbol{x} \rangle - \theta)\Big]_+ \ = \ \mathrm{HL}(\boldsymbol{w}, \theta; \boldsymbol{x}, y).$$

Zero-one loss vs. hinge loss

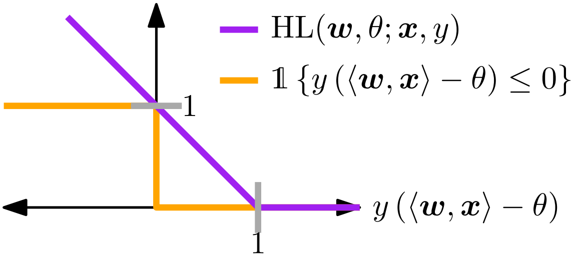


Hinge loss: an upper-bound on zero-one loss.

$$\mathbb{1}\{y(\langle \boldsymbol{w}, \boldsymbol{x}\rangle - \theta) \leq 0\} \;\leq\; \Big[1 - y(\langle \boldsymbol{w}, \boldsymbol{x}\rangle - \theta)\Big]_+ \;=\; \mathrm{HL}(\boldsymbol{w}, \theta; \boldsymbol{x}, y).$$

Soft-margin SVM minimizes an upper-bound on the training error, plus a term that favors large margins.

Zero-one loss vs. hinge loss



Hinge loss: an upper-bound on zero-one loss.

$$\mathbb{1}\{y(\langle \boldsymbol{w}, \boldsymbol{x}\rangle - \theta) \le 0\} \;\le\; \Big[1 - y(\langle \boldsymbol{w}, \boldsymbol{x}\rangle - \theta)\Big]_+ \;=\; \mathrm{HL}(\boldsymbol{w}, \theta; \boldsymbol{x}, y).$$

Soft-margin SVM minimizes an upper-bound on the training error, plus a term that favors large margins.

This is **computationally tractable** (unlike minimizing training error) because the hinge loss is a **convex function** of $(\boldsymbol{w}, \theta)$, and so is $\frac{\lambda}{2}\|\boldsymbol{w}\|_2^2$.

General form

Empirical risk minimization (i.e., minimize training error):

$$\min_{\boldsymbol{w}\in\mathbb{R}^d,\theta\in\mathbb{R}} \qquad \frac{1}{n}\sum_{i=1}^{n}\mathbb{1}\left\{y^{(i)}\left(\langle\boldsymbol{w},\boldsymbol{x}^{(i)}\rangle-\theta\right)\le 0\right\}$$

Soft-margin SVM:

$$\min_{\boldsymbol{w}\in\mathbb{R}^d,\theta\in\mathbb{R}} \qquad \frac{\lambda}{2}\|\boldsymbol{w}\|_2^2+\frac{1}{n}\sum_{i=1}^{n}\mathrm{HL}(\boldsymbol{w},\theta;\boldsymbol{x}^{(i)},y^{(i)})$$

General form

Empirical risk minimization (i.e., minimize training error):

$$\min_{\boldsymbol{w} \in \mathbb{R}^d, \theta \in \mathbb{R}} \qquad \frac{1}{n} \sum_{i=1}^{n} \mathbb{1}\left\{ y^{(i)} \left(\langle \boldsymbol{w}, \boldsymbol{x}^{(i)} \rangle - \theta \right) \leq 0 \right\}$$

Soft-margin SVM:

$$\min_{\boldsymbol{w} \in \mathbb{R}^d, \theta \in \mathbb{R}} \qquad \frac{\lambda}{2} \|\boldsymbol{w}\|_2^2 + \frac{1}{n} \sum_{i=1}^{n} \mathrm{HL}(\boldsymbol{w}, \theta; \boldsymbol{x}^{(i)}, y^{(i)})$$

Generic learning objective:

$$\min_{\boldsymbol{w} \in \mathbb{R}^d, \theta \in \mathbb{R}} \qquad R(\boldsymbol{w}, \theta) + \frac{1}{n} \sum_{i=1}^{n} \ell(\boldsymbol{w}, \theta; \boldsymbol{x}^{(i)}, y^{(i)})$$

- **Regularization**: encodes “learning bias” (e.g., preference for large margins), sometimes promotes stability.
- **Data fitting/empirical loss**: how poorly does the classifier “fit” the data.

Learning via optimization

- Many different choices for regularization and loss.
  - $R(\boldsymbol{w}) \propto \|\boldsymbol{w}\|_2^2$: encourage large margins
  - $R(\boldsymbol{w}) \propto \|\boldsymbol{w}\|_1$: encourage $\boldsymbol{w}$ to be sparse
  - $R(\boldsymbol{w}) \propto \sum_{i=1}^{n} w_i \ln w_i$: "maximum entropy" interpretation
  - $\ell(\boldsymbol{w}, \theta; \boldsymbol{x}, y) = [\langle \boldsymbol{w}, \boldsymbol{x} \rangle - \theta - y]_+^2$
  - $\ell(\boldsymbol{w}, \theta; \boldsymbol{x}, y) = \log(1 + \exp(-y(\langle \boldsymbol{w}, \boldsymbol{x} \rangle - \theta)))$
  - ...
  - Also used beyond classification (e.g., regression, parameter estimation, clustering)
- Often want ℓ be an upper-bound on zero-one loss—i.e., a **surrogate loss**.
- Trade-off parameter λ: usually determine using hold out error or cross validation error.
- Computationally easier when overall objective function is **convex**: possible to efficiently find global minimizer in polynomial time.
- **Next**: techniques for analyzing and solving these optimization problems.