MATH 637: Mathematical Techniques in Data Science
Support vector machines

Dominique Guillot

Departments of Mathematical Sciences
University of Delaware

April 6, 2020

Hyperplanes

Recall:

- A *hyperplane* H in $V = \mathbb{R}^n$ is a subspace of V of dimension $n - 1$ (i.e., a subspace of codimension 1).
- Each hyperplane is determined by a nonzero vector $\beta \in \mathbb{R}^n$ via

$$H = \{x \in \mathbb{R}^n : \beta^T x = 0\} = \operatorname{span}(\beta)^\perp.$$

Hyperplanes

Recall:

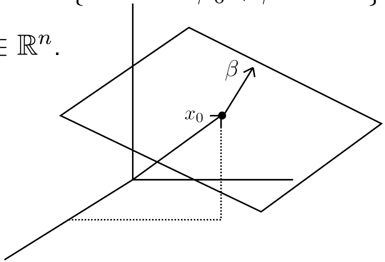
- A *hyperplane* H in $V = \mathbb{R}^n$ is a subspace of V of dimension $n-1$ (i.e., a subspace of codimension 1).
- Each hyperplane is determined by a nonzero vector $\beta \in \mathbb{R}^n$ via

$$H = \{x \in \mathbb{R}^n : \beta^T x = 0\} = \text{span}(\beta)^\perp.$$

- An *affine hyperplane* H in $\mathbb{R}^n$ is a subset of the form

$$H = \{x \in \mathbb{R}^n : \beta_0 + \beta^T x = 0\}$$

where $\beta_0 \in \mathbb{R}, \beta \in \mathbb{R}^n$.



Recall:

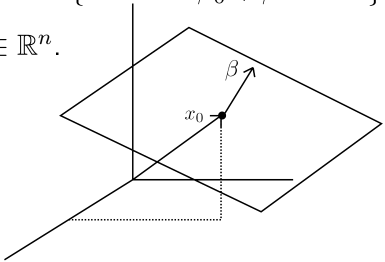
- A *hyperplane* H in $V = \mathbb{R}^n$ is a subspace of V of dimension $n-1$ (i.e., a subspace of codimension 1).
- Each hyperplane is determined by a nonzero vector $\beta \in \mathbb{R}^n$ via

$$H = \{x \in \mathbb{R}^n : \beta^T x = 0\} = \mathrm{span}(\beta)^\perp.$$

- An *affine hyperplane* H in $\mathbb{R}^n$ is a subset of the form

$$H = \{x \in \mathbb{R}^n : \beta_0 + \beta^T x = 0\}$$

where $\beta_0 \in \mathbb{R}, \beta \in \mathbb{R}^n$.

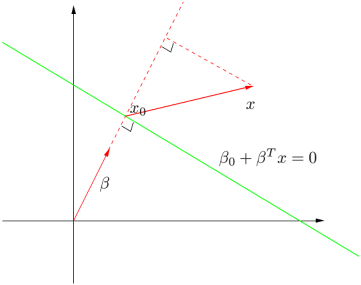


- We often use the term "hyperplane" for "affine hyperplane".

Hyperplanes (cont.)

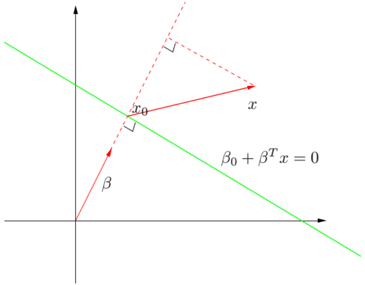
Let

$$H = \{x \in \mathbb{R}^n : \beta_0 + \beta^T x = 0\}.$$



Let

$$H = \{x \in \mathbb{R}^n : \beta_0 + \beta^T x = 0\}.$$



Note that for $x_0, x_1 \in H$,

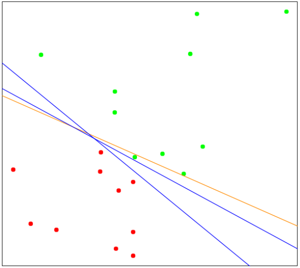
$$\beta^T(x_0 - x_1) = 0.$$

Thus β is perpendicular to H. It follows that for $x \in \mathbb{R}^n$,

$$d(x, H) = \frac{\beta^T}{\|\beta\|}(x - x_0) = \frac{\beta_0 + \beta^T x}{\|\beta\|} = \frac{x^T\beta + \beta_0}{\|\beta\|}.$$

Separating hyperplane

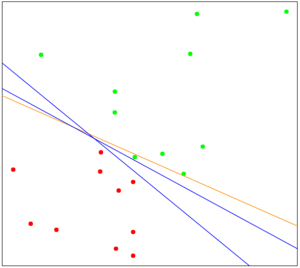
Suppose we have binary data with labels $\{+1, -1\}$. We want to separate data using an (affine) hyperplane.



ESL, Figure 4.14. (Orange = least-squares)

Separating hyperplane

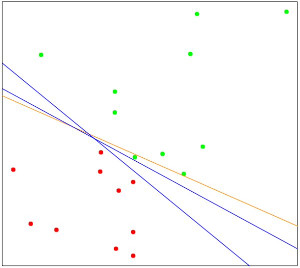
Suppose we have binary data with labels $\{+1, -1\}$. We want to separate data using an (affine) hyperplane.



ESL, Figure 4.14. (Orange = least-squares)

Classify using $G(x) = \mathrm{sgn}(x^T\beta + \beta_0)$.

Suppose we have binary data with labels $\{+1, -1\}$. We want to separate data using an (affine) hyperplane.

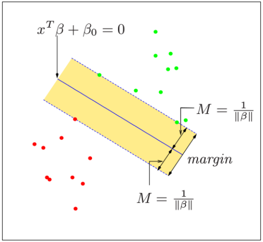


ESL, Figure 4.14. (Orange = least-squares)

Classify using $G(x) = \mathrm{sgn}(x^T\beta + \beta_0)$.

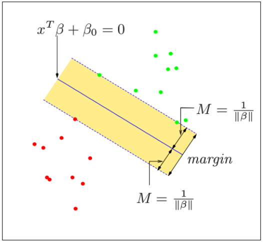
- Separating hyperplane may not be unique.
- Separating hyperplane may not exist (i.e., data may not be separable).

Uniqueness problem: when the data is separable, choose the hyperplane to maximize the "margin" (the "no man's land").



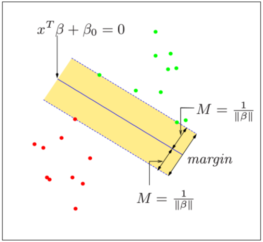
Margins

Uniqueness problem: when the data is separable, choose the hyperplane to maximize the "margin" (the "no man's land").



Data: $(y_i, x_i) \in \{+1, -1\} \times \mathbb{R}^p \qquad (i = 1, \ldots, n)$.
Suppose $\beta_0 + \beta^T x$ is a separating hyperplane with $\|\beta\| = 1$.

Uniqueness problem: when the data is separable, choose the hyperplane to maximize the "margin" (the "no man's land").

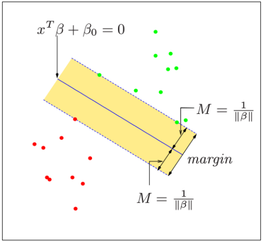


Data: $(y_i, x_i) \in \{+1, -1\} \times \mathbb{R}^p \qquad (i = 1, \ldots, n).$
Suppose $\beta_0 + \beta^T x$ is a separating hyperplane with $\|\beta\| = 1$.
Note that:

$$y_i(x_i^T\beta + \beta_0) > 0 \Rightarrow \text{Correct classification}$$
$$y_i(x_i^T\beta + \beta_0) < 0 \Rightarrow \text{Incorrect classification}$$

Margins

Uniqueness problem: when the data is separable, choose the hyperplane to maximize the "margin" (the "no man's land").



Data: $(y_i, x_i) \in \{+1, -1\} \times \mathbb{R}^p \qquad (i = 1, \ldots, n)$.
Suppose $\beta_0 + \beta^T x$ is a separating hyperplane with $\|\beta\| = 1$.
Note that:

$$y_i(x_i^T\beta + \beta_0) > 0 \Rightarrow \text{Correct classification}$$

$$y_i(x_i^T\beta + \beta_0) < 0 \Rightarrow \text{Incorrect classification}$$

Also, $|y_i(x_i^T\beta + \beta_0)|$ = distance between x and hyperplane (since $\|\beta\| = 1$).

Margins (cont.)

Thus, if the data is separable, we can solve

$$\max_{\beta_0, \beta \in \mathbb{R}^p, \|\beta\|=1} M \quad \text{s.t.} \quad y_i(x_i^T \beta + \beta_0) \geq M \qquad (i = 1, \ldots, n).$$

Margins (cont.)

Thus, if the data is separable, we can solve

$$\max_{\beta_0, \beta \in \mathbb{R}^p, \|\beta\|=1} M \quad \text{s.t.} \quad y_i(x_i^T \beta + \beta_0) \geq M \qquad (i = 1, \ldots, n).$$

We transform the problem into a usual form used in convex optimization.

Thus, if the data is separable, we can solve

$$\max_{\beta_0, \beta \in \mathbb{R}^p, \|\beta\|=1} M \quad \text{s.t.} \quad y_i(x_i^T \beta + \beta_0) \geq M \qquad (i = 1, \ldots, n).$$

We transform the problem into a usual form used in convex optimization.

- We can remove $\|\beta\| = 1$ by replacing the constraint by

$$\frac{1}{\|\beta\|} y_i(x_i^T \beta + \beta_0) \geq M, \quad \text{or equivalently,} \quad y_i(x_i^T \beta + \beta_0) \geq M\|\beta\|.$$

Thus, if the data is separable, we can solve

$$\max_{\beta_0, \beta \in \mathbb{R}^p, \|\beta\|=1} M \quad \text{s.t.} \quad y_i(x_i^T \beta + \beta_0) \geq M \qquad (i = 1, \dots, n).$$

We transform the problem into a usual form used in convex optimization.

- We can remove $\|\beta\| = 1$ by replacing the constraint by

$$\frac{1}{\|\beta\|} y_i(x_i^T \beta + \beta_0) \geq M, \quad \text{or equivalently,} \quad y_i(x_i^T \beta + \beta_0) \geq M\|\beta\|.$$

- We can always rescale (β, β_0) so that $\|\beta\| = 1/M$:

$$\max_{\beta_0, \beta \in \mathbb{R}^p} \frac{1}{\|\beta\|} \quad \text{s.t.} \quad y_i(x_i^T \beta + \beta_0) \geq 1 \qquad (i = 1, \dots, n).$$

Thus, if the data is separable, we can solve

$$\max_{\beta_0, \beta \in \mathbb{R}^p, \|\beta\|=1} M \quad \text{s.t.} \quad y_i(x_i^T \beta + \beta_0) \geq M \qquad (i = 1, \dots, n).$$

We transform the problem into a usual form used in convex optimization.

- We can remove $\|\beta\| = 1$ by replacing the constraint by

$$\frac{1}{\|\beta\|} y_i(x_i^T \beta + \beta_0) \geq M, \quad \text{or equivalently,} \quad y_i(x_i^T \beta + \beta_0) \geq M \|\beta\|.$$

- We can always rescale (β, β_0) so that $\|\beta\| = 1/M$:

$$\max_{\beta_0, \beta \in \mathbb{R}^p} \frac{1}{\|\beta\|} \quad \text{s.t.} \quad y_i(x_i^T \beta + \beta_0) \geq 1 \qquad (i = 1, \dots, n).$$

Equivalently,

$$\min_{\beta_0, \beta \in \mathbb{R}^p} \frac{1}{2}\|\beta\|^2 \quad \text{s.t.} \quad y_i(x_i^T \beta + \beta_0) \geq 1 \qquad (i = 1, \dots, n).$$

Margins (cont.)

Thus, if the data is separable, we can solve

$$\max_{\beta_0,\beta\in\mathbb{R}^p,\|\beta\|=1} M \quad \text{s.t.} \quad y_i(x_i^T\beta+\beta_0) \geq M \qquad (i=1,\dots,n).$$

We transform the problem into a usual form used in convex optimization.

- We can remove $\|\beta\| = 1$ by replacing the constraint by

$$\frac{1}{\|\beta\|}y_i(x_i^T\beta+\beta_0) \geq M, \quad \text{or equivalently,} \quad y_i(x_i^T\beta+\beta_0) \geq M\|\beta\|.$$

- We can always rescale (β, β_0) so that $\|\beta\| = 1/M$:

$$\max_{\beta_0,\beta\in\mathbb{R}^p} \frac{1}{\|\beta\|} \quad \text{s.t.} \quad y_i(x_i^T\beta+\beta_0) \geq 1 \qquad (i=1,\dots,n).$$

Equivalently,

$$\min_{\beta_0,\beta\in\mathbb{R}^p} \frac{1}{2}\|\beta\|^2 \quad \text{s.t.} \quad y_i(x_i^T\beta+\beta_0) \geq 1 \qquad (i=1,\dots,n).$$

We now recognize the problem as a convex optimization problem with a quadratic objective, and linear inequality constraints.

Support vector machines

- The previous problem works well when the data is *separable*. What happens if there is no way to find a margin?

- The previous problem works well when the data is *separable*.

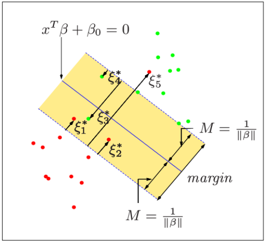
What happens if there is no way to find a margin?

- We allow some points to be on the wrong side of the margin, but

keep control on the error.

- The previous problem works well when the data is *separable*. What happens if there is no way to find a margin?
- We allow some points to be on the wrong side of the margin, but keep control on the error.
- We replace $y_i(x_i^T\beta+\beta_0)\geq M$ by

$$y_i(x_i^T\beta+\beta_0)\geq M(1-\xi_i), \qquad \xi_i\geq 0,$$



- We add a constraint to keep control on the error

- The previous problem works well when the data is *separable*.

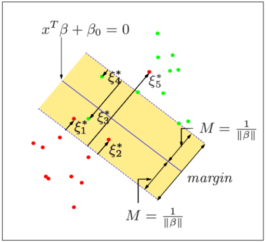
What happens if there is no way to find a margin?

- We allow some points to be on the wrong side of the margin, but

keep control on the error.

- We replace $y_i(x_i^T\beta + \beta_0) \geq M$ by

$$y_i(x_i^T\beta + \beta_0) \geq M(1 - \xi_i), \qquad \xi_i \geq 0,$$



- We add a constraint to keep control on the error

$$\sum_{i=1}^{n} \xi_i \leq C \qquad \text{for some fixed constant } C > 0.$$

Support vector machines (cont.)

The problem becomes:

$$\max_{\beta_0, \beta \in \mathbb{R}^p, \|\beta\|=1} M$$

$$\text{subject to } y_i(x_i^T \beta + \beta_0) \geq M(1 - \xi_i)$$

$$\xi_i \geq 0, \qquad \sum_{i=1}^{n} \xi_i \leq C.$$

Support vector machines (cont.)

The problem becomes:

$$\max_{\beta_0, \beta \in \mathbb{R}^p, \|\beta\|=1} M$$
$$\text{subject to } y_i(x_i^T \beta + \beta_0) \geq M(1 - \xi_i)$$
$$\xi_i \geq 0, \qquad \sum_{i=1}^{n} \xi_i \leq C.$$

As before, we can transform the problem into its "normal" form:

$$\min_{\beta_0, \beta} \frac{1}{2} \|\beta\|^2$$
$$\text{subject to } y_i(x_i^T \beta + \beta_0) \geq 1 - \xi_i$$
$$\xi_i \geq 0, \qquad \sum_{i=1}^{n} \xi_i \leq C.$$

Problem can be solved using standard optimization packages.

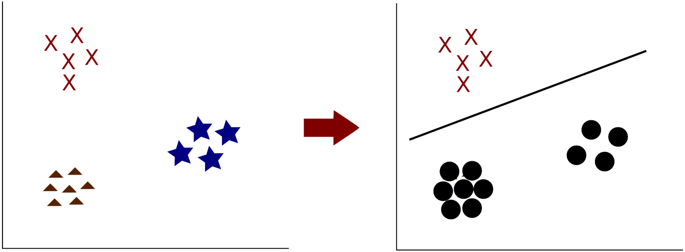
Multiple classes of data

The SVM is a binary classifier. How can we classify data with $K > 2$ classes?

Multiple classes of data

The SVM is a binary classifier. How can we classify data with $K > 2$ classes?

- **One versus all:**(or one versus the rest) Fit the model to separate each class against the remaining classes. Label a new point x according to the model for which $x^T\beta + \beta_0$ is the largest.



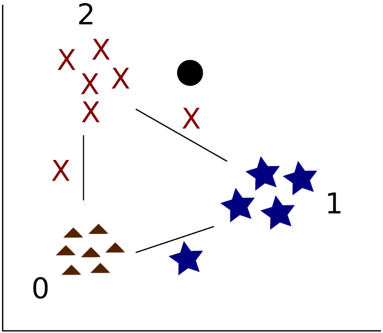
Need to fit the model K times.

Multiple classes of data (cont.)

- **One versus one:**
  1. Train a classifier for each possible **pair** of classes.
     Note: There are $\binom{K}{2} = K(K-1)/2$ such pairs.
  2. Classify a new points according to a **majority vote**: count the number of times the new point is assign to a given class, and pick the class with the largest number.

Multiple classes of data (cont.)

- **One versus one:**
  1. Train a classifier for each possible **pair** of classes. Note: There are $\binom{K}{2} = K(K-1)/2$ such pairs.
  2. Classify a new points according to a **majority vote**: count the number of times the new point is assign to a given class, and pick the class with the largest number.



Need to fit the model $\binom{K}{2}$ times (computationally intensive).