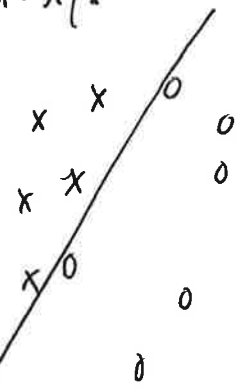
EoSML: Lecture #13

- Support vector machines (SVM) provide a method for classifying linearly separable data: estimate a hyperplane that realizes the separation. The idea can then be extended to non-separable data

ex:



Want to choose a hyperplane which will generalize well. This leads to a geometric optimization problem: want to choose a separating hyperplane that has the largest margin, i.e. space of separation between the two classes.

- More precisely, let $\mathcal{X} \subset \mathbb{R}^D$, $\mathcal{Y} = \{-1, 1\}$, and $f : \mathcal{X} \to \mathcal{Y}$ be a target function we want to learn. Given a sample $S = \{(x_i, y_i)\}_{i=1}^{n} \subset \mathcal{X} \times \mathcal{Y}$ ($\rightarrow f(x_i) = y_i$, deterministic) generated from a (unknown) distribution D, we want to learn the generalization-error minimizing predictor h^*:

$$h^* = \arg\min_h \mathbb{P}_{x \sim D}\left(f(x) \neq h(x) \right).$$

- The (linear) SVM framework has us consider the hypothesis class of linear separators

$$\mathcal{H} = \{ x \mapsto \operatorname{sgn}(w \cdot x + b),\ w \in \mathbb{R}^D,\ b \in \mathbb{R} \}.$$

Note that $\{x \mid wx + b = 0\}$ is a hyperplane, so $\mathcal{H}$ is the family of hyperplane predictors.

- We will start by supposing our training data S is linearly separable, i.e. $\exists\, w^*, b^*$ s.t. the classifier $x_i \mapsto \operatorname{sgn}(w^* \cdot x_i + b^*)$ is perfectly accurate, i.e.

$$\operatorname{sgn}(w^* \cdot x_i + b) = y_i \quad \forall i = 1, \dots, m.$$

$$\Leftrightarrow \quad y_i(w^* \cdot x_i + b) \geq 0 \quad \forall i = 1, \dots, m.$$

- Generically, if there is one such choice of $(w^*, b^*) \in \mathbb{R}^{D+1}$, there are infinitely many, just by slightly perturbing the parameters. We choose a "best" pair by selecting the margin-maximizing pair.

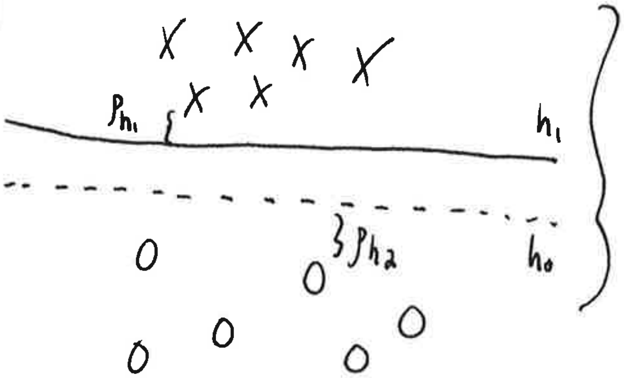
Defn.: Let $h: x \mapsto w \cdot x + b$ be a linear classifier. Its *geometric margin* at x is

$$\rho_h(x) = \text{distance of } x \text{ to } \{x' \mid w \cdot x' + b = 0\}$$

$$= \frac{|w \cdot x + b|}{\|w\|_2}$$

The geometric margin ρ_h of a linear classifier over a sample $S = \{(x_i, y_i)\}_{i=1}^m$ is $\rho_h = \min_{i=1,\dots,m} \rho_h(x_i)$.

- We choose the minimum to ensure ($\rho_h(x)$ large $\Rightarrow$ h ~~well~~ decisively classifies every point).



The dotted line has a larger geometric margin over the sample than does the solid line, i.e. $\rho_{h_2} > \rho_{h_1}$.

- The big question is how to learn these margin-maximizing choices of parameters.

- Consider a separating hyperplane with parameters (w, b). The associated margin can be maximized by considering the optimization problem

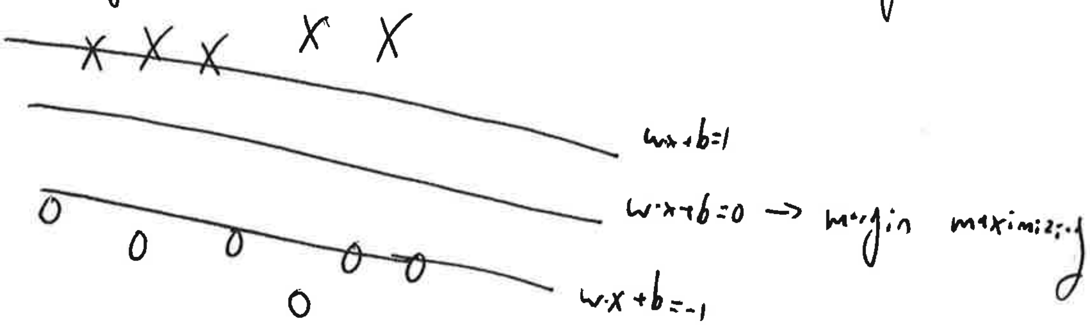
$$\rho = \max_{\substack{(w,b) \in \mathbb{R}^{D+1} \text{ s.t.} \\ y_i(w \cdot x_i + b) \geq 0 \ \forall i=1,\dots,m}} \frac{|w \cdot x_i + b|}{\|w\|_2} = \max_{(w,b) \in \mathbb{R}^{D+1}} \min_{i=1,\dots,m} \frac{y_i(w \cdot x_i + b)}{\|w\|_2}.$$

$\forall \alpha \neq 0$

· Noting that $\{x \mid w\cdot x+b=0\} = \{x \mid (\frac{w}{\alpha})\cdot x + (\frac{b}{\alpha})=0\}$, we may WLOG suppose $\min_{i=1,\dots,m} y_i(w\cdot x_i+b)=1$. This yields a simpler formulation:

$$\rho = \max_{\substack{(w,b)\in\mathbb{R}^{D+1} \text{ s.t.} \\ \min_{i=1,\dots,m} y_i(w\cdot x_i+b)=1}} \frac{1}{\|w\|_2} = \max_{\substack{(w,b)\in\mathbb{R}^{D+1} \text{ s.t.} \\ \forall i=1,\dots,m \; y_i(w\cdot x_i+b)\geq 1}} \frac{1}{\|w\|_2}.$$

· Visually, we can see if $w\cdot x+b=0$ defines the MM hyperplane, then $w\cdot x+b=\pm 1$ define marginal planes:



· Note that the form of the marginal hyperplanes as $w\cdot x+b=\pm 1$ is guaranteed by the assumption that $\min_{i=1,\dots,m} |w\cdot x_i+b|=1$. There are two such marginals (one positive, one negative) because otherwise we could perturb a bit and increase the margin.

· We may write our maximization problem as a minimization problem:

$$\min_{(w,b)\in\mathbb{R}^{D+1}} \frac{1}{2}\|w\|_2^2 \quad \text{s.t. } y_i(w\cdot x_i+b)\geq 1 \quad \forall i=1,\dots,m. \quad ★$$

· The objective function $F(w)=\frac{1}{2}\|w\|_2^2$ is C^∞. Note that $\nabla F = w$

$$\nabla^2 F = I.$$

In particular, F is strictly convex, which is very convenient from an optimization standpoint.

- Moreover, since the constraints $y_i(w \cdot x_i + b) \geq 1$, $i = 1, \dots, m$ are affine, we are guaranteed that ⊛ has a unique solution theoretically, and can be solved practically using algorithms for quadratic programming.

- From an optimization standpoint, it is convenient to introduce the Lagrangian

$$\mathcal{L}(w, b, \alpha) = \frac{1}{2}\|w\|_2^2 - \sum_{i=1}^{m} \alpha_i \left(y_i (w \cdot x_i + b) - 1 \right),$$

where $\alpha = (\alpha_1, \dots, \alpha_m)$, $\alpha_i \geq 0$ are the Lagrange variables (multipliers).

- We may then introduce KKT conditions by setting $\nabla_w \mathcal{L} = 0$, $\nabla_b \mathcal{L} = 0$:

$$\nabla_w \mathcal{L} = 0 \iff w - \sum_{i=1}^{m} \alpha_i y_i x_i = 0 \iff w = \sum_{i=1}^{m} \alpha_i y_i x_i$$

$$\nabla_b \mathcal{L} = 0 \iff -\sum_{i=1}^{m} \alpha_i y_i = 0 \iff \sum_{i=1}^{m} \alpha_i y_i = 0$$

- Moreover, $\alpha_i \left[y_i (w \cdot x_i + b) - 1 \right] = 0 \iff \alpha_i = 0$ or $y_i(w \cdot x_i + b) = 1$

- The condition that $w = \sum_{i=1}^{m} \alpha_i y_i x_i$ combined with $\alpha_i = 0$ or $y_i(w \cdot x_i + b) = 1$ tells us that if x_i appears in the sum defining w (i.e. $\alpha_i \neq 0$) then $y_i(w \cdot x_i + b) = 1$, i.e. x_i is margin minimizing. Such x_i are called <u>support vectors</u>, and they define the solution.

<u>Remark</u>: Support vectors may not be unique, if for example multiple training points lie on a marginal hyperplane.

· The problem $\star$ admits a dual formulation:

$$\mathcal{L} = \frac{1}{2}\left\|\sum_{i=1}^{m}\alpha_i y_i x_i\right\|_2^2 - \sum_{i=1}^{m}\sum_{j=1}^{m}\alpha_i\alpha_j y_i y_j (x_i \cdot x_j) - \sum_{i=1}^{m}\alpha_i y_i b + \sum_{i=1}^{m}\alpha_i$$

$$= \sum_{i=1}^{m}\alpha_i - \frac{1}{2}\sum_{i,j=1}^{m}\alpha_i\alpha_j y_i y_j (x_i \cdot x_j).$$

· This leads us to

$\star\star$

$$\max_{\alpha=(\alpha_1,\dots,\alpha_m)} \sum_{i=1}^{m}\alpha_i - \frac{1}{2}\sum_{i,j=1}^{m}\alpha_i\alpha_j y_i y_j (x_i \cdot x_j) \quad \text{s.t. } \alpha_i \geq 0 \ \forall i=1,\dots,m \text{ and } \sum_{i=1}^{m}\alpha_i y_i = 0.$$

· This is also a nice (e.g. concave) optimization problem that is quadratic in α, so can be handled with quadratic programming algorithms.

· Moreover, strong duality holds, so $\star \Leftrightarrow \star\star$, i.e. we can use the α learned in $\star\star$ to get the solution to $\star$.