SUPPORT VECTOR MACHINES

Idea: explicitly search for large-margin classifiers.

EASY CASE: DATA IS SEPARABLE

$$\min_{\vec{x}, b} \frac{1}{\gamma(\vec{x}, b)}$$

$$\text{s.t.} \quad y_i\left(\langle x_i, \vec{x} \rangle + b\right) \geq 1$$

(How do we optimize this?)

What if there's noise? Give each point some slack, try to optimize combination of slack and margin:

$$\min_{\vec{x}, b, \vec{\xi}} \frac{1}{\gamma(\vec{x}, b)} + C \sum_i \xi_i$$

$$\text{subject to} \quad y_i\left(\langle x_i, \vec{x} \rangle + b\right) \geq 1 - \xi_i$$

$$\xi_i \geq 0$$

(How do you compute margin there?)

FROM LARGE MARGIN TO SMALL WEIGHTS

$$d^+ = \frac{1}{||w||} w \cdot x^+ + b - 1$$

$$d^- = -\frac{1}{||w||} w \cdot x^- - b + 1$$

We can then compute the margin by algebra:

$$\begin{aligned}
\gamma &= \frac{1}{2}\left[d^+ - d^-\right] \\
&= \frac{1}{2}\left[\frac{1}{||w||} w \cdot x^+ + b - 1 - \frac{1}{||w||} w \cdot x^- - b + 1\right] \\
&= \frac{1}{2}\left[\frac{1}{||w||} w \cdot x^+ - \frac{1}{||w||} w \cdot x^-\right] \\
&= \frac{1}{2}\left[\frac{1}{||w||}(+1) - \frac{1}{||w||}(-1)\right] \\
&= \frac{1}{||w||}
\end{aligned}$$

$$\min_{\vec{x}, b, \vec{\xi}} \quad ||\vec{x}||^2 + C \sum_i \xi_i$$

subject to $y_i(\langle x_i, \vec{x} \rangle + b) \geq 1 - \xi_i$

$\xi_i \geq 0$

WHAT ARE THE VALUES OF THE OPTIMAL ξ_i ?

$$\xi_n = \begin{cases} 0 & \text{if } y_n(\boldsymbol{w} \cdot \boldsymbol{x}_n + b) \geq 1 \\ 1 - y_n(\boldsymbol{w} \cdot \boldsymbol{x}_n + b) & \text{otherwise} \end{cases}$$

In other words, the optimal value for a slack variable is *exa* hinge loss on the corresponding example! Thus, we can wr SVM objective as an *unconstrained* optimization problem:

$$\min_{\boldsymbol{w},b} \quad \underbrace{\frac{1}{2}||\boldsymbol{w}||^2}_{\text{large margin}} + \underbrace{C\sum_n \ell^{(\text{hin})}(y_n, \boldsymbol{w} \cdot \boldsymbol{x}_n + b)}_{\text{small slack}}$$

As a consequence, Algorithm 24 finds a large-margin classifier!

Algorithm 24 HingeRegularizedGD(**D**, λ, *MaxIter*)

$\boldsymbol{w} \leftarrow \langle 0, 0, \ldots 0 \rangle$, $b \leftarrow 0$ // initialize weights and bias
for $iter = 1 \ldots MaxIter$ **do**
  $\boldsymbol{g} \leftarrow \langle 0, 0, \ldots 0 \rangle$, $g \leftarrow 0$ // initialize gradient of weights and bias
  for all $(\boldsymbol{x}, y) \in$ **D** **do**
    if $y(\boldsymbol{w} \cdot \boldsymbol{x} + b) \leq 1$ **then**
      $\boldsymbol{g} \leftarrow \boldsymbol{g} + y\,\boldsymbol{x}$ // update weight gradient
      $g \leftarrow g + y$ // update bias derivative
    end if
  end for
  $\boldsymbol{g} \leftarrow \boldsymbol{g} - \lambda \boldsymbol{w}$ // add in regularization term
  $\boldsymbol{w} \leftarrow \boldsymbol{w} + \eta \boldsymbol{g}$ // update weights
  $b \leftarrow b + \eta g$ // update bias
end for
return $\boldsymbol{w}, b$