Machine Learning (CSE 446): Support Vector Machines (continued)

Noah Smith

University of Washington
nasmith@cs.washington.edu

November 20, 2017

Quick Review: Kernels and SVMs

Kernels

A **kernel** function (implicitly) computes:

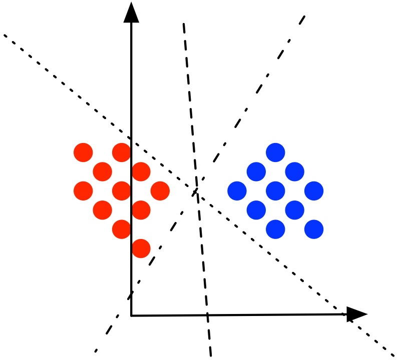
$$K(\mathbf{x}, \mathbf{v}) = \boldsymbol{\phi}(\mathbf{x}) \cdot \boldsymbol{\phi}(\mathbf{v})$$

for some ϕ. Typically it is *cheap* to compute $K(\cdot, \cdot)$, and we never explicitly represent $\boldsymbol{\phi}(\mathbf{v})$ for any vector $\mathbf{v}$.

Some kernels:

$$\begin{aligned}
\text{linear} \quad & K^{\text{linear}}(\mathbf{x}, \mathbf{v}) = \mathbf{x} \cdot \mathbf{v} \\
\text{quadratic} \quad & K^{\text{quad}}(\mathbf{x}, \mathbf{v}) = (1 + \mathbf{x} \cdot \mathbf{v})^2 \\
\text{cubic} \quad & K^{\text{cubic}}(\mathbf{x}, \mathbf{v}) = (1 + \mathbf{x} \cdot \mathbf{v})^3 \\
\text{polynomial} \quad & K_p^{\text{poly}}(\mathbf{x}, \mathbf{v}) = (1 + \mathbf{x} \cdot \mathbf{v})^p \\
\text{radial basis function} \quad & K_\gamma^{\text{rbf}}(\mathbf{x}, \mathbf{v}) = \exp\left(-\gamma \left\|\mathbf{x} - \mathbf{v}\right\|_2^2\right) \\
\text{hyperbolic tangent} \quad & \tilde{K}^{\text{tanh}}(\mathbf{x}, \mathbf{v}) = \tanh(1 + \mathbf{x} \cdot \mathbf{v}) \quad \text{(not a kernel)} \\
\text{all conjunctions} \quad & K^{\text{all conj}}(\mathbf{x}, \mathbf{v}) = \textstyle\prod_{j=1}^{d}(1 + x_j v_j) \quad \text{(for binary features)}
\end{aligned}$$

Choosing a Hyperplane



"Soft-Margin SVM"

$$\min_{\mathbf{w},b,\boldsymbol{\zeta}} \quad \overbrace{\|\mathbf{w}\|_2^2}^{\text{large margin}} \quad + C \overbrace{\sum_{n=1}^{N} \zeta_n}^{\text{small slack}}$$

$$\text{s.t. } y_n \cdot (\mathbf{w} \cdot \mathbf{x}_n + b) \geq 1 - \zeta_n, \forall n$$

$$\zeta_n \geq 0, \forall n$$

(C is a hyperparameter.)

"Soft-Margin SVM"

$$
\begin{aligned}
\min_{\mathbf{w}, b, \boldsymbol{\zeta}} \quad & \overbrace{\|\mathbf{w}\|_2^2}^{\text{large margin}} + C \overbrace{\sum_{n=1}^{N} \zeta_n}^{\text{small slack}} \\
\text{s.t.} \quad & y_n \cdot (\mathbf{w} \cdot \mathbf{x}_n + b) \geq 1 - \zeta_n, \forall n \\
& \zeta_n \geq 0, \forall n
\end{aligned}
$$

(C is a hyperparameter.)

Claim: solving this problem is equivalent to minimizing the hinge loss, with L_2 regularization. Choosing C equates to choosing λ (the regularization strength).

The Dual Form of Soft-Margin SVMs

$$\min_{\boldsymbol{\alpha}} \frac{1}{2} \sum_{n=1}^{N} \sum_{i=1}^{N} \alpha_n \cdot \alpha_i \cdot y_n \cdot y_i \cdot (\mathbf{x}_n \cdot \mathbf{x}_i) - \sum_{n=1}^{N} \alpha_n$$

$$\text{s.t. } 0 \leq \alpha_n \leq C, \forall n$$

This is a **quadratic** problem with “bound” constraints.

Note that now it is possible to kernelize, replacing $\mathbf{x}_n \cdot \mathbf{x}_i$ with $K(\mathbf{x}_n, \mathbf{x}_i)$.

Thinking about the Dual Form

$$\min_{\boldsymbol{\alpha}} \frac{1}{2} \sum_{n=1}^{N} \sum_{i=1}^{N} \alpha_n \cdot \alpha_i \cdot y_n \cdot y_i \cdot K(\mathbf{x}_n, \mathbf{x}_i) - \sum_{n=1}^{N} \alpha_n$$

s.t. $0 \leq \alpha_n \leq C, \forall n$

Thinking about the Dual Form

$$\min_{\boldsymbol{\alpha}} \frac{1}{2} \sum_{n=1}^{N} \sum_{i=1}^{N} \alpha_n \cdot \alpha_i \cdot y_n \cdot y_i \cdot K(\mathbf{x}_n, \mathbf{x}_i) - \sum_{n=1}^{N} \alpha_n$$

s.t. $0 \leq \alpha_n \leq C, \forall n$

Consider n and i such that $y_n = y_i$, so $y_n \cdot y_i = +1$, so that the objective seeks to *decrease* $\alpha_n \cdot \alpha_i \cdot K(\mathbf{x}_n, \mathbf{x}_i)$.

Thinking about the Dual Form

$$\min_{\boldsymbol{\alpha}} \frac{1}{2} \sum_{n=1}^{N} \sum_{i=1}^{N} \alpha_n \cdot \alpha_i \cdot y_n \cdot y_i \cdot K(\mathbf{x}_n, \mathbf{x}_i) - \sum_{n=1}^{N} \alpha_n$$

$$\text{s.t. } 0 \leq \alpha_n \leq C, \forall n$$

Consider n and i such that $y_n = y_i$, so $y_n \cdot y_i = +1$, so that the objective seeks to *decrease* $\alpha_n \cdot \alpha_i \cdot K(\mathbf{x}_n, \mathbf{x}_i)$.

- If $K(\mathbf{x}_n, \mathbf{x}_i)$ is small, then the αs don't matter much.
- If $K(\mathbf{x}_n, \mathbf{x}_i)$ is large ($\mathbf{x}_n$ and $\mathbf{x}_i$ are similar), then one of the αs should be close to zero.

Thinking about the Dual Form

$$\min_{\boldsymbol{\alpha}} \frac{1}{2} \sum_{n=1}^{N} \sum_{i=1}^{N} \alpha_n \cdot \alpha_i \cdot y_n \cdot y_i \cdot K(\mathbf{x}_n, \mathbf{x}_i) - \sum_{n=1}^{N} \alpha_n$$

s.t. $0 \leq \alpha_n \leq C, \forall n$

Consider n and i such that $y_n \neq y_i$, so $y_n \cdot y_i = -1$, so that the objective seeks to *increase* $\alpha_n \cdot \alpha_i \cdot K(\mathbf{x}_n, \mathbf{x}_i)$.

Thinking about the Dual Form

$$\min_{\boldsymbol{\alpha}} \frac{1}{2} \sum_{n=1}^{N} \sum_{i=1}^{N} \alpha_n \cdot \alpha_i \cdot y_n \cdot y_i \cdot K(\mathbf{x}_n, \mathbf{x}_i) - \sum_{n=1}^{N} \alpha_n$$

$$\text{s.t. } 0 \leq \alpha_n \leq C, \forall n$$

Consider n and i such that $y_n \neq y_i$, so $y_n \cdot y_i = -1$, so that the objective seeks to *increase* $\alpha_n \cdot \alpha_i \cdot K(\mathbf{x}_n, \mathbf{x}_i)$.

- If $K(\mathbf{x}_n, \mathbf{x}_i)$ is small, then the αs don't matter much.
- If $K(\mathbf{x}_n, \mathbf{x}_i)$ is large ($\mathbf{x}_n$ and $\mathbf{x}_i$ are similar), then one of the αs should both be large.

A Slightly Different View

When will α_n be nonzero?

A Slightly Different View

When will α_n be nonzero?

Optimization theory says that, at the optimal $\boldsymbol{\alpha}$,

$$\begin{aligned} & \alpha_n \cdot (y_n \cdot (\mathbf{w} \cdot \mathbf{x}_n + b) - 1 + \zeta_n) = 0 \\ \Rightarrow \quad & \alpha_n = 0 \quad \vee \quad y_n \cdot (\mathbf{w} \cdot \mathbf{x}_n + b) - 1 + \zeta_n = 0 \end{aligned}$$

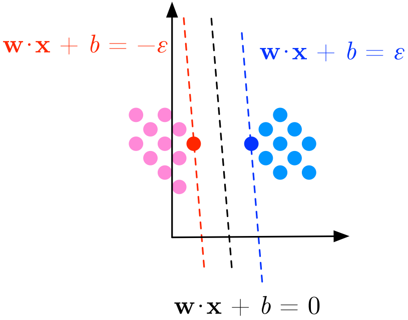
A Slightly Different View

When will α_n be nonzero?

Optimization theory says that, at the optimal $\boldsymbol{\alpha}$,

$$\alpha_n \cdot (y_n \cdot (\mathbf{w} \cdot \mathbf{x}_n + b) - 1 + \zeta_n) = 0$$
$$\Rightarrow \quad \alpha_n = 0 \quad \vee \quad y_n \cdot (\mathbf{w} \cdot \mathbf{x}_n + b) - 1 + \zeta_n = 0$$

So $\alpha_n \neq 0$ only for n where $\mathbf{x}_n$ is precisely on the margin of the hyperplane.



But why are they called “support vector machines”?

The “support vectors” are the data points $\mathbf{x}_n$ where $\alpha_n > 0$.

They “support” the decision boundary.

They are the most “confusable” points; changing them will move the boundary.