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

Noah Smith
© 2017

University of Washington
nasmith@cs.washington.edu

November 20, 2017
Quick Review: Kernels and SVMs
Kernels

A kernel function (implicitly) computes:

\[ K(x, v) = \phi(x) \cdot \phi(v) \]

for some \( \phi \). Typically it is cheap to compute \( K(\cdot, \cdot) \), and we never explicitly represent \( \phi(v) \) for any vector \( v \).

Some kernels:

- linear \( K^{\text{linear}}(x, v) = x \cdot v \)
- quadratic \( K^{\text{quad}}(x, v) = (1 + x \cdot v)^2 \)
- cubic \( K^{\text{cubic}}(x, v) = (1 + x \cdot v)^3 \)
- polynomial \( K_p^{\text{poly}}(x, v) = (1 + x \cdot v)^p \)
- radial basis function \( K^{\text{rbf}}_{\gamma}(x, v) = \exp\left(-\gamma \|x - v\|_2^2\right) \)
- hyperbolic tangent \( \tilde{K}^{\text{tanh}}(x, v) = \tanh(1 + x \cdot v) \) (not a kernel)
- all conjunctions \( K^{\text{all conj}}(x, v) = \prod_{j=1}^{d}(1 + x_j v_j) \) (for binary features)
Choosing a Hyperplane
“Soft-Margin SVM”

\[
\begin{align*}
\min_{w, b, \zeta} & \quad \|w\|^2_2 + C \sum_{n=1}^{N} \zeta_n \\
\text{s.t.} & \quad y_n \cdot (w \cdot x_n + b) \geq 1 - \zeta_n, \forall n \\
& \quad \zeta_n \geq 0, \forall n
\end{align*}
\]

(C is a hyperparameter.)
"Soft-Margin SVM"

\[
\min_{w, b, \zeta} \left\{ \|w\|_2^2 + C \sum_{n=1}^{N} \zeta_n \right\} \\
\text{s.t. } y_n \cdot (w \cdot x_n + b) \geq 1 - \zeta_n, \forall n \\
\zeta_n \geq 0, \forall n
\]

\(C\) is a hyperparameter.

Claim: solving this problem is equivalent to minimizing the hinge loss, with \(L_2\) regularization. Choosing \(C\) equates to choosing \(\lambda\) (the regularization strength).
The Dual Form of Soft-Margin SVMs

\[
\min_{\alpha} \frac{1}{2} \sum_{n=1}^{N} \sum_{i=1}^{N} \alpha_n \cdot \alpha_i \cdot y_n \cdot y_i \cdot (x_n \cdot x_i) - \sum_{n=1}^{N} \alpha_n
\]

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 \(x_n \cdot x_i\) with \(K(x_n, x_i)\).
Thinking about the Dual Form

\[
\min_{\alpha} \frac{1}{2} \sum_{n=1}^{N} \sum_{i=1}^{N} \alpha_n \cdot \alpha_i \cdot y_n \cdot y_i \cdot K(x_n, x_i) - \sum_{n=1}^{N} \alpha_n \\
\text{s.t. } 0 \leq \alpha_n \leq C, \forall n
\]
Thinking about the Dual Form

\[
\min_{\alpha} \frac{1}{2} \sum_{n=1}^{N} \sum_{i=1}^{N} \alpha_n \cdot \alpha_i \cdot y_n \cdot y_i \cdot K(x_n, x_i) - \sum_{n=1}^{N} \alpha_n
\]

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 \(\alpha_n \cdot \alpha_i \cdot K(x_n, x_i)\).
Thinking about the Dual Form

\[
\min_{\alpha} \frac{1}{2} \sum_{n=1}^{N} \sum_{i=1}^{N} \alpha_n \cdot \alpha_i \cdot y_n \cdot y_i \cdot K(x_n, x_i) - \sum_{n=1}^{N} \alpha_n \\
\text{s.t. } 0 \leq \alpha_n \leq C, \forall n
\]

Consider \( n \) and \( i \) such that \( y_n = y_i \), so \( y_n \cdot y_i = +1 \), so that the objective seeks to decrease \( \alpha_n \cdot \alpha_i \cdot K(x_n, x_i) \).

- If \( K(x_n, x_i) \) is small, then the \( \alpha \)s don’t matter much.
- If \( K(x_n, x_i) \) is large (\( x_n \) and \( x_i \) are similar), then one of the \( \alpha \)s should be close to zero.
Thinking about the Dual Form

\[
\min_{\alpha} \frac{1}{2} \sum_{n=1}^{N} \sum_{i=1}^{N} \alpha_n \cdot \alpha_i \cdot y_n \cdot y_i \cdot K(x_n, x_i) - \sum_{n=1}^{N} \alpha_n
\]

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 \(\alpha_n \cdot \alpha_i \cdot K(x_n, x_i)\).
Thinking about the Dual Form

\[
\min_{\alpha} \frac{1}{2} \sum_{n=1}^{N} \sum_{i=1}^{N} \alpha_n \cdot \alpha_i \cdot y_n \cdot y_i \cdot K(x_n, x_i) - \sum_{n=1}^{N} \alpha_n \\
\text{s.t. } 0 \leq \alpha_n \leq C, \forall n
\]

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 \( \alpha_n \cdot \alpha_i \cdot K(x_n, x_i) \).

- If \( K(x_n, x_i) \) is small, then the \( \alpha \)s don’t matter much.
- If \( K(x_n, x_i) \) is large (\( x_n \) and \( x_i \) are similar), then one of the \( \alpha \)s should both be large.
When will $\alpha_n$ be nonzero?
A Slightly Different View

When will $\alpha_n$ be nonzero?

Optimization theory says that, at the optimal $\alpha$,

$$\alpha_n \cdot (y_n \cdot (w \cdot x_n + b) - 1 + \zeta_n) = 0$$

$$\Rightarrow \quad \alpha_n = 0 \quad \vee \quad y_n \cdot (w \cdot x_n + b) - 1 + \zeta_n = 0$$
When will $\alpha_n$ be nonzero?

Optimization theory says that, at the optimal $\alpha$,

$$\alpha_n \cdot (y_n \cdot (w \cdot x_n + b) - 1 + \zeta_n) = 0$$

$$\Rightarrow \alpha_n = 0 \quad \vee \quad y_n \cdot (w \cdot x_n + b) - 1 + \zeta_n = 0$$

So $\alpha_n \neq 0$ only for $n$ where $x_n$ is precisely on the margin of the hyperplane.
But why are they called “support vector machines”?

The “support vectors” are the data points $x_n$ where $\alpha_n > 0$.

They “support” the decision boundary.

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