Statistical Techniques in Robotics (16-831, F09) Lecture #14 (Thursday October 8th)

Support Vector Machines, part 2

Lecturer: Drew Bagnell *Scribe: Alan Kraut*

1 SVM Review

We have a set of linear constraints, that a binary classifier give the right classification on all training data. The linear classifier uses $\text{sign}(w^T f_i)$, where w is a weight vector, and f_i is a particular feature vector. The desired output class is $y_i \in [-1, 1]$.

- These constraints can be expressed as

$$y_i w^T f_i \geq 0 \tag{1}$$

- This series of constraints has several problems. With any real data it will have either infinite or zero solutions. We also can't incrementally update it.
- To find a single solution if there are infinite, we find the w that allows the greatest possible margin. That is, we want to minimize $||w||^2$ subject to

$$y_i w^T f_i \geq 1 \tag{2}$$

- To allow us to find a solution with inconsistent constraints, we introduce a flex variable, ξ. We now want to minimize $\lambda ||w||^2 + \sum_i \xi_i$ subject to

$$y_i w^T f_i \geq 1 - \xi_i, \quad \xi_i \geq 0 \tag{3}$$

- To make this online, we observe that $\xi = \max(0, 1 - y_i w^T f_i)$. This allows us to generate the loss function

$$l_t = \lambda ||w||^2 + \max(0, 1 - y_t w^T f_t) \tag{4}$$

- Our update for w is now as follows.

$$w \leftarrow w - 2\alpha_t \lambda w \tag{5}$$

  And if the output for this time step was incorrect,

$$w \leftarrow w + \alpha_t y_t f_t \tag{6}$$

2 Implementing SVMs

2.1 Selecting α_t

- Stock algorithm would be to set α_t proportional to $\frac{1}{\sqrt{t}}$.
- If we have d elements, each with a maximum value of $|f|_{max}$, the maximum gradient, G, is $\sqrt{d|f|_{max}}$.
- This is not as good as we could do.
- Notice that l_t is an extremely good convex function. It is a quadratic plus a convex function. In the same way all convex functions lie above a line (a subgradient) from every point, l_t lies above a quadratic from every point.
- Specifically, if it is always the case that

$$f(y) \geq f(x) + \frac{H}{2}(y-x)^2 + \nabla f_x^T(y-x) \tag{7}$$

  then $f(x)$ is said to be H-strongly convex.
- In this case l_t is λ-strongly convex.
- If $\alpha_t = \frac{G}{Ht}$, then regret$\leq \frac{G^2}{H}(1+\log t)$. $\log t$ is *really* good, and this learning rate and algorithm is essentially the current best for this class of problem.

2.2 SVMs with Multiple Classes

We can represent problems with more than two classes by having a weight vector, w_i for each class.

- When we get a classification of a particular example (for example, example i is of class 1), we generate a set of constraints that can be expressed as either

$$\begin{array}{rcl} w_1^T f_i & \geq & w_2^T f_i + 1 \\ w_1^T f_i & \geq & w_3^T f_i + 1 \\ w_1^T f_i & \geq & w_4^T f_i + 1 \\ & \cdots & \end{array} \tag{8}$$

  or

$$w_1^T f_i \geq \max_{c \neq i}(w_c^T f_i + 1) \tag{9}$$

- By the same argument as before

$$\xi = \max(0, \max(w_c^T f + 1) - w_1^T f) \tag{10}$$

- We want to update each w by gradient descent on the partial of the cost with respect to that particular w.

- Remember the cost is $l_t = \lambda ||w||^2 + \xi$. We want the update step to be

$$w_c \leftarrow w_c - \partial_{w_c} l_t \tag{11}$$

- In the case that the example was classified correctly, $\partial_{w_c} l_t = 0$. If it was misclassified, there are three cases with different partials: the correct class, the class we incorrectly decided this was an example of, and all others.

$$\begin{aligned}
\partial_{w_c} &= -f_i, \quad y_i = c && (12)\\
\partial_{w_c} &= f_i, \quad c = \underset{c}{\operatorname{argmax}}(w_c^T f_i + 1) && (13)\\
\partial_{w_c} &= 0, \quad \text{otherwise} && (14)
\end{aligned}$$

- That update is for the max representation. If we use the multiple constraints representation it is similar, except we update both w_1 and w_c for all c which violate the constraint.