Lecture 16. Structure and Latent SVM

Prof. Alan Yuille

Spring 2014

Outline

1. Structure SVM
2. Latent SVM

1 Structure SVM

Structure Max-Margin extends binary-classification methods so they can be applied to learn the parameters of an MRF, HMM, SCFG or other methods.

Recall standard SVM, for binary classification,

$$R(\lambda) = \frac{1}{2}||\lambda||^2 + C\sum_{i=1}^{M} \max\langle 0, 1 - y_i\lambda \cdot \phi(x_i)\rangle$$

$\{(y_i, x_i)\}$is training data, and $y_i \in \{\pm 1\}$,

e.g. to get a plane, s.t. $\phi(x) = x$.

Decision rule: $\hat{y}_i(\lambda) = \arg\max_y y\lambda \cdot \phi(x_i) = sign(\lambda \cdot \phi(x_i))$

The task is to minimize $R(\lambda)$ w.r.t λ which maximize the "margin" $\frac{1}{||\lambda||}$.

Here is a more general formulation that can be used if the output variable y is a vector $y = (y_1, \ldots, y_n)$. i.e. it could be the state of an MRF, or HMM, or a SCFG.

$$R(\lambda) = \frac{1}{2}||\lambda||^2 + c\sum_{i=1}^{M} \Delta(y_i, \hat{y}_i(\lambda))$$

decision rule: $\hat{y}_i(\lambda) = \arg\max_y \lambda \cdot \phi(x_i, y)$

the error function $\Delta(y_i, \hat{y}_i(\lambda))$ is any measure of distance between the true solution y_i and the estimate $\hat{y}_i(\lambda)$ to obtain binary value.

Binary is a special case:

(i) set $y_i \in \{-1, 1\}$

(ii) $\phi(x, y) = y\phi(x)$

(iii) $\Delta(y_i, \hat{y}_i(\lambda)) = \max\langle 0, 1 - y_i\lambda \cdot \phi(x_i)\rangle$ Hinge loss because the function is 0 if $y_i\lambda \cdot \phi(x_i) > 1$ i.e. point is on the right side of the margin and the function increases linearly with $\lambda \cdot \phi(x_i)$

(iv) $\hat{y}_i(\lambda) = \arg\max_y y\lambda \cdot \phi(x)$

This more general formulation is

$$R(\lambda) = \frac{1}{2}||\lambda||^2 + C\sum_{i=1}^{M} \Delta(y_i, \hat{y}_i(\lambda))$$

$$\hat{y}_i(\lambda) = \arg\max_y \lambda\phi(y, x_i)$$

(*)This requires an inference algorithm for binary classification inference like in the last few lectures.

(*)Also need to be able to maximized $R(\lambda)$ to find λ. It is hard because the error term $\Delta(y_i, \hat{y}_i(\lambda))$ is a highly complicated function of λ

Modify $R(\lambda)$ to an upper bound $\bar{R}(\lambda)$:

$$\bar{R}(\lambda) = \frac{1}{2}||\lambda||^2 + C\sum_{i=1}^{M} \max_{\hat{y}}\{\Delta(y_i, \hat{y}) + \lambda \cdot \phi(x_i, \hat{y}) - \lambda \cdot \phi(x_i, y_i)\}$$

which is convex in λ.

To get this bounds use two steps:

(Step 1)

$$\max_{\hat{y}} \{\Delta(y_i, \hat{y}) + \lambda \cdot \phi(x_i, \hat{y})\} \geq \Delta(y_i, \hat{y}_i(\lambda)) + \lambda \cdot \phi(x_i, \hat{y}_i(\lambda))$$

(Step 2)

$$\lambda \cdot \phi(x_i, \hat{y}_i(\lambda)) \geq \lambda \cdot \phi(x_i, y_i)$$

Note: bounds are "tight" because if we can find a good solution then $y_i \approx \hat{y}_i(\lambda)$.

How to minimize $R(\lambda)$?

Several algorithms (hot topic)

Some in dual space – like original SVM for binary problem.

Sample: Stochastic gradient descent.

Pick example (x_i, y_i), take derivative of $R(\lambda)$ w.r.t λ

$$\lambda^{t+1} = \lambda^t - \beta^t(\phi(x_i, \hat{y}^t) - \phi(x_i, y_i))$$

where $\hat{y}^t = \arg\max_{\hat{y}} \Delta(y_i, \hat{y}) + \lambda \cdot \phi(x_i, \hat{y}))$

Note: inference algorithm must be adapted to compute this.

2 Latent SVM

How to extend to module with latent (hidden) variables? Denote these variables by h with decision rule $(\hat{y}, \hat{h}) = \arg \max_{(y,h) \in \mathcal{Y}, \mathcal{H}} \lambda \cdot \phi(x, y, h)$

Training data $\langle (x_i, y_i); i = 1, \ldots, M \rangle$. The hidden variables are not known.

Loss function $\Delta(y_i, \hat{y}_i(\lambda), \hat{h}_i(\lambda))$ depends on the truth y_i, the estimate of $\hat{y}_i(\lambda), \hat{h}_i(\lambda)$ from the model

$$R(\lambda) = \frac{1}{2}||\lambda||^2 + C \sum_{i=1}^{M} \Delta(y_i; \hat{y}_i(\lambda), \hat{h}_i(\lambda))$$

nontrivial function of λ replaces $R(\lambda)$ by

$$\bar{R}(\lambda) = \frac{1}{2}||\lambda||^2 + C \sum_{i=1}^{M} \max_{(\hat{y}, \hat{h})} (\Delta(y_i; \hat{y}, \hat{h}) + \lambda \cdot \phi(x_i, \hat{y}, \hat{h})) - \max_{h} \lambda \cdot \phi(x_i, y_i, h)$$

$$f(\lambda) = \max_{(\hat{y}, \hat{h})} (\Delta(y_i; \hat{y}, \hat{h}) + \lambda \cdot \phi(x_i, \hat{y}, \hat{h}))$$

$$g(\lambda) = - \max_{h} \lambda \cdot \phi(x_i, y_i, h)$$

Here $f(\cdot)$ is convex and $g(\cdot)$ is concave.

To show convexity and concavity. Suppose

$$\tau(\lambda) = \sum_{i=1}^{M} \max_{\hat{y}_i} \lambda \cdot \phi(x_i, \hat{y}_i)$$

convex if $\tau(\alpha\lambda_1 + (1 - \alpha)\lambda_2) \leq \alpha\tau(\lambda_1) + (1 - \alpha)\tau(\lambda_2)$

$$\tau(\alpha\lambda_1 + (1 - \alpha)\lambda_2) = \alpha \sum_{i=1}^{M} \max_{\hat{y}_i} \alpha\lambda_1 + (1 - \alpha)\lambda_2), \phi(x_i, \hat{y}_i)$$

$$\alpha\tau(\lambda_1) + (1 - \alpha)\tau(\lambda_2) = \alpha \sum_{i=1}^{M} \max_{y_i} \{\lambda_1, \phi(x_i, \hat{y}_i)\rangle + (1 - \alpha) \sum_{i=1}^{M} \max_{y_i} \lambda_2\phi(x_i, \hat{y}_i)\}$$

but

$$\max_{\hat{y}_i} \alpha\lambda_1\phi(x_i, \hat{y}_i) + \max_{\hat{y}_i} \{(1 - \alpha)\lambda_2\phi(x_i, \hat{y}_i)\} \geq \max_{\hat{y}_i} \{(\alpha\lambda_1 + (1 - \alpha)\lambda_2)\phi(x_i, \hat{y}_i)\}$$

In order to solve the optimization problem we apply the CCCP algorithm.

Two steps:

Step 1:

$$\frac{\partial g(\lambda^t)}{\partial \lambda} = -\phi(x_i, y_i, h^*)$$

where $h^* = \arg\max_h \lambda^t \phi(x_i, y_i, h)$, λ^t is the current estimate of λ. This reduces to a modified SVM with known state:

$$\min_\lambda \frac{1}{2}||\lambda||^2 + C\sum_{i=1}^{M} \max_{(y,h)}\{\lambda \cdot \phi(x_i, y_i, h) + \Delta(y_i, y, h)\} - C\sum_{i=1}^{M} \lambda \cdot \phi(x_i, y_i, h_i^*)$$

Note: similarities to EM:

Step 1 involves estimating the hidden state h_i^*

Step 2 estimate λ

repeat until convergence.

Note: like EM, there is no guarantee that this will converge to the global optimum.

3 Multi-Class Multi-State SVM

$$L_p(\omega, z, \alpha) = \frac{1}{2}|\omega|^2 + c\sum_i z_i - \sum_i \sum_y \alpha_y^i (z_i - l(y, y_i) - \omega\phi(x_i, y) + \omega\phi(x_i, y_i))$$

solution: $\hat{y}(x, \omega) = \arg\max_y \omega\phi(x, y)$

constraint:

$$z_i - l(y, y_i) - \omega\phi(x, y) + \omega(x_i, y_i) \geq 0$$

$$z_i \geq \max_y\{l(y, y_i) + \omega\phi(x_i, y) - \omega\phi(x_i, y_i)\}$$

$$\frac{1}{2}|\omega|^2 + c\sum_i \max_y\{l(y, y_i) + \omega(x_i, y) - \omega(x_i, y_i)\}$$

Note: no need to separately impose $z_i \geq 0$ because if we set $y = y_i$, we see $z_i \geq l(y_i, y) = 0$

Solve the primal problem:

$$\frac{\partial L_p(\omega, z, \alpha)}{\partial \omega} = 0 \Longrightarrow \omega = \sum_i \sum_j \alpha_y^i \{\phi(x_i, y_i) - \phi(x_i, y)\}$$

note: if $\omega\phi(x_i, y_i) > \omega\phi(x_i, y) + l(y_i, y)$ then $\alpha_y^i = 0$

$$\frac{\partial}{\partial z_i} L_p(\omega, z, \alpha) = 0 \Longrightarrow c = \sum_y \alpha_y^i$$

so α_y^i is a probability distribution.

This gives solution $\omega(\alpha)$,$z(\alpha)$, substituting them gives the dual energy:

$$L_x(\alpha) = L_p(\omega(\alpha), z(\alpha), \alpha) = \sum_i \sum_y \alpha_y^i l(y, y_i) - \frac{1}{2}\sum_{i,j}\sum y, z\alpha_y^i \alpha_z^j$$

The case of binary classification can be recovered by setting

$$l(y, y_i) = \begin{cases} 1 & y \neq y_i \\ 0 & y = y_i \end{cases}$$