Lecture 15 - cutting plane

Tuesday, February 26, 2019 14:33

<u>today</u>:
- more SVM struct properties
- M^3-net dual
- cutting plane alg.

more properties of SVM struct dual:

$$w(\alpha) = \frac{1}{\lambda n} \sum_{i=1}^{n} \left(\sum_{\tilde{y} \in \mathcal{Y}_i} \alpha_i(\tilde{y})\, \psi_i(\tilde{y}) \right)$$

let $R_i \triangleq \max_{\tilde{y}} \|\psi_i(\tilde{y})\|_2$

$$\bar{R} = \frac{1}{n} \sum_{i=1}^{n} R_i$$

Then 1) $\|w^*\|_2 \overset{\Delta\text{-ineq.}}{\leq} \frac{1}{\lambda} \frac{1}{n} \sum_i \overbrace{\sum_y \alpha_i(\tilde{y})}^{=1} \underbrace{\|\psi_i(\tilde{y})\|_2}_{\leq R_i}$

$$\leq \frac{1}{\lambda} \left(\frac{1}{n} \sum_i R_i \right) = \frac{\bar{R}}{\lambda}$$

2) kernel trick: $\langle w, \varphi(x,y) \rangle = \frac{1}{\lambda n} \sum_i \sum_{\tilde{y}} \alpha_i(\tilde{y}) \underbrace{\langle \psi_i(\tilde{y}), \varphi(x,y) \rangle}_{K(x^{(i)}, y^{(i)}; x, y) - K(x^{(i)}, \tilde{y}; x, y)}$

$\psi_i(\tilde{y}) = \varphi(x^{(i)}, y^{(i)}) - \varphi(x^{(i)}, \tilde{y})$

$$\|w(\alpha)\|^2 \longrightarrow \alpha^T K \alpha$$

3) suppose scale features $\boxed{\tilde{\varphi} = b\varphi}$

3) suppose scale features $\boxed{\tilde{\psi} = b\psi}$

$$\tilde{H}_i(y; w) = \ell_i(y) - \langle \tilde{w}, \tilde{\psi}_i(y) \rangle$$

$$\tilde{w}^* = \frac{1}{\tilde{\lambda}} \frac{1}{n} \sum_i \sum_{\tilde{y}} \alpha_i^*(\tilde{y}) \underbrace{\tilde{\psi}_i(\tilde{y})}_{b\psi_i(\tilde{y})}$$

let $\boxed{\tilde{\lambda} = b^2 \lambda}$

$$\Rightarrow \tilde{w}^*(\tilde{\alpha}^*) = \frac{1}{b} \left[\frac{1}{\lambda} \frac{1}{n} \sum_{i,\tilde{y}} \tilde{\alpha}_i^*(\tilde{y}) \psi_i(\tilde{y}) \right]$$

use $\tilde{\alpha}^* = \alpha^*$ $\Rightarrow$ $\tilde{w}^* = \frac{w^*}{b}$

$$\Rightarrow \tilde{H}_i(y; \tilde{w}^*) = \ell_i(y) - \langle \frac{w^*}{b}, b\psi_i(y) \rangle = H_i(y; w^*)$$

i.e. $\tilde{\alpha}^*$ is really optimal for the new problem with $\tilde{\psi}$ & $\tilde{\lambda}$ ✓

4) similarly, can show $\tilde{\ell} = b \cdot \ell$ $\Rightarrow$ $\tilde{\lambda} = \frac{\lambda}{b}$ get same sol'n

<u>M^3-net example (dual)</u>: (getting compact dual)

$$w(\alpha) = A\alpha = \sum_i A_i \alpha_i$$

$$\alpha_i \in \Delta_{|\mathcal{Y}_i|}$$

suppose $\psi(y) = \sum_c \psi_c(y_c)$

$$\text{then } A_i \alpha_i = \sum_{\tilde{y}} \alpha_i(\tilde{y}) \psi_i(\tilde{y}) = \sum_{\tilde{y}} \alpha_i(\tilde{y}) \sum_c \psi_{i,c}(\tilde{y}_c)$$

$$= \sum_c \sum_{\tilde{y}_c} \psi_{i,c}(\tilde{y}_c) \left[\sum \alpha_i(y') \right]$$

$$= \sum_C \sum_{\tilde{y}_c} \psi_{i,c}(\tilde{y}_c) \underbrace{\left[\sum_{\substack{y' \text{ s.t.} \\ y'_c = \tilde{y}_c}} \alpha_i(y') \right]}_{\triangleq\, \mu_{i,c}(\tilde{y}_c)}$$

"marginal variable"

$\alpha_i \in \Delta_{|\mathcal{Y}_i|} \Rightarrow \mu_i \in \mathcal{M}_i$

marginal polytope

thus $A_i\alpha_i = \tilde{A}_i\mu_i$ where $(\tilde{A})_{i,(c,y_c)} = \frac{\psi_{i,c}(y_c)}{\lambda n}$

$\hookrightarrow$ # of columns is $\sum_C |\mathcal{Y}_c|$

Similarly, suppose $\ell_i(y) = \sum_C \ell_{i,c}(y_c)$

define $\tilde{b}_{i,c}(y_c) \triangleq \frac{\ell_{i,y}(y_c)}{n}$

$\langle b_i, \alpha_i \rangle = \langle \tilde{b}_i, \mu_i(\alpha_i) \rangle$

(*) thus we can replace $\max_{\alpha_i \in \Delta_{|\mathcal{Y}_i|}} -\lambda \frac{\|A\alpha\|^2}{2} + b^T\alpha$ with $\boxed{\max_{\mu_i \in \mathcal{M}_i} -\lambda \frac{\|\tilde{A}\mu\|^2}{2} + \tilde{b}^T\mu}$

$\rightarrow$ this is a tractable size QP

if $\mathcal{M}_i$ is tractable

M^3-net paper

used "structured SMO" algorithm

block-coordinate ascent using

if G_i was triangulated;

then $\mathcal{M}_i = L_i$ (local consistency polytope) used structured SMO algorithm

block-coordinate ascent using pair of variables on this QP

[similar to "pairwise FW"]

15h15

If G_i was triangulated,

then $M_i = L_i$ (local consistency polytope)

$\max_{\mu_i \in L_i}$ ______

$\rightarrow$ tractable QP?

constraint generation alg.

[Tsochantaridis & al. JMLR 2005]

want to solve $\min_{w,\xi} \frac{\lambda \|w\|^2}{2} + \frac{1}{n}\sum_{i=1}^{n} \xi_i$ (P)

n-slack version

s.t. $\xi_i \geq H_i(y_i; w) \;\; \forall y \in \mathcal{Y}_i$ } exp. # of constraints $\rightarrow$ # variables

$\xi_i \geq 0$

(D)

$\max -\frac{\lambda}{2}\|A\alpha\|^2 + b^T\alpha$

s.t. $\alpha_i \in \Delta_{|\mathcal{Y}_i|}$

Vs.

1-slack version

[ML 2009 paper]

$\min_{w,\xi} \frac{\lambda \|w\|^2}{2} + \xi$ (P)

s.t. $\xi \geq \frac{1}{n}\sum_{i=1}^{n} H_i(\tilde{y}_i; w) \;\; (\forall \tilde{y}_i \in \mathcal{Y}_i)_{i=1,\dots,n}$

$\prod_i |\mathcal{Y}_i|$ constraints

$\frac{1}{n}\left[\sum_{i=1}^{n} \ell_i(\tilde{y}_i) + \langle w, \sum_i \psi_i(\tilde{y}_i) \rangle\right]$

(D)

$\max -\frac{\lambda}{2}\|A\alpha\|^2 + \langle b, \alpha \rangle$

$\alpha \in \Delta_{|\bigtimes_{i=1}^{n} \mathcal{Y}_i|}$

$w(\alpha) = \frac{1}{\lambda n} \sum_{\tilde{y}_{1:n} \in \bigtimes_{i=1}^{n} \mathcal{Y}_i} \alpha(\tilde{y}_{1:n}) \left(\sum_i \psi_i(\tilde{y}_i)\right)$

$$\frac{1}{n}\left[\sum_{i=1}^{n} \ell_i(\tilde{y}_i) + \langle w, \underbrace{\sum_i \psi_i(\tilde{y}_i)}_{O(d)} \rangle\right]$$

$\wedge\ \tilde{y}_{\text{new}} \in \mathcal{Y}_i$

$O(d)$ instead of $O(d \cdot n)$ in n-slack formulation $\Rightarrow$ big memory saving;

n-slack SVMstruct algorithm:

iterate solving QP with more and more constraints

1) start with no constraint on $w \Rightarrow w^{(0)} = 0$, $\xi^{(0)} = 0$

2) repeat: for each i, find $\hat{y}_i = \underset{\tilde{y} \in \mathcal{Y}_i}{\text{argmax}}\ H_i(\tilde{y}; w^{(t)})$ [loss-augmented decoding]

- add $\xi_i \geq H_i(\hat{y}_i; w)$ constraint to QP (if not already there)

then resolve QP(w, ξ) with these constraints to get $w^{(t+1)}, \xi^{(t+1)}$ [e.g. using CVXopt.]

stop when primal-dual gap $\leq \varepsilon$ } note: this takes $O(n)$

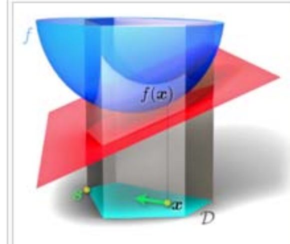
[in 2005, showed that alg. stops after $O(\frac{1}{\varepsilon^2})$ iterations]

refined later to $O(\frac{1}{\varepsilon})$ ~ at least for 1-slack version

Frank-Wolfe algorithm

↳ for smooth constrained optimization

[motivation in an context   dual of SVMstruct $\min_{\alpha_i \in \Delta_{|y_i|}} \frac{\lambda}{2}\|A\alpha\|^2 - b^T\alpha$]



A step of the Frank–Wolfe algorithm

1940s: simplex algorithm to solve LPs

1956: Marguerite Frank & Phil Wolfe

→ non-linear opt. by iterating LPs

Setup: $\min_x f(x)$ s.t. $x \in M$

- f is L-smooth i.e. ∇f is L-Lipschitz and f is convex
- M is convex and bounded set

[more generally, can also get rates for f diff. classes e.g. ∇f is Hölder cts.]

by convexity

$$f(s) \geq \underbrace{f(x_t) + \langle \nabla f(x_t), s - x_t \rangle}_{\text{linear approx. of } f \text{ at } x_t} \quad \forall s \in M$$

FW algorithm:

start with $x_0 \in M$

"linear minimization oracle" LMO

for $t = 0, \ldots$

FW corner

minimizing RHS

compute $s_t = \arg\min_{s \in M} \langle s, \nabla f(x_t) \rangle$

stopping criterion

[let $g_t \triangleq \langle s_t - x_t, -\nabla f(x_t) \rangle$ FW gap   if $g_t < \varepsilon$, output x_t]

$x_{t+1} = (1 - \gamma_t) x_t + \gamma_t s_t$ step-size $\gamma_t \in [0,1]$ (convex combo)

$= x_t + \gamma_t (\underbrace{s_t - x_t}_{d_t})$

end

output x_{t+1}

end
output x_{t+1}

step-size choice: $\gamma_t = \begin{cases} \text{universal choice } \frac{2}{t+2} \\ \text{line search } \gamma_t = \arg\min_{\gamma \in [0,1]} f(x_t + \gamma(s_t - x_t)) \\ \text{adaptive: } \frac{g_t}{L \|s_t - x_t\|^2} \text{ or } \frac{g_t}{C_f} \text{ truncated at 1} \end{cases}$

C_f affine invariant constant

(*) big motivation for FW
is LMO is often much cheaper than projections
and cheap for many sets M appearing in ML

properties: 1) $f(x_t) - \underbrace{\min_{x \in M} f(x)}_{\triangleq f^*} \leq O\left(\frac{1}{t}\right)$

2) FW-gap $g_t \geq f(x_t) - f^*$ → certificate of suboptimality

$\min_{s \leq t} g_s \leq O\left(\frac{1}{t}\right)$ [ie. we will stop in $O(\frac{1}{\varepsilon})$ iterations?]

3) $x_t = \rho_0^t x_0 + \sum_{u=1}^{t} \rho_u^t s_{u-1}$ ⇝ x_t has "sparse" expansion in terms of the FW-corners $\{s_u\}_{u=1}^{t-1}$

where $\sum_{u=0}^{t} \rho_u^t = 1$ $\rho_u^t \geq 0$

(*)(*) "sparse method" → popular in ML

[we'll see can run FW on SVMstruct dual assuming can compute LMO]

(which is loss-augmented decoding here)

4) FW is affine co-variant (like Newton's method)

5) there is a $\Omega\left(\frac{1}{t}\right)$ lower bound for FW-like methods for $t \ll d$