Lecture 20 - variance reduction

Thursday, March 21, 2019 13:27

today: - variance reduction perspective
- application CRF

Variance reduction idea

X & Y be R.V.

goal: estimate $\mathbb{E}X$ using M.C. samples

suppose: $\mathbb{E}Y$ is cheap to compute and Y is correlated with X

consider estimator $\Theta_\alpha \triangleq \alpha(X-Y) + \mathbb{E}Y$ to approximate $\mathbb{E}X$
$\alpha \in [0,1]$

properties: $\mathbb{E}\Theta_\alpha = \alpha\mathbb{E}X + (1-\alpha)\mathbb{E}Y$ $\rightsquigarrow$ unbiased (i.e. $\mathbb{E}\Theta_\alpha = \mathbb{E}X$)
if $\mathbb{E}Y = \mathbb{E}X$ (not interesting)
$\alpha = 1$

variance: $\mathrm{Var}(\Theta_\alpha) = \alpha^2[\mathrm{Var}(X) + \mathrm{Var}(Y) - 2\,\mathrm{cov}(X,Y)]$
variance reduction

for $\alpha = 1$ (unbiased setting) $\Theta_\alpha = X + (\mathbb{E}Y - Y)$
correction

SGD setting:

SGD setting :

X is $\nabla f_i(x_t)$; $\mathbb{E}X$ = batch gradient

SAG/SAGA algorithms : Y is g_i [past stored gradient]

$$\mathbb{E}Y = \frac{1}{n}\sum_i g_i$$

SAG algorithm : $\alpha = \frac{1}{n}$ (biased)

SAGA " : $\alpha = 1$ (unbiased)

α X Y $\mathbb{E}Y$

SAG: $x_{t+1} = x_t - \gamma\left[\frac{1}{n}\nabla f_{i_t}(x_t) - g_{i_t}^t + \frac{1}{n}\sum_j g_j^t\right]$ (biased)

SAGA: $x_{t+1} = x_t - \gamma\left[\nabla f_{i_t}(x_t) - g_{i_t}^t + \frac{1}{n}\sum_j g_j^t\right]$ (unbiased)

($\alpha = 1$)

SVRG: $x_{t+1} = x_t - \gamma\left[\nabla f_{i_t}(x_t) - \nabla f_{i_t}(\tilde{x}_{old}) + \frac{1}{n}\sum_{j=1}^{n}\nabla f_j(\tilde{x}_{old})\right]$ (unbiased)

(stochastic variance reduced gradient)

$\tilde{x}_{old}$ is updated from outer loop

SVRG algorithm :

SVRG algorithm:

for $k=0, \ldots$ (outer loop)

compute $g_{ref} \triangleq \frac{1}{n} \sum_j \nabla f_j(x^{(k)})$

for $t=0, \ldots, T_{max}$

sample i_t

$$x_{t+1}^{(k)} = x_t^{(k)} - \gamma \left[\nabla f_{i_t}(x_t^{(k)}) - \nabla f_{i_t}(x^{(k)}) + g_{ref} \right]$$

end

$x^{(k+1)} = x^{(k)}$

end

questions:

- what is T_{max}?
- what is γ?

original SVRG convergence result: need $\gamma \leq \frac{1}{L}$

"$T_{max} \geq \tilde{\kappa} \frac{L}{\mu} = k$" → to run alg., need to know k;

⇒ not adaptive & local strong convexity

tweak on SVRG:
[Hoffmann & al. NIPS 2015]

$T_{max} \sim \text{Geom}(\_)$

[at inner loop iteration, do batch gradient comp.]

· with prob. $\frac{1}{n}$ ·

then, get same convergence result as SAGA

↳ here, size of inner loop $\mathbb{E}[i_{max}] = n$

overall cost of SVRG $\approx$ 3×(SGD for each n updates)

SAG / SAGA / SVRG (Johnson & Hofmann), rate for convex fct. ($\mu = 0$), get $\min_t \mathbb{E}[f(x_t) - f^*] = O(\frac{1}{t})$

[contrast with $\frac{1}{\sqrt{t}}$ for SGD]

[14h30]

CRF objective:

$\max_{\tilde{y}} l_i(\tilde{y}) - w^T \psi_i(\tilde{y})$

	primal	dual
SVM struct	$\min_w \frac{\lambda \|w\|^2}{2} + \frac{1}{n}\sum_{i=1}^n H_i(w)$ (hinge)	$\max_{\alpha_i \in \Delta(|Y_i|)} -\frac{\lambda \|w(\alpha)\|^2}{2} + \frac{1}{n}\sum_{i=1}^n \alpha_i^T a_i$
CRF:	$\min_w \frac{\lambda \|w\|^2}{2} + \frac{1}{n}\sum_i -\log p(y^{(i)} \mid x^{(i)}; w)$	$\max_{\alpha \in \Delta(|Y_i|)} -\frac{\lambda \|w(\alpha)\|^2}{2} + \frac{1}{n}\sum_{i=1}^n H_i(\alpha_i)$ (entropy)

$\log\left(\sum_{\tilde{y}} \exp(-w^T \psi_i(\tilde{y}))\right)$

$\triangleq -\sum_{\tilde{y}} \alpha_i(\tilde{y}) \log \alpha_i(\tilde{y})$

KKT $\rightarrow$ $w(\alpha) = \frac{1}{\lambda n}\sum_i \sum_{\tilde{y}} \alpha_i(\tilde{y}) \psi_i(\tilde{y})$

$= \frac{1}{\lambda n}\sum_i \sum_{c \in C} \mu_{i,c}(\tilde{y}_c) \psi_{i,c}(\tilde{y}_c)$

(*) at optimality

$\alpha_i^*(y) = p(y \mid x^{(i)}; w(\alpha^*))$

$= \frac{1}{\lambda n} \sum_i \sum_{c \in C_i} \mu_{i,c}(\tilde{y}_c)\, \psi_{i,c}(\tilde{y}_c)$

$p(y \mid x; w) \propto \exp(\langle w, \phi(x,y) \rangle)$

$\alpha_i(y) \to p(y \mid x_i; w(\alpha))$

$\alpha_i^* \in$ interior of $\Delta_{|\mathcal{Y}_i|}$

unlike sparse solution in structured SVM

CRF optimization

- primal objective is smooth & strongly convex [vs non-smooth for SVMstruct]
  - for a while, batch L-BFGS was method of choice [batch $\Rightarrow$ slow for large n]
  - [Collins et al. JMLR 2008]: online exponentiated gradient (OEG)

block-coordinate method on dual; exponentiated gradient step on block

dual fct.

$$\alpha_i(\tilde{y})^{(t+1)} \propto \alpha_i(\tilde{y})^{(t)} \exp\left(-\gamma_t \nabla_{\alpha_i} D(\alpha^{(t)})\right)$$

EG alg $\to$ proximal gradient step using $KL(\alpha \,\|\, \alpha^{(t)})$ as Bregman divergence

$\to$ get linear convergence rate with cheap $O(1)$ updates (like SGD) [vs $O(n)$ for batch method]

[can think of it as variance reduced SGD as well]

SAGA for CRF: $w^{(t+1)} = (1 - \lambda\gamma_t) w^{(t)} - \gamma_t \left[\nabla f_i(w^{(t)}) - g_i^{(t)} + \frac{1}{n} \sum_j g_j^{(t)}\right]$

[Schmidt et al.

[Schmidt & al. AISTATS 2015]

SDCA
stochastic dual coord. ascent.

$$\alpha_{i_t}(\tilde{y})^{(t+1)} = (1-\gamma_t)\alpha_i(\tilde{y})^{(t)} + \gamma_t \underbrace{\tilde{S}_i(\tilde{y})^{(t)}}_{p(\tilde{y} \mid x_i; w(\alpha^{(t)}))}$$

$\gamma_t \in [0,1]$ (stabilize)

[max-marginal inference]

[note: BCFW is a special case of SDCA on SVM-struct obj.]

as a relaxed fixed point update for $\alpha_i^{*} = p(y \mid x_i; w(\alpha^{*}))$

SOA for CRF Lacoste-Julien & al. [UAI 2015]

Proximal gradient method

↳ generalization of projected gradient method to other non-smooth functions

composite framework: $F(w) \triangleq f(w) + \Omega(w)$ where f is convex & L-smooth

- constrained opt. $\Omega(w) = \delta_{\mathcal{M}}(w) \triangleq \begin{cases} 0 & \text{if } w \in \mathcal{M} \\ +\infty & \text{o.w.} \end{cases}$ ("indicator of $\mathcal{M}$")

  Ω is convex but not nec. smooth

- ℓ_1-regularization $\Omega(w) = \|w\|_1$

proximal gradient update:

$$w_{t+1} = \operatorname{argmin}_w \; f(w_t) + \langle \nabla f(w_t), w - w_t \rangle + \frac{1}{\eta} \|w - w_t\|^2 + \Omega(w)$$

$$w_{t+1} = \operatorname*{argmin}_w \underbrace{f(w_t) + \langle \nabla f(w_t), w - w_t \rangle + \frac{1}{2\gamma_t}\|w - w_t\|^2}_{\triangleq B_t(w)} + \Omega(w)$$

if $\gamma_t \leq \frac{1}{L}$ then $f(w) \leq B_t(w)$ $\forall w$

we can rewrite $B_t(w) = \frac{1}{2\gamma_t}\|w - [w_t - \gamma_t \nabla f(w_t)]\|^2 + \text{cst.}$ (by completing the square)

$\Rightarrow$ if $\Omega(w) = \delta_{\mathcal{W}}(w)$, we get the projected gradient alg.

$$w_{t+1} = \operatorname{prox}^{\Omega}_{\gamma_t}(w_t - \gamma_t \nabla f(w_t))$$

$\hookrightarrow$ "proximal operator" $\operatorname{prox}^{\Omega}_{\gamma}(z) \triangleq \operatorname*{argmin}_w \{\Omega(w) + \frac{1}{2\gamma}\|w - z\|^2\}$

could replace with Bregman divergence to get then generalization (e.g. OE6)

like projection, prox operator is non-expansive (ie. 1-Lipschitz)

ie. $\|\operatorname{prox}^{\Omega}_{\gamma}(w) - \operatorname{prox}^{\Omega}_{\gamma}(w')\|_2 \leq \|w - w'\|_2$

$\Rightarrow$ convergence rate of prox. gradient method are same as unconstrained gradient descent