MOHAMED BIN ZAYED
UNIVERSITY OF
ARTIFICIAL INTELLIGENCE

How to use second-order information to speed up optimization methods

Dr. Dmitry Kamzolov
Research Associate

Problem formulation

High-order convex problem

$$\min_{x \in \mathbb{E}} f(x),$$

$f(x)$ is convex function with Lipschitz-continuous gradient and Hessian with constants L_1, L_2

Problem formulation

High-order convex problem

$$\min_{x \in \mathbb{E}} f(x),$$

$f(x)$ is convex function with Lipschitz-continuous gradient and Hessian with constants L_1, L_2

Lipschitz derivative

$$\|\nabla f(x) - \nabla f(y)\|_* \leq L_1 \|x - y\|$$

$$\|\nabla^2 f(x) - \nabla^2 f(y)\|_{op} \leq L_2 \|x - y\|_2$$

Problem formulation

High-order convex problem

$$\min_{x \in \mathbb{E}} f(x),$$

$f(x)$ is convex function with Lipschitz-continuous gradient and Hessian with constants L_1, L_2

Lipschitz derivative

$$\|\nabla f(x) - \nabla f(y)\|_* \leq L_1 \|x - y\|$$

$$\|\nabla^2 f(x) - \nabla^2 f(y)\|_{op} \leq L_2 \|x - y\|_2$$

Lipschitz derivative

$$\|\nabla^2 f(x)\|_{op} \leq L_1$$

$$\|\nabla^3 f(x)\|_{op} \leq L_2$$

Model

Taylor approximation

$$\Omega_1(f,x;y) = f(x) + \langle \nabla f(x), y-x \rangle, y \in E$$

$$\Omega_2(f,x;y) = f(x) + \langle \nabla f(x), y-x \rangle + \frac{1}{2}\left\langle \nabla^2 f(x)(y-x), y-x \right\rangle, y \in E$$

Model

Taylor approximation

$$\Omega_1(f, x; y) = f(x) + \langle \nabla f(x), y - x \rangle, y \in E$$

$$\Omega_2(f, x; y) = f(x) + \langle \nabla f(x), y - x \rangle + \frac{1}{2} \left\langle \nabla^2 f(x)(y - x), y - x \right\rangle, y \in E$$

Upper and lower bound

$$|f(y) - \Omega_p(f, x; y)| \leq \frac{L_p}{(p+1)!} \|y - x\|^{p+1}$$

Model

Taylor approximation

$$\Omega_1(f, x; y) = f(x) + \langle \nabla f(x), y - x \rangle, y \in E$$

$$\Omega_2(f, x; y) = f(x) + \langle \nabla f(x), y - x \rangle + \frac{1}{2} \left\langle \nabla^2 f(x)(y - x), y - x \right\rangle, y \in E$$

Upper and lower bound

$$|f(y) - \Omega_p(f, x; y)| \leq \frac{L_p}{(p+1)!} \|y - x\|^{p+1}$$

Corrolary

$$f(y) \leq \Omega_p(f, x; y) + \frac{L_p}{(p+1)!} \|y - x\|^{p+1}$$

First-order model

Gradient Method

$$x_{k+1} = x_k + \operatorname*{argmin}_{h \in \mathbb{E}} \left\{ f(x_k) + \langle \nabla f(x_k), h \rangle + \frac{L_1}{2} \|h\|^2 \right\}$$

First-order model

Gradient Method

$$x_{k+1} = x_k + \operatorname*{argmin}_{h \in \mathbb{E}} \left\{ f(x_k) + \langle \nabla f(x_k), h \rangle + \frac{L_1}{2} \|h\|^2 \right\}$$

Step

$$\nabla f(x_k) + L_1 h = 0 \qquad h = -\tfrac{1}{L_1} \nabla f(x_k)$$

First-order model

Gradient Method

$$x_{k+1} = x_k + \operatorname*{argmin}_{h \in \mathbb{E}} \left\{ f(x_k) + \langle \nabla f(x_k), h \rangle + \frac{L_1}{2} \|h\|^2 \right\}$$

Step

$$\nabla f(x_k) + L_1 h = 0 \qquad h = -\tfrac{1}{L_1} \nabla f(x_k)$$

Explicit form

$$x_{k+1} = x_k - \tfrac{1}{L_1} \nabla f(x_k)$$

First-order model

Gradient Method

$$x_{k+1} = x_k + \operatorname*{argmin}_{h \in \mathbb{E}} \left\{ f(x_k) + \langle \nabla f(x_k), h \rangle + \frac{L_1}{2} \|h\|^2 \right\}$$

Step

$$\nabla f(x_k) + L_1 h = 0 \qquad h = -\tfrac{1}{L_1} \nabla f(x_k)$$

Explicit form

$$x_{k+1} = x_k - \tfrac{1}{L_1} \nabla f(x_k)$$

Convergence

$$f(x_k) - f(x^*) \le O\left(\frac{L_1 R^2}{k}\right) \qquad R = \|x_0 - x^*\|$$

Second-order Method

Second-order model

$$x_{k+1} = x_k + \operatorname*{argmin}_{h \in \mathbb{E}} \left\{ f(x_k) + \langle \nabla f(x_k), h \rangle + \frac{1}{2} \left\langle \nabla^2 f(x_k) h, h \right\rangle \right\}$$

Second-order Method

Second-order model

$$x_{k+1} = x_k + \operatorname*{argmin}_{h \in \mathbb{E}} \left\{ f(x_k) + \langle \nabla f(x_k), h \rangle + \frac{1}{2} \left\langle \nabla^2 f(x_k) h, h \right\rangle \right\}$$

Step

$$\nabla f(x_k) + \nabla^2 f(x_k) h = 0 \qquad h = -\left[\nabla^2 f(x_k)\right]^{-1} \nabla f(x_k)$$

Second-order Method

Second-order model

$$x_{k+1} = x_k + \operatorname*{argmin}_{h \in \mathbb{E}} \left\{ f(x_k) + \langle \nabla f(x_k), h \rangle + \frac{1}{2} \langle \nabla^2 f(x_k) h, h \rangle \right\}$$

Step

$$\nabla f(x_k) + \nabla^2 f(x_k) h = 0 \qquad h = -\left[\nabla^2 f(x_k)\right]^{-1} \nabla f(x_k)$$

Newton Method [1948, L. Kantorovich]

$$x_{k+1} = x_k - \left[\nabla^2 f(x_k)\right]^{-1} \nabla f(x_k)$$

Second-order Method

Second-order model

$$x_{k+1} = x_k + \operatorname*{argmin}_{h \in \mathbb{E}} \left\{ f(x_k) + \langle \nabla f(x_k), h \rangle + \frac{1}{2} \langle \nabla^2 f(x_k) h, h \rangle \right\}$$

Step

$$\nabla f(x_k) + \nabla^2 f(x_k) h = 0 \qquad h = -\left[\nabla^2 f(x_k)\right]^{-1} \nabla f(x_k)$$

Newton Method [1948, L. Kantorovich]

$$x_{k+1} = x_k - \left[\nabla^2 f(x_k)\right]^{-1} \nabla f(x_k)$$

Damped Newton Method

$$x_{k+1} = x_k - h_k \left[\nabla^2 f(x_k)\right]^{-1} \nabla f(x_k)$$

Cubic Regularized Newton Method

Cubic Regularization [2006, Yu. Nesterov and Boris T Polyak]

$$x_{k+1} = x_k + \operatorname*{argmin}_{h \in \mathbb{E}} \left\{ f(x_k) + \langle \nabla f(x_k), h \rangle + \frac{1}{2} \left\langle \nabla^2 f(x_k) h, h \right\rangle + \frac{L_2}{6} \|h\|^3 \right\}$$

Cubic Regularized Newton Method

Cubic Regularization [2006, Yu. Nesterov and Boris T Polyak]

$$x_{k+1} = x_k + \operatorname*{argmin}_{h \in \mathbb{E}} \left\{ f(x_k) + \langle \nabla f(x_k), h \rangle + \frac{1}{2} \left\langle \nabla^2 f(x_k) h, h \right\rangle + \frac{L_2}{6} \|h\|^3 \right\}$$

Cubic Regularized Newton Method

Cubic Regularization [2006, Yu. Nesterov and Boris T Polyak]

$$x_{k+1} = x_k + \operatorname*{argmin}_{h \in \mathbb{E}} \left\{ f(x_k) + \langle \nabla f(x_k), h \rangle + \frac{1}{2} \left\langle \nabla^2 f(x_k) h, h \right\rangle + \frac{L_2}{6} \|h\|^3 \right\}$$

Step

$$\nabla f(x_k) + \nabla^2 f(x_k) h + \frac{L_2}{2} \|h\| h = 0 \qquad h = -\left[\nabla^2 f(x_k) + \frac{L_2}{2} \|h\| I \right]^{-1} \nabla f(x_k)$$

Cubic Regularized Newton Method

Cubic Regularization [2006, Yu. Nesterov and Boris T Polyak]

$$x_{k+1} = x_k + \operatorname*{argmin}_{h \in \mathbb{E}} \left\{ f(x_k) + \langle \nabla f(x_k), h \rangle + \frac{1}{2} \left\langle \nabla^2 f(x_k) h, h \right\rangle + \frac{L_2}{6} \|h\|^3 \right\}$$

Step

$$\nabla f(x_k) + \nabla^2 f(x_k) h + \frac{L_2}{2} \|h\| h = 0 \qquad h = -\left[\nabla^2 f(x_k) + \frac{L_2}{2} \|h\| I \right]^{-1} \nabla f(x_k)$$

Convergence

$$f(x_k) - f(x^*) \leq O\left(\frac{L_2 R^3}{k^2} \right)$$

Inexact Second-order Methods

Inexact Newton Method

$$x_{k+1} = x_k - h_k B_k^{-1} g_k$$

Inexact Second-order Methods

Inexact Newton Method

$$x_{k+1} = x_k - h_k B_k^{-1} g_k$$

Inexact Cubic Regularization [2017, S. Ghadimi et.al.], [2020, A. Agafonov]

$$x_{k+1} = x_k + \operatorname*{argmin}_{h \in \mathbb{E}} \left\{ f(x_k) + \langle g_k, h \rangle + \frac{1}{2} \langle B_k h, h \rangle + \frac{L_2}{6} \|h\|^3 \right\}$$

Inexact Second-order Methods

Inexact Newton Method

$$x_{k+1} = x_k - h_k B_k^{-1} g_k$$

Inexact Cubic Regularization [2017, S. Ghadimi et.al.], [2020, A. Agafonov]

$$x_{k+1} = x_k + \operatorname*{argmin}_{h \in \mathbb{E}} \left\{ f(x_k) + \langle g_k, h \rangle + \frac{1}{2} \langle B_k h, h \rangle + \frac{L_2}{6} \|h\|^3 \right\}$$

Inexactness

$$\|B_k - \nabla^2 f(x_k)\|_{op} \leq \delta$$

Inexact Second-order Methods

Inexact Newton Method

$$x_{k+1} = x_k - h_k B_k^{-1} g_k$$

Inexact Cubic Regularization [2017, S. Ghadimi et.al.], [2020, A. Agafonov]

$$x_{k+1} = x_k + \operatorname*{argmin}_{h \in \mathbb{E}} \left\{ f(x_k) + \langle g_k, h \rangle + \frac{1}{2} \langle B_k h, h \rangle + \frac{L_2}{6} \|h\|^3 \right\}$$

Inexactness

$$\|B_k - \nabla^2 f(x_k)\|_{op} \leq \delta$$

Convergence

$$f(x_k) - f(x^*) \leq O\left(\frac{\delta R^2}{k}\right) + O\left(\frac{L_2 R^3}{k^2}\right)$$

Examples of Inexactness

Gradient Method

$$B_k = L_1 I$$

Examples of Inexactness

Gradient Method

$$B_k = L_1 I$$

Diagonal Preconditioning

$$B_k = \textbf{diagonal}(\nabla^2 f(x_k))$$

Examples of Inexactness

Gradient Method

$$B_k = L_1 I$$

Diagonal Preconditioning

$$B_k = \textbf{diagonal}(\nabla^2 f(x_k))$$

AdaGrad

$$B_k = \textbf{diag}\left(\sqrt{g_k \odot g_k}\right)$$

Examples of Inexactness

Gradient Method

$$B_k = L_1 I$$

Diagonal Preconditioning

$$B_k = \textbf{diagonal}(\nabla^2 f(x_k))$$

AdaGrad

$$B_k = \textbf{diag}\left(\sqrt{g_k \odot g_k}\right)$$

RMSProp

$$B_k = \sqrt{\beta_2 B_{k-1}^2 + (1-\beta_2)\textbf{diag}\left(g_k \odot g_k\right)}$$

Examples of Inexactness

Diagonal Preconditioning

$$B_k = \textbf{diagonal}(\nabla^2 f(x_k))$$

Examples of Inexactness

Diagonal Preconditioning

$$B_k = \textbf{diagonal}(\nabla^2 f(x_k))$$

OASIS [2021, M. Jahani et.al.]

$$B_k = \beta D_{t-1} + (1-\beta)\textbf{diag}\left(z_k \odot \nabla^2 f(x_k) z_k\right),$$

where z_k is a random vector with Rademacher distribution.

Examples of Inexactness

Diagonal Preconditioning

$$B_k = \textbf{diagonal}(\nabla^2 f(x_k))$$

OASIS [2021, M. Jahani et.al.]

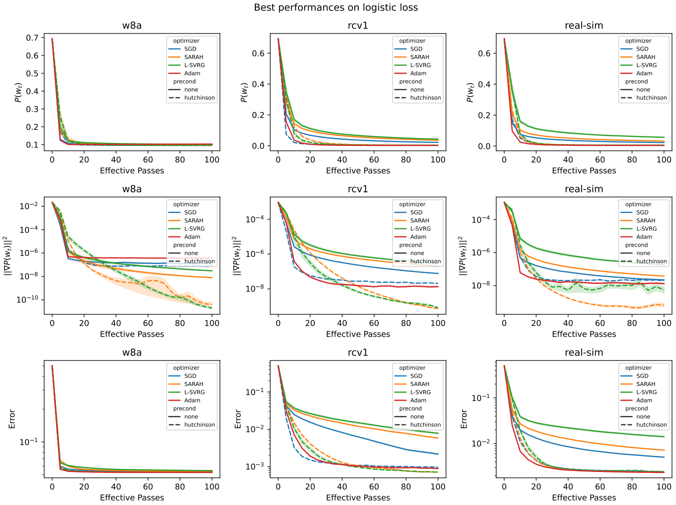
$$B_k = \beta D_{t-1} + (1-\beta)\textbf{diag}\left(z_k \odot \nabla^2 f(x_k) z_k\right),$$

where z_k is a random vector with Rademacher distribution.

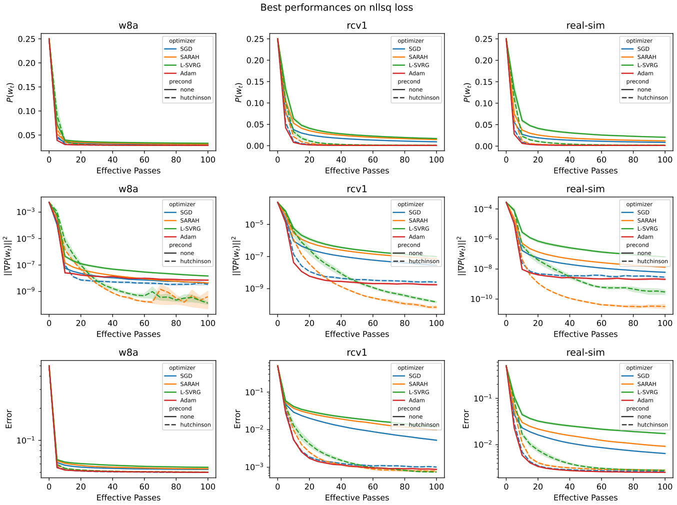
Scaled SARAH + Sclaed L-SVRG [2022, A. Sadiev et.al.]

$$B_k = \beta D_{t-1} + (1-\beta)\textbf{diag}\left(z_k \odot \nabla^2 f(x_k) z_k\right),$$

Scaled SARAH + Sclaed L-SVRG



Scaled SARAH + Sclaed L-SVRG



Examples of Inexactness

L-BFGS

$$B_{k+1} = B_k + \frac{y_k y_k^T}{y_k^T s_k} - \frac{B_k s_k s_k^T B_k^T}{s_k^T B_k s_k}$$

Examples of Inexactness

L-BFGS

$$B_{k+1} = B_k + \frac{y_k y_k^T}{y_k^T s_k} - \frac{B_k s_k s_k^T B_k^T}{s_k^T B_k s_k}$$

L-BFGS

$$H_{k+1} = \left(I - \frac{y_k s_k^T}{y_k^T s_k}\right)^T H_k \left(I - \frac{y_k s_k^T}{y_k^T s_k}\right) + \frac{s_k s_k^T}{y_k^T s_k},$$

where $H_k = B_k^{-1}$

Higher-Order Method

Basic step [Nesterov, 2018]

$$T_{H_p}(x) = \operatorname*{argmin}_y \left\{ \tilde{\Omega}_{p,H_p}(f, x; y) \right\},$$

where

$$\tilde{\Omega}_{p,H_p}(f, x; y) = \Omega_p(f, x; y) + \frac{H_p}{p!} \|y - x\|^{p+1}.$$

For $H_p \geq L_p$ this subproblem is convex and hence implementable.

Higher-Order Method

Basic step [Nesterov, 2018]

$$T_{H_p}(x) = \operatorname*{argmin}_y \left\{ \tilde{\Omega}_{p,H_p}(f, x; y) \right\},$$

where

$$\tilde{\Omega}_{p,H_p}(f, x; y) = \Omega_p(f, x; y) + \frac{H_p}{p!} \|y - x\|^{p+1}.$$

For $H_p \geq L_p$ this subproblem is convex and hence implementable.

Convergence

$$f(x_N) - f(x_*) \leq O\left(\frac{H_p R^{p+1}}{N^p} \right)$$

Superfast Second-Order Method
Nesterov 2018/ Nesterov 2020

1: Choose $x_0 \in E$ and define $A_k = O(k^{p+1})$

2: Define $\psi_0(x) = \frac{1}{p+1}\|x - x_0\|^{p+1}$

3: **for** $k = 1, \ldots, N$ **do**

4: Compute $v_k = \underset{x \in E}{\operatorname{argmin}} \psi_k(x)$ and $y_k = \frac{A_k}{A_{k+1}} x_k + \frac{a_{k+1}}{A_{k+1}} v_k$, where

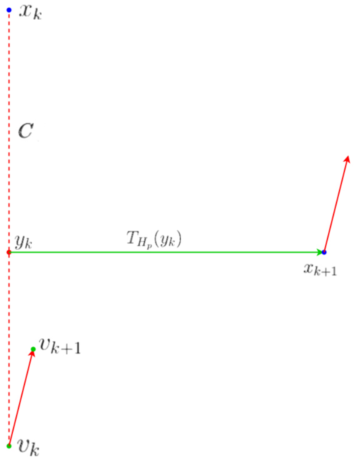
$$a_{k+1} = A_{k+1} - A_k$$

5: Compute $x_{k+1} = T_{H_p}(y_k)$ and update

$$\psi_{k+1} = \psi_k(x) + a_{k+1}[f(x_{k+1}) + \langle \nabla f(x_{k+1}), x - x_{k+1} \rangle]$$

6: **end for**

Superfast Second-Order Method
Nesterov 2018/ Nesterov 2020



Superfast Second-Order Method
Nesterov 2018/ Nesterov 2020

Convergence rate

$$f(x_N) - f(x_*) \le O\left(\frac{H_p R^{p+1}}{N^{p+1}}\right)$$

Hyperfast Second-Order Method
Gasnikov, Bubeck et.al. 2019/ Kamzolov, Gasnikov 2020

Define $A_0 = 0, x_0 = y_0 = 0$

for $k = 0, \dots, N-1$ **do**

Compute a pair $\lambda_{k+1} > 0$ and $y_{k+1} \in \mathbb{R}^d$ such that

$$\frac{1}{2} \le \lambda_{k+1} \frac{L_p \cdot \|x_{k+1} - y_k\|^{p-1}}{(p-1)!} \le \frac{p}{p+1},$$

where

$$x_{k+1} = T_{L_p}(y_k)$$

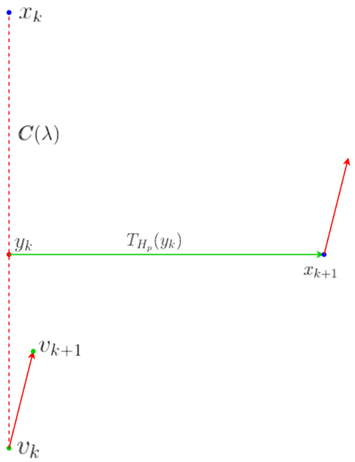
and

$$a_{k+1} = \frac{\lambda_{k+1} + \sqrt{\lambda_{k+1}^2 + 4\lambda_{k+1} A_k}}{2}, y_k = \frac{A_k}{A_{k+1}} x_k + \frac{a_{k+1}}{A_{k+1}} v_k,$$

$$A_{k+1} = A_k + a_{k+1}$$

Update $v_{k+1} = v_k - a_{k+1} \nabla f(x_{k+1})$

Hyperfast Second-Order Method
Gasnikov, Bubeck et.al. 2019/ Kamzolov, Gasnikov 2020



Hyperfast Second-Order Method
Gasnikov, Bubeck et.al. 2019/ Kamzolov, Gasnikov 2020

Convergence rate

$$f(x_N) - f(x_*) \leq \tilde{O}\left(\frac{H_p R^{p+1}}{N^{\frac{3p+1}{2}}}\right)$$

where $\tilde{O}(\cdot)$ means accuracy up to a logarithmic factor

Theoretical Line-search complexity

$$k \leq 30p\log p + \log\left(\frac{H_p\|x^*\|^{p+1}}{\varepsilon}\right)$$

Proximal-Point Method With Segment Search
Nesterov 2020

Set $v_0 = x_0 \in \mathbb{E}$, $H_p > 0$ Đ, $A_0 = 0$

for $k = 0, \dots, k = N-1$ **do**

Compute $x_k^0 \in T_{H_p}(x_k)$. **If** $\langle \nabla f(x_k^0), u_k \rangle \geq 0$, **then** $g_k = \nabla f(x_k^0)$,

Else, $x_k^1 \in T_{H_p}(v_k)$. **If** $\langle \nabla f(x_k^1), u_k \rangle \leq 0$, **then** $g_k = \nabla f(x_k^1)$

Else, find $0 \leq \tau_k^1 \leq \tau_k^2 \leq 1$, $T_k^1 \in T_{H_p}(x_k + \tau_k^1 u_k)$ Đ, $T_k^2 \in T_{H_p}(x_k + \tau_k^2 u_k)$ such that:

$$\textbf{a)}\alpha_k \leq 0 \leq \beta_k \quad \textbf{b)}\gamma_k \alpha_k (\tau_k^1 - \tau_k^2) \leq h(T_k^2),$$

where $\alpha_k = \langle \nabla f(T_k^1), u_k \rangle$, $\beta_k = \langle \nabla f(T_k^2), u_k \rangle$, $\gamma_k = \frac{\beta_k}{\beta_k - \alpha_k}$,

$h(x)$ - special function for step-size computation.

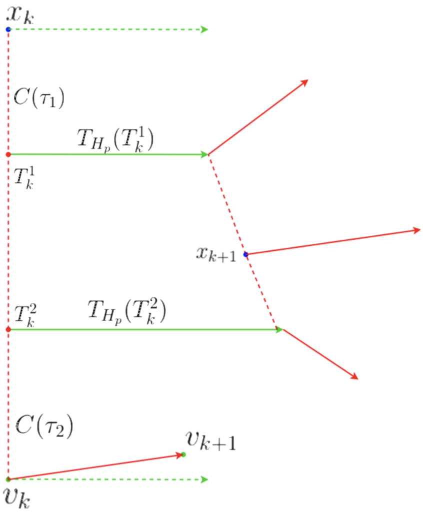
Compute $g_k = \gamma_k \nabla f(T_k^1) + (1-\gamma_k)\nabla f(T_k^2)$ Đ, $x_{k+1} = \gamma_k T_k^1 + (1-\gamma_k) T_k^2$.

Compute $a_{k+1} > 0$ from $\frac{a_{k+1}^2}{A_k + a_{k+1}} = h(x_{k+1})$ and $A_{k+1} = A_k + a_{k+1}$

Compute $v_{k+1} = v_k - a_{k+1} g_k$ and $u_{k+1} = v_{k+1} - x_{k+1}$.

end for

Proximal-Point Method With Segment Search
Nesterov 2020



Proximal-Point Method With Segment Search

Convergence rate

$$f(x_N) - f(x_*) \leq \tilde{O}\left(\frac{H_p R^{p+1}}{N^{\frac{3p+1}{2}}}\right)$$

Theoretical Line-search complexity

$$k \leq 2 + \frac{1}{p} \log\left(\frac{3H_p \|x^*\|^{p+1}}{2\varepsilon}\right)$$

Problem

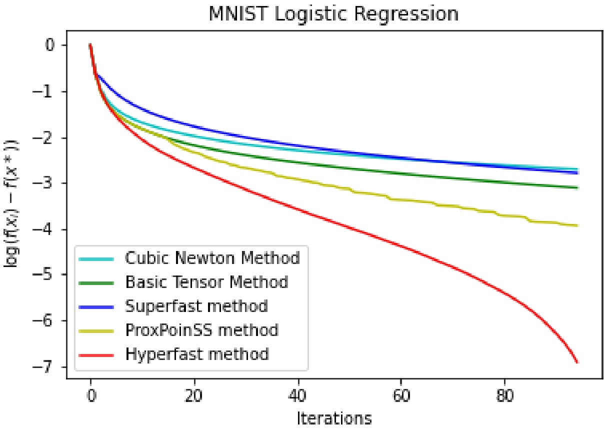
$$f(x) = \frac{1}{M}\sum_{i=1}^{M} \ell(x, \zeta_i) = \frac{1}{M}\sum_{i=1}^{M} \log\left(1 + \exp(-\eta_i x^\top \xi_i)\right),$$

where $M = 5000$, $d = 784$ for MNIST

Implementation Time

Method	Time	Time/iteration
Cubic Newton Method	1073 sec.	10.73 sec./iter.
Basic Tensor Method	1079 sec.	10.79 sec./iter.
Superfast Method	1102 sec.	11.02 sec./iter.
ProxPointSS Method	1746 sec.	17.46 sec./iter.
Hyperfast Method	1579 sec.	15.79 sec./iter.

Experiments



Experiments

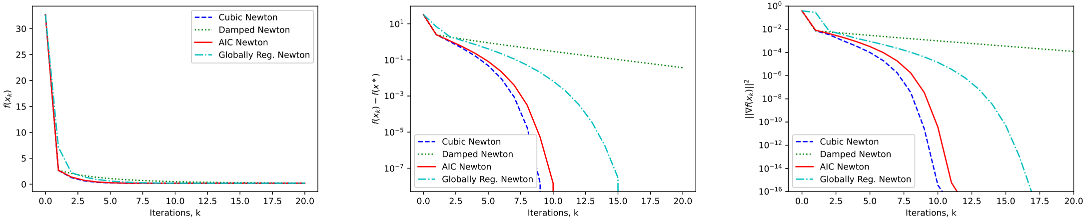


Figure: Comparison of regularized Newton methods and Damped Newton method for logistic regression task on MNIST dataset (10 models for i vs. other digits problems with argmax aggregation).

Contacts

Code is available

https://github.com/OPTAMI/OPTAMI

My contacts

kamzolov.dmitry@mbzuai.ac.ae

MOHAMED BIN ZAYED
UNIVERSITY OF
ARTIFICIAL INTELLIGENCE

Mohamed bin Zayed
University of Artificial Intelligence
Masdar City
Abu Dhabi
United Arab Emirates