Cubic Regularization technical report

Ioannis Tsingalis
Department of Informatics
Aristotle University

September 30, 2021

I. Notations and Definitions

Where needed, the dependence on t is omitted for simplicity. The spectrum of a symmetric $d \times d$ Hessian matrix $\mathbf{H}$ is denoted by $\lambda(\mathbf{H}) = \{\lambda_i(\mathbf{H})\}_{i=1}^{d}$. Suppose that the eigenvalues are numbered in decreasing order, i.e.,

$$\lambda_1(\mathbf{H}) \geq \lambda_2(\mathbf{H}) \geq \cdots \geq \lambda_d(\mathbf{H}). \tag{1}$$

If $\mathbf{H}$ is indefinite, i.e.,

$$\lambda_d(\mathbf{H}) < 0 \quad \text{and} \quad \lambda_i(\mathbf{H}) > 0, \tag{2}$$

where $i < d$, then $f(\mathbf{x})$ is non-convex.

II. Main Algorithm

In this section, the Cubic Regularization (CR) technique is described. Let

$$T_M(x) \triangleq \arg\min_y \left[\langle g_x, y - x \rangle + \frac{1}{2}\langle y - x, H_x(y - x)\rangle + \frac{M}{6}\|y - x\|_2^3 \right], \tag{3}$$

where $g_x = \nabla f(x)$ and $H_x = \nabla^2 f(x)$. For a specific time point t we have $g_{x_t} = \nabla_x f(x_t)$ and $H_{x_t} = \nabla^2 f(x_t)$, and $x_{t+1} = T_M(x_t)$.

Algorithm 1 Cubic Regularization of Newton Method

Input: $x_0 \in \mathbb{R}^n$, L_0

Define $T_M : x \to T_M(x)$
Initialize $t \leftarrow 0$, $M_0 \leftarrow L_0$
do
  while $f\left(T_{M_t}(x_t)\right) > f(x_t)$ **do**
    $M_t = 2M_t$
  end while
  $M_{t+1} = M_t$
  $x_{t+1} = T_{M_t}(x_t)$
while not $\mu_{M_k}(x_{t+1}) < \epsilon$ [a]

a.

$$\mu_M(x) = \max\left\{ \sqrt{\frac{2}{L+M}\|\nabla_x f(x)\|}, -\frac{2}{2L+M}\lambda_d(\nabla_x^2 f(x)) \right\}$$

III. Solving $T_M(x)$

In this section, the procedure to obtain the solution of (3) is described. Subscripts x and t are removed for notational simplicity. Setting $h = y - x$, (3) can be written as

$$\min_{h \in \mathbb{R}^n} \underbrace{\left[g^T h + \frac{1}{2} h^T H h + \frac{M}{6}\|h\|_2^3 \right]}_{m_M(h)}. \tag{4}$$

Even though (4) is non-convex, it is shown in [2, Section 5.1] that the minimizer of (4) is a global minimizer. This follows from the equivalence of

$$\max_{r \in \mathcal{D}} \underbrace{\left[-\frac{1}{2}\langle g, \left(H + \frac{Mr}{2} \right)^{-1} g \rangle - \frac{M}{12} r^3 \right]}_{u(r)}, \tag{5}$$

with (4), where $\mathcal{D} = \left\{r \in \mathbb{R} \mid H + \frac{Mr}{2} I \succ 0, r \geq 0\right\}$. Assume that r^* is the minimizer of (5). Then the minimizer of (4) is $h^* = h(r^*)$. r^* is obtained by solving $u'(\lambda) = 0$ which is equivalent to

$$\psi(r) = \|h(r)\| - r = 0, \tag{6}$$

with

$$h(r) = -\left(H + \frac{Mr}{2} I\right)^{-1} g, \quad r \in \mathcal{D}. \tag{7}$$

Setting $\lambda = \frac{M}{2} r$, (6) and (7) can be written as

$$\psi(\lambda) = \|h(\lambda)\| - \frac{2\lambda}{M} = 0 \tag{8}$$

and

$$h(\lambda) = -\left(H + \lambda I\right)^{-1} g = -H(\lambda)^{-1} g, \tag{9}$$

respectively, with $H(\lambda) = H + \lambda I$. Thus, to obtain r^*, the non-linear equation in (8) has to be solved. In [1, Section 7], it is shown that solving (8) directly has drawbacks and the solution of

$$\phi(\lambda) = \frac{1}{\|h(\lambda)\|} - \frac{M}{2\lambda} = 0 \tag{10}$$

is preferred.

Lemma III.1. *Suppose $g \neq 0$. Then, the function $\phi(\lambda)$ is strictly increasing, when $\lambda > \lambda_n$ and concave. Its first derivative is*

$$\phi'(\lambda) = -\frac{\langle h(\lambda), \partial_\lambda h(\lambda) \rangle}{\|h(\lambda)\|_3^2} + \frac{M}{2\lambda^2}, \tag{11}$$

where

$$\partial_\lambda h(\lambda) = -H(\lambda)^{-1} h(\lambda). \tag{12}$$

Given the factorization $H(\lambda) = LL^T$, the numerator of $\phi'(\lambda)$ can be written as $\langle h(\lambda), \partial_\lambda h(\lambda) \rangle = -\|w\|^2$, where w is the solution of $Lw = h(\lambda)$. ◇

Proof. A procedure similar to that in [1, Lemma 7.3.1] can be followed to get the results. □

To solve (10), a variety of methods can be applied [1, Section 7]. For example, when the eigensolution is cheap [1, Algorithm 7.3.6] can be applied. Algorithm 2 is [1, Algorithm 7.3.6] where $\phi(\lambda)$ and $\phi'(\lambda)$ are defined in (10) and (11), respectively.

Algorithm 2 Subproblem solution of Cubic Regularization of Newton Method

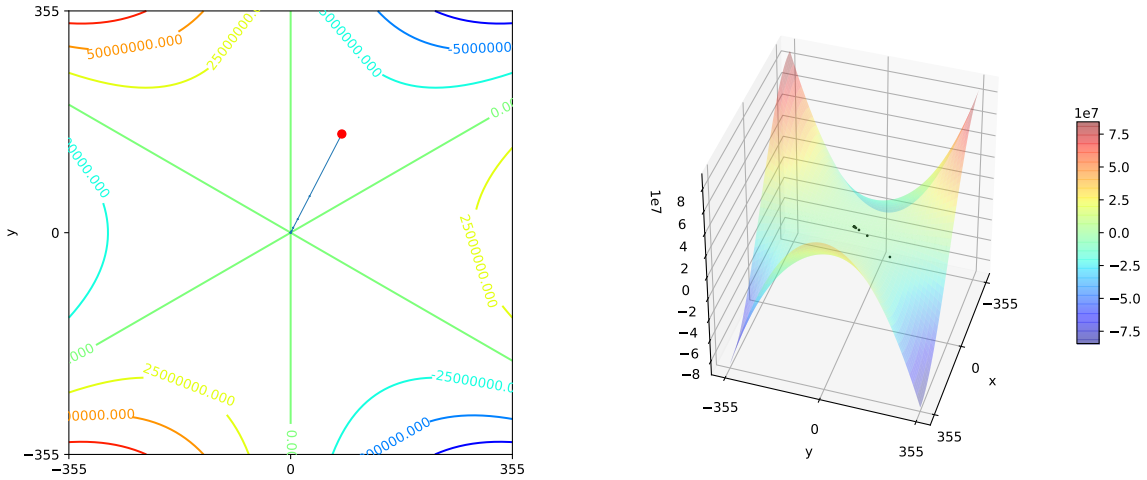
Input: H, g, M

Output: s

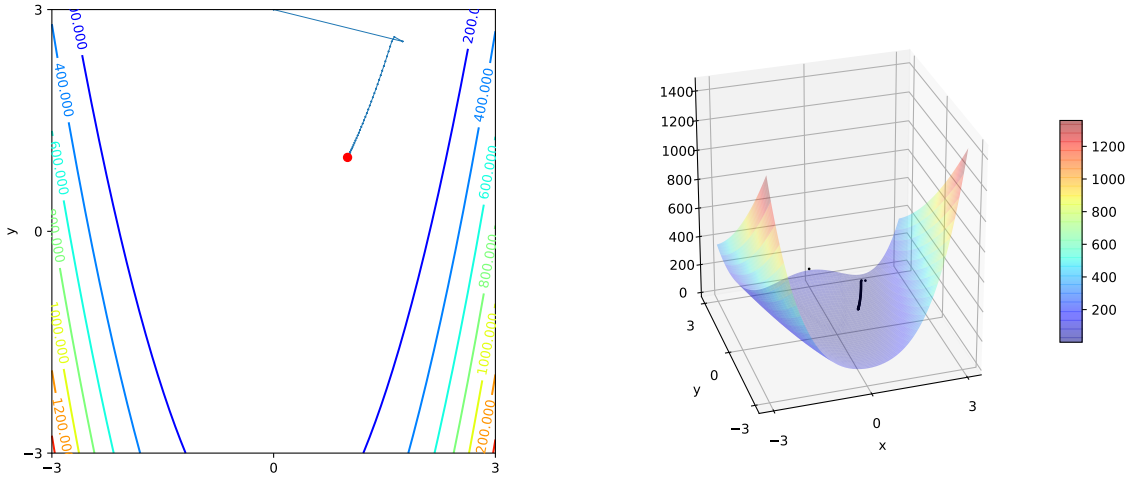
Initialize $\kappa_{\text{easy}} \in (0, 1)$
repeat
  if $H \succ 0$ **then**
    $\lambda \leftarrow 0$
  else
    $\lambda \leftarrow -\lambda_d^+$ ▷ λ_d^+ is barely smaller that λ_d
  end if
  Factorize $H(\lambda) = LL^T$
  Solve $LL^T h(\lambda) = -g$
  if $\|h(\lambda)\|_2 \leq \frac{2\lambda}{M}$ **then**
    if λ is 0 or $\|h(\lambda)\|_2 = \frac{2\lambda}{M}$ **then**
      `return` $h(\lambda)$
    else
      Solve $(A - \lambda_1)u_1 = 0$ w.r.t. u_1 ▷ Get the corresponding eigenvector.
      Solve $\|h(\lambda) + \alpha u_1\|_2 = \frac{2\lambda}{M}$ w.r.t. α
      $h(\lambda) \leftarrow h(\lambda) + \alpha u_1$
      `return` $h(\lambda)$
    end if
  else
    while $\left|\|h(\lambda)\|_2 - \frac{2\lambda}{M}\right| < \kappa_{\text{easy}} \frac{2\lambda}{M}$ **do**
      Solve $Lw = h(\lambda)$
      $\lambda \leftarrow \lambda - \frac{\phi(\lambda)}{\phi'(\lambda)}$ ▷ $\phi(\lambda)$ and $\phi'(\lambda)$ are defined in (10) and (11), respectively
    end while
    `return` $h(\lambda)$
  end if

IV. Experiments

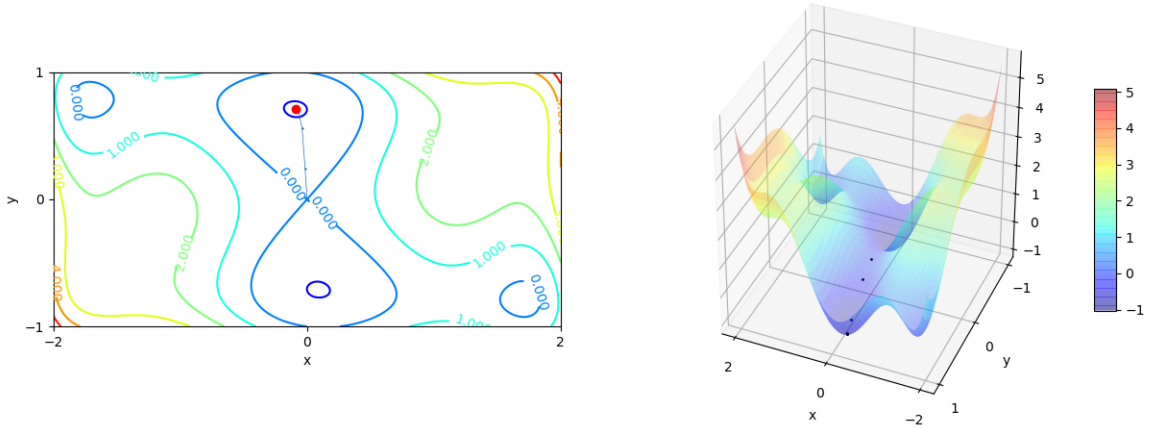
A. Toy Examples



(a) Monkey function: $f(x, y) = x^3 - 3xy^2$.



(b) Rosenbrock function: $f(x, y) = (\alpha - x)^2 + \beta(y - x^2)^2$ with $\alpha = 1$, $\beta = 10$.



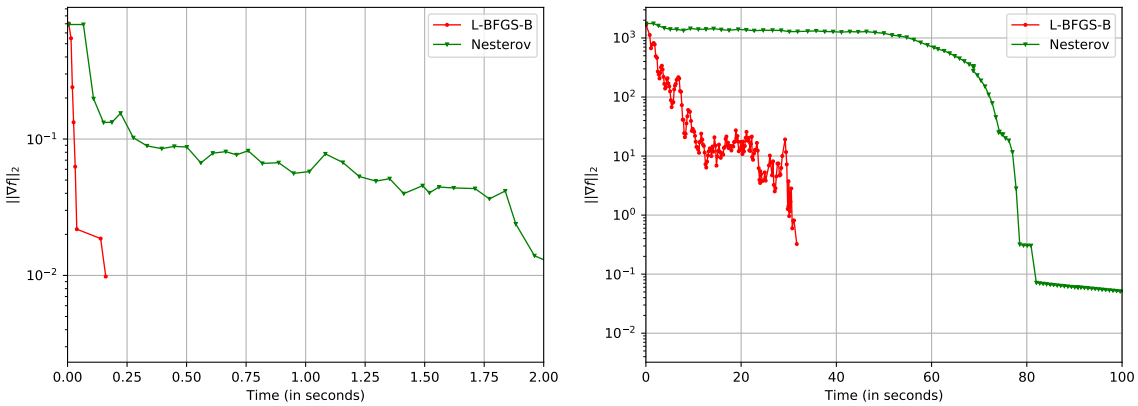
(c) Six-hump camel function: $\left(4 - 2.1x^2 + \frac{x^4}{3}\right) x^2 + xy + (4y^2 - 4)y^2$.

Fig. 1: Application of Cubic Regularization in different functions.

B. Real Data

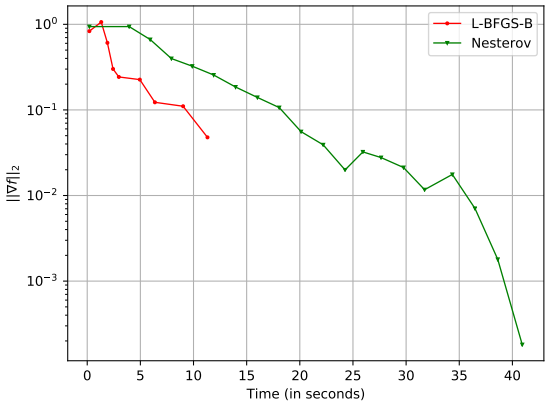
In this section, CR is applied to binary Logistic Regression with a non-convex regularizer, i.e.,

$$\min_{\mathbf{w}\in\mathbb{R}^d} -\left[\frac{1}{N}\sum_{i=1}^{N} y_i \ \log\left(\frac{1}{1+e^{-w^T x}}\right) + (1-y_i)\ \log\ \left(\frac{e^{w^T x}}{1+e^{-w^T x}}\right)\right] + \alpha\sum_{j=1}^{d}\frac{w_i^2}{1+w_i^2}. \tag{13}$$



(a) a9a

(b) ijcnn1



(c) MNIST

Fig. 2: Average $\|\nabla f\|_2$ for non-convex logistic regression ($\alpha = 0.1$) on a9a (N=32.561, d=123), covtype (N= 58.1012, d=54), and MNIST (N= 60.000 training / 10.000 testing, d=784) after 20 executions.

Methods	Datasets	
	a9a	MNIST
L-BFGS-B	78.71	84.59
CR	77.32	84.73

TABLE I: Average classification accuracy for non-convex logistic regression ($\alpha = 0.1$) on a9a (N=32.561, d=123), and MNIST (N= 60.000 training / 10.000 testing, d=784) after 20 executions.

REFERENCES

[1] A. R Conn, N. IM Gould, and P. L Toint. *Trust region methods*. SIAM, 2000.

[2] Y. Nesterov and B. T Polyak. Cubic regularization of newton method and its global performance. *Mathematical Programming*, 108(1):177–205, 2006.