A Damped Newton Method Achieves Global $\mathcal{O}\left(\frac{1}{k^2}\right)$ and Local Quadratic Convergence Rate

Anonymous Author(s)
Affiliation
Address
email

Abstract

In this paper, we present the first stepsize schedule for Newton method resulting in fast global and local convergence guarantees. In particular, a) we prove an $\mathcal{O}\left(\frac{1}{k^2}\right)$ global rate, which matches the state-of-the-art global rate of cubically regularized Newton method of Polyak and Nesterov (2006) and of regularized Newton method of Mishchenko (2021) and Doikov and Nesterov (2021), b) we prove a local quadratic rate, which matches the best-known local rate of second-order methods, and c) our stepsize formula is simple, explicit, and does not require solving any subproblem. Our convergence proofs hold under affine-invariant assumptions closely related to the notion of self-concordance. Finally, our method has competitive performance when compared to existing baselines, which share the same fast global convergence guarantees.

1 Introduction

Second-order methods are the backbone of much of scientific computing. They have been used for centuries (Kantorovich [1948], Moré [1978], Ypma [1995]). The number of extensions and applications is enormous; only the survey of trust regions and quasi-Newton methods from two decades ago [Conn et al., 2000] cited over a thousand papers. Research of second-order methods has been flourishing up to this day [Nesterov et al., 2018].

Machine Learning is very closely connected to scientific computing. Machine Learning is mostly built upon first-order optimizers due to their scalability properties (cheap iterates, parallelizable). Therefore, first order methods attracted a lot of attention, researchers have been obtaining strong results in many related areas: variance reduction (Roux et al. [2012],Gower et al. [2020]), preconditioning, acceleration (Nesterov [2013],d'Aspremont et al. [2021]) distributed/federated computation (Konečný et al. [2016], Kairouz et al. [2021]), decentralized computation [Koloskova et al., 2020],. . . However, the convergence of first-order methods always depends on the conditioning of the underlying problem. Improving conditioning is fundamentally impossible without using higher-order information.

Removing conditioning dependence is possible by incorporating information about the Hessian. This leads to second-order methods. Their most compelling advantage is that they can converge extremely quickly, usually in just a few iterations.

1.1 Newton's method: benefits and limitations

For instance, one of the most famous and classical algorithms in optimization, Newton's method, takes iterates of form

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

Its iterates satisfy the recursion $\|\nabla f(x_{k+1})\| \leq c \|\nabla f(x_k)\|^2$ (for a constant $c > 0$), which means that Newton's method converges locally quadratically. However, convergence of Newton method is

Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

limited to only to the neighborhood of the solution. It is well-known that when initialized far from optimum, Newton can diverge, both in theory and practice (Jarre and Toint [2016], Mascarenhas [2007]). We can explain intuition why this happens. Update rule of Newton (1) was chosen to minimize right hand side of Taylor approximation

$$f(x) \approx f(y) + \langle \nabla f(y), x - y \rangle + \frac{1}{2} \left\langle \nabla^2 f(y)(x-y), x-y \right\rangle. \tag{2}$$

The main problem is that Taylor approximation is not an upper bound, and therefore, global convergence of Newton method is not guaranteed.

1.2 Towards a fast globally convergent Newton's method

Even though superlinear local convergence rates are very common, global convergence guarantees of any form are surprisingly rare. Many papers proposed globalization strategies, essentially all of them require some combination of the following: line-search, trust regions, damping/truncation, regularization, or solving a subproblem in each iteration. Some popular globalization strategies prevent an increase in functional value. However, this turned out to be insufficient for convergence to the optimum. Jarre and Toint [2016], Mascarenhas [2007] designed simple functions (strictly convex with compact level sets) so that Newton method with Armijo stepsizes does not converge to the optimum. For clarity, we will use the term *convergence guarantees* to talk about convergence to the exact optimum (excluding methods that can get stuck in cycles) on convex functions with smooth gradients and Hessians.

To this day, virtually all known global convergence guarantees are for regularized Newton methods with, which can be written as

$$x_{k+1} = x_k - \alpha_k \left(\nabla^2 f(x_k) + \lambda_k \mathbf{I} \right)^{-1} \nabla f(x_k), \tag{3}$$

where $\lambda_k \geq 0$. Parameter λ_k is also known as Levenberg-Marquardt regularization [Moré, 1978], which was first introduced for a nonlinear least-squares objective. For simplicity, we disregard differences in the objectives for the literature comparison. Motivation behind update (3) is to replace Taylor approximation in (2) by an upper bound. In particular, cubic Newton method [Nesterov and Polyak, 2006] assumes that function f has Lipschitz-continuous Hessian,

$$\|\nabla^2 f(x) - \nabla^2 f(y)\|_2 \leq L_2 \|x - y\|_2 \tag{4}$$

to upper bound Taylor approximation eq. (2) as

$$f(x) \leq f(y) + \langle \nabla f(y), x - y \rangle + \frac{1}{2} \left\langle \nabla^2 f(y)(x-y), x-y \right\rangle + \frac{L_2}{6} \|x - y\|_2^3. \tag{5}$$

Next iterate of cubic Newton can be written as a minimizer of right hand side of (5)[1]

$$x_{k+1} = \operatorname*{argmin}_{y \in \mathbb{E}} \left\{ f(x_k) + \langle \nabla f(x_k), y - x_k \rangle + \frac{1}{2} \left\langle \nabla^2 f(x_k)(y - x_k), y - x_k \right\rangle + \frac{L_2}{6} \|y - x_k\|_2^3 \right\}. \tag{6}$$

This is almost identical to the step of our newly-proposed algorithm *AIC Newton* (Algorithm 1)[1]

$$x_{k+1} = \operatorname*{argmin}_{y \in \mathbb{E}} \left\{ f(x_k) + \langle \nabla f(x_k), y - x_k \rangle + \frac{1}{2} \left\langle \nabla^2 f(x_k)(y - x_k), y - x_k \right\rangle + \frac{M_x}{6} \|y - x_k\|_{x_k}^3 \right\}. \tag{7}$$

The difference between the update of Cubic Newton and *AIC Newton* is that we measure the cubic regularization term in the local Hessian norms. This seemingly negligible perturbation turned out to be of a great significance for two reasons **a)** model in (7) is affine-invariant, **b)** surprisingly, the next iterate of (7) lies in the direction of Newton method step and is obtainable without regularizer λ_k (*AIC Newton* just needs to set stepsize α_k). We elaborate on both of these points later in the paper.

Cubic Newton method (6) can be equivalently expressed in form (3) with $\alpha_k = 1, \lambda_k = L_2 \|x_k - x_{k+1}\|$. This algorithm is a first algorithm with proven global convergence rate $\mathcal{O}\left(k^{-2}\right)$. However, since such λ_k depends on x_{k+1}, resulting algorithm requires solving implicit problem each iteration. Next work showing convergence rate of regularized Newton method [Polyak, 2009] avoided implicit steps by choosing $\lambda_k \propto \|\nabla f(x_k)\|$. However, this came with a tradeoff for slower convergence rate, $\mathcal{O}\left(k^{-1/4}\right)$. Finally, Mishchenko [4/2021] and Doikov and Nesterov [12/2021] improved upon both of these works by using explicit regularization $\alpha_k = 1$, $\lambda_k \propto \sqrt{L_H \|\nabla f(x_k)\|}$, and proving rate $\mathcal{O}\left(k^{-2}\right)$.

[1] Where $\mathbb{E}$ a d-dimensional Euclidean space, defined in Section 2.1.

Table 1: A summary of regularized Newton methods with global convergence guarantees. We consider algorithms with updates of form $x_{k+1} = x_k - \alpha_k \left(\nabla^2 f(x_k) + \lambda_k \mathbf{I}\right)^{-1} \nabla f(x_k)$. For simplicity of comparison, we disregard differences in objectives and assumptions. We assume Hessian smoothness, convexity, local strong convexity and bounded level sets. For regularization parameter holds $\lambda_k \geq 0$ and stepsize satisfy $0 < \alpha_k \leq 1$.

Algorithm	**Regularizer** $\lambda_k \propto$	**Stepsize** $\alpha_k =$	**Affine**[(1)] **invariant? (alg., ass., rate)**	**Avoids line search?**	**Global convergence rate**	**Local**[(2)] **convergence exponent**	**Reference**
Newton	0	1	(✓, ✗, ✗)	✓	✗	2	Kantorovich [1948]
Newton	0	1	(✓, ✓, ✓)	✓	✗	2	Nesterov and Nemirovskii [1994]
Damped Newton B	0	$\frac{1}{1+G_1}$ [(4)]	(✓,✓,✓)	✓	$\mathcal{O}\left(k^{-\frac{1}{2}}\right)$	**2**	Nesterov et al. [2018, (5.1.28)]
Damped Newton C	0	$\frac{1+G_1}{1+G_1+G_1^2}$ [(4)]	(✓,✓,✓)	✓	✗	**2**	Nesterov et al. [2018, $(5.2.1)_C$]
Cubic Newton	$L_2\|x_{k+1} - x_k\|_2$	1	(✗, ✗, ✗)	✗	$\mathcal{O}\left(\mathbf{k^{-2}}\right)$	**2**	Nesterov and Polyak [2006], Griewank [1981], Doikov and Nesterov [2021]
Locally Reg. Newton	$\|\nabla f(x_k)\|_2$	1	(✗, ✗, ✗)	✓	✗	**2**	Polyak [2009]
Globally Reg. Newton	$\|\nabla f(x_k)\|_2$	$\frac{\mu+\|\nabla f(x_k)\|_2}{L_1}$ [(3)]	(✗, ✗, ✗)	✗	$\mathcal{O}\left(k^{-\frac{1}{4}}\right)$	**2**	Polyak [2009]
Globally Reg. Newton	$\sqrt{L_2\|\nabla f(x_k)\|_2}$	1	(✗, ✗, ✗)	✓	$\mathcal{O}\left(\mathbf{k^{-2}}\right)$	$\frac{3}{2}$	Mishchenko [4/2021] Doikov and Nesterov [12/2021]
AIC Newton (Algorithm 1)	0	$\frac{-1+\sqrt{1+2G}}{G}$ [(4)]	(✓, ✓, ✓)	✓	$\mathcal{O}\left(\mathbf{k^{-2}}\right)$	**2**	**This work**

[(1)] In triplets, we report whether algorithm, used assumptions, convergence rate are affine-invariant, respectively.
[(2)] For a Lyapunov function Φ^k and a constant c, we report exponent β of $\Phi(x_{k+1}) \leq c\Phi(x_k)^\beta$.
[(3)] f is a μ - strongly convex with L_1-Lipshitz continuous gradient.
[(4)] For simplicity, we denote $G \stackrel{\text{def}}{=} L_2\|\nabla f(x_k)\|^*_{x_k}$ and $G_1 \stackrel{\text{def}}{=} M_f\|\nabla f(x_k)\|^*_{x_k}$.

1.3 *AIC Newton* as a damped Newton method

In this work, we investigate global convergence for most basic globalization strategy, stepsized Newton method without any regularizer ($\lambda_k = 0$). This algorithm is also referred as damped (or truncated) Newton method; it can be written as

$$x_{k+1} = x_k - \alpha_k \nabla^2 f(x_k)^{-1} \nabla f(x_k).$$

Resulting algorithm was investigated in detail as an interior-point method. Nesterov et al. [2018] shows quadratic local convergence for stepsizes $\alpha_1 \stackrel{\text{def}}{=} \frac{1}{1+G_1}, \alpha_2 \stackrel{\text{def}}{=} \frac{1+G_1}{1+G_1+G_1^2}$, where[2] $G_1 \stackrel{\text{def}}{=} M_f\|\nabla f(x_k)\|^*_{x^k}$. Our algorithm *AIC Newton* is also damped Newton method with similar stepsize, $\alpha = \frac{-1+\sqrt{1+2G}}{G}$, where[2] $G \stackrel{\text{def}}{=} L_H\|\nabla f(x_k)\|^*_{x_k}$. Mentioned stepsizes $\alpha_1, \alpha_2, \alpha$ share two characteristics. Firstly, all of them depends on gradient computed in the dual norm and scaled by a smoothness constant (G_1 or G). Secondly, all of these stepsizes converge to 1 from below (for $\hat{\alpha} \in \{\alpha_1, \alpha_2, \alpha\}$ holds $0 < \hat{\alpha} \leq 1$ and $\lim_{x \to x_*} \hat{\alpha} = 1$). We show comparison of stepizes of *AIC Newton* with damped Newton methods [Nesterov et al., 2018] in Figure 3. Our algorithm uses stepsize bigger by orders of magnitude (see Appendix A).

The main difference between already established stepsizes α_1, α_2 and our stepsize α are resulting global convergence rates. While stepsize α_2 does not lead to a global convergence rate , and α_1 leads to rate $\mathcal{O}\left(k^{-1/2}\right)$, our stepsize α leads to a significantly faster, $\mathcal{O}\left(k^{-2}\right)$ rate. Our rate matches best known global rates for regularized Newton methods. We manage to achieve these results by carefully choosing assumptions. While rates for α_1 and α_2 follows from standard self-concordance , our assumptions are a consequence of a slightly stronger version of self-concordance. We will discuss this difference in detail later.

We summarize important properties of regularized Newton methods with fast global convergence guarantees and damped Newton methods in Table 1 .

[2]For $\|\nabla f(x_k)\|^*_{x_k} = \nabla f(x_k)^T \nabla^2 f(x^k)^{-1} \nabla f(x_k)$, as it is defined in Section 2.1

2 Contributions

To summarize novelty in our work, we present a novel algorithm *AIC Newton*. Our algorithm can be interpreted in two viewpoints **a)** as a regularized Newton method (version of cubic Newton method), **b)** as a damped Newton method. *AIC Newton* enjoys the best properties of these two worlds:

• **Fast global convergence**: *AIC Newton* converges globally with rate $\mathcal{O}\left(k^{-2}\right)$, which matches state-of-the-art global rate for all regularized Newton methods. Furthermore, it is the first such rate for Damped Newton method.

• **Fast local convergence:** In addition to the fast global rate, *AIC Newton* decreases gradient norms locally in quadratic rate. This result matches the best-known rates for both regularized Newton algorithms and damped Newton algorithms.

• **Implementability:** Formula for *AIC Newton* stepsize is simple. It is both explicit and directly computable (without the need to run a subroutine or solving a subproblem).

• **Simplicity:** Previous works on Newton regularizations can be viewed as a popular global-convergence fix for the Newton method. We propose an even simpler fix in the form of a stepsize schedule.

• **Practical performance:** We show that in practice, *AIC Newton* outperforms all algorithms sharing same convergence guarantees: Cubic Newton [Nesterov and Polyak, 2006] and Globally Regularized Newton [Mishchenko, 4/2021] and Doikov and Nesterov [12/2021], and fixed stepsize Damped Newton method.

• **Geometric properties:** We analyze *AIC Newton* under more geometrically natural assumptions than what is commonly used for the regularized Newton methods. We use an affine-invariant version of Hessian smoothness, closely related to self-concordance. Our algorithm preserves affine-invariance obtained from assumptions throughout the convergence.

Rest of the paper is structured as following. In Section 2.1 we introduce our notation. In Section 3, we discuss algorithm *AIC Newton*, affine-invariant properties and self-concordance. In Sections 4, 5 we show global and local convergence guarantees (resp.). In Section 6 we present an empirical comparison of *AIC Newton* with other algorithms sharing fast global convergence.

2.1 Minimization problem & notation

In the paper, we consider a d-dimensional Euclidean space $\mathbb{E}$. It's dual space, $\mathbb{E}^*$, is composed of all linear functionals on $\mathbb{E}$. For a functional $g \in \mathbb{E}^*$, we denote by $\langle g, x \rangle$ its value at $x \in \mathbb{E}$.

We consider the following convex optimization problem:

$$\min_{x \in \mathbb{E}} f(x), \tag{8}$$

where $f(x) \in C^2$ is a convex function with continuous first and second derivatives and positive definite hessian. We assume that the problem has a unique minimizer $x_* \in \operatorname*{argmin}_{x \in \mathbb{E}} f(x)$. Note, that

$$\nabla f(x) \in \mathbb{E}^*, \quad \nabla^2 f(x) h \in \mathbb{E}^*$$

Now, we introduce different norms for spaces $\mathbb{E}$ and $\mathbb{E}^*$. Denote $x \in \mathbb{E}, g, h \in \mathbb{E}^*$. For a self-adjoint positive-definite operator $\mathbf{H} : \mathbb{E} \to \mathbb{E}^*$, we can endow these spaces with conjugate Euclidean norms:

$$\|x\|_{\mathbf{H}} \stackrel{\text{def}}{=} \langle \mathbf{H}x, x \rangle^{1/2}, \; x \in \mathbb{E}, \qquad \|g\|^*_{\mathbf{H}} \stackrel{\text{def}}{=} \left\langle g, \mathbf{H}^{-1} g \right\rangle^{1/2}, \; g \in \mathbb{E}^*.$$

For identity $\mathbf{H} = \mathbf{I}$, we get classical 2-norm $\|x\|_2 = \|x\|_{\mathbf{I}} = \langle x, x \rangle^{1/2}$. For local Hessian norm $\mathbf{H} = \nabla^2 f(x)$, we use shortened notation

$$\|h\|_x \stackrel{\text{def}}{=} \left\langle \nabla^2 f(x) h, h \right\rangle^{1/2}, \; h \in \mathbb{E}, \qquad \|g\|^*_x \stackrel{\text{def}}{=} \left\langle g, \nabla^2 f(x)^{-1} g \right\rangle^{1/2}, \; g \in \mathbb{E}^*. \tag{9}$$

Specific case $\mathbb{E} \leftarrow \mathbb{R}^d$: In this particular scenario H is a symmetric positive definite matrix.

3 New Algorithm: Affine-Invariant Cubic Newton

Finally, we are ready to present algorithm *AIC Newton*. It is damped Newton method with updates

$$x_{k+1} = x_k - \alpha_k \nabla^2 f(x_k)^{-1} \nabla f(x_k), \tag{10}$$

with stepsize $\alpha_k \stackrel{\text{def}}{=} \frac{-1+\sqrt{1+2L_H \|\nabla f(x_k)\|^*_{x_k}}}{L_H \|\nabla f(x_k)\|^*_{x_k}}$, as summarized in Algorithm 1. Stepsize satisfy $\alpha_k \leq 1$ (from AG inequality, see Technical lemmas in Appendix B). Also $\lim_{x_k \to x_*} \alpha_k = 1$, hence (10) converges to Newton method. Next, we are going to discuss geometric properties of our algorithm.

Algorithm 1 *AIC Newton*: Affine-Invariant Cubic Newton

1: **Requires:** Initial point x^0, constant L_H
2: **for** $k = 0, 1, 2 \dots$ **do**
3: $\quad \alpha_k = \frac{-1+\sqrt{1+2L_f \|\nabla f(x_k)\|_{x_k}}}{L_f \|\nabla f(x_k)\|_{x_k}}$
4: $\quad x_{k+1} = x_k - \alpha_k \left[\nabla^2 f(x_k)\right]^{-1} \nabla f(x_k)$ // Note that $x_{k+1} \stackrel{(17)}{=} S(x_k)$.
5: **end for**

3.1 Geometric properties: affine invariance

One of the main geometric properties of the Newton method is *affine invariance*, invariance to affine transformations of variables. Let $A \in \mathbb{R}^{d \times d}$ be a non-degenerate linear transformation. Consider function $\phi(y) = f(Ay)$. By affine transformation, we denote $f(x) \to \phi(y) = f(Ay), x \to A^{-1}y$.

Significance of norms: Note that local Heessian norm that we use $\|h\|^2_x$ is affine-invariant because

$$\|z\|^2_y = \left\langle \nabla^2 \phi(y) z, z \right\rangle = \left\langle A^T \nabla^2 f(Ay) A z, z \right\rangle = \left\langle (\nabla^2 f(x) h, h \right\rangle = \|h\|^2_x,$$

where $h = Az$. On the other hand, $\|h\|^2_2$ is not affine-invariant because

$$\|z\|^2 = \langle z, z \rangle = \left\langle A^{-1}h, A^{-1}h \right\rangle = \|A^{-1}h\|^2_2.$$

With respect to geometry, the most natural norm is local Hessian norm, $\|h\|^2_x$. From affine invariance follows that for this norm, the level sets $\{x \in \mathbb{E} \,|\, f(x) \leq c\}$ are balls centered around x (all directions have the same scaling). In comparison, for the l_2 norm, scaling is dependent on eigenvalues of the Hessian. In terms of convergence, in l_2 norms one direction can significantly dominate others and slow-down algorithm.

Significance for algorithms: Algorithms that are not affine-invariant can suffer from chosen coordinate system. This is the case for cubic Newton, as its model (4) is bound to base-dependent l_2 norm. Same is true for any other method regularized with $\|h\|_2$.

On the other hand, (damped) Newton methods have affine-invariant models, and hence are independent of the chosen coordinate system. We prove this claim in following lemma (note: $\alpha_k = 1$ and α_k from (10) are affine-invariant).

Lemma 1 (Lemma 5.1.1 Nesterov et al. [2018])**.** *Let the sequence $\{x_k\}$ be generated by a damped Newton method with affine-invariant stepsize α_k, applied to the function f:*

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

For function $\phi(y)$, damped Newton method generates $\{y_k\}$:

$$y_{k+1} = y_k - \alpha_k \left[\nabla^2 \phi(y_k)\right]^{-1} \nabla \phi(y_k),$$

with $y_0 = A^{-1}x_0$. Then $y_k = A^{-1}x_k$.

Significance in assumptions: self-concordance

We showed that damped Newton methods preserve affine-invariance through iterations. It is more fitting to analyze these methods under affine-invariant assumptions. Affine-invariant version of smoothness, *self-concordance*, was introduced in Nesterov and Nemirovskii [1994].

Definition 1. *Convex function $f(x) \in C^3$ is called self-concordant if*

$$|D^3 f(x)[h]^3| \leq M_f \|h\|_x^3, \tag{11}$$

where for any $p \geq 1$, we denote $D^p f(x)[h]^p \stackrel{def}{=} D^p f(x)[h, \ldots, h]$ as a directional p-th order derivative[a] of $f(x)$ along $h \in \mathbb{E}$, $i = 1, \ldots, p$.

[a]For example, $D^1 f(x)[h] = \langle \nabla f(x), h \rangle$ and $D^2 f(x)[h]^2 = h^T \nabla^2 f(x) h$.

Both sides of inequality are affine-invariant. This assumption corresponds to a big class of optimization methods called interior-point methods. Self-concordance implies uniqueness of the solution, as stated in following proposition.

Proposition 1 (Theorem 5.1.16, Nesterov et al. [2018])**.** *Let a self-concordant function f be bounded below. Then it attains its minimum at a single point.*

Rodomanov and Nesterov [2021] introduced stronger version of this assumption.

Definition 2. *Convex function $f(x)$ is called strongly self-concordant if*

$$\nabla^2 f(y) - \nabla^2 f(x) \preceq L_f \|y - x\|_z \nabla^2 f(w), \quad \forall y, x, z, w \in \mathbb{R}^d. \tag{12}$$

For our analysis, we introduce definitions for functions between self-concordant and strongly self-concordant.

Definition 3. *Convex function $f(x)$ is called semi-strongly self-concordant if*

$$\nabla^2 f(y) - \nabla^2 f(x) \preceq L_f \|y - x\|_x \nabla^2 f(x), \quad \forall y, x \in \mathbb{R}^d. \tag{13}$$

Definition 4. *Convex function $f(x)$ is called weakly-strongly self-concordant if*

$$\left\langle (\nabla^2 f(y) - \nabla^2 f(x))(y - x), (y - x) \right\rangle \leq L_f \|y - x\|_x^3, \quad \forall x, y \in \mathbb{R}^d. \tag{14}$$

Note that all of the Definitions 1 - 4 are affine-invariant and

$$\begin{aligned} &\textit{strong self-concordance} \subseteq \textit{semi-strong self-concordance} \\ &\subseteq \textit{weakly-strong self-concordance} \subseteq \textit{self-concordance}. \end{aligned}$$

These notions are related to the convexity and smoothness; strong concordance follows from function L_2-Lipschitz continuous Hessian and strong convexity.

Lemma 2 (Example 4.1., Rodomanov and Nesterov [2021])**.** *Let $C : \mathbb{E} \to \mathbb{E}^*$ be a self-adjoint positive definite operator. Suppose there exist $\mu > 0$ and $L_2 \geq 0$ such that the function f is μ-strongly convex and its Hessian is L_2-Lipschitz continuous with respect to the norm $\| \cdot \|_C$. Then f is strongly self-concordant with constant $L_f = \frac{L_2}{\mu^{3/2}}$.*

From assumptions to algorithm

From weakly-strong self-concordance we can get a second-order bounds on the function and model.

Lemma 3. *Assume $f(x)$ is weakly-strongly self-concordant then*

$$\left| f(y) - f(x) - \langle \nabla f(x), y - x \rangle - \tfrac{1}{2} \left\langle \nabla^2 f(x)(y - x), (y - x) \right\rangle \right| \leq \tfrac{L_f}{6} \|y - x\|_x^3. \tag{15}$$

Consequently, we have upper bound for function value in form

$$f(y) \leq f(x) + \langle \nabla f(x), y - x \rangle + \tfrac{1}{2} \left\langle \nabla^2 f(x)(y - x), y - x \right\rangle + \tfrac{L_f}{6} \|y - x\|_x^3. \tag{16}$$

We can define iterates of optimization algorithm to be minimizers of the right hand side of (16),

$$S(x) \stackrel{\text{def}}{=} x + \operatorname*{argmin}_{h \in \mathbb{E}} \left\{ f(x) + \langle \nabla f(x), h \rangle + \tfrac{1}{2} \left\langle \nabla^2 f(x) h, h \right\rangle + \tfrac{L_f}{6} \|h\|_x^3 \right\}, \tag{17}$$

$$x_{k+1} = S(x_k). \tag{18}$$

It turns out that subproblem (17) is easy to solve. To get an explicit solution of the subproblem, we compute its gradient w.r.t. h. For solution h^*, it should be equal to zero.

$$\nabla f(x) + \nabla^2 f(x) h^* + \tfrac{L_f}{2} \|h^*\|_x \nabla^2 f(x) h^* = 0 \tag{19}$$

$$h^* = -\left[\nabla^2 f(x)\right]^{-1} \nabla f(x) \cdot \left(\tfrac{L_f}{2} \|h^*\|_x + 1 \right)^{-1} \tag{20}$$

We get that step (18) has the same direction as a Damped Newton and scaled by $\alpha_k = \left(\frac{L_f}{2} \|h^*\|_x + 1 \right)^{-1}$. Now, we substitute h^* from (20) to (19).

$$\nabla f(x) - \nabla f(x) \alpha_k - \tfrac{L_f}{2} \left(\nabla f(x)^T \left[\nabla^2 f(x)\right]^{-1} \nabla f(x) \right)^{1/2} \nabla f(x) \alpha_k^2 = 0$$

$$\nabla f(x) \left(1 - \alpha_k - \tfrac{L_f}{2} \|\nabla f(x)\|_x^* \alpha_k^2 \right) = 0 \tag{21}$$

We solve the quadratic equation (21) for α_k, and obtain explicit formula for stepsizes of *AIC Newton*, as (10). We formalize this connection in theorem, for further explanation see proof in Appendix B .

Theorem 1. *Update of* AIC Newton (10)

$$x_{k+1} = x_k - \alpha_k \nabla^2 f(x_k)^{-1} \nabla f(x_k), \qquad \textit{where} \qquad \alpha_k = \frac{-1 + \sqrt{1 + 2 L_H \|\nabla f(x_k)\|_{x_k}^*}}{L_H \|\nabla f(x_k)\|_{x_k}^*},$$

is a minimizer of upper bound (17)*,* $x_{k+1} = S(x_k)$.

Remark: In main paper body, we show convergence through self-concordance. However, it is also possible to obtain convergence without directly using self-concordance. See Appendix B.

4 Global Convergence

Next, we focus on global convergence guarantees. We will utilize the following assumption:

Assumption 1 (Bounded level sets)**.** *The objective function f has a finite optimum x_* such that Also, the diameter of the level set $\mathcal{L}(x_0) \stackrel{\text{def}}{=} \{x \in \mathbb{E} : f(x) \le f(x_0)\}$ is bounded by some constant $D_2 > 0$ in l_2 norm* [a]*,* $\max_{x \in \mathcal{L}(x_0)} \|x - x^*\|_2 \le D_2 < +\infty$.

[a] We state it in l_2 norm to be verified more easily. In proofs, we use D in Hessian norms, (24).

Our analysis proceeds as follows. Firstly, we show that one step of the algorithm decreases function value, and secondly, we use the technique from [Ghadimi et al., 2017] to show that multiple steps lead to $\mathcal{O}\left(k^{-2}\right)$ global convergence. To prove the global convergence, we need additional lemma.

Lemma 4 (One step globally)**.** *Assume that function $f(x)$ is weakly-strongly self-concordant convex function with positive-definite Hessian and we set next model as* (17)*. Then for any $x \in \mathbb{E}$, we have*

$$f(S(x)) \le \min_{y \in \mathbb{E}} \left\{ f(y) + \tfrac{L_f}{3} \|y - x\|_x^3 \right\}. \tag{22}$$

This lemma implies that function value decreases each step (take $y \leftarrow x$). Using notation of Assumption 1, $x_k \in \mathcal{L}(x_0)$ for any $k \ge 0$. Also, from (14) for $y \leftarrow x$ and $x \leftarrow x_*$, we get

$$\|x - x_*\|_x \le \left(\|x - x_*\|_{x_*}^2 + L_f \|x - x_*\|_{x_*}^3 \right)^{\frac{1}{2}}. \tag{23}$$

We denote those distances D and R,

$$D \stackrel{\text{def}}{=} \max_{t \in [0; k+1]} \|x_t - x_*\|_{x_t} \qquad \text{and} \qquad R \stackrel{\text{def}}{=} \max_{x \in \mathcal{L}(x_0)} \left(\|x - x_*\|_{x_*}^2 + L_f \|x - x_*\|_{x_*}^3 \right)^{\frac{1}{2}}. \tag{24}$$

They are both affine-invariant and from (23) R upper bounds D. While R depends only on the level set $\mathcal{L}(x_0)$, D can be used to obtain more tight inequalities. D_2 is the common distance, but unusable as it would ruin affine-invariant properties.

Theorem 2. *Let $f(x)$ be a weakly-strongly self-concordant convex function with positive-definite Hessian and Assumption 1 holds. Then, after $k+1$ iterations of Algorithm 1, we have the following convergence rate:*

$$f(x_{k+1}) - f(x_*) \leq O\left(\frac{L_f D^3}{k^2}\right) \leq O\left(\frac{L_f R^3}{k^2}\right). \tag{25}$$

Proof of the theorem is in Appendix B.2 As a result of Theorem 2, Algorithm 1 converges globally with a fast rate $\mathcal{O}\left(k^{-2}\right)$. We can now proceed with quadratic analysis locally.

5 Local Convergence

For local quadratic convergence we need following additional lemmas.

Lemma 5 (Local strongly convexity)**.** *For convex M_f-self-concordant function f in the neighborhood of the solution holds:*

$$\nabla^2 f(x_k)^{-1} \preceq c\nabla^2 f(x_{k+1})^{-1}, \quad \textit{where} \quad c \overset{def}{=} \left(1 - \frac{M_f}{2}\sqrt{\alpha_k}\|\nabla f(x_k)\|^*_{x_k}\right)^{-2} \sim \frac{36}{25}. \tag{26}$$

With assumptions properly introduced, we are ready to present our analysis. Firstly, we show gradient norm decrease and secondly, we chain it for multiple consecutive iterates to obtain convergence rate.

Lemma 6 (One step decrease locally)**.** *Assume that the function is semi-strongly self-concordant. For norms of the gradients hold*

$$\|\nabla f(x_{k+1})\|^*_{x_k} \leq \frac{3}{2}L_f\alpha_k^2\|\nabla f(x_k)\|^{*2}_{x_k} < \frac{3}{2}L_f\|\nabla f(x_k)\|^{*2}_{x_k} \tag{27}$$

Using Lemma 5, we shift the gradient bound to respective norms,

$$\|\nabla f(x_{k+1})\|^*_{x_{k+1}} \leq \frac{3\sqrt{c}}{2}L_f\alpha_k^2\|\nabla f(x_k)\|^{*2}_{x_k} < \frac{3\sqrt{c}}{2}L_f\|\nabla f(x_k)\|^{*2}_{x_k}. \tag{28}$$

Gradient norm decrease $\|\nabla f(x_{k+1})\|^*_{x_{k+1}} \leq \|\nabla f(x_k)\|^*_{x_k}$ holds for $\|\nabla f(x_k)\|^*_{x_k} \leq \frac{6\sqrt{c}}{L_f(1-3\sqrt{c})^2}$.

Hence neighborhood of local convergence is $\|\nabla f(x)\|^*_x \leq \frac{6\sqrt{c}}{L_f(1-3\sqrt{c})^2}$. One step of *AIC Newton* decreases gradient norm quadratically. Multiple steps leads to following decrease.

Theorem 3 (Local convergence rate)**.** *Assume function f is semi-strongly self-concordant, and μ-strongly convex with L_1-Lipschitz continuous gradient. Starting point x_0 is in the neighborhood of the solution such that $\|\nabla f(x_0)\|^*_{x_0} \leq \frac{6\sqrt{c}}{L_f(1-3\sqrt{c})^2}$. For $k \geq 0$, we have quadratic decrease of the gradient norms,*

$$\|\nabla f(x_k)\|^*_{x_k} \leq \left(\frac{3\sqrt{c}}{2}L_f\right)^k \left(\|\nabla f(x_0)\|^*_{x_0}\right)^{2^k}. \tag{29}$$

6 Numerical Experiments

In this section, we evaluate proposed *AIC Newton* (Algorithm 1) algorithm on the logistic regression task and second-order lower bound function. We compare it with regularized Newton methods sharing fast global convergence guarantees: cubic Newton method [Nesterov and Polyak, 2006], and globally regularized Newton method ([Mishchenko, 4/2021, Doikov and Nesterov, 12/2021]) with L_2-constant. Because *AIC Newton* has a form of a damped Newton method, we also compare it with Damped Newton with fixed (tuned) stepsize $\alpha_k = \alpha$. We report decrease in function value $f(x_k)$, function value suboptimality $f(x_k) - f(x_*)$, and l_2 norm of the gradient $\|\nabla f(x_k)\|^2$.

Logistic regression: For first part, we solve the following empirical risk minimization problem:

$$\min_{x\in\mathbb{R}^d}\left\{f(x) = \frac{1}{m}\sum_{i=1}^m \log\left(1 - e^{-b_i a_i^T x}\right) + \frac{\mu}{2}\|x\|_2^2\right\},$$

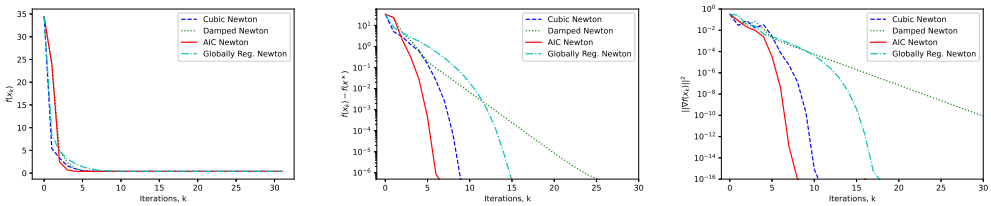

Figure 1: Comparison of regularized Newton methods and Damped Newton method for logistic regression task on *a9a* dataset.

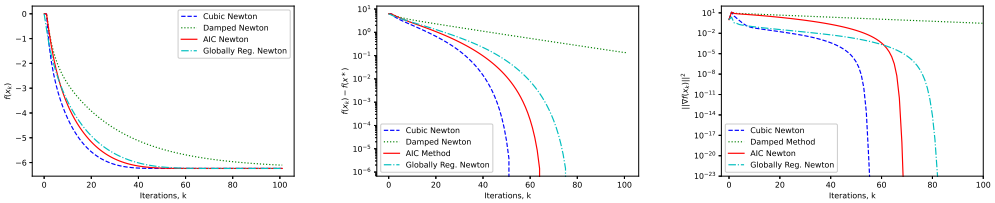

Figure 2: Comparison of regularized Newton methods and Damped Newton method for *second-order lower bound function*.

where $\{(a_i, b_i)\}_{i=1}^m$ are given data samples described by features a_i and class $b_i \in \{-1, 1\}$. In Figure 1, we consider task of classification images on dataset *a9a* [Chang and Lin, 2011]. Number of features for every data sample is $d = 123$, $m = 20000$. We take staring point $x_0 \stackrel{\text{def}}{=} 10[1, 1, \ldots, 1]^T$ and $\mu = 10^{-3}$. Our choice is differ from $x_0 = 0$ (equal to all zeroes) to show globalisation properties of methods ($x_0 = 0$ is very close to the solution, as Newton method converges in 4 iterations). Parameters of all methods are fine-tuned, we choose parameters L_f, L_2, α (of *AIC Newton*, cubic Newton, Damped Newton, resp.) to largest values having monotone decrease in reported metrics. Fine-tuned values are $L_f = 0.97$ $L_2 = 0.000215$, $\alpha = 0.285$, note that for $L_f = 0.965$, $L_2 = 0.00021$, $\alpha = 0.286$ respective methods are non-monotone.

The Figure 1 demonstrates that *AIC Newton* converges slightly faster than Cubic Newton method, notably faster than Globally Regularised Newton and significantly faster than Damped Newton.

Second-order lower bound function: For second part we solve the following minimization problem:

$$\min_{x \in \mathbb{R}^d} \left\{ f(x) = \frac{1}{d} \sum_{j=1}^{d} \left| [Ax]_j \right|^3 - x_1 + \frac{\mu}{2} \|x\|_2^2 \right\}, \quad \text{where} \quad A = \begin{pmatrix} 1 & -1 & 0 & \ldots & 0 \\ 0 & 1 & -1 & \ldots & 0 \\ & & \ldots & \ldots & \\ 0 & 0 & \ldots & 1 & -1 \\ 0 & 0 & \ldots & 0 & 1 \end{pmatrix}.$$

This function is a lower bound for a class of functions with Lipshitz continuous Hessian (4) with additional regularization [Nesterov and Polyak, 2006, Nesterov, 2021]. In Figure 2, we take $d = 20$, $x_0 = 0$ (equal to all zeroes). Parameters L_f, L_2, α are fine-tuned to largest values having monotone decrease in reported metrics: $L_f = 662$ $L_2 = 0.662$, $\alpha = 0.0172$. For $L_f = 660$, $L_2 = 0.66$, $\alpha = 0.0175$, respective methods are non-monotone.

Figure 2 demonstrates that *AIC Newton* converges slightly slower than Cubic Newton method, slightly faster than Globally Regularised Newton, and significantly faster than Damped Newton.

To summarize experimental results, *AIC Newton* has practical global convergence. It is competitive with cubic Newton method, and faster than globally regularised Newton method. Note, that cubic Newton's iteration needs additional line-search, so one iteration of Cubic Newton is computationally harder than one iteration of *AIC Newton*. We add more comparison to Appendix A.

References

Chih-Chung Chang and Chih-Jen Lin. LIBSVM: a library for support vector machines. *ACM transactions on intelligent systems and technology (TIST)*, 2(3):1–27, 2011.

Andrew R Conn, Nicholas IM Gould, and Philippe L Toint. *Trust region methods*. SIAM, 2000.

Alexandre d'Aspremont, Damien Scieur, and Adrien Taylor. Acceleration methods. January 2021.

Nikita Doikov and Yurii Nesterov. Gradient regularization of Newton method with Bregman distances. *arXiv preprint arXiv:2112.02952*, 12/2021.

Nikita Doikov and Yurii Nesterov. Local convergence of tensor methods. *Mathematical Programming*, pages 1–22, 2021.

Saeed Ghadimi, Han Liu, and Tong Zhang. Second-order methods with cubic regularization under inexact information. *arXiv preprint arXiv:1710.05782*, 2017.

Robert M Gower, Mark Schmidt, Francis Bach, and Peter Richtárik. Variance-reduced methods for Machine Learning. *Proceedings of the IEEE*, 108(11):1968–1983, 2020.

Andreas Griewank. The modification of Newton's method for unconstrained optimization by bounding cubic terms. Technical report, Technical report NA/12, 1981.

Florian Jarre and Philippe L Toint. Simple examples for the failure of Newton's method with line search for strictly convex minimization. *Mathematical Programming*, 158(1):23–34, 2016.

Peter Kairouz, H Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Kallista Bonawitz, Zachary Charles, Graham Cormode, Rachel Cummings, et al. Advances and open problems in Federated Learning. *Foundations and Trends® in Machine Learning*, 14(1–2):1–210, 2021.

Leonid Vital'evich Kantorovich. Functional analysis and applied mathematics. *Uspekhi Matematicheskikh Nauk*, 3(6):89–185, 1948.

Anastasia Koloskova, Nicolas Loizou, Sadra Boreiri, Martin Jaggi, and Sebastian Stich. A unified theory of decentralized SGD with changing topology and local updates. In *International Conference on Machine Learning*, pages 5381–5393. PMLR, 2020.

Jakub Konečný, H Brendan McMahan, Felix X Yu, Peter Richtárik, Ananda Theertha Suresh, and Dave Bacon. Federated Learning: Strategies for improving communication efficiency. *arXiv preprint arXiv:1610.05492*, 2016.

Walter F Mascarenhas. On the divergence of line search methods. *Computational & Applied Mathematics*, 26(1):129–169, 2007.

Konstantin Mishchenko. Regularized Newton method with global $\mathrm{O}(1/k^2)$ convergence. *arXiv preprint arXiv 2112.02089*, 4/2021.

Jorge J Moré. The Levenberg-Marquardt algorithm: implementation and theory. In *Numerical analysis*, pages 105–116. Springer, 1978.

Yurii Nesterov. *Introductory lectures on Convex optimization: a basic course*, volume 87. Springer, 2013.

Yurii Nesterov. Implementable tensor methods in unconstrained convex optimization. *Mathematical Programming*, 186(1):157–183, 2021.

Yurii Nesterov and Arkadii Nemirovskii. *Interior-point polynomial algorithms in Convex programming*. SIAM, 1994.

Yurii Nesterov and Boris T Polyak. Cubic regularization of Newton method and its global performance. *Mathematical Programming*, 108(1):177–205, 2006.

Yurii Nesterov et al. *Lectures on Convex optimization*, volume 137. Springer, 2018.

Roman A Polyak. Regularized Newton method for unconstrained Convex optimization. *Mathematical Programming*, 120(1):125–145, 2009.

Anton Rodomanov and Yurii Nesterov. Greedy quasi-Newton methods with explicit superlinear convergence. *SIAM Journal on Optimization*, 31(1):785–811, 2021.

Nicolas Roux, Mark Schmidt, and Francis Bach. A stochastic gradient method with an exponential convergence rate for finite training sets. *Advances in Neural Information Processing Systems*, 25, 2012.

Tianxiao Sun and Quoc Tran-Dinh. Generalized self-concordant functions: A recipe for Newton-type methods. *Mathematical Programming (2018)*, March 2017.

Tjalling J Ypma. Historical development of the Newton–Raphson method. *SIAM review*, 37(4): 531–551, 1995.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes]
   (c) Did you discuss any potential negative societal impacts of your work? [N/A] Paper is of theoretical nature.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes] We highlighted all assumptions.
   (b) Did you include complete proofs of all theoretical results? [Yes] See Appendix B
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] For source code, see supplementary materials.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See Sections, 6, A
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [N/A] We compare deterministic algorithms.
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [N/A] We used 0 GPU hours.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes]
   (b) Did you mention the license of the assets? [N/A] We use publicly available datasets.
   (c) Did you include any new assets either in the supplemental material or as a URL? [No]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]