1 Methodology

In this section, we will discuss the methodology and it main idea of this algorithm. Let's define a minimize objective function $\arg\min_{x\in\Omega} f(x)$, where Ω is the sample space which may constrained or unconstrained.

1.1 Newton's algorithm

The standard form of Newton's algorithm use the second order Taylor expansion. In the standard Newton's algorithm, given a current iterate $x^t \in \Omega$, where Ω is the sample space of x^t and $\Omega \subseteq \mathbb{R}^d$, it generate the new iterate x^{t+1} with the following formula.

$$x^{t+1} = \arg\min_{x\subseteq\Omega}\{\frac{1}{2} < x - x^t, \nabla^2 f(x^t)(x - x^t) > + < \nabla f(x^t), x - x^t >\} \tag{1}$$

where $< \cdot, \cdot >$ means the inner product of two vectors. Suppose that we have available a Hessian matrix square root $\nabla^2 f(x)^{\frac{1}{2}}$, it is a $n \times d$ matrix which have the property $(\nabla^2 f(x)^{\frac{1}{2}})^T \nabla^2 f(x)^{\frac{1}{2}} = \nabla^2 f(x)$, for $n \geq rank(\nabla^2 f(x))$. At this present, let's consider a function of the form $f(x) = g(Ax)$ where $A \in \mathbb{R}^{n\times d}$ and the function $g : \mathbb{R}^n \to \mathbb{R}$ can be written as $g(Ax) = \sum_{i=1}^n g_i(< a_i, x >)$. In this case, a square root of Hessian matrix is given by $n \times d$ matrix $\nabla^2 f(x)^{\frac{1}{2}} = diag\{g_i^{''}(< a_i, x >)\}_{i=1}^n A$.

In terms of this notation, the ordinary Newton update can be re-written as

$$x^{t+1} = \arg\min_{x\subseteq\Omega}\{\frac{1}{2}||\nabla^2 f(x^t)^{\frac{1}{2}}(x - x^t)||_2^2 + < \nabla f(x^t), x - x^t >\} \tag{2}$$

and we can use those formula to find the globe minimize value in the convex function and convex sample space.

1.2 Newton Sketch algorithm

1.2.1 Fully Sketch Newton

In this section, we will discuss the fully sketch newton method. For a sketch matrix $S \in \mathbb{R}^{m\times n}$, it is an *isotropic sketch matrix*, satisfying the relation $\mathrm{E}[S^T S] = I_n$. For the first step, we need to choose the sketch dimension m. Then, the Newton sketch algorithm generates a sequence of iterates $\{x^t\}_0^\infty$ according to the recursion.

$$x^{t+1} = \arg\min_{x\subseteq\Omega}\{\frac{1}{2}||S^t\nabla^2 f(x^t)^{\frac{1}{2}}(x - x^t)||_2^2 + < \nabla f(x^t), x - x^t >\} \tag{3}$$

where $S^t \in \mathbb{R}^{m\times n}$ is an *independent realization of a sketching matrix*. When the problem is unconstrained, i.e., $\Omega = \mathbb{R}^d$ and the matrix $(\nabla^2 f(x^t)^{\frac{1}{2}})^T (S^t)^T S^t \nabla^2 f(x^t)^{\frac{1}{2}}$ is invertible, the Newton sketch update takes the simpler form to

$$x^{t+1} = x^t - ((\nabla^2 f(x^t)^{\frac{1}{2}})^T (S^t)^T S^t \nabla^2 f(x^t)^{\frac{1}{2}})^{-1} \nabla f(x^t) \tag{4}$$

1.2.2 Partially Sketched Newton

Given an additive decomposition of the form $f(x) = f_0(x) + g(x)$, we perform a sketch of the Hessian $\nabla^2 f_0(x)$ but remain the exact form of the Hessian $\nabla^2 g(x)$. This conducts the partially sketched update

$$x^{t+1} = \arg\min_{x\in\Omega}\{\frac{1}{2}(x - x^t)^T Q^t (x - x^t) + < \nabla f(x^t), x - x^t >\} \tag{5}$$

where $Q^t = (S^t \nabla^2 f_0(x^t)^{\frac{1}{2}})^T S^t \nabla^2 f_0(x^t)^{\frac{1}{2}} + \nabla^2 g(x^t)$

The main idea of *Sketch Newton algorithm* is used a sketch matrix $S \in \mathbb{R}^{m\times n}$ where $m \ll n$ to reduce the dimension of original Hessian matrix $\nabla^2 f(x)$. The sketch dimension m can be chosen to be substantially smaller than n, in this case the sketched Newton updates will be much cheaper than a standard Newton update.

The intuition of the *Sketch Newton algorithm* is that: in the iterate x^{t+1}, the random objective function

$$\Phi(x; S^t) = \frac{1}{2}||S^t \nabla^2 f(x^t)^{\frac{1}{2}}(x - x^t)||_2^2 + < \nabla f(x^t), x - x^t >$$

whose expectation is $\mathrm{E}(\Phi(x; S^t))$, taking average over the isotropic sketch matrix S^t, is equal to the *Standard Newton objective*

$$\Phi(x) = \frac{1}{2}||\nabla^2 f(x^t)^{\frac{1}{2}}(x - x^t)||_2^2 + < \nabla f(x^t), x - x^t >$$

In this case, this algorithm can be seen as the stochastic form of Newton update.

2 Experimental Details

We will apply this method only on Logistic Regression because of the limitation of computing resource. The goal of this experiment is to compare the consuming time and the convergence ability of Sketch Newton and Standard Newton and the influence of different reduced dimensions to convergence ability of Sketch Newton.

2.1 Set up

2.1.1 Set up for comparing two algorithms

For Sketch Newton algorithm, the reduced dimension in this project which we selected is 100. We simulated a data matrix $X \in \mathbb{R}^{N\times K}$ from Standard Normal Distribution $\mathcal{N}(0, 1)$, where N represents the samples numbers and K denotes the coefficients number. In this project, we selected $N = 1000$ and $K = 100$.We randomly established the labels of each samples from Bernoulli distribution. The probability is $p = 0.5$ for label 1. The initial coefficients $\vec{\beta}^0 \in \mathbb{R}^K$ randomly build from Standard Normal Distribution $\mathcal{N}(0, 1)$. The learning rate for Sketch Newton is $1e^{-7}$. **For Standard Newton algorithm**, we used the same data matrix, same labels and same initial coefficients. The learning rate for Standard Newton is $1e^{-6}$. The max iteration of those two algorithms are all 25000 times.

2.1.2 Set up for exploring different reduced dimensions

We selected $N = 500$ and $K = 50$ for this data matrix. The reduced dimensions which we selected are $[5, 25, 45, 65, 85, 105]$. We randomly established the labels of each samples from Bernoulli distribution. The probability is $p = 0.5$ for label 1. The initial coefficients $\vec{\beta}^0 \in \mathbb{R}^K$ randomly build from Standard Normal Distribution $\mathcal{N}(0, 1)$. The learning rate for those Sketch Newton are all $1e^{-3}$. The max iterations are all set 25000.

2.2 Results

The comparison between Sketch Newton and Standard Newton and the effect of different reduced dimensions are showed in the plots (Because the limitation of pages, those plots are showed in appendix). We used the mean MSE loss for the predict vector and truth labels to present the convergence ability.

From figure 1 we can know that there contains fluctuation in the Sketch Newton method and the convergence ability of Sketch Newton method is much better than the Standard Newton method in the same iterations even though the learning rate of Sketch Newton is 10 times smaller than Standard Newton. From the theory 3.3.1 we can know that this is in our expectation. The consuming time of Sketch Newton is 46 minutes with 56 seconds. The consuming time of Standard Newton is 1 hour 10 minutes with 53 seconds. Those phenomenons indicate that this Sketch Newton algorithm transcends the Standard Newton method prodigious, regardless on the consuming time or the convergence ability.

From figure 2 we can know that with different reduced dimension, the convergence ability is raised when reduced dimension increased and then declined when reduced dimension becomes huge. The best reduced dimension in those experiments is 45. If the reduced dimensions too small, the information contains in the sketched Hessian matrix is small. So, it is hard to estimate the truth value that the convergence ability is low. However, with the dimension increasing, the raising rate of information is decreased but the random factor in Sketch Newton algorithm will effect the estimation ability more seriously. So, the convergence ability will decrease slightly.

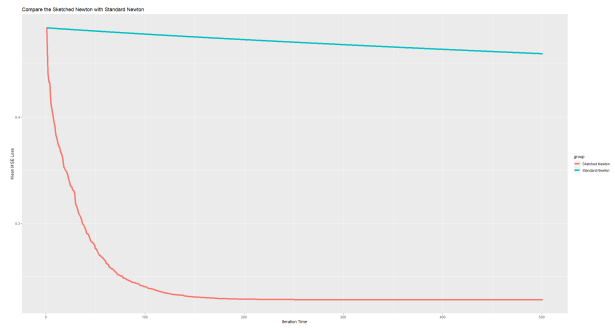

Figure 1: *The comparison between those two algorithms. Red line represents the Sketch Newton. Blue line represents the Standard Newton.*

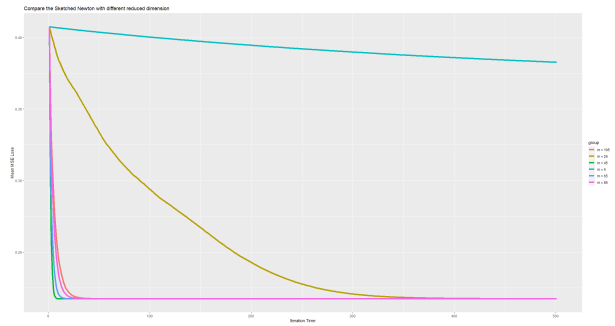

Figure 2: *The comparison between different reduced dimensions.*