情報数理科学 VII　最適化の手法

松島　慎

2020 年 1 月 7 日

1　凸解析の基本概念

Definition 1. 関数 $f : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ が凸関数であるとは

$$f(tx + (1-t)x') \leq tf(x) + (1-t)f(x') \tag{1}$$

を全ての x, x'、 $t \in [0,1]$ で満たすこと。f が凹関数とは $-f$ が凸関数であること。

ここで凸関数の重要な性質を三つあげる。一つ目は、勾配が下限関数を定めることである。

Property 1 (微分可能な凸関数の下限)**.** 微分可能な関数 f が凸関数であることは

$$f(x) \geq f(y) + \nabla f(y)^\top (x - y) \tag{2}$$

が全ての x, y で成立することと同値。

二つ目は、勾配に単調性があることである。

Property 2 (凸関数の単調性)**.** 微分可能な関数 f が凸関数であることは

$$(\nabla f(x) - \nabla f(y))^\top (x - y) \geq 0 \tag{3}$$

が全ての x, y で成立することと同値。

Proof. 略 □

最後の三つ目は、ヘッセ行列に半正定値性があることである。

Property 3 (凸関数の二回微分)**.** 二回微分可能な関数 f が凸関数であることは

$$y^\top \nabla^2 f(x) y \geq 0 \tag{4}$$

が全ての x, y で成立することと同値。ここで $\nabla^2 f(x)$ は f の x でのヘッセ行列。さらにこれは $\nabla^2 f(x)$ が全ての x で半正定値であることと同値。

Proof. 略 □

次に μ-強凸関数を定義する。関数が強凸関数であることを仮定することで、多くの場合より速い収束が達成でき、解析も簡単になる。理論的な立場では強凸の仮定を外したり緩めたりすることも重要であるが、ここでは f が強凸であることを仮定する。

Definition 2 (μ-強凸関数 (μ-strongly convex function))**.** $\mu > 0$ に対して、関数 f が μ-強凸関数であるとは

$$f(tx + (1-t)x') \leq tf(x) + (1-t)f(x') - \frac{\mu}{2} t(1-t) \left\| x - x' \right\|^2$$

をすべての x, x'、 $t \in [0,1]$ で満たすこと

強凸関数にもやはり対応する三つの性質がある。ある関数が L-滑凸関数であることと対照的に、ある関数が強凸関数であるということは、各点で曲率 μ の二次関数が下限を設定することを意味する。

Property 4 (μ-強凸関数の下限)**.** 微分可能な関数 f が μ-強凸関数であることは

$$f(x) \geq f(y) + \nabla f(y)^\top (x - y) + \frac{\mu}{2} \left\| x - y \right\|^2$$

が全ての x, y で成立することと同値。

Proof. 略 □

さらに勾配に対する次の性質も成り立つ。これも単調性の強さを評価することができる。

Property 5 (μ-強凸関数の強単調性)**.** 微分可能な関数 f が μ-強凸関数であることは

$$(\nabla f(x) - \nabla f(y))^\top (x - y) \geq \mu \left\| x - y \right\|^2 \tag{5}$$

が全ての x, y で成立することと同値。

Proof. 略 □

さらにヘッセ行列に対し次の性質が成り立つ。

Property 6 (μ-強凸関数の二回微分)**.** 二回微分可能な関数 f が μ-強凸関数であることは

$$y^\top \nabla^2 f(x) y \geq \mu \left\| y \right\|^2 \tag{6}$$

が全ての x, y で成立することと同値。さらにこれは $\nabla^2 f(x) - \mu I$ が全ての x で半正定値であることと同値。

Proof. 略 □

これらは次の事実からわかる。

Property 7 (μ-強凸関数と凸関数の関係)**.** $f(x)$ が μ-強凸関数であることは $f(x) - \frac{\mu}{2} \left\| x \right\|^2$ が凸関数であることと同値。

Proof. 略 □

特に、本稿では各 f_i が L-滑凸関数である場合に限定する。

Definition 3 (L-滑凸関数 (L-smooth convex function))**.** 微分可能な凸関数が L-滑凸関数 *(L-smooth convex function)* である

とは $L > 0$ に対して、

$$\|\nabla f(x) - \nabla f(y)\| \leq L \|x - y\| \tag{7}$$

が全ての x, y で成立すること。

各 f_i が L-滑凸関数な場合、f 自体も L-滑凸関数となる。

L-滑凸関数にも凸関数の三つの性質に対応する特徴づけがある。一つ目はある関数が L-滑凸関数であることは、各点で曲率 L の二次関数が上限を設定することを意味する。すなわち次のことが言える。

Property 8 (L-滑凸関数の上限)**.** 微分可能な凸関数が L-滑凸関数 *(L-smooth convex function)* であることは,

$$f(x) \leq f(y) + \nabla f(y)^\top (x - y) + \frac{L}{2} \|x - y\|^2 \tag{8}$$

が全ての x, y で成立することと同値。

Proof. 略 □

Property 9. 微分可能な関数 f が μ-強凸関数であることは

$$(\nabla f(x) - \nabla f(y))^\top (x - y) \leq L \|x - y\|^2 \tag{9}$$

が全ての x, y で成立することと同値。

Proof. 略 □

最後に三つ目は、

Property 10 (L-滑凸関数の二回微分)**.** 二回微分可能な関数 f が L-滑凸関数であることは

$$y^\top \nabla^2 f(x) y \leq L \|y\|^2 \tag{10}$$

が全ての x, y で成立することと同値。さらにこれは $LI - \nabla^2 f(x)$ が全ての x で半正定値であることと同値。

Proof. 略 □

これらは次の事実からわかる。

Property 11 (L-滑凸関数と凸関数の関係)**.** $f(x)$ が L-滑凸関数であることは $\frac{L}{2}\|x\|^2 - f(x)$ が凸関数であることと同値。

Proof. 略 □

全体の関数 f が強凸関数である場合、最適化問題 (11) の最小解の存在が保証される。κ を L/μ で定義する。全体の f に関してこれを条件数とよび、値は常に 1 以上になる。関数が最適化しやすいかどうかはこの値に大きく依存し、一般に大きい方が最適化が難しくなる。このことは以下で紹介する理論的結果にも反映されており、実験的にも簡単に観察することができる。

2 最適化法

以下では各種の最適化問題の解を求めるためのアルゴリズムを紹介する。ここでは簡単のために各 f_i が L-滑凸関数でありかつ

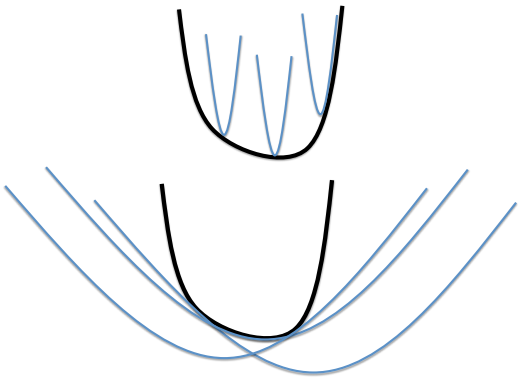

図 1 L-滑凸関数と μ-強凸関数のイメージ。Property 8 と Property 4 の右辺は黒線の関数 $f(x)$ に関して、青線に示すような、任意に選んだ点 y で接する曲率一定の二次関数を定義している。これが全ての x に関して上限になっているというのが L-滑凸関数の必要十分条件であり、下限になっているというのが μ- 強凸関数の必要十分条件となる。

μ-強凸関数である場合の最小化問題、

$$\text{minimize} \qquad (f(x) =) \frac{1}{n} \sum_{i=1}^{n} f_i(x) \tag{11}$$

を考える。例えば、ある 2 値ラベルつきデータ $\{x_i, y_i\}_{i=1,..,n} \in \left(\mathbb{R}^d \times \{-1, 1\}\right)^n$ と正則化係数 λ に対する L2 ロジスティック回帰

$$\frac{1}{n} \sum_{i=1}^{n} \left(\frac{\lambda}{2} \|w\|^2 + \log\left(1 + \exp(-y_i {x_i}^\top w)\right) \right) \tag{12}$$

や L2 サポートベクターマシン（L2SVM）

$$\frac{1}{n} \sum_{i=1}^{n} \left(\frac{\lambda}{2} \|w\|^2 + \left(\max\left(0, 1 - y_i {x_i}^\top w\right)\right)^2 \right) \tag{13}$$

は（w の関数として）今回考える仮定を満たしている。また回帰のためのデータ $\{x_i, y_i\}_{i=1,..,n} \in \left(\mathbb{R}^d \times \mathbb{R}\right)^n$ に関するリッジ回帰

$$\frac{1}{n} \sum_{i=1}^{n} \left(\frac{\lambda}{2} \|w\|^2 + \left(y_i - {x_i}^\top w\right)^2 \right) \tag{14}$$

も仮定を満たす。全体の関数 f が強凸関数である場合、最適化問題 (11) の最小解の存在が保証される。κ を L/μ で定義する。全体の f に関してこれを条件数とよび、値は常に 1 以上になる。関数が最適化しやすいかどうかはこの値に大きく依存し、一般に大きい方が最適化が難しくなる。このことは以下で紹介する理論的結果にも反映されており、実験的にも簡単に観察することができる。

2.1 勾配法

このとき以下のことがわかる。

Theorem 1 ([Nesterov (2004)])**.** L-滑凸かつ μ-強凸である f に対し、$\eta < \frac{2}{L+\mu}$ とした場合の勾配法のアルゴリズムが生成する

Algorithm 1 勾配降下法のアルゴリズム

Set x_0
for $t = 0, 1, \ldots, T-1$ **do**
  $x_{t+1} = x_t - \eta \nabla f(x_t)$
end for
Return x_T

点列 $\{x_t\}_{t=0,\ldots}$ に対し、

$$\|x_t - x^*\|^2 \leq \left(1 - \frac{2L\eta}{1+\kappa}\right)^t \|x_0 - x^*\|^2 \tag{15}$$

が成り立つ。

すなわち一次の勾配降下法のアルゴリズムで生成される点列 x_t は最適解 x^* との二乗距離 $\|x_t - x^*\|^2$ が指数的に減少する。このような場合、アルゴリズムは一次収束するまたは線形収束するという。一般にある t_0 で

$$\|x_t - x^*\|^2 \leq C\epsilon(t) \qquad (\forall t \geq t_0) \tag{16}$$

ととなる場合、$\|x_t - x^*\|^2$ の収束レートが $\mathcal{O}(\epsilon(t))$ であるという。よって勾配降下法の収束レートは $\mathcal{O}(\exp(-t))$ である。t に関する依存性だけをみれば、勾配の情報を用いて達成しうる収束レートとしては最速であることがわかっている。一方で、ある近似精度 ϵ に対し、$\|x_t - x^*\|^2 \leq \epsilon$ となるために必要な反復回数 t を考える。上述の定理 1 によると、

$$\begin{aligned}
\|x_t - x^*\|^2 \leq \epsilon &\Leftarrow \left(1 - \frac{2}{1+\kappa}\right)^t \|x_0 - x^*\|^2 \leq \epsilon \\
&\Leftarrow e^{-\frac{2}{1+\kappa}t} \|x_0 - x^*\|^2 \leq \epsilon \\
&\Leftrightarrow -\frac{2}{1+\kappa} t \leq \log\left(\frac{\epsilon}{\|x_0 - x^*\|^2}\right) \\
&\Leftrightarrow t \geq \frac{1+\kappa}{2} \log\left(\frac{\|x_0 - x^*\|^2}{\epsilon}\right)
\end{aligned}$$

として $\frac{1+\kappa}{2}\log(\frac{\|x_0-x^*\|^2}{\epsilon})$ が十分な量としてとれる。このような時、特に ϵ だけに着目してみれば反復のコンプレキシティが $\mathcal{O}\left(\log(\frac{1}{\epsilon})\right)$ であるという。また、ϵ と κ に着目すれば、コンプレキシティは $\mathcal{O}\left(\kappa \log(\frac{1}{\epsilon})\right)$ である。収束レートに関しても同様の事情があって、これらの表現はどの変数に注目しているか明示されないことも多いので、注意が必要である。

2.2 確率的勾配法

確率的勾配法（Stochastic Gradient Method）はしばしば確率的勾配降下法（Stochastic Gradient Descent Method）と呼ばれる。そのため SGD と略記されることが多いが厳密には「降下法」の分類には属さない。本稿では確率的勾配法とは次のアルゴリズムを指す。

このときアルゴリズムは初期値 x_0, 学習率 η と確率変数列 i_t による。そのため、$t \geq 1$ に対し、x_t は確率変数列となるため、解析では $\|x_t - x^*\|^2$ ではなくその期待値 $\mathbb{E}\|x_t - x^*\|^2$ を考える。

ここで、次の仮定を置く。

Algorithm 2 確率的勾配法のアルゴリズム

Set x_0
for $t = 0, 1, \ldots, T-1$ **do**
  Sample i_t uniformly from $\{1, \ldots, n\}$.
  $x_{t+1} = x_t - \eta \nabla f_{i_t}(x_t)$
end for
Return x_T

Assumption 2.

$$\mathbb{E}\|\nabla f_{i_t}(x^*)\|^2 \leq \sigma^2 \tag{17}$$

が成立するとする[*1]。

σ によって $\mathbb{E}\|x_t - x^*\|^2$ の上限の形は大きく変わっていく。σ が小さければ収束は一般に速くなる。例えば、全ての i で $f_i(x) = f(x)$ であれば、

$$\mathbb{E}\|\nabla f_{i_t}(x^*)\|^2 = \mathbb{E}\|\nabla f(x^*)\|^2 = 0 \tag{19}$$

であるが、この場合、確率的勾配法は勾配降下法と一致するので、収束レートも同様であるべきである。

ここではステップサイズ一定の場合を考える。確率的勾配法はある時点で、$x_t = x^*$ となっていたとしても、次の反復では別の点に動いていることになるので、どうしても半径 $\mathcal{O}(\eta\sigma)$ の球程度の範囲で動き回ることになってしまう。ただし、そのような球を見つけることは指数的に速く可能である。そのことを表すのが次の定理である。

Theorem 4 ([Needel et al.(2015)])**.** 確率的勾配法は $\eta < \frac{1}{L}$ としたとき、*Assumption 2* のもとで、任意の L-滑かつ μ-強凸である f と、$x_0 \in \mathbb{R}^d$ に対して、

$$\mathbb{E}\|x_t - x^*\|^2 \leq (1 - 2\eta\mu(1-\eta L))^t \|x_0 - x^*\|^2 + \frac{\eta\sigma^2}{\mu(1-\eta L)} \tag{20}$$

である。

例えば $\eta = \frac{1}{2L}$ とした場合、次のような結果を得る。

Theorem 5. 確率的勾配法は $\eta = \frac{1}{2L}$ としたとき、*Assumption 2* のもとで、任意の L-滑かつ μ-強凸である f と、$x_0 \in \mathbb{R}^d$ に対して、

$$\mathbb{E}\|x_t - x^*\|^2 \leq \left(1 - \frac{1}{2\kappa}\right)^t \|x_0 - x^*\|^2 + \frac{\sigma^2}{\mu L} \tag{21}$$

である。

[*1] いくつかの文献で見られる以下の仮定はさらに強いので注意して区別する必要がある。

Assumption 3. 任意の x で

$$\mathbb{E}\|\nabla f_{i_t}(x)\|^2 \leq \sigma^2 \tag{18}$$

例えば、$\mathbb{R}^d$ 全域で考えている今の場合では、強凸かつ L-平滑な関数はこの仮定を満たすことができない。

また、ステップサイズを小さく設定すれば、任意の精度で解を求めることが可能である。

Theorem 6. 確率的勾配法は任意の平滑かつ強凸である f と、$x_0 \in \mathbb{R}^d$ に対して、*Assumption 2* のもとで、$\eta = \frac{\mu\epsilon}{2\epsilon\mu L + 2\sigma^2}$ ととれば、任意の $t \geq t_0 = 2\log\left(\frac{2\|x_0 - x^*\|^2}{\epsilon}\right)\left(\frac{L}{\mu} + \frac{\sigma^2}{\mu^2\epsilon}\right)$ に対し、

$$\mathbb{E}\left\|x_t - x^*\right\|^2 \leq \epsilon \tag{22}$$

となる。

勾配降下法の際とは異なり達成したい近似精度 ϵ によってステップサイズ η を変えなければならない。このような場合も反復のコンプレキシティが $\mathcal{O}\left(\log\left(\frac{2\|x_0 - x^*\|^2}{\epsilon}\right)\left(\frac{L}{\mu} + \frac{\sigma^2}{\mu^2\epsilon}\right)\right)$ であるという。

2.3 確率的分散縮小勾配法

確率的勾配法では式 (21) における確率的勾配の分散 σ^2 にかかわる項が収束を妨げているということが言える。ステップサイズを小さく設定すれば、任意の精度で解を求めることが可能であるが他に分散を縮小させる方法はないか？このような動機で Johnson らにより発見されたのが以下の SVRG (Stochastic Variance Reduced Gradient, 確率的分散縮小勾配法) と呼ばれるアルゴリズムである。SVRG においては二重の反復（ループ）

Algorithm 3 確率的分散縮小勾配法（SVRG）のアルゴリズム

```
Set x_0
for t = 0, 1, ..., T − 1 do
    g_t = (1/n) Σ_i ∇f_i(x_t)
    for s = 0, 1, ..., T_inner − 1 do
        Sample i_s uniformly from {1, ..., n}.
        x_{t,s+1} = x_{t,s} − η(g_t + ∇f_{i_s}(x_{t,s}) − ∇f_{i_s}(x_t))
    end for
    x_{t+1} = x_{t,T_inner}
end for
return x_T
```

が存在する。内側の反復において確率的勾配法と同様の更新式を適用しているが、単純に勾配 $\nabla f_{i_s}(x_{t,s})$ を確率的勾配として使うのではなく、$g_t + \nabla f_{i_s}(x_{t,s}) - \nabla f_{i_s}(x_t)$ を確率的勾配として扱っていることがわかる。これが確率的勾配となること、すなわち $\mathbb{E}(g_t + \nabla f_{i_s}(x_{t,s}) - \nabla f_{i_s}(x_t)) = \nabla f(x_{t,s})$ となることは $g_t = \mathbb{E}\nabla f_{i_s}(x_t)$ という関係があることからすぐにわかる。さらに、この項には分散縮小性があるということがわかる。すなわち、現在の点 x_t や $x_{t,s}$ が最適解に近づくにつれ、確率的勾配の分散も縮小していくと言うことである。[*2]

Theorem 7 ([Johnson and Zhang (2013)])**.** 任意の L-滑かつ μ-強凸である f と、$x_0 \in \mathbb{R}^d$ があるとする。*Algorithm 3* は $\eta < \frac{1}{2L}$、T_{inner} に関し

$$\frac{1}{2\mu\eta T_{\text{inner}}(1/2 - \eta L)} + \frac{\eta L}{1/2 - \eta L} < 1 \tag{23}$$

が成立する場合、左辺の値を ρ とし以下が成り立つ。

$$\mathbb{E}f(x_t) - f(x^*) \leq \rho^t \left(f(x_0) - f(x^*)\right). \tag{24}$$

上の定理から次のことが簡単にわかる。

Theorem 8. *Algorithm 3* は $\eta = \frac{1}{6L}$ に関し、

$$18\kappa < T_{\text{inner}} \tag{25}$$

が成立する場合、以下が成り立つ。任意の L-滑かつ μ-強凸である f と、$x_0 \in \mathbb{R}^d$ に対して、

$$\mathbb{E}\left\|x_t - x^*\right\|^2 \leq \kappa^2\left(\frac{9\kappa}{T_{\text{inner}}} + \frac{1}{2}\right)^t \left\|x_0 - x^*\right\|^2. \tag{26}$$

よって（外部）反復のコンプレキシティは $\mathcal{O}\left(\log\left(\frac{\kappa^2\|x_0 - x^*\|^2}{\epsilon}\right)\right)$ である。外部反復一回にかかる計算量は $\mathcal{O}\left(n + T_{\text{inner}}\right)$ であるから、全体の計算量は

$$\mathcal{O}\left(\left(n + T_{\text{inner}}\right)\log\left(\frac{\kappa^2\left\|x_0 - x^*\right\|^2}{\epsilon}\right)\right)$$

となる。特に κ, ϵ など n 以外の変数を定数とみなし、$n \to \infty$ の状況を考えると、$\mathcal{O}\left(n\log\left(\frac{\kappa^2\|x_0 - x^*\|^2}{\epsilon}\right)\right)$ となる。一方で、κ 以外の変数を定数とみなし $\kappa \to \infty$ の状況を考える場合、$T_{\text{inner}} = \mathcal{O}(\kappa)$ とし $\mathcal{O}\left(\kappa\log\left(\frac{\kappa^2\|x_0 - x^*\|^2}{\epsilon}\right)\right)$ となる。多くの文献ではこれらを総合して表現するために SVRG の計算量評価を $\mathcal{O}\left((n+\kappa)\log\left(\frac{1}{\epsilon}\right)\right)$ と書くことが多い。勾配降下法の計算量を考えると、各反復は $\mathcal{O}(n)$ の計算量がかかるため勾配降下法の計算量は $\mathcal{O}\left(n\kappa\log\left(\frac{\|x_0 - x^*\|^2}{\epsilon}\right)\right)$ となる。そのため勾配法に比べても、SVRG は計算量の n も κ も十分に大きい場合は計算量が抑えられると期待できる。

[*2] 厳密に言えば、ここで $v_{t,s}$ に分散縮小性があるとは、どのような δ でも $\|x_{t,s} - x^*\| + \|x_t - x^*\| \leq \epsilon$ ならば、$\mathbb{E}\|v_{t,s}\|^2 \leq \delta$ となるような ϵ が存在するということである。

2.4 L-平滑でない凸関数の最適化法

次に L-平滑でない関数の最小化を考える。ここでは最初化したい関数 F が以下のように書けるとする。

$$F(x) = f(x) + h(x) \tag{27}$$

ここで f は μ 強凸かつ L 平滑、h は（微分できるとは限らない）凸関数である。例としては LASSO 回帰などがあげられる。

$$\frac{1}{n}\sum_{i=1}^{n}\left(\frac{\lambda}{2}\|w\|_1 + \left(y_i - {x_i}^\top w\right)^2\right) \tag{28}$$

また制約付きの最適化問題もこの問題であると考えることができる。例えば制約 $\|w\| \leq 1$ のもとでの関数 f の最小化は

$$h(x) = \begin{cases} 0 & \|w\| \leq 1 \\ \infty & \|w\| > 1 \end{cases} \tag{29}$$

と考えればよい。

2.4.1 近接勾配降下法

近接勾配降下法のアルゴリズムは以下のようなものである。近

Algorithm 4 近接勾配降下法のアルゴリズム

Set x_0
for $t = 0, 1, \ldots, T-1$ **do**
  $x_{t+1} = \mathrm{argmin}_x \nabla f(x_t)^\top (x - x_t) + \frac{1}{2\eta}\|x_t - x\|^2 + h(x)$
end for
Return x_T

接勾配降下法は毎回の反復で $\min_x a(x-y)^2 + h(x)$ のような最小化問題を解く必要がある。これは単純な計算で達成できる場合、近接勾配降下法は勾配降下法とほぼ変わらない計算量で反復を行うことができる。以下にそのような例を挙げる。

$$\mathrm{argmin}\,\frac{1}{2}(x-y)^2 + \lambda|x| = \begin{cases} y-\lambda & \lambda < y \\ 0 & -\lambda \leq y \leq \lambda \\ y+\lambda & y < -\lambda \end{cases} \tag{30}$$

また、

$$\Omega_c(x) = \begin{cases} \infty & |x| > c \\ 0 & |x| \leq c \end{cases} \tag{31}$$

の時、

$$\mathrm{argmin}\,\frac{1}{2}(x-y)^2 + \Omega_c(x) = \begin{cases} c & c < y \\ y & -c \leq y \leq c \\ -c & y < -c \end{cases} \tag{32}$$

よって h により具体的な更新の式は変わる。上述のような簡単な式で書けず、最小化問題を別のアルゴリズムを用いて解かなければいけないこともある。そのような場合は近接勾配降下法はあまり適切ではない。

$h = 0$ の時、近接勾配降下法は勾配降下法と一致する。そもそも勾配降下法は L 平滑な関数の持つ上限関数 $f(x_t) + \nabla f(x_t)^\top (x - x_t) + \frac{1}{2\eta}\|x - x_t\|^2$ を利用して上限関数を各更新で最小化しているといえる。近接勾配降下法はこの上限関数を利用してやはり上限となっている関数

$$f(x_t) + \nabla f(x_t)^\top (x - x_t) + \frac{1}{2\eta}\|x - x_t\|^2 + h(x) \tag{33}$$

を最小化している。

理論的な結果としては以下の結果がある。つまり近接勾配降下法は勾配法と同様のコンプレキシティで線形収束する。

Theorem 9 ([Tayler et al.(2018)])**.** L-滑凸かつ μ-強凸である f と凸な h に対し、$\eta < \frac{2}{L+\mu}$ とした場合の勾配法のアルゴリズムが生成する点列 $\{x_t\}_{t=0,\ldots}$ に対し、

$$\|x_t - x^*\|^2 \leq \kappa\,(1 - \mu\eta)^{2t}\,\|x_0 - x^*\|^2 \tag{34}$$

が成り立つ。

2.4.2 確率的近接勾配降下法

問題が以下のように書けるとする。

$$F(x) = \frac{1}{n}\sum f_i(x) + h(x) \tag{35}$$

この場合確率的勾配法と同様な形で確率的近接勾配降下法を考えることができる。

Algorithm 5 確率的近接勾配降下法のアルゴリズム

Set x_0
for $t = 0, 1, \ldots, T-1$ **do**
  Sample i_t uniformly from $\{1, \ldots, n\}$.
  $x_{t+1} = \mathrm{argmin}_x \nabla f_{i_t}(x_t)^\top (x - x_t) + \frac{1}{2\eta}\|x_t - x\|^2 + h(x)$
end for
Return x_T

確率的近接勾配降下法も確率勾配降下法と同様のコンプレキシティを持つ。すなわち η を $O(\epsilon)$ に設定し、$O(\log\left(\frac{\|x_0 - x^*\|^2}{\epsilon}\right)(\kappa + \epsilon^{-1}))$ 回反復することで $\mathbb{E}x_t - x^* \leq \epsilon$ を得る。

2.4.3 確率的分散縮小近接勾配法（prox-SVRG）のアルゴリズム

分散縮小も h がない場合と同様にして行うことができる。この時アルゴリズムは線形収束し、かつ適当に T_{inner} をとれば反復のコンプレキシティは $O(\log(\frac{1}{\varepsilon}))$ となる。

Theorem 10 ([Xiao and Zhang (2014)])**.** 任意の L-滑かつ μ-強凸である f と、$x_0 \in \mathbb{R}^d$ があるとする。*Algorithm 2.4.3* は $\eta < \frac{1}{4L}$、T_{inner} に関し

$$\frac{1}{4\mu\eta T_{\mathrm{inner}}(1/4 - \eta L)} + \frac{\eta L}{1/4 - \eta L}\frac{T_{\mathrm{inner}} + 1}{T_{\mathrm{inner}}} < 1 \tag{36}$$

が成立する場合、左辺の値を ρ とし以下が成り立つ。

$$\mathbb{E}f(x_t) - f(x^*) \leq \rho^t\,(f(x_0) - f(x^*))\,. \tag{37}$$

参考文献

[Nesterov (2004)] Y. Nesterov, *"Introductory lectures on convex optimization: A Basic course."* Springer, 2004.

Algorithm 6 確率的分散縮小近接勾配法（prox-SVRG）のアルゴリズム

Set x_0
for $t = 0, 1, \ldots, T-1$ **do**
  $g_t = \frac{1}{n} \sum_i \nabla f_i(x_t)$
  for $s = 0, 1, \ldots, T_{\text{inner}} - 1$ **do**
    Sample i_s uniformly from $\{1, \ldots, n\}$.
    $g_{t,s} = g_t + \nabla f_{i_s}(x_{t,s}) - \nabla f_{i_s}(x_t)$
    $x_{t,s+1} = \operatorname{argmin} g_{t,s}^{\top}(x - x_{t,s}) + \frac{1}{2\eta} \|x_{t,s} - x\|^2 + h(x)$
  end for
  $x_{t+1} = T_{\text{inner}}^{-1} \sum_{s=1}^{T_{\text{inner}}} x_{t,s}$
end for
return x_T

[Nesterov (2012)] Y. Nesterov. *"Efficiency of coordinate descent methods on huge-scale optimization problems"* SIAM Journal on Optimization, 22(2), pp 341–362, 2012.

[Richtárik and Takáč (2014)] P. Richtarik and M. Takac. *"Iteration complexity of randomized block-coordinate descent methods for minimizing a composite function"* Mathematical Programming, 144(1-2), pp 1–38, 2014.

[Lu and Xiao (2015)] Zhaosong Lu and Lin Xiao. *"On the complexity analysis of randomized block-coordinate descent methods"* Mathematical Programming, 152(1), pp 615–642, 2015.

[Johnson and Zhang (2013)] R. Johnson, T. Zhang. *"Accelerating stochastic gradient descent using predictive variance reduction."* Advances in neural information processing systems, 2013.

[Needel et al.(2015)] D. Needel, N. Srebro, and R. Ward. *"Stochastic Gradient Descent, Weighted Sampling, and the Randomized Kaczmarz Algorithm"*. arXiv:1310.5715, 2015.

[Tayler et al.(2018)] Taylor, A. B., Hendrickx, J. M., and Glineur, F. *"Exact worst-case convergence rates of the proximal gradient method for composite convex minimization."* Journal of Optimization Theory and Applications, 178(2), pp. 455-476, 2018.

[Xiao and Zhang (2014)] Lin Xiao and Tong Zhang. *"A proximal stochastic gradient method with progressive variance reduction."* SIAM Journal on Optimization, 24(4), 2057-2075, 2014 .