机器学习中的优化方法

张华清

交叉信息研究院
清华大学

2024.08.06

目录

Table of Contents

参考资料

Sébastien Bubeck: Convex Optimization: Algorithms and Complexity 第 1 章、第 3 章。
Yurii Nesterov: Lectures on Convex Optimization 第 2 章。

优化

用一句话概括“优化”这个领域在做什么，大概就是“在可行集合上最小化目标函数”。用数学语言来描述，即为：

$$\begin{aligned} \min_{x} \quad & f(x) \\ \text{s.t.} \quad & x \in Q. \end{aligned}$$

优化

用一句话概括“优化”这个领域在做什么，大概就是“在可行集合上最小化目标函数”。用数学语言来描述，即为：

$$\min_{x} \quad f(x)$$
$$\text{s.t.} x \in Q.$$

本课程中，我们关注的是连续的优化问题，即变量 x 是连续的。

优化

用一句话概括“优化”这个领域在做什么，大概就是“在可行集合上最小化目标函数”。用数学语言来描述，即为：

$$\min_{x} \quad f(x)$$
$$\text{s.t.} x \in Q.$$

本课程中，我们关注的是连续的优化问题，即变量 x 是连续的。我们为什么要关注优化？只需举一个和课程主题相关的例子：考虑神经网络的训练，我们所做的只不过是求解损失函数的最小值。

优化

初看起来，解函数最小值不是什么难事——求导，令导数 $= 0$ 即可。然而，当 $f(x)$ 定义在高维空间时，事情便会开始棘手。

优化

初看起来，解函数最小值不是什么难事——求导，令导数 $= 0$ 即可。然而，当 $f(x)$ 定义在高维空间时，事情便会开始棘手。
几十年来，人们针对不同问题，提出了各种优化算法。

- 单纯形法
- 椭球法
- 内点法
- 梯度下降法
- ……

优化

初看起来，解函数最小值不是什么难事——求导，令导数 $= 0$ 即可。然而，当 $f(x)$ 定义在高维空间时，事情便会开始棘手。
几十年来，人们针对不同问题，提出了各种优化算法。

- 单纯形法
- 椭球法
- 内点法
- 梯度下降法
- ……

课程简介

本课程中，我们将主要介绍以梯度下降法为代表的一阶算法。人工智能模型训练所用到的优化方法几乎都为一阶算法。（可以想象，对于一个有 *100B* 参数的神经网络，其损失函数定义在 *100B* 维空间上，计算/存储 *Hessian* 矩阵是不可接受的。）

课程简介

本课程中，我们将主要介绍以梯度下降法为代表的一阶算法。人工智能模型训练所用到的优化方法几乎都为一阶算法。（可以想象，对于一个有 *100B* 参数的神经网络，其损失函数定义在 *100B* 维空间上，计算/存储 *Hessian* 矩阵是不可接受的。）
我们将从理论上证明三种一阶方法（梯度下降法、随机梯度下降法、条件梯度法）的收敛速度，并将简要介绍复杂度下界的概念和证明方法。

课程简介

本课程中，我们将主要介绍以梯度下降法为代表的一阶算法。人工智能模型训练所用到的优化方法几乎都为一阶算法。（可以想象，对于一个有 *100B* 参数的神经网络，其损失函数定义在 *100B* 维空间上，计算/存储 *Hessian* 矩阵是不可接受的。）
我们将从理论上证明三种一阶方法（梯度下降法、随机梯度下降法、条件梯度法）的收敛速度，并将简要介绍复杂度下界的概念和证明方法。
让我们从梯度下降法开始。

梯度下降法

梯度下降法 (gradient descent, GD) 大概是最简单也最重要的一阶优化算法，是很多其他一阶算法的母体，如解决带限制问题的投影梯度下降法 (projected gradient descent)，Nesterov 加速梯度下降法，以及应用于神经网络训练的随机梯度下降法（stochastic gradient descent, SGD）等。

梯度下降法

梯度下降法 (gradient descent, GD) 大概是最简单也最重要的一阶优化算法，是很多其他一阶算法的母体，如解决带限制问题的投影梯度下降法 (projected gradient descent)，Nesterov 加速梯度下降法，以及应用于神经网络训练的随机梯度下降法（stochastic gradient descent, SGD）等。
该算法要追溯到柯西于 1847 年的工作，但对其复杂度的分析则是近百年间的事。

梯度下降法

梯度下降法 (gradient descent, GD) 大概是最简单也最重要的一阶优化算法，是很多其他一阶算法的母体，如解决带限制问题的投影梯度下降法 (projected gradient descent)，Nesterov 加速梯度下降法，以及应用于神经网络训练的随机梯度下降法（stochastic gradient descent, SGD）等。
该算法要追溯到柯西于 1847 年的工作，但对其复杂度的分析则是近百年间的事。

梯度下降法

我们已很熟悉梯度下降法的流程：从初始点 x_0 出发，每次沿着梯度的反方向走一小步。用数学语言描述，即为：

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

其中 $\eta > 0$ 被称为学习率。

梯度下降法

我们已很熟悉梯度下降法的流程：从初始点 x_0 出发，每次沿着梯度的反方向走一小步。用数学语言描述，即为：

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

其中 $\eta > 0$ 被称为学习率。
那么，是否可能从理论角度，分析梯度下降法的复杂度？

梯度下降法

可以想象，没有算法能高效地对任意函数求最小值。我们需要对目标函数做出一些假设。比如，我们至少希望目标函数是连续的。其他常见假设包括可导性、(强) 凸性、Lipschitz 连续性、光滑性等等。下面，我们来简要介绍一下这些假设。

Table of Contents

凸性

定义 (凸集合)

我们称一个集合 $\mathcal{X} \subset \mathbb{R}^n$ 是凸的，如果对于任意两点 $x, y \in \mathcal{X}$ 及任意 $\gamma \in [0, 1]$，有

$$\gamma x + (1 - \gamma) \in \mathcal{X}$$

凸性

定义 (凸集合)

我们称一个集合 $\mathcal{X} \subset \mathbb{R}^n$ 是凸的，如果对于任意两点 $x, y \in \mathcal{X}$ 及任意 $\gamma \in [0,1]$，有

$$\gamma x + (1-\gamma) \in \mathcal{X}$$

定义 (凸函数)

我们称一个函数 $f\colon \mathcal{X} \to \mathbb{R}$ 是凸的，如果其定义域 $\mathcal{X}$ 是凸的，并且对于定义域内任意两点 $x, y \in \mathcal{X}$ 及任意 $\gamma \in [0,1]$，有

$$f((1-\gamma)x + \gamma y) \leq (1-\gamma)f(x) + \gamma f(y)$$

凸性

命题 (凸函数的等价定义)

对于定义在凸集合 $\mathcal{X} \subset \mathbb{R}^n$ 上的函数 $f: \mathcal{X} \to \mathbb{R}$,

- 若 f 可导，则 f 是凸函数当且仅当

$$f(y) \geq f(x) + \langle \nabla f(x), y - x \rangle, \forall x, y \in \mathcal{X}$$

凸性

命题 (凸函数的等价定义)

对于定义在凸集合 $\mathcal{X} \subset \mathbb{R}^n$ 上的函数 $f: \mathcal{X} \to \mathbb{R}$,

- 若 f 可导，则 f 是凸函数当且仅当

$$f(y) \geq f(x) + \langle \nabla f(x), y - x \rangle, \forall x, y \in \mathcal{X}$$

- 若 f 二阶可导，则 f 是凸函数当且仅当 f 的 *Hessian* 矩阵半正定。

Remark

$f(x) + \langle \nabla f(x), y - x \rangle$ 是在 x 处对 $f(\cdot)$ 的线性估计（下界）。

凸性

命题（凸函数的等价定义）

对于定义在凸集合 $\mathcal{X} \subset \mathbb{R}^n$ 上的函数 $f: \mathcal{X} \to \mathbb{R}$，

- 若 f 可导，则 f 是凸函数当且仅当

$$f(y) \geq f(x) + \langle \nabla f(x), y - x \rangle, \forall x, y \in \mathcal{X}$$

- 若 f 二阶可导，则 f 是凸函数当且仅当 f 的 *Hessian* 矩阵半正定。
- （琴生不等式）$f(\mathbb{E}x) \leq \mathbb{E}f(x)$

Remark

$f(x) + \langle \nabla f(x), y - x \rangle$ 是在 x 处对 $f(\cdot)$ 的线性估计（下界）。

凸性

命题 (凸函数的性质)

对于定义在凸集合 $\mathcal{X} \subset \mathbb{R}^n$ 上的凸函数 $f\colon \mathcal{X} \to \mathbb{R}$,

- f 的下水平集 *(sublevel set)* $\{x|f(x) \leq a\}$, $a \in \mathbb{R}$ 是凸集合。特别地，f 的全局最小值点构成的集合 $\arg\min f$ 是凸的。

凸性

命题 (凸函数的性质)

对于定义在凸集合 $\mathcal{X} \subset \mathbb{R}^n$ 上的凸函数 $f\colon \mathcal{X} \to \mathbb{R}$,

- f 的下水平集 *(sublevel set)* $\{x|f(x) \leq a\}$, $a \in \mathbb{R}$ 是凸集合。特别地，f 的全局最小值点构成的集合 $\arg\min f$ 是凸的。
- f 的任何局部最小值点都是全局最小值点。

凸性

命题（凸函数的性质）

对于定义在凸集合 $\mathcal{X} \subset \mathbb{R}^n$ 上的凸函数 $f\colon \mathcal{X} \to \mathbb{R}$，

- f 的下水平集 *(sublevel set)* $\{x|f(x) \le a\}$，$a \in \mathbb{R}$ 是凸集合。特别地，f 的全局最小值点构成的集合 $\arg\min f$ 是凸的。
- f 的任何局部最小值点都是全局最小值点。

可以想象，局部最小值点推出全局最小值点的性质很有帮助——大多数一阶算法只能用到函数局部的信息（摸黑下山）。事实上，对于非凸函数，即使求解局部最小值点已是 NP-难问题。

凸性

机器学习中的很多问题都是凸的，如

- 线性回归
- 逻辑斯蒂回归
- 支持向量机

[1]图片来源：https://www.cs.umd.edu/%7Etomg/projects/landscapes/

凸性

机器学习中的很多问题都是凸的，如

- 线性回归
- 逻辑斯蒂回归
- 支持向量机

然而，现代神经网络是非凸的。

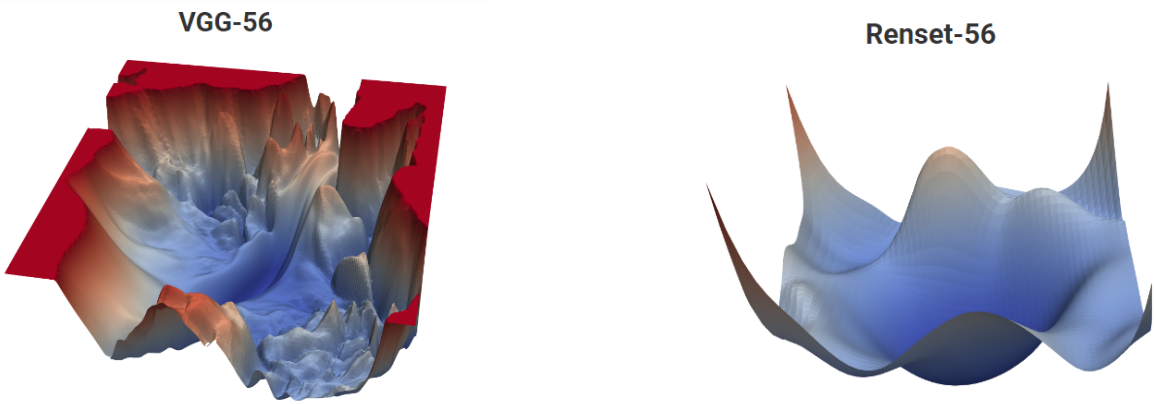
[1]图片来源：https://www.cs.umd.edu/%7Etomg/projects/landscapes/

凸性

机器学习中的很多问题都是凸的，如

- 线性回归
- 逻辑斯蒂回归
- 支持向量机

然而，现代神经网络是非凸的。



图：神经网络的 landscape[1]

[1]图片来源：https://www.cs.umd.edu/%7Etomg/projects/landscapes/

强凸性

定义 (强凸函数)

我们称一个可导函数 $f\colon \mathcal{X} \to \mathbb{R}$ 是 μ-强凸的，如果其定义域 $\mathcal{X}$ 是凸的，并且对于定义域内任意两点 $x, y \in \mathcal{X}$ 及任意 $\gamma \in [0, 1]$，有

$$f(y) \geq f(x) + \langle \nabla f(x), y - x \rangle + \frac{\mu}{2}\|y - 2\|^2$$

强凸性

定义 (强凸函数)

我们称一个可导函数 $f: \mathcal{X} \to \mathbb{R}$ 是 μ-强凸的，如果其定义域 $\mathcal{X}$ 是凸的，并且对于定义域内任意两点 $x, y \in \mathcal{X}$ 及任意 $\gamma \in [0, 1]$，有

$$f(y) \geq f(x) + \langle \nabla f(x), y - x \rangle + \frac{\mu}{2} \|y - 2\|^2$$

Remark

$f(x) + \langle \nabla f(x), y - x \rangle + \frac{\mu}{2} \|y - 2\|^2$ 是在 x 处对 $f(\cdot)$ 的二次估计（下界）。

强凸性

命题 (强凸函数的等价定义)

- f 为 μ-强凸等价于 $f(x) - \frac{\mu}{2}\|x\|^2$ 是凸函数。

强凸性

命题 (强凸函数的等价定义)

- f 为 μ-强凸等价于 $f(x) - \frac{\mu}{2}\|x\|^2$ 是凸函数。
- 若 f 二阶可导，则 f 是凸函数当且仅当 f 的 *Hessian* 矩阵 $\nabla^2 f \succeq \mu I$。

强凸性

命题 (强凸函数的等价定义)

- f 为 μ-强凸等价于 $f(x) - \frac{\mu}{2}\|x\|^2$ 是凸函数。
- 若 f 二阶可导，则 f 是凸函数当且仅当 f 的 *Hessian* 矩阵 $\nabla^2 f \succeq \mu I$。

命题 (强凸函数的性质)

强凸函数有唯一的全局最小值点。

Lipschitz 连续性

定义

我们称一个函数 $f: \mathcal{X} \subset \mathbb{R}^n \to \mathbb{R}$ 是 M-Lipschitz 连续的，如果

$$|f(x) - f(y)| \leq M\|x - y\|, \forall x, y \in \mathcal{X}$$

光滑性

定义

我们称一个可导函数 $f: \mathcal{X} \subset \mathbb{R}^n \to \mathbb{R}$ 是 L-光滑的，如果

$$\|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\|, \forall x, y \in \mathcal{X}$$

光滑性

定义

我们称一个可导函数 $f: \mathcal{X} \subset \mathbb{R}^n \to \mathbb{R}$ 是 L-光滑的，如果

$$\|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\|, \forall x, y \in \mathcal{X}$$

命题 (光滑函数的性质)

若 $f: \chi \subset \mathbb{R}^n \to \mathbb{R}$ 是 L-光滑的，则

$$f(x)+\langle \nabla f(x), y-x\rangle - \frac{L}{2}\|y-x\|^2 \leq f(y) \leq f(x)+\langle \nabla f(x), y-x\rangle + \frac{L}{2}\|y-x\|^2$$

Remark

$f(x) + \langle \nabla f(x), y - x \rangle \pm \frac{L}{2}\|y - 2\|^2$ 是在 x 处对 $f(\cdot)$ 的二次估计（上/下界）。

Remark

$f(x) + \langle \nabla f(x), y - x \rangle \pm \frac{L}{2}\|y - 2\|^2$ 是在 x 处对 $f(\cdot)$ 的二次估计（上/下界）。

直观上，一阶算法是用梯度对函数做估计。所以我们希望函数的梯度变化不太大——光滑性即描述了这件事。

光滑性

当 f 同时满足凸性和光滑性时，我们能得到 f 的更紧的一个下界

引理 (1)

若 f 是凸函数，且满足 L-光滑，则

$$f(y) \geq f(x) + \langle \nabla f(x), y - x \rangle + \frac{1}{2L} \|\nabla f(y) - \nabla f(x)\|^2$$

光滑性

证明.

取 $z = y - \frac{1}{L}(\nabla f(y) - \nabla f(x))$。则

$$
\begin{aligned}
& f(x) - f(y) \\
=& f(x) - f(z) + f(z) - f(y \\
\leq& \langle \nabla f(x), x - z \rangle + \langle \nabla f(y), z - y \rangle + \frac{\beta}{2}\|z - y\|^2 \\
=& \langle \nabla f(x), x - y \rangle + \langle \nabla f(x) - \nabla f(y), y - z \rangle + \frac{1}{2L}\|\nabla f(y) - \nabla f(x)\|^2 \\
=& \langle \nabla f(x), x - y \rangle - \frac{1}{2L}\|\nabla f(y) - \nabla f(x)\|^2
\end{aligned}
$$

光滑性

引理

$$\langle \nabla f(x) - \nabla f(y), x - y \rangle \geq \frac{1}{L} \|\nabla f(x) - \nabla f(y)\|^2$$

证明.

上一引理交换 x, y 后相加即得证。 □

Table of Contents

梯度下降法

我们已做好了充足准备！下面，我们从理论角度对梯度下降法地收敛速度进行分析。

梯度下降法

我们已做好了充足准备！下面，我们从理论角度对梯度下降法地收敛速度进行分析。
事实上，对于不同的函数类（即，对于目标函数做出的假设），梯度下降法会给出不同的收敛速度，部分结果如下表所示：

表：梯度下降法的收敛速度

函数类	收敛到	收敛速度
凸，M-Lip 连续	最小值点	$f(x_T)-f(x^*)=O(\frac{M\|x^*-x_0\|}{\sqrt{T}})$
凸，L-光滑	最小值点	$f(x_T)-f(x^*)=O(\frac{L\|x^*-x_0\|^2}{T})$
μ-强凸，L-光滑	最小值点	$\|x_T-x^*\|^2\le\exp(-\frac{t}{L/\mu})\|x^*-x_0\|^2$
非凸，L-光滑	一阶稳定点	$\|\nabla f(x)\|\le O\left(\sqrt{\frac{L}{T}(f(w_0)-f(w^*)}\right)$

梯度下降法

下面，我们以“凸且光滑”为例进行证明。部分其他情形留作作业。我们记 x^* 为 f 的最小值点，$f(x^*)$ 是 f 的最小值。

定理 (梯度下降法)

假设定义在 $\mathbb{R}^n$ 上的目标函数 f 满足凸性，且为 L-光滑。考虑如下的迭代法 $x_{k+1} = x_k - \eta \nabla f(x_k)$。当 $\eta = \frac{1}{L}$ 时，我们有：

$$f(x_T) - f^* \leq \frac{2\beta \|x_0 - x^*\|^2}{T+1}$$

梯度下降法

下面，我们以“凸且光滑”为例进行证明。部分其他情形留作作业。我们记 x^* 为 f 的最小值点，$f(x^*)$ 是 f 的最小值。

定理 (梯度下降法)

假设定义在 $\mathbb{R}^n$ 上的目标函数 f 满足凸性，且为 L-光滑。考虑如下的迭代法 $x_{k+1} = x_k - \eta \nabla f(x_k)$。当 $\eta = \frac{1}{L}$ 时，我们有：

$$f(x_T) - f^* \leq \frac{2\beta \|x_0 - x^*\|^2}{T+1}$$

Remark

注意梯度下降法适用于无限制的优化问题。若要求凸集合 $\mathcal{X}$ 上的最小值 (或 f 的定义域即为 $\mathcal{X}$)，可以使用投影梯度下降法：只需在 gradient step 后，向 $\mathcal{X}$ 做投影，即 $x_{k+1} = \Pi_{\mathcal{X}}(x_k - \eta \nabla f(x_k))$。收敛速度不变。

梯度下降法

证明.

步骤 1：对一步迭代的分析。

$$\begin{aligned} f(x_{k+1}) - f(x_k) &= f(x_k - \frac{1}{L}\nabla f(x_k)) - f(x_k) \\ &\leq \langle \nabla f(x_k), -\frac{1}{L}\nabla f(x_k) \rangle + \frac{L}{2}\|\frac{1}{L}\nabla f(x_k)\|^2 \\ &= -\frac{1}{2L}\|\nabla f(x_k)\|^2 \end{aligned}$$

梯度下降法

证明.

步骤 1：对一步迭代的分析。

$$\begin{aligned} f(x_{k+1}) - f(x_k) &= f(x_k - \frac{1}{L}\nabla f(x_k)) - f(x_k) \\ &\leq \langle \nabla f(x_k), -\frac{1}{L}\nabla f(x_k)\rangle + \frac{L}{2}\|\frac{1}{L}\nabla f(x_k)\|^2 \\ &= -\frac{1}{2L}\|\nabla f(x_k)\|^2 \end{aligned}$$

步骤 2：对 suboptimality 的分析：考虑 $\delta_k = f(x_k) - f^*$。有

$$\delta_{k+1} - \delta_k \leq -\frac{1}{2L}\|\nabla f(x_k)\|^2.$$

梯度下降法

证明 (Cont'd).

步骤 3：用 δ 来 lower bound $\|\nabla f(x_k)\|^2$。直观上，$\|\nabla f(x_k)\|$ 不会太小，否则 δ_k 会很小。

$$\delta_k \leq \langle x_k - x^*, \nabla f(x^*) \rangle \leq \|x_k - x^*\| \|\nabla f(x_k)\|$$

则

$$\|\nabla f(x_k)\| \geq \frac{\delta_k}{\|x_k - x^*\|}$$

梯度下降法

证明 (Cont'd).

步骤 4：证明 $\|x_k - x^*\| \le \|x_0 - x^*\|$，**则** $\|\nabla f(x_k)\| \ge \frac{\delta_k}{\|x_k - x^*\|} \ge \frac{\delta_k}{\|x_0 - x^*\|}$。

$$\begin{aligned}\|x_{k+1} - x^*\|^2 &= \|x_k - \frac{1}{L}\nabla f(x_k) - x^*\|^2 \\ &= \|x_k - x^*\|^2 - \frac{2}{L}\langle x_k - x^*, \nabla f(x_k)\rangle + \frac{1}{L^2}\|\nabla f(x_k)\|^2\end{aligned}$$

下面利用之前证明的引理：
$\langle \nabla f(x) - \nabla f(y), x - y\rangle \ge \frac{1}{L}\|\nabla f(x) - \nabla f(y)\|^2$。**带入** $x = x_k, y = x^*$，**注意到** $\nabla f(x^*) = \mathbf{0}$，**我们有：**

$$\langle x_k - x^*, \nabla f(x_k)\rangle \ge \frac{1}{L}\|\nabla f(x_k)\|^2$$

梯度下降法

证明 (Cont'd).

则 $\|x_{k+1} - x_0\|^2 \leq \|x_k - x^*\|^2 - \frac{1}{L^2}\|\nabla f(x_k)\|^2 \leq \|x_k - x^*\|^2$。归纳即得 $\|x_k - x^*\|^2 \leq \|x_0 - x^*\|^2$。

步骤 5：解关于 δ_k 的递推式。结合步骤 2，3，4，我们有：

$$\delta_{k+1} \leq \delta_k - \frac{1}{2L\|x_0 - x^*\|^2}\delta_k^2$$

则 $\frac{1}{\delta_k} \leq \frac{1}{\delta_{k+1}} - \frac{1}{2L\|x_0 - x^*\|^2}\frac{\delta_k}{\delta_{k+1}} \leq \frac{1}{\delta_{k+1}} - \frac{1}{2L\|x_0 - x^*\|^2}$。累加推出

$$\delta_T \leq \frac{2L\|x_0 - x^*\|^2}{T}$$

□

Polyak 动量法与 Nesterov 加速算法

何时梯度下降法的表现会不好？考虑下图情况，梯度下降给出的方向与到最小值点的方向相差甚远。

[2]图片来源：Princeton ELE522。也可以玩https://distill.pub/2017/momentum/

Polyak 动量法与 Nesterov 加速算法

何时梯度下降法的表现会不好？考虑下图情况，梯度下降给出的方向与到最小值点的方向相差甚远。借鉴物理上动量的概念：一个大质量的小球在滚落时会具有惯性/动量，进而抗拒速度的改变。我们在迭代式中加上一个动量项 $\beta\|x_k - x_{k-1}\|$，其中 β 通常取 $(0, 1]$ 间常数：

$$x_{k+1} = x_k - \eta\nabla f(x_k) + \beta\|x_k - x_{k-1}\|$$

图：梯度下降法与动量法[2]

[2]图片来源：Princeton ELE522。也可以玩https://distill.pub/2017/momentum/

Polyak 动量法与 Nesterov 加速算法

Polyak's momentum method 能在某些情况下能起到加速作用，然后其收敛性**没有**保证。
1989 年，Nesterov 在动量法的基础上提出了 Nesterov 加速算法。该算法对于凸且光滑的函数类的收敛速度为 $O(\frac{1}{T^2})$，达到理论下界。其迭代式为：

$$x_{k+1} = (x_k + \beta_{k-1}(x_k - x_{k-1})) - \frac{1}{L}\nabla f(x_k + \beta_{k-1}(x_k - x_{k-1}))$$

该算法复杂度证明较复杂，留作作业。

Table of Contents

随机梯度下降法

在机器学习问题中，损失函数往往形如 $L = \frac{1}{N}\sum_{i=1}^{N} l(\theta, x_i)$，其中 θ 是模型的参数，x_i 是第 i 个数据点。当训练集大小 N 很大时，计算全梯度 ∇L 时的开销巨大，无法接受。

随机梯度下降法

在机器学习问题中，损失函数往往形如 $L = \frac{1}{N}\sum_{i=1}^{N} l(\theta, x_i)$，其中 θ 是模型的参数，x_i 是第 i 个数据点。当训练集大小 N 很大时，计算全梯度 ∇L 时的开销巨大，无法接受。
因此，每次迭代时，人们常从训练集中采样一个子集（batch）S，用 $\frac{1}{|S|}\sum_{i\in S}\nabla l(\theta, x_i)$ 作为梯度的（无偏）估计。这时，原本的梯度下降法即变为随机梯度下降法：

$$x_{k+1} = x_k - \eta G_k$$

其中 $\mathbb{E}[G_k] = \nabla f(x_k)$。我们进一步假设 $Var(G_k) \leq \sigma^2$。

随机梯度下降法

定理

假设定义在 $\mathbb{R}^n$ 上的目标函数 f 满足凸性，且为 L-光滑。考虑如下的迭代法 $x_{k+1} = x_k - \eta \nabla G_k$，其中 $\mathbb{E}[G_k] = \nabla f(x_k)$，$Var(G_k) \leq \sigma^2$，$\eta = \frac{1}{L}$。我们有：

$$\mathbb{E}[f(\bar{x}_T)] \leq f^* + \frac{\|x_0 - x^*\|^2}{2T\eta} + \eta\sigma^2$$

其中 $\bar{x}_T = \frac{\sum_{i=1}^{T} x_i}{T}$

随机梯度下降法

证明.

步骤 1：对一步迭代的分析——期望意义上每次迭代都进步。

$$
\begin{aligned}
\mathbb{E}[f(x_{k+1})] &\leq f(x_k) + \mathbb{E}\langle \nabla f(x_k), x_{k+1} - x_k \rangle + \mathbb{E}\left[\frac{L}{2}\|x_{k+1} - x_k\|^2\right] \\
&= f(x_k) + \langle \nabla f(x_k), -\eta \nabla f(x_k) \rangle + \frac{L\eta^2}{2}\mathbb{E}\|G_k\|^2 \\
&\leq f(x_k) - \eta\|\nabla f(x_k)\|^2 + \frac{L\eta^2}{2}(\|\nabla f(x_k)\|^2 + Var(G_k)) \\
&\leq f(x_k) - \eta(1 - \frac{L\eta}{2})\|f(x_k)\|^2 + \frac{L\eta^2}{2}\sigma^2 \\
&\leq f(x_k) - \frac{\eta}{2}\|f(x_k)\|^2 + \frac{\eta}{2}\sigma^2
\end{aligned}
$$

随机梯度下降法

证明 (Cont'd).

步骤 2：联系 $f(x_k) - f(x^*)$ 与 $\|x_k - x^*\|$

$$\begin{aligned}
\mathbb{E}f(x_{k+1}) - f(x^*) &\le f(x_k) - f^* - \frac{\eta}{2}\|\nabla f(x_k)\|^2 + \frac{\eta}{2}\sigma^2 \\
&\le \langle \nabla f(x_k), x_k - x^* \rangle - \frac{\eta}{2}\|\nabla f(x_k)\|^2 + \frac{\eta}{2}\sigma^2 \text{ (convexity)} \\
&\le \mathbb{E}\langle G_k, x_k - x^* \rangle - \frac{\eta}{2}(\mathbb{E}\|G_k\|^2 - \sigma^2) + \frac{\eta}{2}\sigma^2 \ (\mathbb{E}G_k = \nabla f(x_k)) \\
&= \mathbb{E}\left[\langle G_k, x_k - x^* \rangle - \frac{\eta}{2}\|G_k\|^2\right] + \eta\sigma^2
\end{aligned}$$

随机梯度下降法

证明 (Cont'd).

配方:

$$
\begin{aligned}
&\langle G_k, x_k - x^* \rangle - \frac{\eta}{2}\|G_k\|^2 \\
=&\frac{1}{\eta}\langle x_k - x_{k+1}, x_k - x^* \rangle - \frac{1}{2\eta}\|x_k - x_{k+1}\|^2 \\
=&\frac{1}{2\eta}(2\langle x_k - x_{k+1}, x_k - x^* \rangle - \|x_k - x_{k+1}\|^2 - \|x_k - x^*\|^2 + \|x_k - x^*\|^2) \\
=&\frac{1}{2\eta}(\|x_k - x^*\|^2 - \|x_{k+1} - x^*\|^2)
\end{aligned}
$$

随机梯度下降法

证明 (Cont'd).

带入整理:

$$\mathbb{E}f(x_{k+1}) - f(x^*) \leq \frac{1}{2\eta}\mathbb{E}(\|x_k - x^*\|^2 - \|x_{k+1} - x^*\|^2) + \eta\sigma^2$$

步骤 3: Telescoping（**累加**）。

$$\begin{aligned}\frac{1}{T}\sum_{k=0}^{T-1}(\mathbb{E}f(x_{k+1}) - f(x^*)) &\leq \frac{1}{2T\eta}(\|x_0 - x^*\|^2 - \mathbb{E}\|x_T - x^*\|) + \eta\sigma^2 \\ &\leq \frac{1}{2T\eta}\|x_0 - x^*\|^2 + \eta\sigma^2\end{aligned}$$

应用琴生不等式，即得 $\mathbb{E}[f(\bar{x}_T)] \leq f^* + \frac{\|x_0 - x^*\|^2}{2T\eta} + \eta\sigma^2$。 □

随机梯度下降法

推论

当 $\frac{\|x_0-x^*\|}{\sqrt{2T}\sigma} \leq \frac{1}{L}$ 时，设 $\eta = \frac{\|x_0-x^*\|}{\sqrt{2T}\sigma}$，有

$$\mathbb{E}[f(\bar{x}_T)] \leq f^* + \frac{\sqrt{2}\|x_0 - x^*\|\sigma}{\sqrt{T}}$$

随机梯度下降法

推论

当 $\frac{\|x_0-x^*\|}{\sqrt{2T}\sigma} \leq \frac{1}{L}$ 时，设 $\eta = \frac{\|x_0-x^*\|}{\sqrt{2T}\sigma}$，有

$$\mathbb{E}[f(\bar{x}_T)] \leq f^* + \frac{\sqrt{2}\|x_0 - x^*\|\sigma}{\sqrt{T}}$$

对于凸且光滑的函数类，梯度下降法的收敛速度为 $\frac{1}{T}$，而随机梯度下降法则为 $\frac{1}{\sqrt{T}}$。从推导中可看出，根号来自于平衡噪声的方差。有一类被称为方差缩减（variance reduction）的技术，可使 SGD 的收敛速度与 GD 一样快，如 SAG,SAGA,SVRG 等。

随机梯度下降法的优势

最大的优势当然是减小计算全梯度所需要的巨大代价。除此之外，随机梯度下降法还有一些额外好处。

随机梯度下降法的优势

最大的优势当然是减小计算全梯度所需要的巨大代价。除此之外，随机梯度下降法还有一些额外好处。

- 非凸优化中，噪声会带来一些帮助：SGD 可以跳出局部最小值点或鞍点，而 GD 不能。这导致 SGD 可能求解出比 GD 更好的解。
- 神经网络 +SGD 是 “universal learner”，而神经网络 +GD 不是。

Table of Contents

条件梯度法

在之前的讨论中，我们默认自变量的取值范围是 $\mathbb{R}^d$；一个自然的问题是，如果我们对自变量的取值范围加以限制，该如何处理？

前面我们已简要减少了投影梯度法。接下来，我们介绍与梯度下降法稍显不同的条件梯度法（Conditional Gradient Method/Frank-Wolfe Method）。

问题设定

我们希望解决如下优化问题：

$$\min_{x \in Q} \quad (x)$$

其中 $Q \subset \mathbb{R}^n$ 是一个凸而紧的集合（紧 $=$ 闭 $+$ 有界），f 是一个 L-光滑的可微凸函数

算法

- 取步长 $\{\gamma_k = \frac{2}{k+2}\}_k$。
- 选取任意初始点 $x_0 \in Q$。
- 迭代 $k = 0 \dots T-1$:
  - 计算 $s_k = \arg\min_{s \in Q} \langle \nabla f(x_k), s \rangle$
  - 计算 $x_{k+1} = (1-\gamma_K)x_k + \gamma_k s_k$

Intuition: 计算 $s_k = \arg\min_{s \in Q} \langle \nabla f(x_k), s \rangle$，等价于在 Q 上优化 x_k 处对 f 的线性估计 $f(x_k) + \langle \nabla f(x_k), s - x_k \rangle$。然后往该线性近似问题的最优解方向走一小步，并且步长越来越小。

收敛复杂度

定理

假设目标函数 f 满足凸性，且为 L-光滑，并假设可行集合 Q 为凸而紧的集合。设 Q 的直径 $\max_{x,y\in Q}\|x-y\| = D$。考虑如上描述的条件梯度法，我们有：

$$f(x_T) - f(x^*) \leq \frac{2LD^2}{K+1}$$

其中 $x^* = \arg\min_{x\in Q} f(x)$。

条件梯度法

证明.

步骤 1：对一步迭代的分析：证明
$f(x_{k+1}) - f(x^*) \leq (1-\gamma_k)(f(x_k) - f(x^*)) + \frac{1}{2}\gamma_k^2 L D^2$。

条件梯度法

证明.

步骤 1：对一步迭代的分析：证明
$f(x_{k+1}) - f(x^*) \leq (1-\gamma_k)(f(x_k) - f(x^*)) + \frac{1}{2}\gamma_k^2 L D^2$。
根据光滑性，我们有

$$f(x_{k+1}) \leq f(x_k) + \langle \nabla f(x_k), x_{k+1} - x_k \rangle + \frac{1}{2} L \|x_{k+1} - x_k\|^2$$

条件梯度法

证明.

步骤 1：对一步迭代的分析：证明
$f(x_{k+1}) - f(x^*) \leq (1-\gamma_k)(f(x_k) - f(x^*)) + \frac{1}{2}\gamma_k^2 L D^2$。
根据光滑性，我们有

$$f(x_{k+1}) \leq f(x_k) + \langle \nabla f(x_k), x_{k+1} - x_k \rangle + \frac{1}{2} L \|x_{k+1} - x_k\|^2$$

而 $x_{k+1} = x_k + \gamma_k(s_k - x_k)$，所以：

$$\begin{aligned} f(x_{k+1}) &\leq f(x_k) + \gamma_k \langle \nabla f(x_k), s_k - x_k \rangle + \frac{1}{2}\gamma_k^2 L \|s_k - x_k\|^2 \\ &\leq f(x_k) + \gamma_k \langle \nabla f(x_k), s_k - x_k \rangle + \frac{1}{2}\gamma_k^2 L D^2. \end{aligned}$$

条件梯度法

证明 (Cont'd).

下面，我们该利用 $s_k = \arg\min_{s\in Q}\langle \nabla f(x_k), s\rangle = \arg\min_{s\in Q}\langle \nabla f(x_k), s - x_k\rangle$ 了。注意 s_k 是这个线性优化问题的最优解，我们带入 $s = x^*$，并用一下凸性：

$$\langle \nabla f(x_k), s - x_k\rangle \leq \langle \nabla f(x_k), x^* - x_k\rangle \leq f(x^*) - f(x_k)$$

条件梯度法

证明 (Cont'd).

下面，我们该利用 $s_k = \arg\min_{s\in Q}\langle \nabla f(x_k), s\rangle = \arg\min_{s\in Q}\langle \nabla f(x_k), s - x_k\rangle$ 了。注意 s_k 是这个线性优化问题的最优解，我们带入 $s = x^*$，并用一下凸性：

$$\langle \nabla f(x_k), s - x_k\rangle \le \langle \nabla f(x_k), x^* - x_k\rangle \le f(x^*) - f(x_k)$$

带入上式：

$$f(x_{k+1}) \le f(x_k) + \gamma_k(f(x^*) - f(x_k)) + \frac{1}{2}\gamma_k^2 LD^2$$

整理即得 $f(x_{k+1}) - f(x^*) \le (1-\gamma_k)(f(x_k) - f(x^*)) + \frac{1}{2}\gamma_k^2 LD^2$。

条件梯度法

证明 (Cont'd).

第二步:

定义 $A_k = k(k+1)$。**带入** $\gamma_k = \frac{2}{k+2}$ **后，两边同乘** A_{k+1}，**得到**

$$\begin{aligned} A_{k+1}(f(x_{k+1} - f(x^*)) &\leq A_k(f(x_k) - f(x^*)) + \frac{2(k+1)}{k+2}LD^2 \\ &\leq A_k(f(x_k) - f(x^*)) + 2LD^2 \end{aligned}$$

条件梯度法

证明 (Cont'd).

第二步：
定义 $A_k = k(k+1)$。带入 $\gamma_k = \frac{2}{k+2}$ 后，两边同乘 A_{k+1}，得到

$$\begin{aligned} A_{k+1}(f(x_{k+1} - f(x^*)) &\leq A_k(f(x_k) - f(x^*)) + \frac{2(k+1)}{k+2} LD^2 \\ &\leq A_k(f(x_k) - f(x^*)) + 2LD^2 \end{aligned}$$

Telescoping：

$$A_K(f(x_K) - f(x^*)) \leq A_0(f(x_0) - f(x^*)) + 2KLD^2 = 2KLD^2$$

整理即得： $f(x_K) - f(x^*) \leq \frac{2LD^2}{K+1}$

Table of Contents

下界

在优化理论中，我们不仅想对某类问题提出更高效的算法（即，更好的复杂度上界），同时也希望能证明复杂度下界（即，任何算法都至少需要怎样的迭代次数，才能解决某类问题）。

下界

在优化理论中，我们不仅想对某类问题提出更高效的算法（即，更好的复杂度上界），同时也希望能证明复杂度下界（即，任何算法都至少需要怎样的迭代次数，才能解决某类问题）。
为研究复杂度下界，我们不仅需要定义问题（函数类），还需要对研究的算法类做出限定。下面，我们给出一个对一阶算法的假设。绝大多数一阶算法都满足此假设。

zero-respecting algorithms

定义

我们称一个迭代算法是 zero-respecting 的一阶算法，如果其生成的迭代序列 $\{x_k\}_{k\geq 0}$ 满足：

$$x_k \in x_0 + \mathrm{Span}(\nabla f(x_0), \nabla f(x_1), \cdots, \nabla f(x_{k-1}))$$

zero-respecting algorithms

定义

我们称一个迭代算法是 zero-respecting 的一阶算法，如果其生成的迭代序列 $\{x_k\}_{k\geq 0}$ 满足：

$$x_k \in x_0 + \mathrm{Span}(\nabla f(x_0), \nabla f(x_1), \cdots, \nabla f(x_{k-1}))$$

GD,momentum method 都满足此假设。

zero-respecting algorithms

定义

我们称一个迭代算法是 zero-respecting 的一阶算法，如果其生成的迭代序列 $\{x_k\}_{k\geq 0}$ 满足：

$$x_k \in x_0 + \mathrm{Span}(\nabla f(x_0), \nabla f(x_1), \cdots, \nabla f(x_{k-1}))$$

GD,momentum method 都满足此假设。

为何被称为 “zero-respecting”？若 $x_0 = 0$，且对于任意 $k < t$，$\nabla f(x_k)_i = 0$，则 $x_{t,i} = 0$。接下来的证明就要用到这个性质。

复杂度下界

接下来，我们以凸且光滑的函数类为例，给出 zero-respecting 的一阶算法解决此类问题的复杂度下界：

定理

对于任意 $T > 0, L > 0$，存在一个 L-光滑的凸函数，使得对于任意 *zero-respecting* 的一阶算法，我们都有：

$$\min_{1 \le k \le T} f(x_k) - f^* \ge \frac{3L}{32} \frac{\|x_0 - x^*\|}{(t+1)^2}$$

也就是说，任意 zero-respectin 的一阶算法都无法取得优于 $\Theta\left(\frac{L\|x_0 - x^*\|}{T^2}\right)$ 的收敛速度——因此 Nesterov 加速算法已是“最优算法”。

复杂度下界

证明.

我们显式构造出一个 “hard problem”:
我们构造 $f_d(x) = \frac{L}{8}x^T A_d x - \frac{L}{4}x_{(1)} : \mathbb{R}^d \to \mathbb{R}$

$$(A_d)_{i,j} = \begin{cases} 2 & \text{for} \quad i = j \\ -1 & \text{for} \quad |i-j| = 1 \\ 0 & \text{for} \quad \text{otherwise.} \end{cases}$$

复杂度下界

证明.

我们显式构造出一个 “hard problem”:
我们构造 $f_d(x) = \frac{L}{8}x^T A_d x - \frac{L}{4}x_{(1)} : \mathbb{R}^d \to \mathbb{R}$

$$(A_d)_{i,j} = \begin{cases} 2 & \text{for} \quad i = j \\ -1 & \text{for} \quad |i-j| = 1 \\ 0 & \text{for} \quad \text{otherwise.} \end{cases}$$

取问题的维度 $n \geq 2T+1$。构造 $f(x) = f_n(x)$。

复杂度下界

证明 (Cont'd).

步骤 1：我们验证： $0 \succeq A_n \succeq 4I_n$：

$$
\begin{aligned}
x^T A_n x &= 2\sum_{i=1}^{n} x_{(i)}^2 - 2\sum_{i=1}^{n-1} x_{(i)}x_{(i+1)} \\
&= \sum_{i=1}^{k-1}(x_{(i)} - x_{(i+1)})^2 + x_{(1)}^2 + x_{(k)}^2 \geq 0 \\
&\leq \sum_{i=1}^{k-1}(2x_{(i)}^2 + 2x_{(i+1)}^2) + x_{(1)}^2 + x_{(k)}^2 \leq 4\sum_{i=1}^{k-1} x_{(i)}^2
\end{aligned}
$$

则 $f(x)$ 为凸函数，且为 L-光滑。

复杂度下界

证明 (Cont'd).

步骤 2：验证：若 $x_t(i) = 0, i = t, \cdots, n$，则 $\nabla f(x_t)_{(i)} = 0, i = t+1, \cdots, n$。这是因为：

$$\nabla f(x) = \frac{L}{4}(Ax - e_1)$$

则由 zero-respecting 的性质，我们有 $x_{t+1,(i)} = 0, i = t+1, \cdots, n$

复杂度下界

证明 (Cont'd).

步骤 2：验证：若 $x_t(i) = 0, i = t, \cdots, n$，则 $\nabla f(x_t)_{(i)} = 0, i = t+1, \cdots, n$。这是因为：

$$\nabla f(x) = \frac{L}{4}(Ax - e_1)$$

则由 zero-respecting 的性质，我们有 $x_{t+1,(i)} = 0, i = t+1, \cdots, n$

即，每迭代一步，只有一个分量被激活（变为非零）。

复杂度下界

证明 (Cont'd).

步骤 3：Bound $f(x_t) - f^*$，$t \leq T = \frac{n-1}{2}$。

复杂度下界

证明 (Cont'd).

步骤 3：Bound $f(x_t) - f^*$，$t \leq T = \frac{n-1}{2}$。
由于 x_t 只有前 t 个分量非零，我们有 $f(x_t) = f_n(x_t) = f_T(x_t)$。

复杂度下界

证明 (Cont'd).

步骤 3：Bound $f(x_t) - f^*$，$t \le T = \frac{n-1}{2}$。
由于 x_t 只有前 t 个分量非零，我们有 $f(x_t) = f_n(x_t) = f_T(x_t)$。
对于 $f_d(x)$，其最小值点由 $\nabla f_d(x^{d*}) = 0$ 给出。解得

$$x^{d*}_{(i)} = 1 - \frac{i}{d+1}$$

带入得到

$$f^*_d = -\frac{L}{8}(1 - \frac{1}{d+1})$$

复杂度下界

证明 (Cont'd).

则

$$f^* = f_n^* = -\frac{L}{8}(1 - \frac{1}{n+1}) = -\frac{L}{8}(1 - \frac{1}{2T+2}),$$

$$\min_{1 \le k \le T} f(x_k) \ge f_T^* = -\frac{L}{8}(1 - \frac{1}{T+1})$$

复杂度下界

证明 (Cont'd).

则

$$f^* = f_n^* = -\frac{L}{8}(1 - \frac{1}{n+1}) = -\frac{L}{8}(1 - \frac{1}{2T+2}),$$
$$\min_{1\le k\le T} f(x_k) \ge f_T^* = -\frac{L}{8}(1 - \frac{1}{T+1})$$

有

$$\min_{1\le k\le T} f(x_k) - f^* \ge \frac{L}{16(T+1)}$$

复杂度下界

证明 (Cont'd).

注意到

$$\|x^*\|^2 = \sum_{i=1}^{2T+1} (1 - \frac{i}{2T+2})^2 \leq \frac{2T+2}{3}$$

我们得到

$$\min_{1 \leq k \leq T} f(x_k) - f^* \geq \frac{3L\|x^*\|^2}{32(T+1)^2}$$

。

谢谢大家！