中国科学技术大学

本科毕业论文

基于随机梯度法的机器学习优化算法收敛性态分析

作者姓名：	郑滕飞
学　　号：	PB20000296
专　　业：	信息与计算科学
导师姓名：	卢建良 实验师
完成时间：	2024 年 2 月 24 日

摘要

随着模型的复杂、样本个数的增大，对所有样本进行迭代的梯度方法变得不可接受，因此需要采用随机抽样构建成随机梯度法，并对其进行收敛性态的分析。在强凸函数中，其展现了非常好的期望收敛性，而对一般函数亦有收敛性的结论。为了进一步降低随机梯度法的噪声，可以采取动态采样、梯度聚合与迭代平均的手段，它们在理论上可以有效降低方差的影响，实践中也使得损失曲线变得光滑。此外，为了改善其收敛性，可以考虑直接引入海森阵估算的二阶方法，如无海森牛顿法等，或利用保持估算正定性的估算方案，如自然梯度法等。对它们收敛性态的理论与实践分析表明，二阶方法的不稳定性相对高，但在最优值附近收敛较快，因此，可以在自然梯度法降噪的基础上提出一个结合普通随机梯度的步长自适应二阶方法，其在实践中表现出了良好的收敛性态。

关键词：随机梯度法；收敛性态；自然梯度法

Abstract

As the model becomes more complex and the number of samples increases, the gradient method of iterating over all the samples becomes unacceptable. So, using random sampling, it needs to be constructed as a stochastic gradient method, with its convergence state be analyzed. It exhibits better convergence state in strongly convex functions, and convergence is also concluded for general functions. In order to further reduce the noise of the stochastic gradient method, dynamic sampling, gradient aggregation and iterative averaging can be adopted, which can reduce the effect of variance in theory and make the loss curve smooth in practice. In addition, in order to improve its convergence, one can consider directly introducing second-order methods for Hessian matrix estimation, such as the Hessian-free Newton method, or utilizing estimation schemes that maintain the positive qualitative nature of the estimation, such as the natural gradient method. The theoretical and practical analyses of their convergence states show that the second-order methods have relatively high instability but converge faster near the optimal value, so a step-size adaptive second-order method incorporating ordinary stochastic gradient can be proposed on the basis of noise reduction by the natural gradient method, which shows good convergence states in practice.

Key Words: stochastic gradient method; convergence state; natural gradient method

目录

第 1 章　前言

随着科技的发展，我们正在面对越来越大的数据量与越来越复杂的机器学习模型。由于机器学习问题最终往往转化为优化，即使是单个图像进行风格迁移这类的相对低复杂度问题，也可能涉及到数十万维空间中的优化[1]，而对涉及训练的问题，还可能使用数万量级的样本[2]。然而，传统的一阶或二阶下降算法每次迭代都需要对所有样本在整个空间进行梯度的计算，单次迭代所消耗的算力过大，甚至往往是不可接受的，于是，我们必须寻求更好的迭代策略。

一个直观的想法是，每次从样本中通过适当的抽样方法抓取一个相对较小的批次，并以此进行迭代，这就引出了抽样进行梯度下降的随机梯度法。容易得到，简单随机抽样后梯度的期望等于真实梯度，但由于误差的累计，这并不能直接带来良好的收敛性。

后文将证明，从数学上，在对迭代方差进行适当控制的情况下，目标函数满足一定条件（如强凸性）时可以直接证明随机梯度法的期望收敛性，对条件更弱的情况也存在相应更弱的收敛性结论。不过，普通的随机梯度法的方差较大，易受噪声影响，且收敛速度不及线性，需要从克服噪声与加速收敛两个方向进行改进：

1. 直观来说，抽样更全面可以实现降噪，但也会带来开销的增大，这其中需要进行权衡。此外，当步长较小时，还可以利用之前迭代中已计算过的结果来加入估计，这就衍生出了梯度聚合、迭代平均等方法；

2. 为了加速收敛，引入二阶导信息是常用的手段，但直接计算仍然过于复杂，因此通常考虑只计算部分的二阶信息估计全部，或直接以一阶信息估计二阶信息（拟牛顿法）。其中，高斯-牛顿法巧妙地解决了拟牛顿法可能非正定的问题，在实际应用中效果较好。值得注意的是，机器学习中常用的函数，如 ReLU，很可能是并不具有可微性的，这时二阶条件的使用反而可能导致更多问题。

现实中，机器学习应用的随机梯度法往往结合了不同改进手段，也导致了对其的收敛性与收敛速率分析变得更加复杂。为了确定不同改进对收敛性态的影响，我们既需要理论上的分析，也需要在实践中进行测试。本文旨在对现有的基于随机梯度法的各类优化算法进行理论分析与实际测试，以评价各方法的收敛性态，并在此基础上提出可能的综合与改进方案。

第 2 章　随机梯度法

第 1 节　基本算法

§2.1.1　问题描述

机器学习中，在样本空间 X 到输出空间 Y 存在某个真实映射 y，而我们希望在给定的函数空间 $\mathcal{H}$ 中学习一个最优的 h_{opt}，也即给定真实世界中 X 上的概率分布 f，其应满足

$$h_{\text{opt}} = \operatorname*{argmin}_{h\in\mathcal{H}} R[h], \quad R[h] = E_f[I_{h(x)=y(x)}]$$

这里 argmin 指使右端存在最小值的参数，而 $I_{h(x)=y(x)}$ 在 $h(x)=y(x)$ 时为 1，否则为 0。由于真实的 y 需要通过样本估算，假设所有样本为 $x_i, y_i, i=1,\ldots,N$，则可估算

$$R[h] \approx \sum_{i=1}^{N} I_{y_i=h(x_i)}$$

更准确来说，在 Y 连续的情况下，我们一般用损失函数 $l(h(x),y)$ 度量差距，而非直接进行 0 或 1 的区分。此外，由于我们会对一些函数有偏好（例如更光滑的函数），一般需要与正则化项结合进行度量，例如考虑优化目标为

$$F[h] = R[h] + \lambda\Omega[h], \quad R[h] = \frac{1}{N}\sum_{i=1}^{N} l(h(x_i),y_i)$$

这里 $\Omega[h]$ 为正则化项，只与 h 的形式有关，与样本无关。

下面我们假设 h 可以被参数向量 w 确定，从而写为 $h_w(x)$。由此，记 $F(w)=F[h_w]$，有

$$F(w) = \frac{1}{N}\sum_{i=1}^{N} f_w(\xi_i), \quad \xi_i=(x_i,y_i), \quad f_w(x,y) = l(h_w(x),y) + \lambda\Omega[h_w]$$

一般假设 h 对 w 有较好的光滑性，至少存在一阶或二阶导数，此时将上方的 f 也看作 w 的函数较为方便，即记

$$f_i(w) = f_w(\xi_i)$$

至此，我们将对泛函的优化问题转化为了一个求函数最小值的问题。我们以 Logistic 回归为例，若取正则化为二范数正则化，考虑均方误差，则可以取

$$h_w(x) = \frac{1}{1+\mathrm{e}^{-w^Tx^{(1)}}}, \quad \Omega[h_w] = w^Tw, \quad f_i(w) = \left(y_i - \frac{1}{1+\mathrm{e}^{-w^Tx_i^{(1)}}}\right)^2 + \lambda w^Tw$$

其中 $x^{(1)}$ 表示 x 最前增添一个为 1 的分量得到的向量，对应拟合的截距。对所有样本求和后平均，即能得到需要优化的函数 $F(w)$。

§2.1.2 梯度方法

一般利用梯度方法求最小值的模型为，对目标函数 $h(x)$，每次在最速下降方向（负梯度）方向取一个步长 α，进行迭代

$$x_{n+1} = x_n - \alpha \nabla_x h(x_n)$$

直到 $\nabla_x h(x_n)$ 的模长充分小。

设第 k 步步长为 α_k，对 w 进行第 k 步梯度下降的公式应为

$$w_{k+1} = w_k - \alpha_k \nabla F(w_k) = w_k - \frac{\alpha_k}{N} \sum_{i=1}^{N} \nabla f_i(w_k)$$

这里 w_k, w_{k+1} 为列向量，如无特殊说明，此后的 ∇ 表示对 w 的梯度，并排成列向量的形式。

如前所述，当样本量很大时，对所有样本计算梯度并求和的开销极大，因此必须考虑抽样。基础的迭代方式为：

算法 1 (基础迭代)**.** 对每次抓取的样本 $\xi_1, \ldots, \xi_n$，先抽取一个下标 i_k，再进行迭代

$$w_{k+1} = w_k - \alpha_k \nabla f_{i_k}(w_k)$$

虽然这个想法非常简单，但其正确性并不显然。总的来说，由于等概率抽取后选择的方向的期望是下降方向 $\nabla F(w_k)$，w_k 期望于下降至 F 的极小点。后续我们会讨论利用降噪方法与二阶信息改进随机梯度法，而本章则会关注对方法本身的分析。除了基础迭代以外，自然也可以抽取多个下标，这也在我们分析的范畴内：

算法 2 (多样本迭代)**.** 对每次抓取的样本 $\xi_1, \ldots, \xi_n$，抽取 n_k 个下标，记为 $\{k,1\}, \ldots, \{k,n_k\}$，再进行迭代

$$w_{k+1} = w_k - \alpha_k \frac{1}{n_k} \sum_{i=1}^{n_k} \nabla f_{k,i}(w_k)$$

或

$$w_{k+1} = w_k - \alpha_k \frac{1}{n_k} H_k \sum_{i=1}^{n_k} \nabla f_{k,i}(w_k)$$

这里 H_k 为对称正定阵，可由牛顿法或拟牛顿法产生[3]。

方便起见，用 $w_{k+1} = w_k - \alpha_k g(w_k, \xi_k)$ 表示任何一种迭代步，对多样本迭代算法，ξ_k 是一列样本。三种方法合称为（简单）随机梯度法，简称 SG 方法。由于对方法合理性的要求，我们必须先行证明收敛性。

第 2 节　收敛分析

§2.2.1　迭代下降性

为了证明 SG 方法的收敛，由于涉及梯度，自然需要先保证目标函数 F 具有一定的光滑性质，即：

定义 1 (Lipschitz 连续、梯度 Lipschitz 函数)**.** 称一个映射 $\phi : D \to \mathbb{R}^n$ 具有界为 L 的 Lipschitz 连续性，若其满足

$$\|\phi(w) - \phi(\bar{w})\| \leq L\|w - \bar{w}\|, \quad \forall w, \bar{w} \in D$$

而若一个函数 F 在定义域上可微且其梯度 Lipschitz 连续，则称其为界为 L 的梯度 Lipschitz 函数。

如上要求后，有结论：

定理 1 (梯度 Lipschitz-SG 迭代下降性)**.** 在 $F(w)$ 有界为 L 的梯度 Lipschitz 连续性时，函数值下降有结论（这里 E_{ξ_k} 表示 ξ_k 随机抽取下的期望）

$$E_{\xi_k}[F(w_{k+1})] - F(w_k) \leq -\alpha_k \nabla F(w_k)^T E_{\xi_k}[g(w_k, \xi_k)] + \frac{\alpha_k^2 L}{2} E_{\xi_k}\left[\|g(w_k, \xi_k)\|^2\right] + O(\alpha_k^3)$$

证明. 对 $F(w_{k+1})$ 在 w_k 处作二阶泰勒展开后代入计算对比原式，发现只需证明

$$x^T \nabla^2 F(w) x \leq L\|x\|^2$$

对任何 x, w 成立。若否，假设对 w_0, x_0 不成立，则考虑泰勒展开

$$\nabla F(w_0 + \alpha x_0) - \nabla F(w_0) = \alpha \nabla^2 F(w_0) x_0 + O(\alpha^2)$$

当 α 充分小时即有

$$\alpha x_0^T (\nabla F(w_0 + \alpha x_0) - \nabla F(w_0)) = \alpha^2 x_0^T \nabla^2 F(w_0) x_0 + O(\alpha^3) > L\|\alpha x_0\|^2$$

记 $y = \alpha x_0$，$z = \nabla F(w_0 + \alpha x_0) - \nabla F(w_0)$。由于 $y^T z > L\|y\|^2$，又由柯西不等式 $y^T z < \|y\|\|z\|$ 即得 $\|z\| > L\|y\|$，与 Lipschitz 条件矛盾。 □

直观来看,等概率随机时的任何一种迭代步,都有 $E_{\xi_k}[g(w_k, \xi_k)]$ 为 $\nabla F(w_k)$ 或 $H_k \nabla F(w_k)$，于是非驻点处只要 α_k 充分小，就能保证函数值期望的下降。由此过程对任何一次迭代都成立，$E_{\xi_1,\ldots,\xi_{k-1}}[w_k]$ 一定是逐渐下降的，这就说明了收敛性。

若作出进一步的假设，还有更好的结论：

定理 2 (二阶矩条件-SG 迭代下降性)**.** 在 $F(w)$ 有界为 L 的梯度 Lipschitz 连续性时，进一步要求（Var_{ξ_k} 表示方差）：

- 存在 $\mu_G \geq \mu > 0$，使得对一切 k 有

$$\nabla F(w_k)^T E_{\xi_k}[g(w_k, \xi_k)] \geq \mu \|\nabla F(w_k)\|^2$$

$$E_{\xi_k}[\|g(w_k, \xi_k)\|] \leq \mu_G \|\nabla F(w_k)\|$$

- 存在 $M \geq 0$ 与 $M_V \geq 0$，使得对一切 k 有

$$\operatorname*{Var}_{\xi_k}[\|g(w_k, \xi_k)\|] \leq M + M_V \|\nabla F(w_k)\|^2$$

记 $M_G = M_V + \mu_G^2$，则函数值下降有结论

$$E_{\xi_k}[F(w_{k+1})] - F(w_k) \leq -\left(\mu\alpha_k - \frac{\alpha_k^2 L}{2} M_G\right) \|\nabla F(w_k)\|^2 + \frac{\alpha_k^2 L}{2} M + O(\alpha_k^3)$$

证明. 根据方差定义计算知 $E_{\xi_k}\left[\|g(w_k, \xi_k)\|^2\right] \leq M + M_G \|\nabla F(w_k)\|^2$，从而直接由上一定理代入不等式化简即可。 □

§2.2.2 强凸函数

根据以上两个关于下降性的定理，只要给函数补充适当的条件，就能得到收敛性的结论。首先，我们假设迭代序列 $\{w_k\}$ 始终在 F 有下界的某开集内，最优值存在，记为 F_*。

定义 2 (强凸函数). 对函数 $F : \mathbb{R}^n \to \mathbb{R}$，若其可微，且存在 $c > 0$ 使得

$$\forall x, y \in \mathbb{R}^n, \quad F(y) \geq F(x) + \nabla F(x)^T (y - x) + \frac{c}{2} \|y - x\|^2$$

则称 F 是参数为 c 的强凸函数。

值得注意的是，对比凸函数的一阶条件可知强凸函数一定是严格凸函数，从而其最优解若存在则唯一。

定理 3 (强凸函数性质). 若函数 F 是参数为 c 的强凸函数，且在 w^* 取到最小值 F_*，则有

$$F(w) - F_* \leq \frac{1}{2c} \|\nabla F(w)\|^2$$

证明. 根据强凸函数的条件配方有

$$F(w) - F_* \leq -\nabla F(w)^T (w^* - w) - \frac{c}{2} \|w^* - w\|^2 = -\left\| \frac{1}{\sqrt{2c}} \nabla F(w) + \sqrt{\frac{c}{2}} \right\|^2 + \frac{1}{2c} \|\nabla F(w)\|^2$$

从而得证。 □

定理 4 (强凸函数-固定步长 SG 算法收敛性)**.** 若 F 是参数为 c 的强凸函数，且 SG 迭代满足二阶矩条件。假设步长 α_k 恒定为 $\bar{\alpha}$，满足

$$0 < \bar{\alpha} \leq \min\left\{\frac{\mu}{LM_G}, \frac{1}{c\mu}\right\}$$

则迭代过程中（这里 E 代表每一步都随机抽取下的期望，对 w_k，由于只有这之前的影响，因此为 $E_{\xi_1,\ldots,\xi_{k-1}}$）

$$E[F(w_k) - F_*] \leq \frac{\bar{\alpha}LM}{2c\mu} + (1 - c\mu\bar{\alpha})^{k-1}\left(F(w_1) - F_* - \frac{\bar{\alpha}LM}{2c\mu}\right)$$

证明. 根据二阶矩条件下的结论与定理 3 即有（根据范围可得 $\mu\bar{\alpha} - \frac{\bar{\alpha}^2 L}{2}M_G > \frac{\mu\bar{\alpha}}{2}$）

$$E_{\xi_k}[F(w_{k+1})] - F(w_k) \leq -\frac{\mu\bar{\alpha}}{2}\|\nabla F(w_k)\|^2 + \frac{\bar{\alpha}^2 L}{2}M + O(\bar{\alpha}^3)$$

$$\leq -c\mu\bar{\alpha}(F(w_k) - F_*) + \frac{\bar{\alpha}^2 L}{2}M + O(\bar{\alpha}^3)$$

同加 $F(w_k) - F_*$ 后取期望变形即

$$E[F(w_{k+1}) - F_*] \leq (1 - c\mu\bar{\alpha})E[F(w_k) - F_*] + \frac{\bar{\alpha}^2 LM}{2} + O(\bar{\alpha}^3)$$

从而忽略高阶小量 $O(\bar{\alpha}^3)$ 展开计算知

$$E[F(w_{k+1}) - F_*] \leq (1 - c\mu\bar{\alpha})^k E[F(w_1) - F_*] + \frac{\bar{\alpha}^2 LM}{2}\sum_{n=0}^{k-1}(1 - c\mu\bar{\alpha})^n$$

$$= (1 - c\mu\bar{\alpha})^k E[F(w_1) - F_*] + \frac{\bar{\alpha}LM}{2c\mu}(1 - (1 - c\mu\bar{\alpha})^k)$$

$$= \frac{\bar{\alpha}LM}{2c\mu} + (1 - c\mu\bar{\alpha})^k\left(F(w_1) - F_* - \frac{\bar{\alpha}LM}{2c\mu}\right)$$

这就是结论的形式。 □

理论来说，取使 $c\mu\bar{\alpha} < 1$ 的 $\bar{\alpha}$，最终误差即会趋于 $\frac{\bar{\alpha}LM}{2c\mu}$。根据 M 的定义，若 $g(w_k, \xi_k)$ 的方差能被 $\|\nabla F(w_k)\|^2$ 线性控制，即有 $M = 0$，可以趋于最优解，否则，必须选取较小的步长来保证结果在容差范围的收敛性。自然地，实践中的梯度下降方法会在接近最优点时减少步长，此时会有结论：

定理 5 (强凸函数-变步长 SG 算法收敛性)**.** 若 F 是参数为 c 的强凸函数，且 SG 迭代满足二阶矩条件。假设步长 $\alpha_k = \frac{\beta}{\gamma + k}$ 满足 $\beta > \frac{1}{c\mu}, \gamma > 0, \alpha_1 \leq \frac{\mu}{LM_G}$，则迭代过程中

$$E[F(w_k) - F_*] \leq \frac{\nu}{\gamma + k}, \nu = \max\left\{\frac{\beta^2 LM}{2(\beta c\mu - 1)}, (\gamma + 1)(F(w_1) - F_*)\right\}$$

证明. 利用归纳法，首项由于 $\nu \geq (\gamma+1)(F(w_1)-F_*)$ 知成立，递推中与上一定理相同忽略 $O(\alpha_k^3)$ 有

$$E[F(w_{k+1})-F_*] \leq (1-\mu\alpha_k c)E[F(w_k)-F_*] + \frac{\alpha_k^2 LM}{2}$$

继续计算得右侧为（第一个不等号利用 $\nu \geq \frac{\beta^2 LM}{2(\beta c\mu-1)}$ 放缩，参数范围限制了符号）

$$\frac{2(\gamma+k)\nu - 2\mu\beta c\nu + \beta^2 LM}{2(\gamma+k)^2} \leq \frac{2(k+\gamma)\nu - 2\nu}{2(\gamma+k)^2} = \frac{\gamma+k-1}{(\gamma+k)^2}\nu < \frac{\nu}{\gamma+k+1}$$

即得证。 □

§2.2.3 一般函数

虽然我们在强凸函数时得到了充分良好的收敛性，遗憾的是，一般的机器学习问题中，遇到的函数往往是非凸的，极小值与驻点会对算法产生很大的影响。不过，依然可以从梯度角度刻画出 SG 算法的收敛性：

定理 6 (一般函数-固定步长 SG 算法收敛性)**.** 若 SG 迭代满足二阶矩条件，且步长 α_k 恒定为 $\bar{\alpha}$，满足 $0 < \bar{\alpha} \leq \frac{\mu}{LM_G}$，则迭代过程中

$$E\left[\sum_{k=1}^{K} \|\nabla F(w_k)\|^2\right] \leq \frac{K\bar{\alpha}LM}{\mu} + \frac{2(F(w_1)-F_*)}{\mu\bar{\alpha}}$$

从而

$$\lim_{K\to\infty} E\left[\frac{1}{K}\sum_{k=1}^{K} \|\nabla F(w_k)\|^2\right] \leq \frac{\bar{\alpha}LM}{\mu}$$

证明. 与强凸函数时相同忽略 $O(\bar{\alpha}^3)$ 得到

$$E_{\xi_k}[F(w_{k+1})] - F(w_k) \leq -\frac{\mu\bar{\alpha}}{2}\|\nabla F(w_k)\|^2 + \frac{\bar{\alpha}^2 L}{2}M$$

作期望后累加即有

$$F_* - F(w_1) \leq E[F(w_{k+1})] - F(w_1) \leq \frac{\mu\bar{\alpha}}{2}\sum_{k=1}^{K}\|\nabla F(w_k)\|^2 + \frac{K\bar{\alpha}^2 L}{2}M$$

移项变形得结论。 □

数列前 n 项平均值的极限称为数列的 Cesaro 平均，也即此收敛性是针对 Cesaro 平均而言。对变步长情况有结论：

定理 7 (一般函数-变步长 SG 算法收敛性)**.** 若 SG 迭代满足二阶矩条件，且步长 $\alpha_k > 0$ 满足

$$\sum_{k=1}^{\infty}\alpha_k = \infty, \sum_{k=1}^{\infty}\alpha_k^2 < \infty$$

则迭代过程中

$$\sum_{k=1}^{\infty}\alpha_k E\big[\|\nabla F(w_k)\|^2\big]<\infty$$

证明. 由已证的

$$E_{\xi_k}[F(w_{k+1})]-F(w_k)\le-\left(\mu\alpha_k-\frac{\alpha_k^2L}{2}M_G\right)\|\nabla F(w_k)\|^2+\frac{\alpha_k^2L}{2}M+O(\alpha_k^3)$$

忽略 $O(\alpha_k^3)$ 取梯度累加，减去确定收敛的 $\sum_{k=1}^{\infty}\frac{\alpha_k^2L}{2}M$ 得到下式收敛：

$$-\mu\sum_{k=1}^{\infty}\alpha_k E\big[\|\nabla F(w_k)\|^2\big]+\frac{LM_G}{2}\sum_{k=1}^{\infty}\alpha_k^2E\big[\|\nabla F(w_k)\|^2\big]M$$

由于 $\sum_{k=1}^{\infty}\alpha_k^2<\infty$，必然有 $\lim_{k\to\infty}\alpha_k=0$，因此某项后 $\alpha_k<\frac{\mu}{LM_G}$，由有限项不影响收敛性可不妨设 $\alpha_1<\frac{\mu}{LM_G}$，这时级数成为负项级数，由收敛满足

$$-\mu\sum_{k=1}^{\infty}\alpha_k E\big[\|\nabla F(w_k)\|^2\big]+\frac{LM_G}{2}\sum_{k=1}^{\infty}\alpha_k^2E\big[\|\nabla F(w_k)\|^2\big]M>-\infty$$

而根据刚才的假设

$$-\frac{\mu}{2}\sum_{k=1}^{\infty}\alpha_k E\big[\|\nabla F(w_k)\|^2\big]>-\mu\sum_{k=1}^{\infty}\alpha_k E\big[\|\nabla F(w_k)\|^2\big]+\frac{LM_G}{2}\sum_{k=1}^{\infty}\alpha_k^2E\big[\|\nabla F(w_k)\|^2\big]M>-\infty$$

即证明了定理中的收敛性。 □

进一步地，我们可以得到：

定理 8 (下极限的估计). 满足上一定理条件时有（$\liminf$ 表示下极限，即所有有极限子列的极限下限）

$$\liminf_{k\to\infty}E\big[\|\nabla F(w_k)\|^2\big]=0$$

证明. 记数列为 $\{a_n\}$。若否，由于数列恒正，其下极限 $c>0$，则取 $\epsilon=\frac{c}{2}$，必然存在 N 使得 $n>N$ 时有 $a_n>\epsilon$ (否则能取出一个均 $<\epsilon$ 的子列，其下极限 $\le\epsilon<c$，矛盾)，而由于 $\sum_{k=1}^{\infty}\alpha_k=\infty$，这意味着

$$\sum_{k=1}^{\infty}\alpha_k a_k\ge\epsilon\sum_{k=1}^{\infty}\alpha_k-\sum_{k=1}^{N}\alpha_k a_k=\infty$$

矛盾。 □

此结论还有更多的推论，如：

定理 9 (一般函数-SG 算法梯度收敛性). 在 SG 算法与步长满足上述条件时，有

- 记 X_K 为 1 到 K 中随机抽取一个下标的随机变量，抽到 k 的概率正比于 α_k，则 $\|\nabla F(w_{X_K})\|$ 在 $K \to \infty$ 时依概率收敛于 0。
- 进一步假设 F 二阶可微且 $\|\nabla F(w)\|^2$ 作为 w 的函数是梯度 Lipschitz 连续的，则有

$$\lim_{k\to\infty} E\left[\|\nabla F(w_k)\|^2\right] = 0$$

证明. 对第一个结论，定理 7 即表示 $K \to \infty$ 时 $\|\nabla F(w_{X_K})\|^2 \sum_{i=1}^{K} \alpha_i$ 的期望有界，而由 $\sum_{i=1}^{K} \alpha_i$ 无穷时无界，利用概率论知识可证明 $\|\nabla F(w_{X_K})\|^2$ 依概率收敛于 0，从而 $\|\nabla F(w_{X_K})\|$ 依概率收敛于 0。

对第二个结论，由定理 8 知只需说明极限存在。与定理 1 类似可得到下降性的证明，从而其必然单调减，即得结论。 □

第 3 节　实际测试

§2.3.1　线性回归

考虑带二范数正则化的线性回归，也即取

$$f_i(w) = (y_i - w^T x_i^{(1)})^2 + \lambda w^T w$$

根据强凸函数的定义与凸函数的一阶定义可发现，若 G 是参数为 c 的强凸函数，H 是凸函数，则 $H + \lambda G$ 是参数为 λc 的强凸函数，因此，取 $G(w) = w^T w, H(w) = (y_i - w^T x_i^{(1)})^2$，可算出 $f_i(w)$ 是参数为 λ 的强凸函数，类似组合即得到 F 是参数为 λ 的强凸函数，于是能拥有较好的理论收敛性。

直接计算可以得到

$$\nabla f_i(w) = 2(w^T x_i^{(1)} - y_i)x_i^{(1)} + 2\lambda w$$

由此，在 $y = 1 + 2x_1 + 3x_2 - x_3$ 上随机添加高斯噪声，并尝试用随机梯度法求解，记得到的系数 w_0, w_1, w_2, w_3 与真实系数相比的损失为

$$L = (w_0 - 1)^2 + (w_1 - 2)^2 + (w_2 - 3)^2 + (w_3 + 1)^2$$

随机生成 100000 个样本，每次抽取一个进行迭代，分别取步长为 $\alpha_k = \frac{1}{999+k}$ 的逐渐衰减步长与 $\alpha_k = 0.0001$ 的恒定步长，令正则化系数为 0，初始 w 各分量均为 0，每次迭代中，先均匀抽取下标 i，再计算

$$w_{k+1} = w_k - \alpha_k \nabla f_i(w_k)$$

进行 30000 次迭代得到的损失如**图 1** 。总体来说，步长下降时的收敛效果好于恒定步长时，恒定步长后期出现的振荡更加明显，符合理论分析中的期望收敛性。

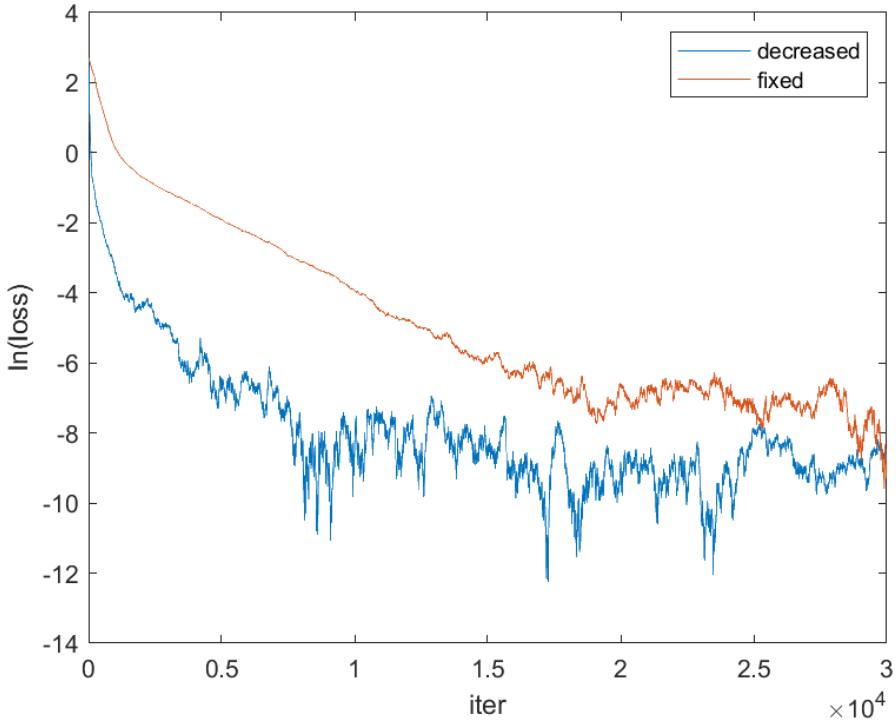

图 1: 单个抽样迭代

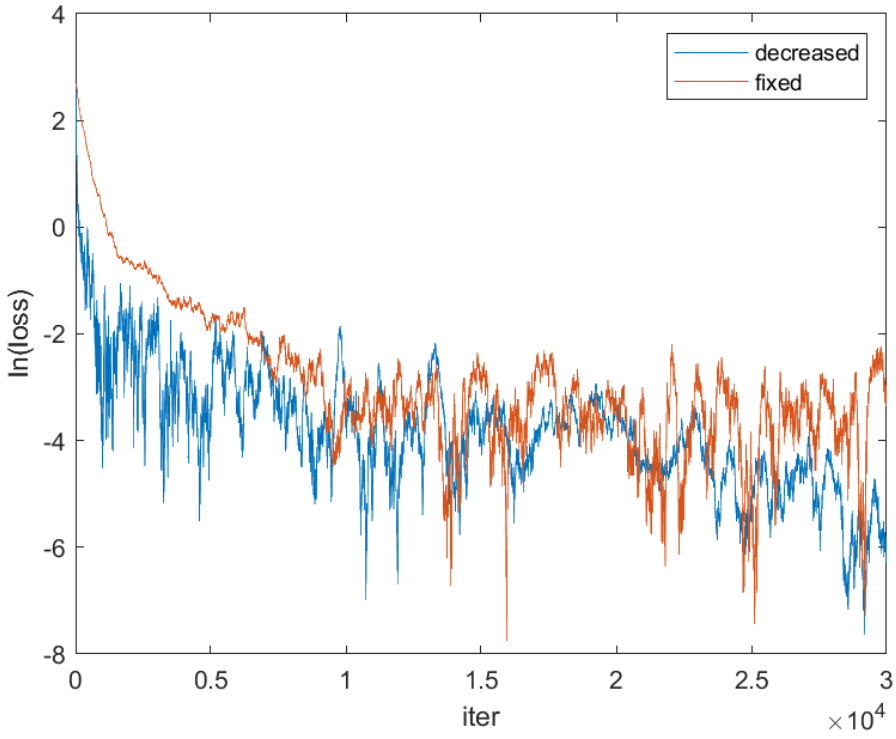

图 2: 噪声增强

增大噪声强度为 10 倍后的结果如**图 2**，由于纵轴代表 $\ln L$，当噪声增强时，可以明显发现两种迭代方法由于对单个样本的依赖，都会变得更加难以收敛，这与理论分析的结果一

致，也即抽样方差的增强会引起结果的不稳定。

根据机器学习时的经验，正则化项有助于对抗样本中的噪声，因此，下面将研究正则化项的作用。尝试在噪声增强后添加正则化项 $\lambda = 0.00001$ 后，可以得到**图 3** 。

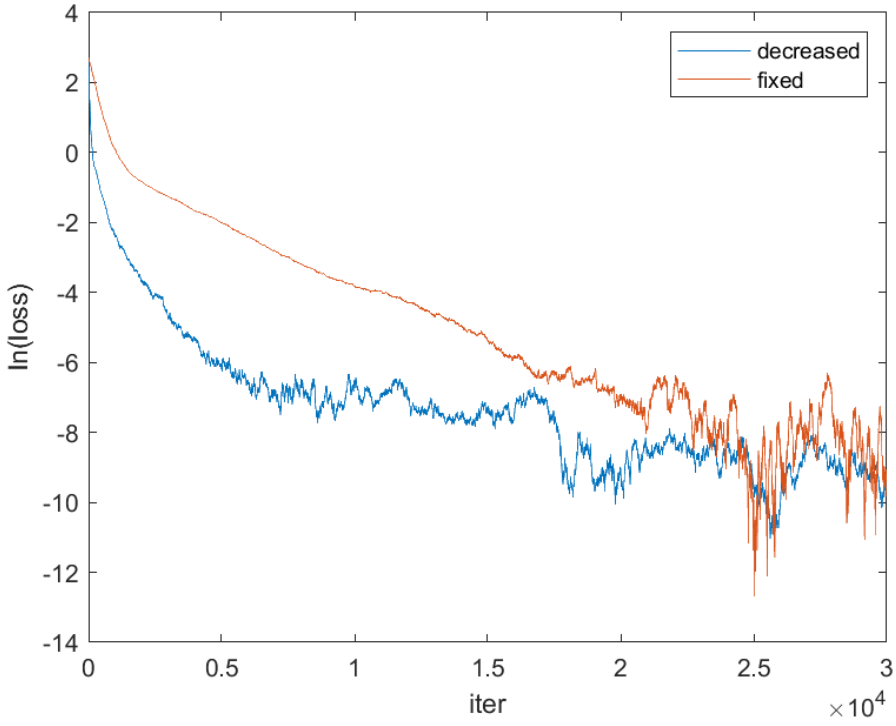

图 3: 加入正则化

增大正则化系数为 0.0001、0.01、1 的结果是**图 4** 。

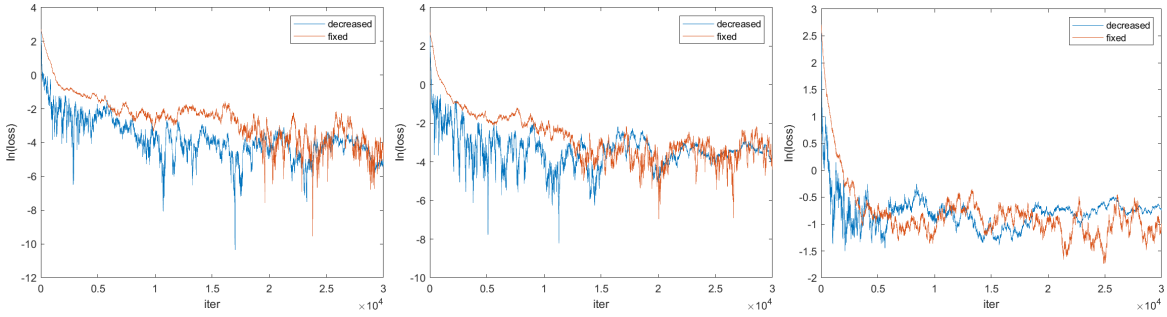

图 4: 正则化系数影响

从中可以看出，正则化增大时的确让迭代变得更加光滑，但也同时降低了收敛的效果，需要根据实际噪声的强度选取合适的系数。

§2.3.2 岭回归

岭回归与线性回归的差异在于，其正则化利用一范数进行，从而可以保证稀疏性。取 $G(w) = \|w\|_1$ 时，假设绝对值在 0 点处导数为 0（这实质上是取了最小范数子梯度），可以得到

$$\nabla f_i(w) = 2(w^T x_i^{(1)} - y_i)x_i^{(1)} + \lambda \text{sign}(w)$$

岭回归可以更容易得到稀疏解，从而剔除无效特征。为发挥其特性，考虑

$$\vec{x} = (x_1, \ldots, x_5), \quad y = 2 + x_1 - x_3 + 3x_4$$

生成 10000 个样本并对应添加随机误差，取正则化系数为 0.01，可以得到**图 5** 的效果。

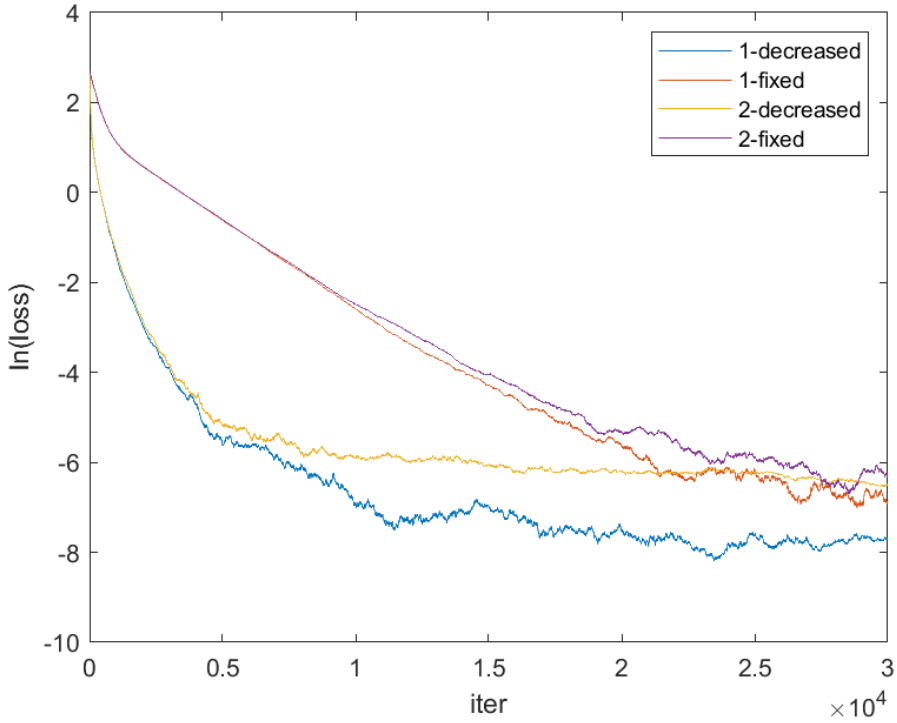

图 5: 岭回归

图例中 2 开头的两条曲线代表二范数正则化的结果，1 开头则代表岭回归的结果，固定步长与递减步长对应的步长与上一部分中相同。从中可以看出，存在无效特征时，岭回归的效果优于二范数正则化——即使它会导致函数不再具有真正的一阶可微性。

图 5 中，随机梯度法是每次取多个平均实现的，取的个数（即批次的大小）为 3。增大批次大小为 10、20、30 得到的效果如**图 6** 。由此可知，当步长取定时，批次大小并不会过多影响收敛速度，但会显著影响振荡的程度。随着批次的逐渐增大，迭代过程越能保证单调的下降性，尤其是在步长逐渐衰减时，增大批次可以得到近乎光滑下降的曲线。

下面，我们观察步长增大时批次大小对收敛的影响，**图 7** 与**图 8** 是初始步长提升为原本的 3 倍与 10 倍后在批次大小 10、20、30 时的收敛结果。

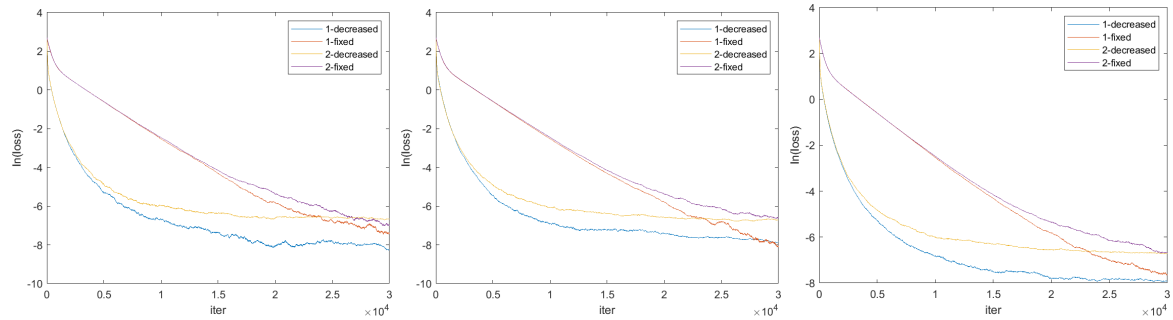

图 6: 批次大小影响

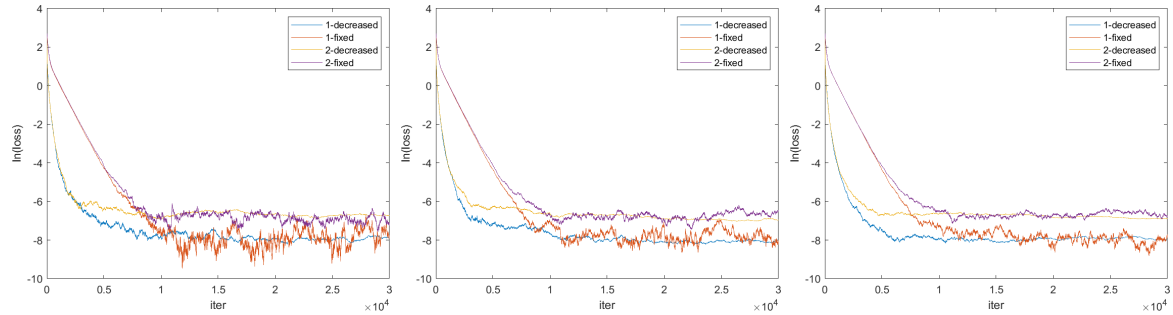

图 7: 三倍步长效果

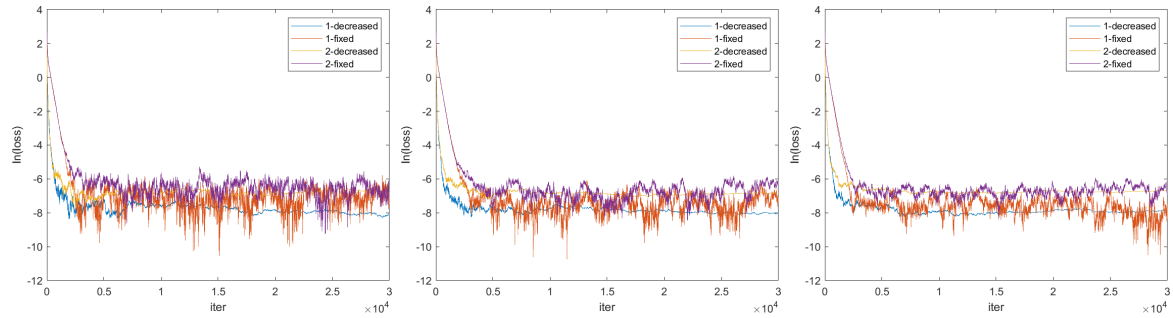

图 8: 十倍步长效果

可以看出，当批次大小增大时，可以适当增大步长以提升收敛速率。不过，具体何种批次大小与步长下可以达到计算最少次梯度时收敛，还需要根据实际情况进行调整。

§2.3.3 Logisitic 回归

最后，我们考虑用 Logisitic 回归进行线性二分类问题的拟合。设真实标签为

$$y = \begin{cases} 1 & a^T x^{(1)} > 0 \\ 0 & a^T x^{(1)} < 0 \end{cases}, \quad a = (7, 2, -1, 0, 1, 3, -2)^T$$

由于测量误差，训练集中的 100000 个样本中有 5% 标签被翻转。而测试集中有 2000 个样本，标签均为真实结果，以测试集中的误分类率作为损失。

采用 Logisitic 回归进行判别，学习系数 $w \in \mathbb{R}^7$，若 2.1.1 中的 $h_w(x) > 0.5$，也即 $w^T x_i^{(1)} > 0$，则认为标签是 1，否则认为标签是 0。二范数正则化下，记 $z = w^T x_i^{(1)}$，可知

$$f_i(w) = \left(y_i - \frac{1}{1+\mathrm{e}^{-z}}\right)^2 + \lambda w^T w, \quad \nabla f_i(w) = 2\left(\frac{1}{1+\mathrm{e}^{-z}} - y_i\right)\frac{\mathrm{e}^{-z}}{(1+\mathrm{e}^{-z})^2}x_i^{(1)} + 2\lambda w$$

此函数并非凸函数，理论来说未必具有良好的收敛性。取正则化系数 0.001，批次大小为 5，逐渐衰减步长为 $\alpha_k = \frac{1}{99+k}$，恒定步长 $\alpha_k = 0.001$，进行 2000 次迭代可以得到**图 9** 。

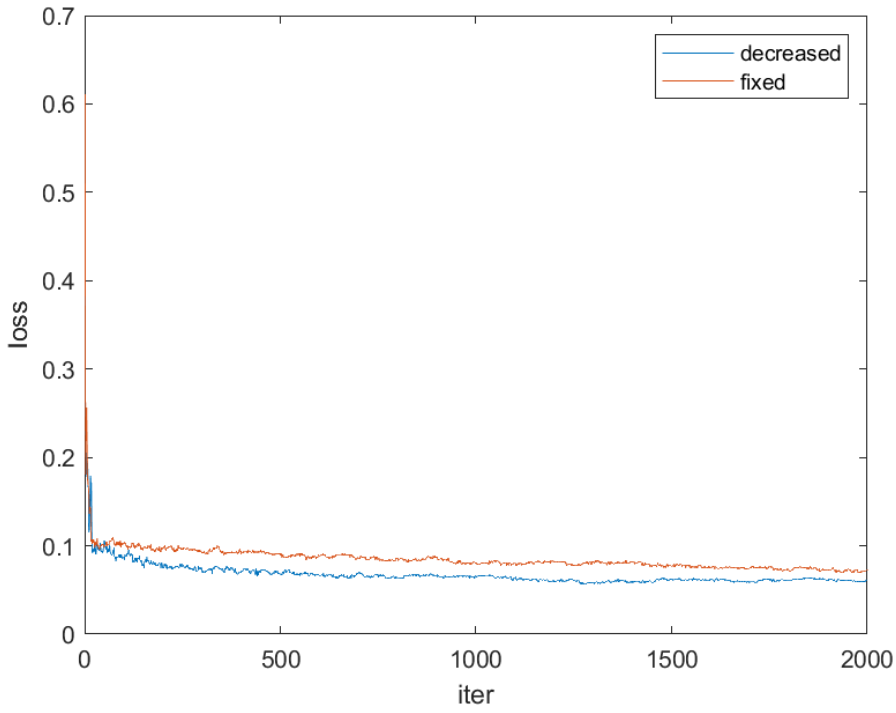

图 9: Logisitic 回归迭代

从此图中也可以看出随机梯度法的优越性：即使样本空间很大，只要其分布均匀，抽取其中一小部分就能得到收敛。实际测试中，批次大小为 10 时此例子的收敛次数大约为 150 次，远低于一次完整迭代的样本量。

不过，一个显著的问题是，即使在噪音率并不高的情况下，迭代中也仍有很明显的振荡，导致难以判断收敛情况。若将噪声率调高为 20%，这一振荡的问题会变得极为明显，如**图 10** 。振荡严重时，我们几乎无法判断迭代是否已经收敛，因此也无法给出合适的结束条件。步长衰减可以让振荡同时衰减，但我们仍然无法保证收敛结果的稳定。

另一个有趣的结果是，若将批次取得充分大，例如取为 100，则固定步长的迭代反而会比步长衰减结果更好——即使它更加不稳定，如**图 11** 。从理论上分析，后一种情况的出现是由于在大批次下，更大的步长是可以被允许的，而步长衰减时设定的初始步长相对更小，导致收敛变得极为缓慢。这也启示我们，在固定步长下进行一定次数的迭代，至收敛相对充分后再换用衰减步长是一个较好的选择。

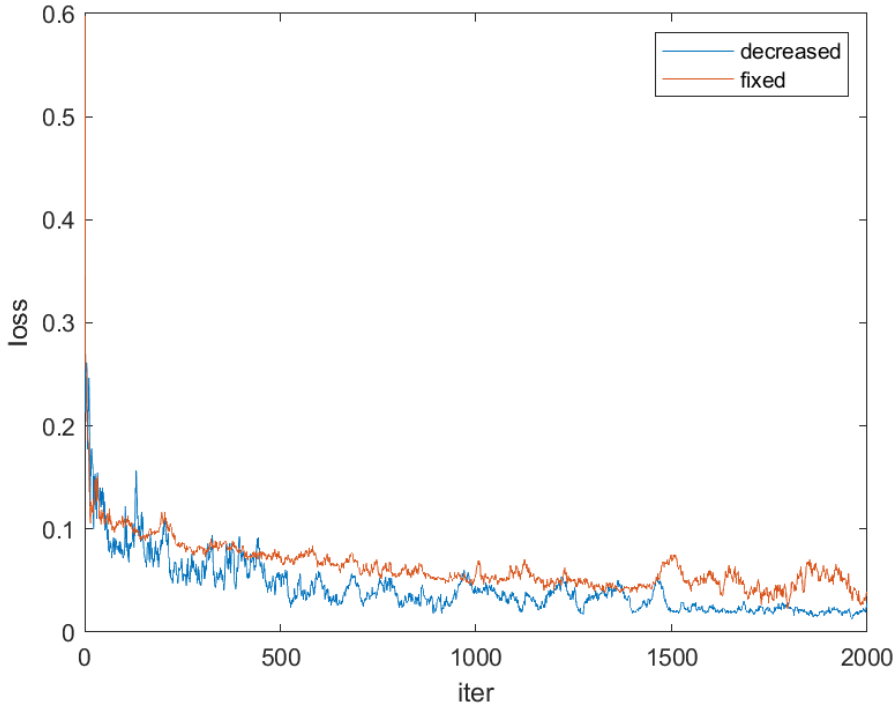

图 10: 高噪声率下的迭代

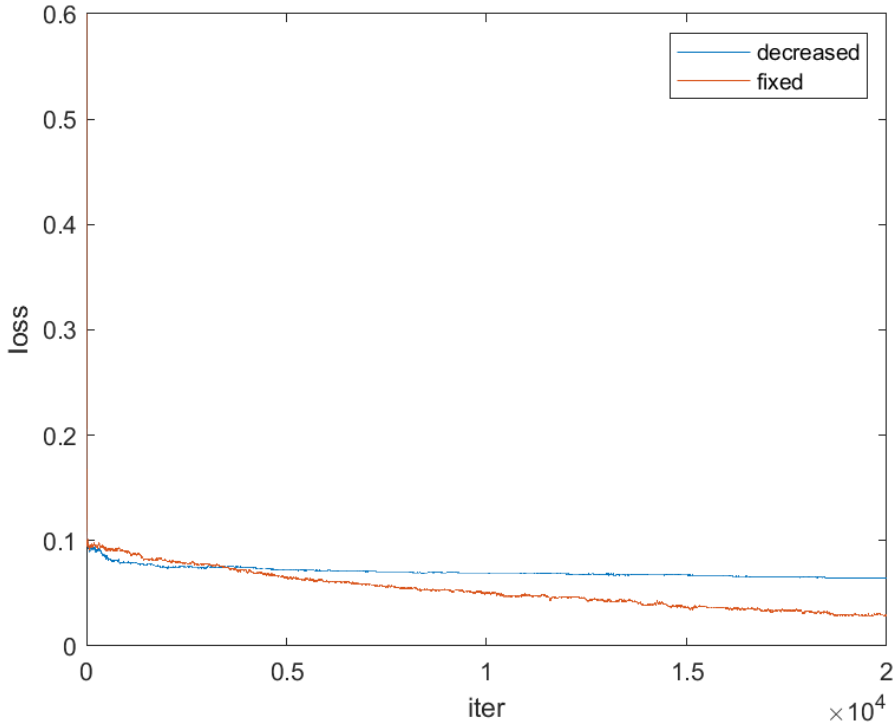

图 11: 高噪声率下的大批次迭代

总的来说，步长越大，收敛的速度相对越快，但噪声的影响也就越明显。接下来两章的策略即是从减弱噪声与加快收敛两方面进行改进的。

第 3 章　噪声控制

之前的分析中，我们证明了原始的随机梯度法在各种情况下的下降与收敛性质，也进行了一些实际测试。可以发现，由于随机抽取样本进行的估算可能比起真实的梯度有很大误差（也即估算的噪声），我们必须通过各种手段减小噪声以获取更好的结果。增添噪声处理的随机梯度法一般称为批次梯度法，接下来主要考察三种常用的手段：

- **动态采样法**，即通过视情况逐步增加迭代过程中抽取的样本个数 n_k 来实现降噪[4]；
- **梯度聚合法**，即存储之前迭代中的梯度估计值，并在每次迭代中更新其中一部分，接着将搜索方向定义为这些估计值的加权平均值，从而提高搜索方向的质量；
- **迭代平均法**，即维护一个优化过程中迭代的平均值以减小噪声。

第 1 节　动态采样

§3.1.1　理论分析

回顾之前得到的

$$E_{\xi_k}[F(w_{k+1})] - F(w_k) \leq -\alpha_k \nabla F(w_k)^T E_{\xi_k}[g(w_k,\xi_k)] + \frac{\alpha_k^2 L}{2} E_{\xi_k}\left[\|g(w_k,\xi_k)\|^2\right] + O(\alpha_k^3)$$

如果 $E_{\xi_k}\left[\|g(w_k,\xi_k)\|^2\right]$ 可以快速下降，噪声就并不会影响收敛。事实上，可以证明只要方差 $\mathrm{Var}_{\xi_k}[g(w_k,\xi_k)]$ 减小足够快，就能有良好的收敛性态：

定理 10 (强凸函数-方差衰减收敛性)**.** 在 $F(w)$ 有界为 L 的梯度 Lipschitz 连续性与参数为 c 的强凸性时，进一步要求：

- 存在 $\mu_G \geq \mu > 0$，使得对一切 k 有

$$\nabla F(w_k)^T E_{\xi_k}[g(w_k,\xi_k)] \geq \mu \|\nabla F(w_k)\|^2$$

$$E_{\xi_k}[\|g(w_k,\xi_k)\|] \leq \mu_G \|\nabla F(w_k)\|$$

- 存在 $M \geq 0$ 与 $\zeta \in (0,1)$，使得对一切 k 有

$$\operatorname*{Var}_{\xi_k}[\|g(w_k,\xi_k)\|] \leq M\zeta^{k-1}$$

步长 α_k 恒定为 $\bar{\alpha}$，满足

$$0 < \bar{\alpha} \leq \min\left\{\frac{\mu}{L\mu_G^2}, \frac{1}{c\mu}\right\}$$

则迭代过程中 k 充分大时有

$$E[F(w_k) - F_*] \leq \omega \rho^{k-1}, \quad \omega = \max\left\{\frac{\bar{\alpha} LM}{c\mu}, F(w_1) - F_*\right\}, \quad \rho = \max\left\{1 - \frac{c\mu\bar{\alpha}}{2}, \zeta\right\}$$

证明. 由于 $E_{\xi_k}\left[\|g(w_k,\xi_k)\|^2\right] \le M\zeta^{k-1} + \mu_G^2\|\nabla F(w_k)\|^2$，代入可以类似之前得到

$$E_{\xi_k}[F(w_{k+1})] - F(w_k) \le -\left(\mu\bar{\alpha} - \frac{\bar{\alpha}^2 L}{2}\mu_G^2\right)\|\nabla F(w_k)\|^2 + \frac{\bar{\alpha}^2 L}{2}M\zeta^{k-1} + O(\bar{\alpha}^3)$$

再根据 $\bar{\alpha} < \frac{\mu}{L\mu_G^2}$ 放缩并省略 $O(\bar{\alpha}^3)$ 有

$$E_{\xi_k}[F(w_{k+1})] - F(w_k) \le -\frac{\mu\bar{\alpha}}{2}\|\nabla F(w_k)\|^2 + \frac{\bar{\alpha}^2 L}{2}M\zeta^{k-1}$$

再根据强凸函数性质得

$$E_{\xi_k}[F(w_{k+1})] - F(w_k) \le -c\mu\bar{\alpha}(F(w_k) - F_*) + \frac{\bar{\alpha}^2 L}{2}M\zeta^{k-1}$$

同加并取期望有

$$E[F(w_{k+1}) - F_*] \le (1 - c\mu\bar{\alpha})E[F(w_k) - F_*] + \frac{\bar{\alpha}^2 L}{2}M\zeta^{k-1}$$

利用归纳法，第一项满足要求，若 $E[F(w_k) - F_*] \le \omega\rho^{k-1}$，则

$$E[F(w_{k+1}) - F_*] \le (1 - c\mu\bar{\alpha})\omega\rho^{k-1} + \frac{\bar{\alpha}^2 L}{2}M\zeta^{k-1}$$

由于 $\zeta \le \rho, \frac{\bar{\alpha}LM}{c\mu} \le \omega,$，即有

$$E[F(w_{k+1}) - F_*] \le (1 - c\mu\bar{\alpha})\omega\rho^{k-1} + \frac{c\mu\bar{\alpha}\omega}{2}M\rho^{k-1} = \left(1 - \frac{c\mu\bar{\alpha}}{2}\right)\omega\rho^{k-1} \le \omega\rho^k$$

□

在实践中，根据这些量通常的范围，步长 $\bar{\alpha}$ 的取值一般是符合要求的。为了说明这个定理与动态采样的关系，我们先考虑如下的问题：

定理 11 (随机采样方差性质)**.** 已知 n 个数 $x_1, \ldots, x_n$，记随机变量 X_k 为从它们之中等概率随机抽取 k 个不同的数的均值，则

$$\mathrm{Var}(X_k) = \frac{n-k}{k(n-1)}\,\mathrm{Var}(x)$$

证明. 由于同平移不影响问题结论，可不妨设这些数均值为 0，则抽取 $x_{i_1}, \ldots, x_{i_k}$ 的概率为 $\frac{1}{C_n^k}$，误差为 $\frac{\sum_{t=1}^k x_{i_t}}{k}$。于是方差为

$$\frac{1}{C_n^k}\sum_{i_1,\ldots,i_k}\left(\frac{x_{i_1} + \cdots + x_{i_k}}{k}\right)^2$$

由于乘积中只含有二次项与交叉项，分析可得其为

$$\frac{1}{k^2 C_n^k}\left(C_{n-1}^{k-1}\sum_i x_i^2 + 2C_{n-2}^{k-2}\sum_{i<j} x_i x_j\right) = \frac{1}{nk}\left(\sum_i x_i^2 + \frac{2(k-1)}{n-1}\sum_{i<j} x_i x_j\right)$$

假定 $\sum_i x_i = 0$ 后，平方得 $\sum_i x_i^2 + 2\sum_{i<j} x_i x_j = 0$，消去交叉项有其为

$$\frac{n-k}{k(n-1)} \frac{\sum_i x_i^2}{n} = \frac{n-k}{k(n-1)} \operatorname{Var}(x)$$

□

此方差恒小于 $\frac{\operatorname{Var}(x)}{k}$，且在 n 远大于 k 时逼近 $\frac{\operatorname{Var}(x)}{k}$。这说明，只要我们每次取的样本个数指数增长，就能使方差指数下降。在这种最简单的思路下可以得到算法：

算法 3 (动态采样法-迭代)**.** 初始给定 $\tau > 1$，对每次抓取的样本 $\xi_1, \ldots, \xi_n$，抽取 $n_k = \lceil \tau^k \rceil$ 个下标，记为 $\{k,1\}, \ldots, \{k,n_k\}$，再进行迭代

$$w_{k+1} = w_k - \alpha_k \frac{1}{n_k} \sum_{i=1}^{n_k} \nabla f_{k,i}(w_k)$$

根据上方分析，它完全符合方差衰减的条件，因此在 F 是强凸函数时能以指数速度逼近。值得注意的是，由于采样大小增长的要求，每次迭代的运算时间也在以指数形式增长。为了证明此时仍然能起到加快收敛的效果，我们再给出一个结论：

定理 12 (动态采样法-收敛速率)**.** 假设动态采样的随机梯度法满足 F 的强凸性与二阶矩性质，且 $1 < \tau \leq \left(1 - \frac{c\mu\bar{\alpha}}{2}\right)^{-1}$，则存在 C, D 使得对任何 ϵ，达到

$$E[F(w_k) - F_*] \leq \epsilon$$

的误差需要的梯度计算次数不超过 $C\frac{1}{\epsilon} + D$ 次。

证明. 根据动态采样法的定义，更新至 w_{k+1} 需要的运算次数至多为 (最后一项由向上取整产生)

$$\tau + \tau^2 + \cdots + \tau^k + k = \frac{\tau^{k+1} - \tau}{\tau - 1} + k$$

另一方面，注意到这里的 τ 与收敛性定理中 $\frac{1}{\zeta}$ 一致，根据范围要求可知 $\rho = \zeta = \frac{1}{\tau}$，于是有

$$E[F(w_k) - F_*] \leq \frac{\omega}{\tau^{k-1}}$$

考虑 ϵ 比 $\frac{\omega}{\tau^{k-1}}$ 略小，则必须计算 w_{k+1} 才能保证，代入消去 k 可得达到 ϵ 需要的计算次数至多为

$$\frac{\frac{\omega}{\epsilon}\tau^2 - \tau}{\tau - 1} + \log_\tau \frac{\omega}{\epsilon} + 1 = \frac{\tau^2 \omega}{\tau - 1}\frac{1}{\epsilon} + \log_\tau \omega - \frac{1}{\tau - 1} + \frac{\ln \frac{1}{\epsilon}}{\ln \tau}$$

由 $x > 0$ 时 $\ln x < x$ 可取 $C = \frac{\tau^2\omega}{\tau - 1} + \frac{1}{\ln \tau}, D = \log_\tau \omega - \frac{1}{\tau - 1}$。 □

也即，事实上其可以保证与时间反比的收敛速率。根据 C 的表达式，可以得到理论上使得 C 取到下界的 τ，但实际中的批次扩大倍数一般通过实验得到。

§3.1.2 实际测试

仍然采用 Logisitic 回归的例子，并且考虑较大的步长使得振荡明显，批次大小恒为 1 时如**图 12** 所示，固定步长几乎无法收敛。对接下来的两种降噪手段，初始情况仍利用此参数选取作为对比。

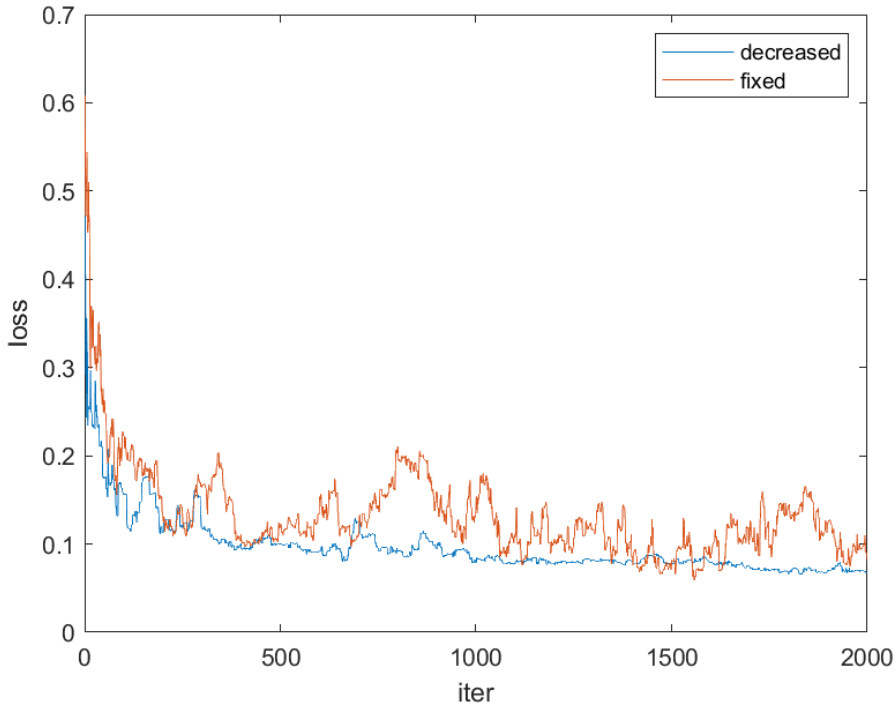

图 12: 无降噪迭代

采用动态采样的思路后，取初始批次大小为 1，扩大倍数分别为 1.001、1.002、1.004，可以得到结果为**图 13** 。

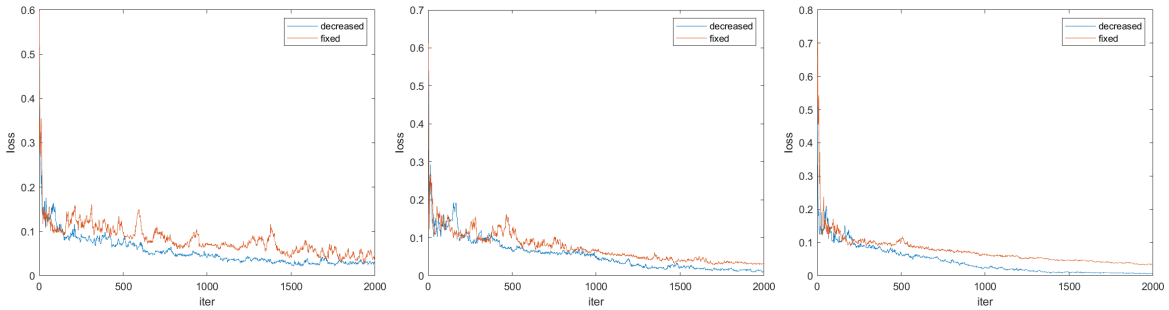

图 13: 动态采样法

从中确实可以看出，动态采样有效减少了结果中的振荡，使得收敛性更加易于判别，也能达到随时间更快的收敛速率。扩大倍数为 1.004 的动态采样法在 2000 次迭代后，几乎可以完全精确地判断分类边界，而估算可知这仅相当于整体进行了 10 次左右的迭代。

第 2 节　梯度聚合

§3.2.1　理论分析

比起每次抓取新的更多样本，梯度聚合法选择利用已经计算出的估计值。其中一个完整的例子是随机方差衰减梯度 [stochastic variance reduced gradient, SVRG] 算法[5]。

算法 4 (SVRG 算法)**.** 算法流程为：

1. 给定初始 w_1，步长 $\alpha > 0$ 与正整数 m，令 $k = 1$。
2. 计算 $\nabla F(w_k)$，并记 $\tilde{w}_1 = w_k$。
3. 重复执行 m 次 (记当前次数为 j)：从 1 到 n 中抽取一个下标 i_j，并更新

$$\tilde{w}_{j+1} = \tilde{w}_j - \alpha\big(\nabla f_{i_j}(\tilde{w}_j) - (\nabla f_{i_j}(w_k) - \nabla F(w_k))\big)$$

4. 选择一种方式计算 w_{k+1}：
   - $w_{k+1} = \tilde{w}_{m+1}$
   - $w_{k+1} = \frac{1}{m}\sum_{j=1}^{m} \tilde{w}_{j+1}$
   - 从 1 到 m 中抽取一个 j，记 $w_{k+1} = \tilde{w}_{j+1}$
5. 判定是否终止，若否则 $k = k + 1$，回到第二步。

证明. 我们先证明 $\tilde{g}_j = \nabla f_{i_j}(\tilde{w}_j) - (\nabla f_{i_j}(w_k) - \nabla F(w_k))$ 构成 $\nabla F(\tilde{w}_j)$ 的无偏估计。

$$E_{i_j}[\tilde{g}_j] = E_{i_j}[\nabla f_{i_j}(\tilde{w}_j)] - E_{i_j}[\nabla f_{i_j}(w_k)] + \nabla F(w_k)$$

于是 $E_{i_j}[\tilde{g}_j] = \nabla F(\tilde{w}_j) - \nabla F(w_k) + \nabla F(w_k)$，得证。

另一方面，由于第二项的稳定作用，在 $\tilde{w}_j$ 与 w_k 距离不远时，$\nabla f_{i_j}(\tilde{w}_j) - \nabla f_{i_j}(w_k)$ 不会太大，可以说明 $\mathrm{Var}_{i_j}[\tilde{g}_j]$ 比 $\mathrm{Var}_{i_j}[\nabla f_{i_j}(\tilde{w}_j)]$ 更小，于是如此构造的 $\tilde{g}_j$ 有更好的性质。　□

这个算法中，我们每次先计算完整的梯度，再以其作为估计对 w_k 进行多次更新，并集成结果。相较计算完整梯度后直接更新，它能在同样的更新范围中达到更精细的更新。其收敛速率有结论：

定理 13 (SVRG-收敛速率)**.** 在 SVRG 算法中，若 F 是参数为 c 的强凸函数与界为 L 的梯度 Lipschitz 函数，设步长 α 与内循环迭代次数 m 满足

$$\rho = \frac{1}{1 - 2\alpha L}\left(\frac{1}{mc\alpha} + 2L\alpha\right) < 1$$

则对后两种方式更新的 w_{k+1} 有 $E[F(w_{k+1}) - F(w^*)] \leq \rho E[F(w_k) - F(w^*)]$，即 $\{w_k\}$ 满足线性收敛速率。

值得注意的是，SVRG 一次外循环需要计算 $2m+n$ 个梯度，因此外循环计算开销和直接进行梯度下降同量级。若能把核心部分 $\nabla F(w)$ 的计算简化，我们就能获得更高的效率。在计算机中，一个常用的想法是用空间换时间，即通过储存已经计算的结果来加快后续结果的计算速度。此处这样的方法称为随机平均梯度 [Stochastic Average Gradient, SAG] 算法，而更有效的是其加速版本，SAGA 算法[6]：

算法 5 (SAGA 算法)**.** 算法流程为：

1. 给定初始 w_1，计算向量 $\tilde{g}_i = \nabla f_i(w_1), i = 1, \ldots, n$。

2. 步长 $\alpha > 0$，令 $k = 1$。

3. 从 1 到 n 中抽取一个下标 j，计算

$$g_k = \nabla f_j(w_k) - \tilde{g}_j + \frac{1}{n}\sum_{i=1}^{n} \tilde{g}_i$$

4. 更新 $\tilde{g}_j = g_k, w_{k+1} = w_k - \alpha g_k$。

5. 判定是否终止，若否则 $k = k + 1$，回到第二步。

算法中，$\tilde{g}_j$ 记录的是 f_j 最近计算的一次梯度，并以此作为真实梯度的估计。由于定义，与上个算法相同可知 g_k 是无偏估计，并一定程度减小了方差。对其收敛速率则有结论：

定理 14 (SAGA-收敛速率)**.** 在 SAGA 算法中，若 F 是参数为 c 的强凸函数与界为 L 的梯度 Lipschitz 函数，设步长 $\alpha = \frac{1}{2(cn+L)}$，则有

$$E[\|w_k - w^*\|^2] \leq \left(1 - \frac{c}{2(cn+L)}\right)^k \left(\|w_1 - w^*\|^2 + \frac{n(F(w_1) - F(w^*))}{cn+L}\right)$$

事实上，SAGA 算法也可以采取其他的初始化策略，如先用普通随机梯度法进行一些迭代等。虽然此算法在高维时有很大的空间开销，在一些特殊问题时仍可以简化，如当 $f_i(w) = \hat{f}(\alpha_i^T w)$ 时，有 $\nabla f_i(w) = \hat{f}'(\alpha_i^T w)\alpha_i$，由此只要存储了每个 α_i，只需要额外保存一个标量 $\hat{f}'(\alpha_i^T w)$ 即可，这在一些回归模型中常会出现。

虽然上述梯度聚合方法在达到指定误差时的收敛比普通 SG 算法快，但它们事实上并不明显优于 SG 算法。类似之前对计算时间的分析，对参数为 c 的强凸函数与界为 L 的梯度 Lipschitz 函数 F，记 $\kappa = \frac{L}{c}$，当 n 维时欲达到 ϵ 误差，直接 SG 算法的梯度计算次数正比于 $\frac{\kappa^2}{\epsilon}$，而 SAGA 与 SVRG 则正比于 $-(n+\kappa)\ln\epsilon$。随着 n 的增大，梯度聚合方法的计算次数会变得更多。

§3.2.2 实际测试

在 10000、100000、1000000 个样本的训练集用 SAGA 算法进行 Losigitic 回归，得到的结果如**图 14** 。

样本集越大，梯度聚合法需要的初始计算次数就越多，但迭代中的效果也会越好，这符合理论的推导结果。事实上，此现象在步长逐渐衰减时尤为明显。

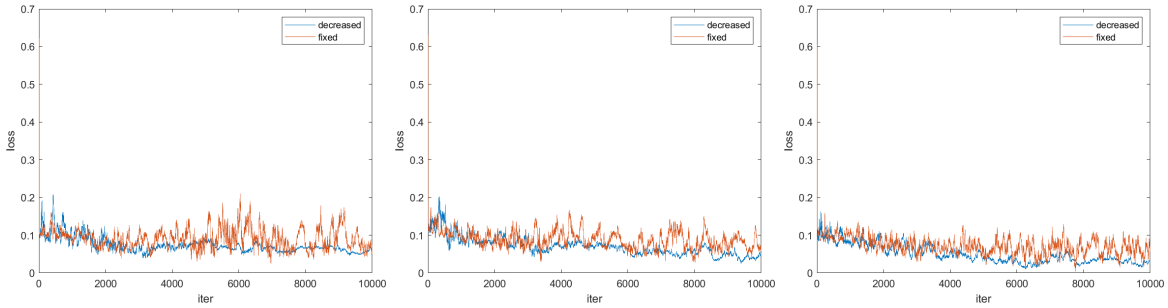

图 14: 改变样本集大小

值得一提的是，由于 SAGA 算法本质是一种单个抽样的算法，其步长应该相对取小才能使结果的振荡减弱。以 100000 大小的训练集为例，步长从大到小的效果如**图 15** 。

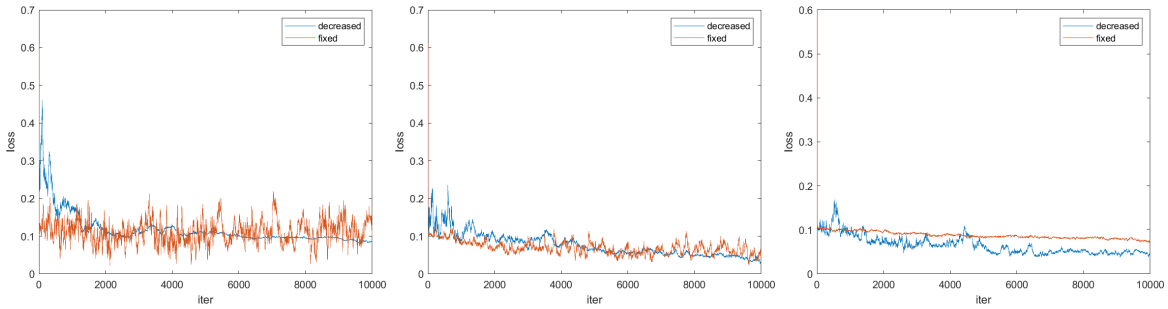

图 15: 改变步长大小

对固定步长而言，步长越小可以使振荡越小，收敛性质也会更好，但速度会有一定程度放慢。对衰减步长，性质是类似的，只要保证衰减速率一定，初始步长可以取得较大。在此例子中，相比预处理的时间（100000 次梯度计算），实际的迭代时间相对短，因此可取更小步长以保证良好的收敛性。

第 3 节 迭代平均

§3.3.1 理论分析

迭代平均缘于一个简单的想法：由于普通 SG 方法在每次迭代中抽取一些，若能将这些抽取的结果平均，亦能减少噪声：

算法 6 (迭代平均法-迭代)**.** 给定步长 $\alpha_k > 0$，每次迭代中，先计算 $w_{k+1} = w_k - \alpha_k g(w_k, \xi_k)$，再令最终更新为

$$\tilde{w}_{k+1} = \frac{1}{k+1}\sum_{j=1}^{k+1} w_j$$

其计算过程与普通 SG 算法完全相同，只是没有选择最新计算出的点，而是选择了所有计算出点的平均。对它的收敛速率有结论：

定理 15 (迭代平均-收敛速率)**.** 对符合衰减速率 $O\left(k^{-a}\right), a \in (0.5, 1)$ 的步长 α_k，普通 SG 算法迭代平均后的收敛速率满足

$$E[\|w_k - w^*\|^2] = O\left(k^{-a}\right), E[\|\tilde{w}_k - w^*\|^2] = O\left(k^{-a}\right)$$

证明. $E[\|w_k - w^*\|^2]$ 的估算与之前衰减步长 SG 算法收敛性完全相同，而后者的估算只需利用柯西不等式

$$\left(\frac{a_1 + a_2 + \cdots + a_n}{n}\right)^2 \leq \frac{a_1^2 + \cdots + a_n^2}{n}$$

代入 $a_i = \|w_i - w^*\|$ 即可知右侧可以被左侧的 Cesaro 平均控制，而当原数列为 $O\left(k^{-a}\right)$ 时，根据数学分析知识可计算得 Cesaro 平均与原数列的收敛速率相同，从而得证。 □

由此即可知迭代平均法改善了收敛效果。

§3.3.2 实际测试

分别考虑批次大小为 1、3，以迭代平均法进行 10000 次迭代得到**图 16** 。

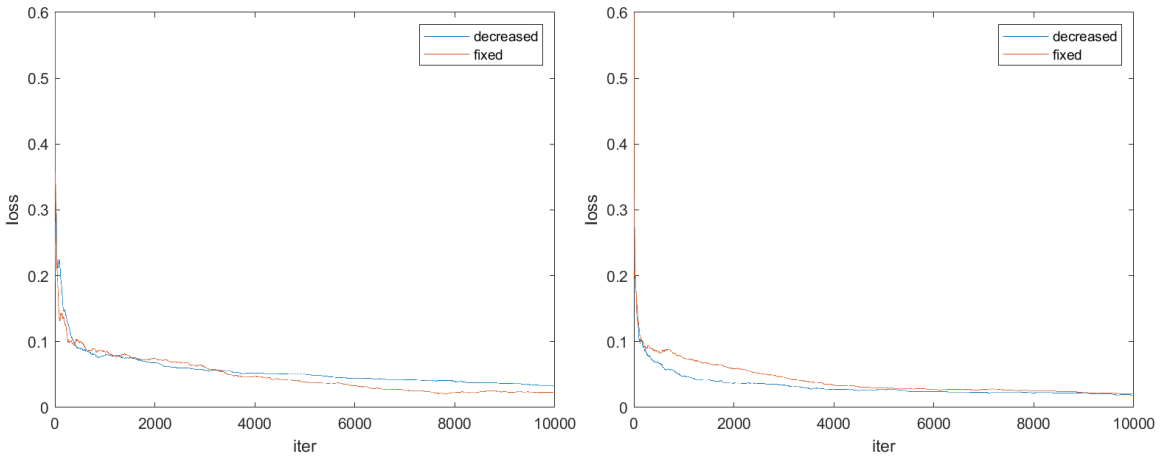

图 16: 迭代平均法

无论是从理论推导还是实际测试中，迭代平均法对光滑性的改善都是十分明显的。并且，虽然表面上它会将迭代计算的成本从迭代次数的一次量级提升到二次量级，但实际上通过

$$\tilde{w}_{k+1} = \frac{k\tilde{w}_k + w_{k+1}}{k+1}$$

完全可以在一次量级解决。

第 4 章　二阶手段

另一种对 SG 算法的改进方式为利用上 F 的二阶信息。就像梯度类方法利用二阶信息后成为牛顿类方法，利用二阶信息的随机算法称为**随机牛顿法**。由于牛顿类方法往往在最优解附近有二阶收敛速率，随机牛顿法一般有更好的收敛效果。

我们主要考虑五种随机牛顿方法，其中，无海森牛顿法与自然梯度法需要批次有一定规模时才有效，而剩下的对角缩放、拟牛顿与高斯-牛顿方法则无需此限制。

第 1 节　拟牛顿法

§4.1.1　无海森牛顿法

考虑一般牛顿法的迭代过程：

$$w_{k+1} = w_k + \alpha_k s_k, \quad \nabla^2 F(w_k) s_k = -\nabla F(w_k)$$

事实上，我们并不需要精确求解这个方程，只需要进行一些近似求解（如利用共轭梯度法），只要保证解的接近性，即可满足超线性的收敛速度。在共轭梯度法近似求解的过程中，不需要显式计算出海森阵，只会出现它与向量的乘积，因此称为无海森牛顿法。

另一方面，由于此方法不要求精确求解，对 ∇F 与 $\nabla^2 F$ 都可以抽取一些 f 并平均估算 (一般估算 $\nabla^2 F$ 取的样本个数更少)。当然，取样过多会引起计算速度降低，取样过少则导致估算并不准确，这其中需要一些经验性的选择。

算法 7 (采样无海森牛顿算法)**.** 算法流程为：

1. 给定初始 w_1，$\rho \in (0,1), \eta \in (0,1), \gamma > 1$，正整数 m，令 $k = 1$。

2. 从 1 到 n 中抽取一族下标 S_k，并从 S_k 中抽取一部分 $S_k^{(H)}$。计算 (这里 # 代表元素个数)

$$g_k = \nabla f_{S_k}(w_k) = \frac{1}{\#S_k} \sum_{i \in S_k} \nabla f(w_k, \xi_i)$$

$$H_k = \nabla^2 f_{S_k^{(H)}}(w_k) = \frac{1}{\#S_k^{(H)}} \sum_{i \in S_k} \nabla^2 f(w_k, \xi_i)$$

3. 用共轭梯度法求解 $H_k s = g_k$，直到达到迭代次数上限 m 或误差满足 $\|H_k s + g_k\| \le \rho \|g_k\|$。

4. 令 $\alpha_k = 1$，若

$$f_{S_k}(w_k + \alpha_k s_k) \le f_{S_k}(w_k) + \eta \alpha_k g_k^T s_k$$

则放大 $\alpha_k = \gamma \alpha_k$，取使其能成立的最大可能 $\alpha_k = \gamma^{p_k}$ 作为更新步长。

5. 更新 $w_{k+1} = w_k + \alpha_k s_k$，判定是否终止，若否则 $k = k + 1$，回到第二步。

当噪声较大时，$\#S_k^{(H)}$ 必须取得较大才能避免海森阵导致迭代出现问题，因此 $\#S_k$ 较大时才可能采用此方法。此外，虽然对精确计算 ∇F 与 $\nabla^2 F$ 的无海森牛顿法可以证明收敛，加入采样后并没有办法保证超线性的收敛速率。

更多时候，问题是非凸的，这样的搜索很容易出现问题，因此需要在上方第四步调整为：用共轭梯度法求解 $H_k s = g_k$，直到达到迭代次数上限 m 或误差满足 $\|H_k s + g_k\| \leq \rho \|g_k\|$ 或 s_k 是负曲率方向，即 $s_k^T H_k s_k < 0$。当然，比起处理不正定的海森阵，我们更希望能维持一个正定或半正定的估计。

§4.1.2 随机拟牛顿法

为克服上述阻碍，我们考虑无约束优化中直接估计海森阵的拟牛顿法，其更新形式满足

$$w_{k+1} = w_k - \alpha_k H_k \nabla F(w_k)$$

其中 H_k 为对 $\nabla^2 F(w_k)^{-1}$ 的估算。

其中常用的 BFGS 方法，利用对 H_k 的动态更新，在仅使用一阶信息时通过估计二阶信息在局部达到了超线性收敛速率。然而，由于 H_k 不具有稀疏性 (即使真实的海森阵是稀疏的)，大规模计算中其存储开销非常大，我们需要考虑有限记忆策略，例如不需要显式计算出 H_k 的 L-BFGS 方法[7]。具体来说，我们记录集合 P，其中储存着若干对 s 与 y，并每次根据其中的 s 与 y 计算出 $H_k g$：

算法 8 (L-BFGS 迭代)**.** 给定包含 m 对 $(s_0, y_0), \ldots, (s_{m-1}, y_{m-1})$ 的集合 P，则根据 BFGS 迭代产生的矩阵为

$$H_0 = I$$

$$H_{k+1} = \left(I - \frac{y_k s_k^T}{s_k^T y_k}\right)^T H_k \left(I - \frac{y_k s_k^T}{s_k^T y_k}\right) + \frac{s_k s_k^T}{s_k^T y_k}$$

由于每个 $I - \frac{y_k s_k^T}{s_k^T y_k}$ 或 $\frac{s_k s_k^T}{s_k^T y_k}$ 与向量的乘积均可只通过向量运算得到，展开 $H_m g$ 可得到不含矩阵运算的计算方式。

即使这样，引入随机后利用采样估算的梯度 $g(w_k, \xi_k)$ 进行的更新

$$w_{k+1} = w_k - \alpha_k H_k g(w_k, \xi_k)$$

也会遇到诸多问题：

1. 理论局限性：引入随机后，此方法无法达到线性收敛速率，与一般的 SG 算法并无量级上的区别。不过，由于常数上的改进，随机采样下的拟牛顿法仍然存在意义。可以证明，当 $H_k \to \nabla^2 F(w^*)^{-1}$ 时，其常数较一般随机梯度法更好。

2. 较长的单次迭代时间：设 m 为 L-BFGS 方法的记忆周期 (通常为 5)，d 为计算 $g(w_k, \xi_k)$ 所需的操作次数，则 $H_k g(w_k, \xi_k)$ 的计算需要 $4md$ 次计算，也即直接计算 $g(w_k, \xi_k)$ 的 20 倍左右。为解决此问题，一般估计 $g(w_k, \xi_k)$ 需要采用相对较小的批次，如设定为 256。

3. 估计海森阵的训练过程：估算 H_k 需要用到 $g(w_k, \xi_k)$ 与 $g(w_{k+1}, \xi_{k+1})$ 之间的差，但二者都是估计而成，可能造成很大的误差。

4. 误差的累计：与普通拟牛顿法一样每步都对 H_k 进行更新可能没有必要，或反而引起误差累积导致估计更加不准确。

对后两个问题有不同的解决方法，例如采取更好的对 y_k 的估计方式。

算法 9 (随机拟牛顿法-估计 y_k)**.** 已知 w_{k+1} 与 w_k 时，有 $s_k = w_{k+1} - w_k$，进一步假设已选出 S_k 并估计了梯度 $g(w_k, \xi_k) = \nabla f_{S_k}(w_k)$，可由如下方法估计对应的 $y_k = \nabla F(w_{k+1}) - \nabla F(w_k)$：

1. 令 $y_k = \nabla f_{S_k}(w_{k+1}) - \nabla f_{S_k}(w_k)$；

2. 从 S_k 中抽取一部分 $S_k^{(H)}$，令 $y_k = \nabla^2 f_{S_k^{(H)}}(w_k) s_k$。

前者通过相同采样的方式避免了过大的误差，后者则根据 $y = Hs$ 的要求计算 y_k，将梯度计算与海森阵更新**解耦** [decouple]。综合估计 y_k 的方法，我们可以得到完整的随机拟牛顿法：

算法 10 (随机拟牛顿法)**.** 算法流程为：

1. 给定初始 w_1，正整数 m，恒正的步长序列 α_k。令 P 为空集，$k = 1$。

2. 抽取 S_k，并计算

$$g_k = \nabla f_{S_k}(w_k)$$

$$w_{k+1} = w_k - \alpha_k H_k g_k$$

这里 $H_k g_k$ 由 P 中的 (s, y) 通过 L-BFGS 方法得出。

3. 判定是否需要更新 H_k，若是则继续，否则令 $k = k + 1$，回到第二步。

4. 任选一种方式估计 y_k，并将 (s_k, y_k) 添加到 P 中，若 P 中数量超过 m，丢弃其中下标最小的 (s_i, y_i)。

5. 令 $k = k + 1$，回到第二步。

§4.1.3 实际测试

由于测试 LBFGS 的特点需要的规模较大，我们采用 Pytorch 自带的 LBFGS 求解器与基于一阶信息的 Adam 进行对比。考虑风格迁移问题[1]，采用论文中算法在 VGG 网络的各层级进行迭代优化时，假设原图片与目标图片为**图 17**，下面考虑不同优化器的效果。

采用 Adam 优化器进行风格迁移在 0、100、200、300、400、500 次迭代的效果如**图 18**，而损失函数（根据论文，损失分为内容与风格两部分，其和为总损失）如**图 19** 。

图 17: 原图片与目标图片

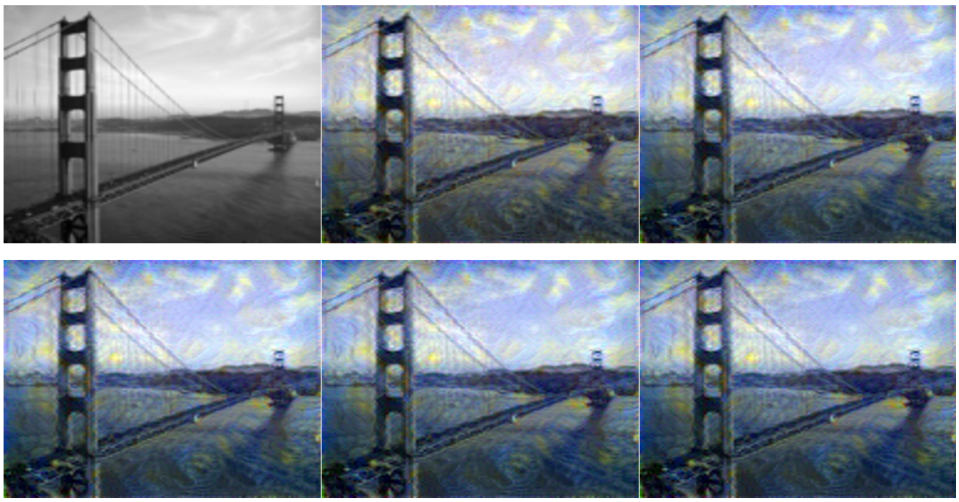

图 18: 一阶迭代方法

可以看出，一阶优化器的整体效果是较为光滑的，损失在迭代中逐渐收敛后几乎保持不变。采用 LBFGS 优化器后，迭代效果与损失函数如**图 20** 、**图 21** 。由于随机方法的引入，其速度事实上要快于 Adam 优化器，但这也导致它更加不光滑。此外，作为拟牛顿法，其有时需要进行重新启动才能保证较好的收敛，如 400 次迭代左右的尖峰。

不过，对比两种方法也能发现，实际应用中 LBFGS 的收敛速度是高于梯度方法的，这也展示了二阶方法在收敛性上的优势。在优化目标具有较好光滑性时，引入二阶信息可以起到大幅加速收敛的效果。

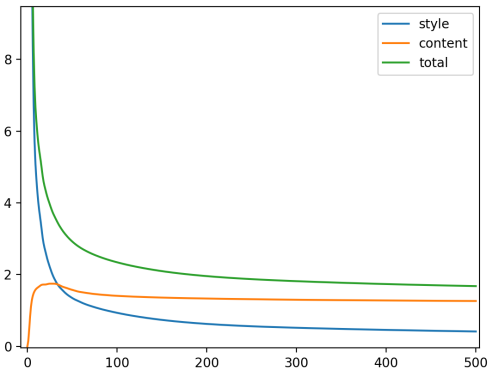

图 19: 一阶迭代损失函数

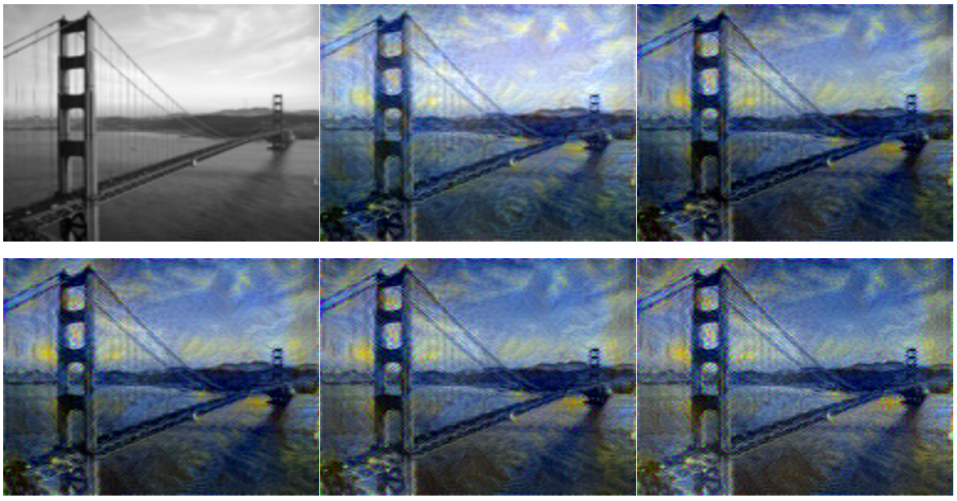

图 20: LBFGS 迭代方法

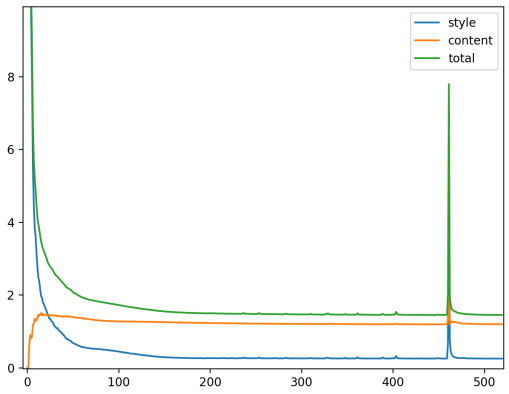

图 21: LBFGS 迭代损失函数

第 2 节　半正定性的保持

虽然随机拟牛顿法避免了对海森阵的复杂计算与存储，仍然存在更新过程中未必正定导致迭代误差很大的问题。本节的主题即为解决此问题。

§4.2.1 高斯-牛顿法

高斯-牛顿方法从估算的角度作出了改进，其在真实的海森阵非正定时也能保证估计过程的半正定性。

回到基础定义，对 $\xi = (\xi_x, \xi_y)$ 与参数 w 产生的预测函数 h_w，考虑最小二乘误差

$$f_w(\xi) = l(h_w(\xi_x), \xi_y) = \frac{1}{2}\|h_w(\xi_x) - \xi_y\|^2$$

在给定 ξ 时记 h_w 对 w 的 Jacobi 阵为 $\mathcal{J}_{h_\xi}(w)$，则有

$$h_w(\xi_x) = h_{w_k}(\xi_x) + \mathcal{J}_{h_\xi}(w_k)(w - w_k) + O(\|w - w_k\|^2)$$

于是计算可得 $f_w(\xi)$ 忽略二阶项近似为

$$\frac{1}{2}\|h_{w_k}(\xi_x) - \xi_y\|^2 + \big(h_{w_k}(\xi_x) - \xi_y\big)^T \mathcal{J}_{h_\xi}(w_k)(w - w_k) + \frac{1}{2}(w - w_k)^T \mathcal{J}_{h_\xi}(w_k)^T \mathcal{J}_{h_\xi}(w_k)(w - w_k)$$

若将其看作对 f 的展开，海森阵即为 $\mathcal{J}_{h_\xi}(w_k)^T \mathcal{J}_{h_\xi}(w_k)$。由此思想，我们可以作近似：

定义 3 (高斯-牛顿阵)**.** 对一族下标 $S_k^{(H)}$，其对最小二乘误差产生的高斯-牛顿阵为

$$G_{S_k^{(H)}}(w_k) = \frac{1}{\#S_k^{(H)}} \sum_{i \in S_k^{(H)}} \mathcal{J}_{h_{\xi_i}}(w_k)^T \mathcal{J}_{h_{\xi_i}}(w_k)$$

对一般的凸误差函数 $l(y_1, y_2)$，高斯-牛顿阵则为

$$G_{S_k^{(H)}}(w_k) = \frac{1}{\#S_k^{(H)}} \sum_{i \in S_k^{(H)}} \mathcal{J}_{h_{\xi_i}}(w_k)^T \nabla_h^2 l(h, \cdot) \mathcal{J}_{h_{\xi_i}}(w_k)$$

定理 16 (正定性)**.** 对任何凸误差函数，高斯-牛顿阵半正定，且对 $\lambda > 0$ 有 $\lambda I + G_{S_k^{(H)}}(w_k)$ 正定。

证明. 由凸函数二阶条件可知 $\nabla_h^2 l(h, \cdot)$ 半正定，从而根据定义可知 $G_{S_k^{(H)}}(w_k)$ 半正定，进一步得到 $\lambda I + G_{S_k^{(H)}}(w_k)$ 正定。 □

由此，一般增加 λI 后得到正定的矩阵作为海森阵的近似，从而进行迭代。此方法的计算成本取决于预测函数的维度。事实上，在机器学习中，计算随机梯度向量 $\nabla f_\xi(w)$ 通常不需要明确计算 Jacobi 矩阵的所有行，此外，也可以有其他低成本解决高斯-牛顿迭代问题的方案。

§4.2.2 对角缩放法

为了进一步降低每次迭代运算次数，可以采用对角缩放法。在高斯-牛顿法中，我们虽然能够近似海森阵，却仍然需要近似其逆才能计算出迭代更新的方向。而对角缩放法的思路则是迭代更新高斯-牛顿阵，并直接将其逆近似为对角元对应取逆的对角阵，具体来说为：

算法 11 (对角缩放法-迭代)**.** 若每一步取出样本 ξ_k，对应梯度为 $g(w_k,\xi_k)$，当前高斯-牛顿阵对角元构成的向量为 G_k，则迭代过程先计算 w_{k+1}(下标 i 表示第 i 个分量，对所有分量如此计算)：

$$w_{k+1,i}=w_{k,i}-\frac{\alpha}{G_{k,i}+\mu}g(w_k,\xi_k)_i$$

再更新 G_k (下标 ii 即第 i 个对角元)：

$$G_{k+1,i}=(1-\lambda)G_{k,i}+\lambda\big(\mathcal{J}_{\xi_{k+1}}(w_{k+1})^T\mathcal{J}_{\xi_{k+1}}(w_{k+1})\big)_{ii}$$

这里 μ 为给定正数，避免对角元太小，α 为步长因子，$\lambda\in(0,1)$ 为更新权重。

此算法中，对于高斯-牛顿阵利用权重衰减的方法进行近似，且只保存与更新其对角元，并完全用对角元近似 Hg_k。比起二阶方法，它其实更接近对 g_k 进行一定幅度调整的一阶方法。

除了这样直接通过 Jacobi 阵计算对角元与其倒数外，也可以采用近似计算的方案。例如，在拟牛顿法中，可以考虑由当前的 s_k,y_k 迭代更新 H_k 的对角元 $H_{k,i}$：

$$H_{k+1,i}=(1-\lambda)H_{k,i}+\lambda\mathrm{Proj}\left(\frac{s_{k,i}}{y_{k,i}}\right)$$

这里 Proj 代表向某个正区间 $[a,b]$ 投影，即将小于 a 的置为 a，大于 b 的置为 b，其余不变。然而，由于 H_k 无法控制，直接运用这样的方法是噪声较大且难以纠正的，于是需要寻求更好的办法。

其中一个想法是，迭代更新海森阵 G_k 而非其倒数 H_k，再直接计算倒数，更新方式为

$$G_{k+1,i}=G_{k,i}+\mathrm{Proj}\left(\frac{y_{k,i}}{s_{k,i}}\right)$$

注意对角缩放法的迭代步骤可发现，Proj 的投影保证了实际更新的步长是以 $O\big(\frac{1}{k}\big)$ 下降的。

§4.2.3 自然梯度法

我们最后考察的优化方法是自然梯度法[8]，其想法为在空间 $\mathcal{H}$ 中（而非参数空间下）进行梯度下降迭代，因此得名。

假设参数空间中所有的函数都是密度函数，即 $\int h_w(x)\mathrm{d}x=1$ 且 $h_w(x)\geq 0$，由此可以定义 E_{h_w} 等，并假设其充分正则 (由归一化条件可知此式为 0)：

$$\forall t>0,\int\frac{\partial^t}{\partial w^t}h_w(x)\mathrm{d}x=\frac{\partial^t}{\partial w^t}\int h_w(x)\mathrm{d}x$$

这里 $\frac{\partial^t}{\partial w^t}$ 指的是对任一种对不同分量求总计 t 阶导数的方式。

我们先定义 KL 散度：

定义 4 (KL 散度). 对两个非零的密度函数 h_1, h_2，可定义 h_2 对 h_1 的 KL 散度 [Kullback-Leibler divergence] 为

$$D_{KL}(h_1\|h_2) = E_{h_1}\left[\ln\frac{h_1(x)}{h_2(x)}\right]$$

值得注意的是，$D_{KL}(h_1\|h_2)$ 未必与 $D_{KL}(h_2\|h_1)$ 相同，因此这并非对称的定义。。

定理 17 (KL 散度的性质). $D_{KL}(h_1\|h_2) \geq 0$，且其取到 0 当且仅当 h_1 与 h_2 几乎处处相等。

证明. 由于 $-\ln x$ 是严格凸函数，对和为 1 的正数 $\lambda_1, \ldots, \lambda_n$ 与正数 $x_1, \ldots, x_n$，由琴生不等式有

$$\sum_i -\lambda_i \ln x_i \geq -\ln\left(\sum_i \lambda_i x_i\right)$$

且等号成立当且仅当 x_i 均相等。

将其连续化即得，对满足 $\int f(x)\mathrm{d}x = 1$ 的正函数 $f(x)$ 与正函数 $g(x)$，有

$$\int -f(x)\ln g(x)\mathrm{d}x \geq -\ln\int f(x)g(x)\mathrm{d}x$$

且等号成立当且仅当 $g(x)$ 几乎处处恒定。

于是有

$$D_{KL}(h_1\|h_2) = \int -\ln\frac{h_2(x)}{h_1(x)}h_1(x)\mathrm{d}x \geq -\ln\int\frac{h_2(x)h_1(x)}{h_1(x)}\mathrm{d}x = -\ln 1 = 0$$

由于 h_1, h_2 均为归一化的正函数，$\frac{h_1}{h_2}$ 几乎处处恒定可得只能恒定为 1，从而得证。 □

将 $h_{w+\delta w}(x)$ 对 δw 展开到二阶，可计算得

$$D_{KL}(h_w\|h_{w+\delta w}) = -\delta w^T E_{h_w}\left[\nabla_w \ln h_w(x)\right] - \frac{1}{2}\delta w^T E_{h_w}\left[\nabla_w^2 \ln h_w(x)\right]\delta w + O(\|\delta w\|^3)$$

定理 18 (期望估算). （省略期望下标 h_w）

$$E\left[\nabla_w \ln h_w(x)\right] = 0, \quad -E\left[\nabla_w^2 \ln h_w(x)\right] = E\left[\nabla_w \ln h_w(x)\nabla_w \ln h_w(x)^T\right]$$

证明. 根据充分正则性，写成期望形式，对 $t = 1, 2$ 可知

$$E\left[\frac{1}{h_w(x)}\nabla_w h_w(x)\right] = 0, \quad E\left[\frac{1}{h_w(x)}\nabla_w^2 h_w(x)\right] = 0$$

由第一个式子内部可直接写成 $\nabla_w \ln h_w(x)$ 可知 $E\left[\nabla_w \ln h_w(x)\right] = 0$，而进一步计算可知

$$E\left[\nabla_w^2 \ln h_w(x)\right] = E\left[\frac{1}{h_w(x)}\nabla_w^2 - \frac{1}{h_w^2(x)}\nabla_w h_w(x)\nabla_w h_w(x)^T\right]$$

由求和中第一项为 0 即得证。 □

从而，记 $G(w)=E\left[\nabla_w \ln h_w(x)\nabla_w \ln h_w(x)^T\right]$，即有

$$D_{KL}(h_w\|h_{w+\delta w})=\frac{1}{2}\delta w^T G(w)\delta w+O(\|\delta w\|^3)$$

下面研究其如何应用于迭代。

我们考虑每次限制变化范围的优化

$$w_{k+1}=\underset{w}{\operatorname{argmin}}\left\{F(w)\mid D_{KL}(h_{w_k}\|h_w)\le\eta_k^2\right\}$$

根据推导，此约束可近似成 $\frac{1}{2}(w-w_k)^T G(w_k)(w-w_k)\le\eta_k^2$。将硬约束化为软约束，可看作优化目标增加项 $\frac{1}{2\alpha_k}(w-w_k)^T G(w_k)(w-w_k)$，$\alpha_k$ 越大则步长越大，再将 $F(w)$ 近似为 $F(w_k)+\nabla F(w_k)^T(w-w_k)$，最终变为

$$w_{k+1}=\underset{w}{\operatorname{argmin}}\left\{\nabla F(w_k)^T(w-w_k)+\frac{1}{2\alpha_k}(w-w_k)^T G(w_k)(w-w_k)\right\}$$

求解可得到：

算法 12 (自然梯度法-迭代)**.** 每次迭代中，给定步长 $\alpha_k>0$，抽取一族下标 S_k，估算 $G(w_k)$ 为 ($\xi_{i,x}$ 代表样本 ξ_i 中的 x)

$$\tilde{G}(w_k)=\frac{1}{\#S_k}\sum_{i\in S_k}\nabla_w \ln h_{w_k}(\xi_{i,x})\nabla_w \ln h_{w_k}(\xi_{i,x})^T$$

并更新

$$w_{k+1}=w_k-\alpha_k G(w_k)^{-1}\nabla F(w_k)$$

与高斯-牛顿法类似可证 $\tilde{G}(w_k)$ 半正定，于是使用中可添加 λI 以保证正定性。

§4.2.4 实际测试

不妨考虑 y 为一维的情况，假设预测函数为二次函数

$$h_{w_1,w_2,w_3}(x)=(w_1^Tx)^2-(w_2^Tx)^2+w_3^Tx$$

容易验证此函数对应损失函数非凸，真实海森阵在大部分情况下非正定。

直接计算可知

$$\nabla_{w_1}h(x)=2(w_1^Tx)x,\quad \nabla_{w_2}h(x)=-2(w_2^Tx)x,\quad \nabla_{w_3}h(x)=x$$

高斯-牛顿法中，将 $\nabla_w h(x)\nabla_w h(x)^T$ 在采样中平均以得到对 G 的估算，而自然梯度法将

$$\nabla_w \ln h(x)\nabla_w \ln h(x)^T=\frac{1}{h^2(x)}\nabla_w h(x)\nabla_w h(x)^T$$

作为对 G 的估算。添加 λI 后计算其逆，结合采样估算梯度

$$\nabla_w f_i(w) = 2(h(x_i) - y_i)\nabla_w h(x_i)$$

即可得到迭代。这里的 λ 事实上包含了某些正则化的功能，因此无需再特别进行正则化。

令

$$a_1 = (1.1, 2.1, 0.3, -0.5, 0, 0, 1)^T$$

$$a_2 = (2, 0.1, 1.3, 0.2, -1, 0, -1)^T$$

$$a_3 = (1, -2, 1, -1.1, 0, 3, 0)^T$$

生成函数为

$$y = (a_1^T x + \epsilon_1)^2 - (a_2^T x + \epsilon_2)^2 + a_3^T x + \epsilon_3$$

其中 $\epsilon_{1,2,3}$ 为独立的随机误差。

这个例子中，由于结果的形式，$\nabla_w f_i(w)$ 的前 14 个分量只要 $w_1^T x = 0, w_2^T x = 0$ 即会为 0，因此这个例子中十分容易落入局部最优点，一阶方法无法对前 14 个分量进行有效更新。

以全 1 进行初始化（全 0 进行初始化的影响将在之后讨论），估算梯度的批次大小为 50，估算海森阵的批次大小为 30，正则化系数 $\lambda = 0.1$，高斯-牛顿法步长取为 0.001，自然梯度法步长取为 0.00001，进行 10000 次迭代后可结果如**图 22**，纵轴为均方误差的对数。可以看出，步长相对高的高斯牛顿法虽然在刚开始迭代时下降较快，但收敛结果上不如自然梯度法，这也暗示着自然梯度法是对海森阵更好的估计。

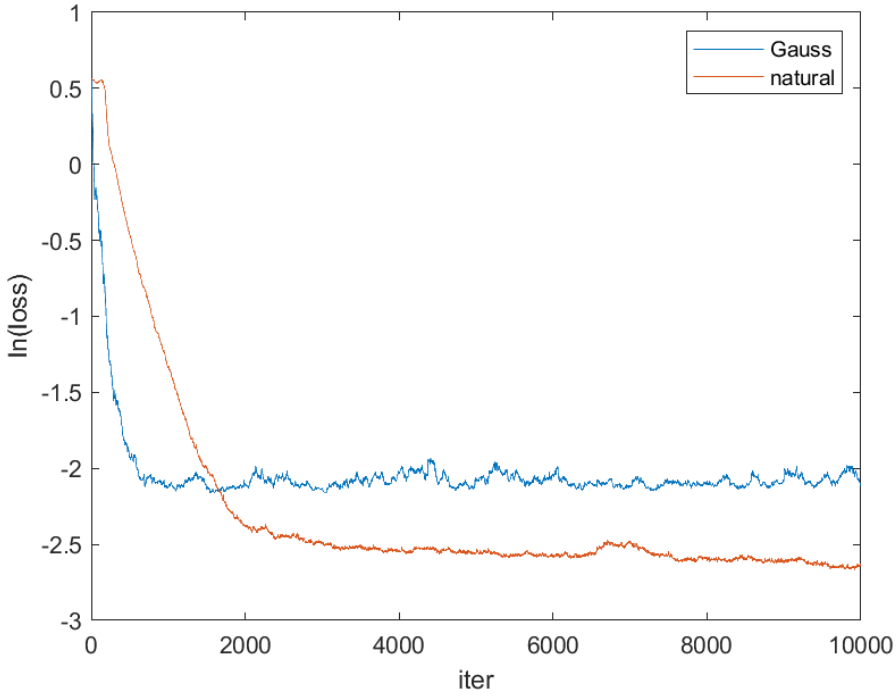

图 22: 基础结果

将正则化系数取为 0.001、0.05、1 的效果如**图 23** 。可以看出，自然梯度法受正则化的影响十分剧烈，若正则化系数太低会因为矩阵接近奇异而趋于发散，太高则会降低收敛速度。相比之下，高斯-牛顿法的矩阵更稳定，因此需要正则化部分的影响偏小。

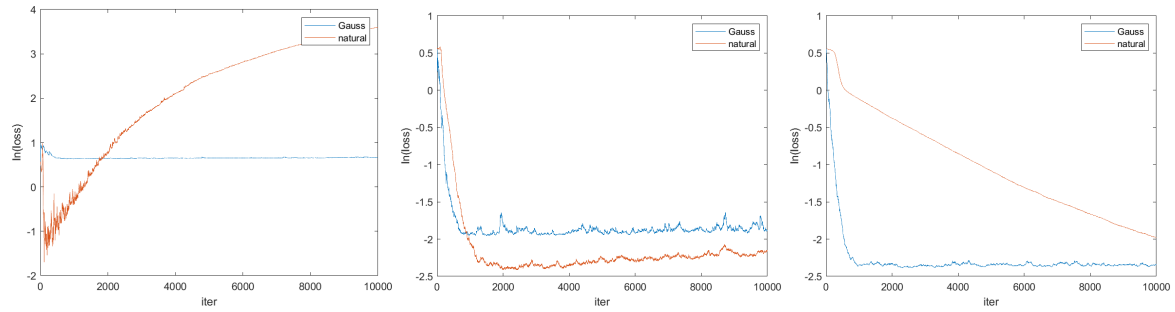

图 23: 正则化影响

尝试将步长进行衰减，可以得到**图 24** 。总体来说，步长衰减对牛顿迭代的效果差别不大，这是由于海森阵的估计与梯度的作用已经提供了较决定性的步长因子。因此，一般无需再额外进行步长的衰减。

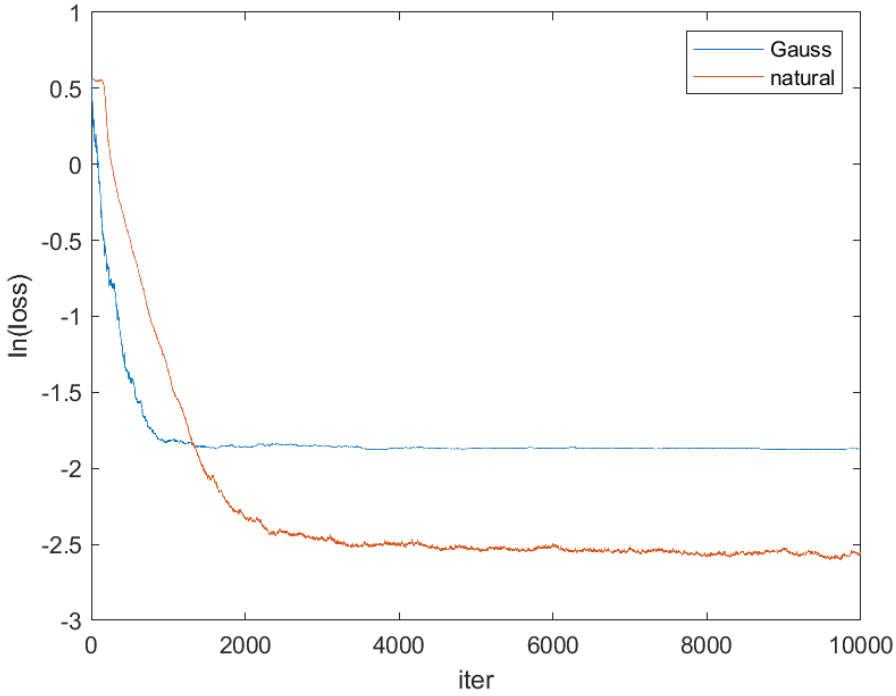

图 24: 步长衰减结果

第 5 章　批次牛顿法

以上两章中，我们讨论了对随机梯度法进行降噪以及引入二阶信息形成随机牛顿法。本章中，我们将二者结合，尝试给出一个通用性更好的方案。此时的理论分析会变得十分困难，

但我们仍可以通过实践测定方法的合理性。

第 1 节　牛顿法的降噪

由于对海森阵的估计亦存在噪声，对随机牛顿法仍然可以采用之前的降噪策略。除了梯度聚合法需要依赖梯度本身，动态采样与迭代平均都可以直接实现。

§5.1.1　动态采样

设初始估算梯度的批次大小为 50，估算海森阵的批次大小为 30，每次乘 1.001 进行扩大采样，上限为 300、180，迭代 3000 次的结果如**图 25** 。

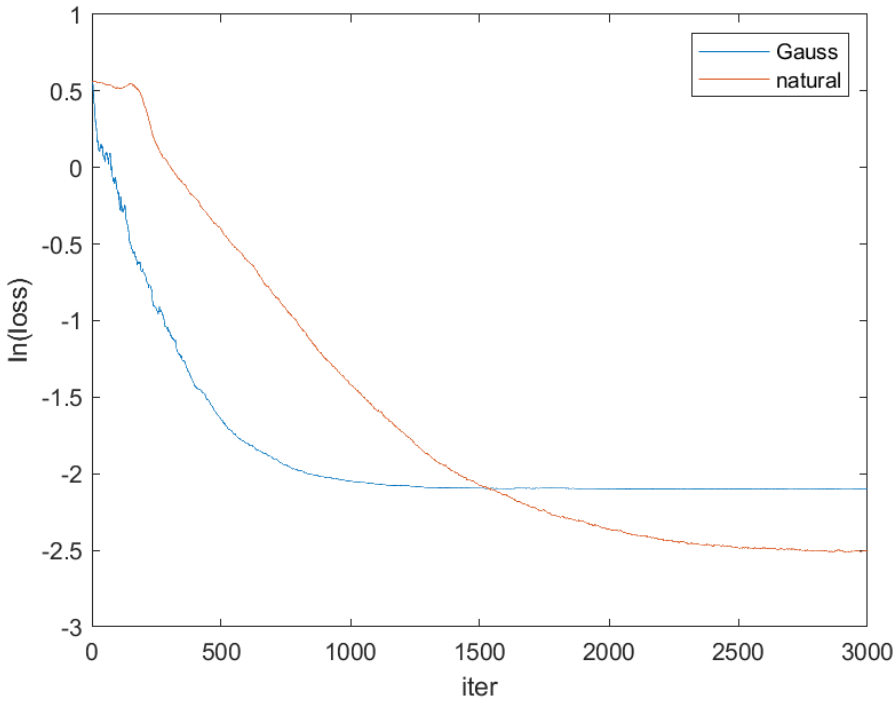

图 25: 二阶动态采样

对比两条曲线可以发现，对输入更加敏感的自然梯度法在动态采样下显著降低了振荡程度，而更加稳定的高斯-牛顿法振荡程度的降低并不明显。尝试使用更小的批次，如估算梯度 40、估算海森阵 30，或估算梯度 30、估算海森阵 20 进行迭代，可以发现结果如**图 26** 。自然梯度法的迭代效果几乎不受影响，但高斯-牛顿法在样本较少时很容易收敛到了并不是真正最小点的驻点。

不过，由于需要用若干个秩为 1 矩阵的和估算真正的海森阵，考虑到可逆性的要求，估算海森阵的样本个数不能更小，这也导致动态采样法对收敛效果的改进有限——由于初始批次大小必须充分大，动态采样自始至终都保持在同一个量级上，逐渐增大的批次大小导致的计算开销增大更值得注意。

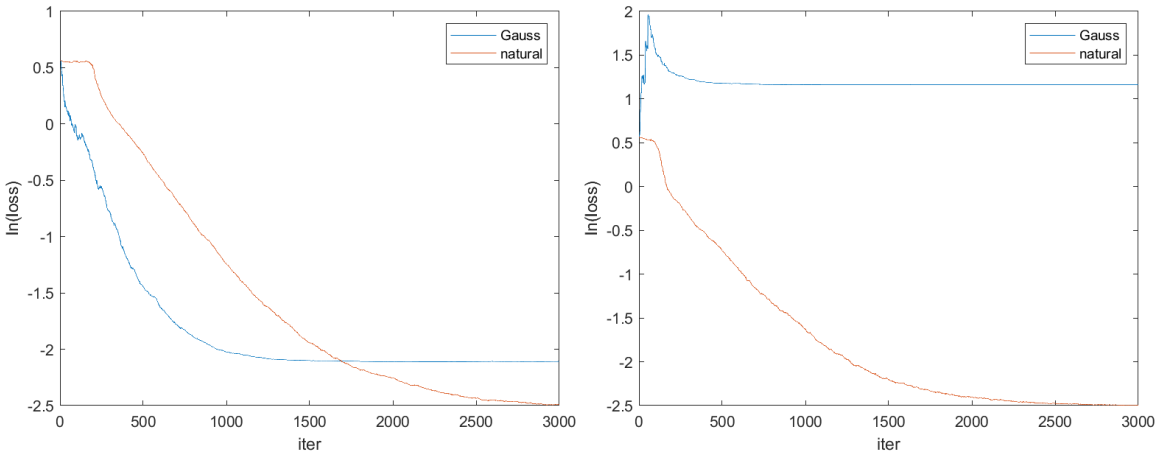

图 26: 降低批次大小

§5.1.2 迭代平均

与一阶方法时完全类似进行迭代平均的效果如**图 27** 。虽然收敛速度有所降低，但无论是对高斯-牛顿法还是自然梯度法，其都非常显著地增加了结果的光滑程度。

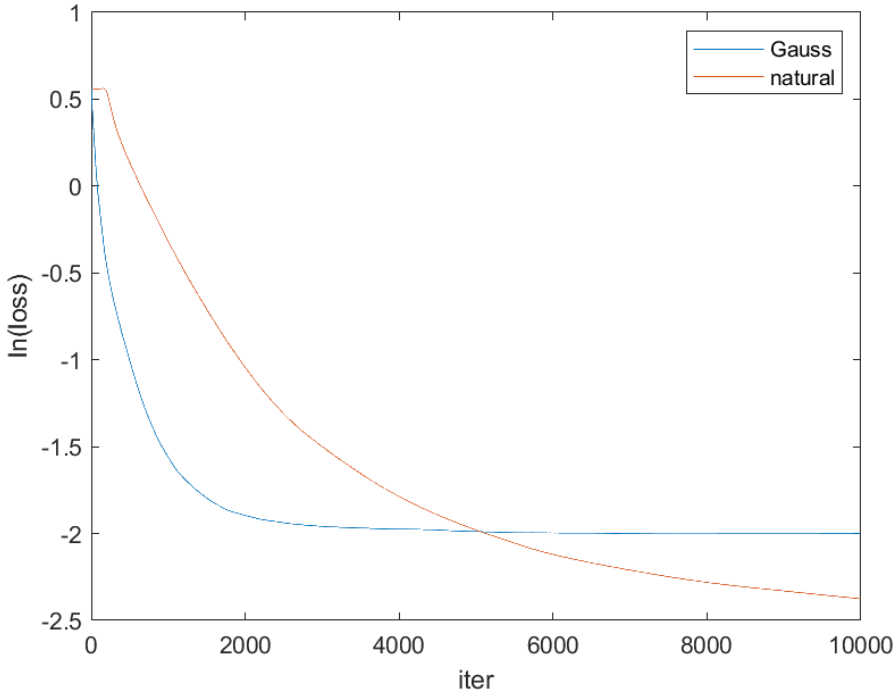

图 27: 二阶迭代平均

将步长与迭代次数都翻倍，可以得到较充分收敛的结果，如**图 28** 。

高斯-牛顿法很快达到了最低点，随即开始过拟合，而进一步迭代测试可发现自然梯度法在误差对数接近 -8 时方才开始过拟合，展现出了很好的逼近性质，整体对比如**图 29** 。

综合上述讨论，自然梯度法结合迭代平均是计算开销和收敛效果都相对较好的方案。此

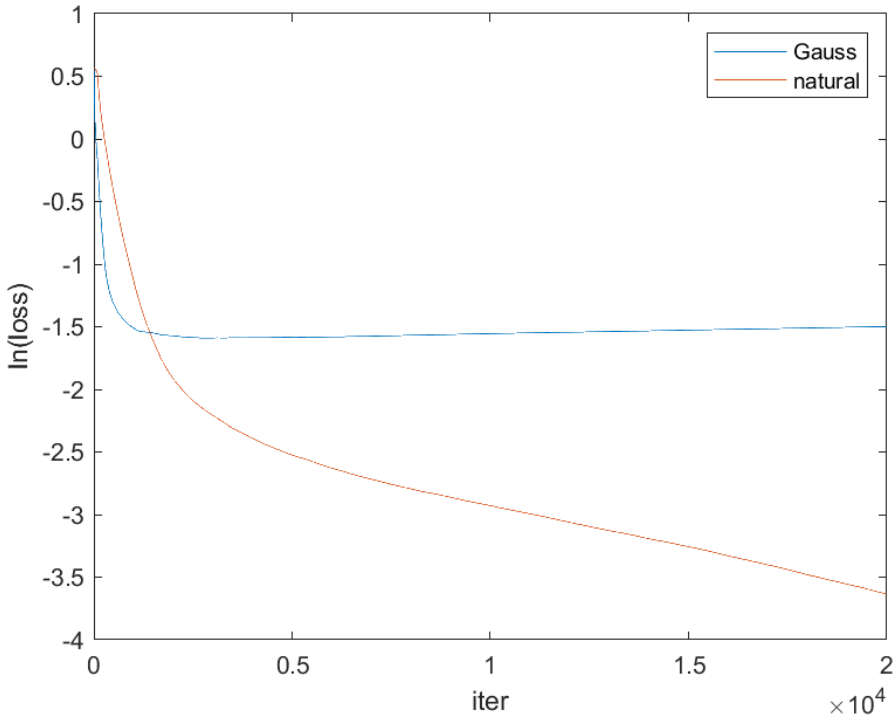

图 28: 充分迭代平均的随机牛顿法

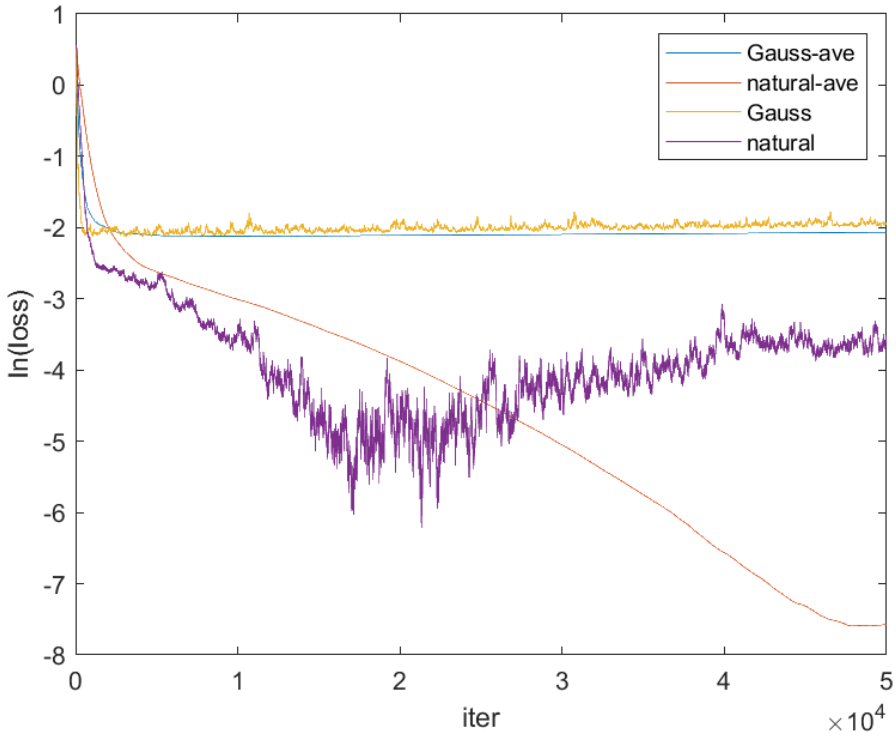

图 29: 有无迭代平均的两种二阶方法对比

后无特殊说明时，所有方法都采用了迭代平均策略。

第 2 节　迭代稳定性

除了噪声的处理，二阶方法另一个需要考虑的问题是，在存在局部极小值，或未必有二阶信息时，迭代能否稳定进行。

§5.2.1　非凸迭代

首先，对于之前的例子，我们还需要考虑不同位置出发的迭代。若初始为全 0 的位置，$h_i(w) = 0$，自然梯度法根本无法进行迭代。因此，必须在 $h_i(w)$ 充分小时利用高斯-牛顿法估算梯度，由此得到的全 0 出发的结果如**图 30** 。

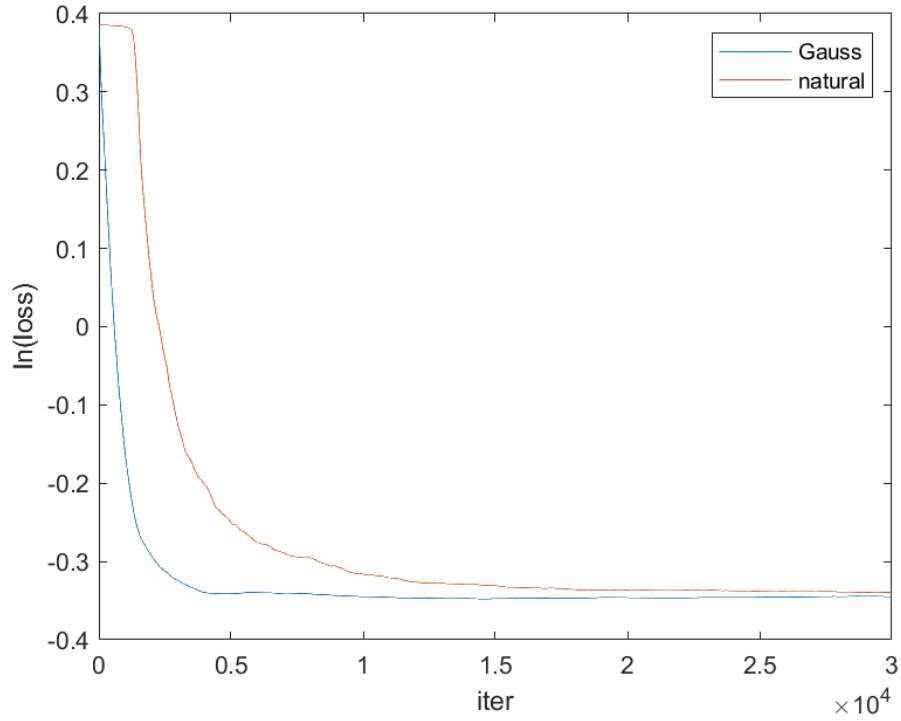

图 30: 奇异点出发的二阶迭代

事实上，由于前 14 个分量的所有梯度均为 0，迭代过程中它们亦保持恒 0，不会发生变化，从而导致无法正常收敛，于是损失居高不下。机器学习中，解决此问题的常用方案是随机初始化策略。由于落在驻点附近的概率相对很小，只要采取随机初始化，即可避免出现无法迭代的情况。不过，随机初始化的效果并不稳定，也可能落入其他的驻点，如**图 31** 。

另一个常用的策略是迭代过程中添加随机的扰动——某种意义上，随机初始化即是在迭代开头添加了随机扰动。例如，从全 0 出发，每 1000 次迭代添加一个方差恒定的正态分布。

几次测试的效果如**图 32** ，可以发现，这时虽然避免了完全落入前 14 个分量为 0 的点，但是否能改善收敛仍然是随机的，因此，若不能找到好的起始点，二阶方法在非凸情况的稳定性可能较差。

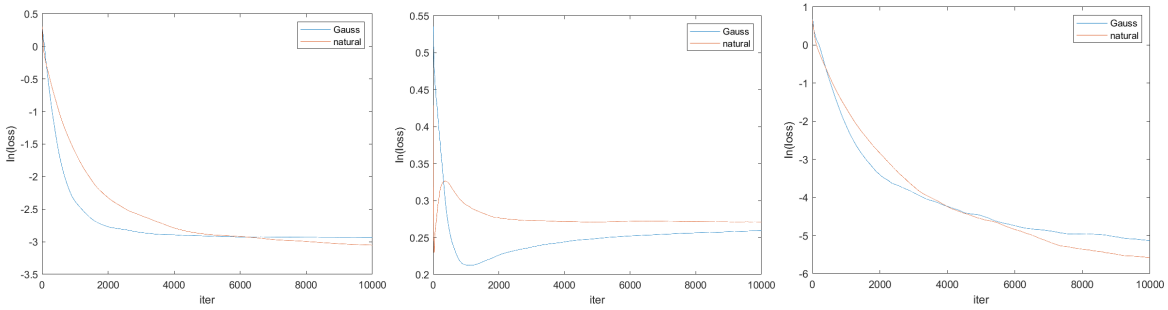

图 31: 几次随机初始化的结果

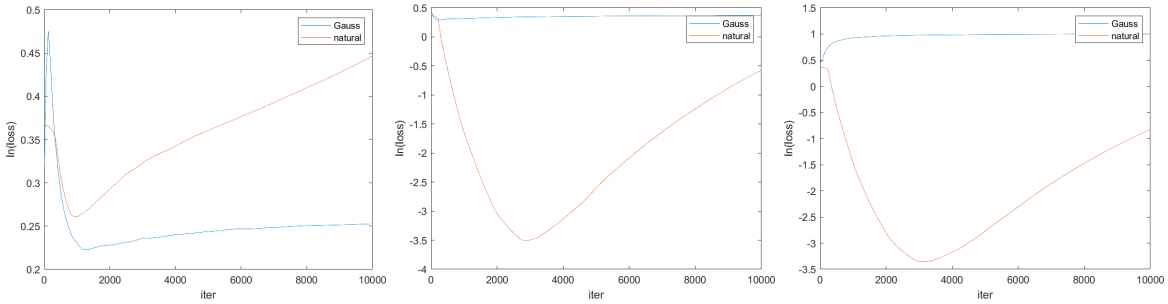

图 32: 随机扰动后的二阶方法

§5.2.2 非二阶情况

此外，原函数不存在有意义二阶导的情况也可能影响二阶方法的效果。考虑之前例子中令

$$h_{w_1,w_2,w_3}(x) = (w_1^T x)^2 - |w_2^T x| + w_3^T x$$

y 同理更改，由此仍然可以利用高斯-牛顿法与自然梯度法计算，从全 1 出发，迭代结果为**图 33**，可以看到，自然梯度法落入了驻点，而高斯-牛顿法较好地进行了迭代。

随机初始化，多次测试后可以得到**图 34**。总的来说，二阶方法的不稳定性较高，尤其容易受初始点与函数形式的影响。不过，无论是高斯-牛顿法还是自然梯度法，由于实质上都只使用了一阶信息，对函数二阶光滑性的要求并不算高，即使对不具有良好光滑性的函数，也能在合适的初值下得到相对好的收敛结果。

然而，即使如此，二阶方法的迭代稳定性问题仍然难以解决。理论计算可知，牛顿法在离全局最优点充分近时是具有二阶收敛速率的，但它并不能保证在远处能快速接近最优点。反之，梯度方法可以快速接近最优点，但附近的收敛速度较慢。为了克服二阶方法的迭代稳定性问题，我们需要从此思路出发将一阶、二阶方法进行结合。

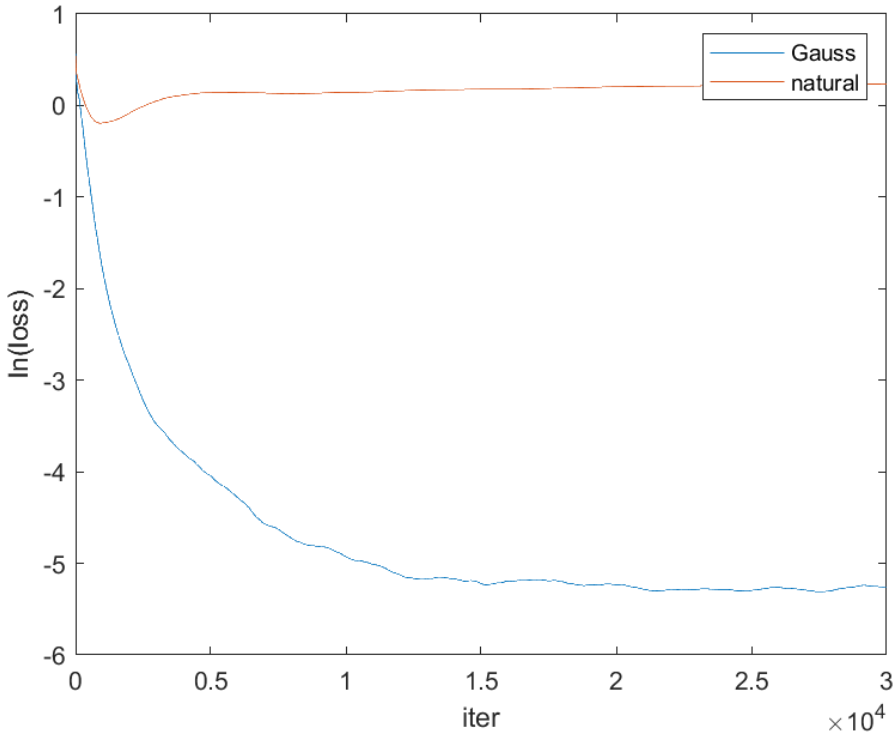

图 33: 含绝对值的二阶迭代

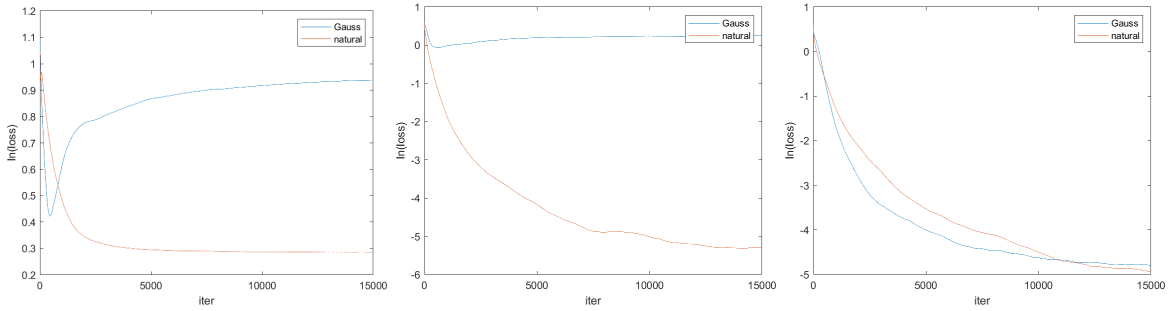

图 34: 随机初始化的含梯度二阶迭代

第 3 节　复合方法

§5.3.1　近端牛顿法

回到二阶光滑、全 1 初始化的例子，进行充分迭代后，对比高斯-牛顿法、自然梯度法与普通的随机梯度法，可得到**图 35** 。

此结果符合之前的理论分析，也即远处一阶方法收敛性更好，而近处二阶方法收敛性更好。采用更新幅度作为表征，若直接梯度法的更新幅度相对较小则选用对应的步长更小、也更精细的自然梯度法，即可以得到综合的结果。考虑一个方差较大的随机初始化（从而导致初始可能距离很远）的例子，充分迭代后的实际效果如**图 36** ，可以看出，两种方法结合后明显改善了收敛的速率。（这里自然梯度法收敛较慢是出于精细调整的目的降低了步长。）

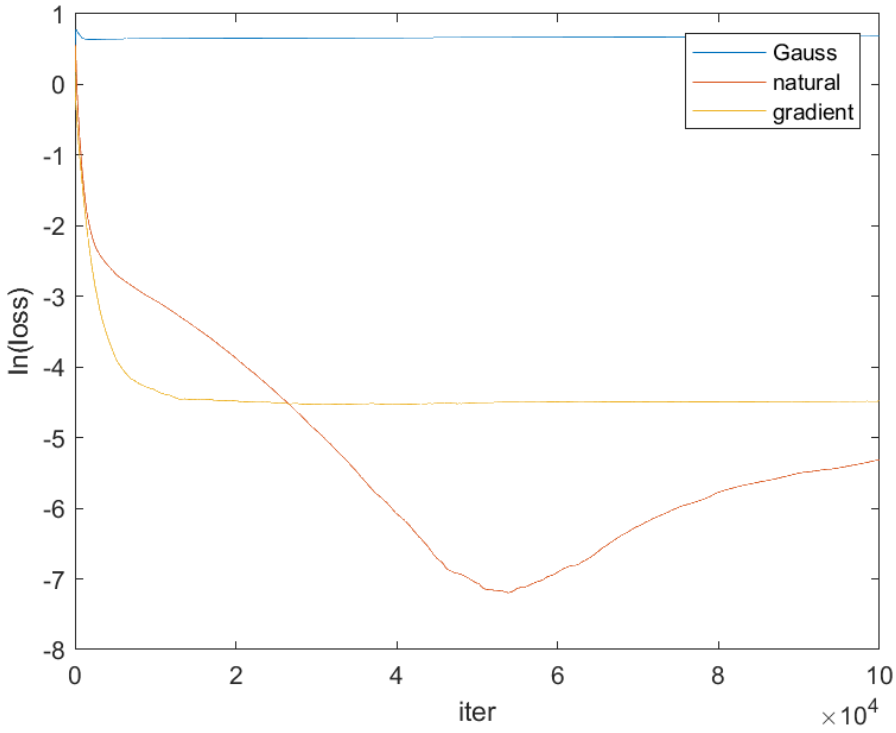

图 35: 一阶、二阶方法对比

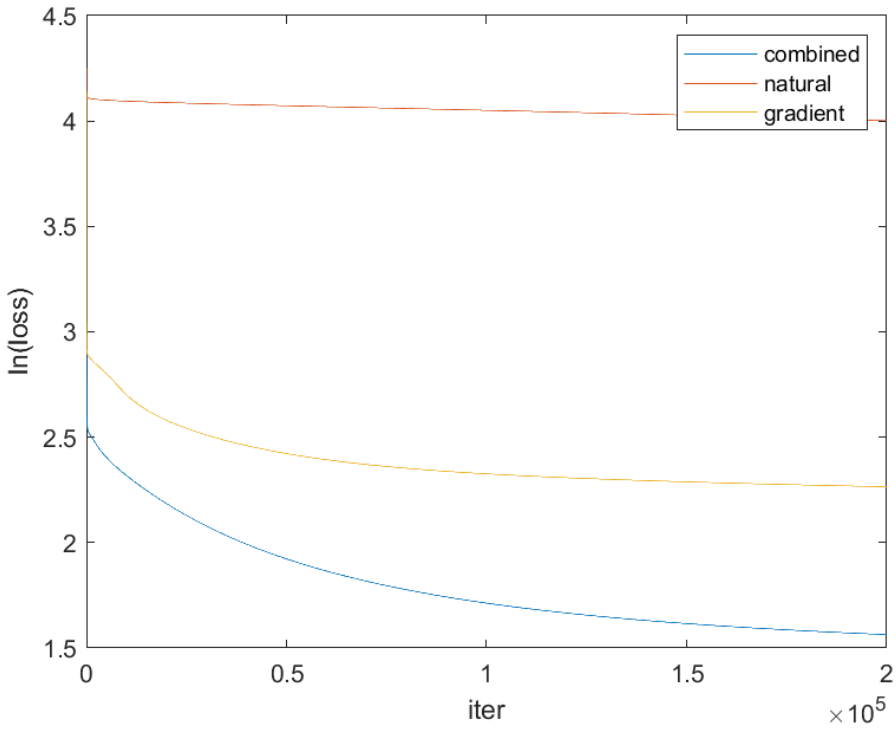

图 36: 综合方法

不过，初始距离很远的情况下，迭代过程中也更容易被更远处的局部最优点捕获，导致难以充分逼近，这时仍然需要结合前一部分的随机扰动策略以保证迭代的持续进行。

§5.3.2 自适应二阶方法

最后，结合上述讨论，我们给出一个具有自适应特性的二阶方法，以保证收敛速率与收敛的准确性。

算法 13 (自适应二阶方法). 算法流程为：

1. 给定初始 w_1，某初始较低的步长 α_b，记 $\alpha_0 = \alpha_0^n = \alpha_b$，估算梯度与估算二阶项的批次大小 m、n，令 $k = 1$。

2. 从样本中抽取 m 个构成 S_k，并计算

$$g_k = \frac{1}{\#S_k} \sum_{i \in S_K} \nabla f_i(w_k)$$

3. 从 S_k 中抽取 n 个构成 T_k，并计算

$$G_k = \frac{1}{\#T_k} \sum_{i \in T_k} \nabla_w \ln h_{w_k}(\xi_{i,x}) \nabla_w \ln h_{w_k}(\xi_{i,x})^T$$

   这里右侧的求和中，若某项的 $h_{w_k}(\xi_{i,x})$ 过小，可替换为 $\nabla_w h_{w_k}(\xi_{i,x}) \nabla_w h_{w_k}(\xi_{i,x})^T$ 以避免奇异。

4. 计算牛顿方向 $g_k^n = G_k^{-1} g_k$。

5. 记 $w_{k+1}(\alpha) = w_k - \alpha g_k$，从 $\alpha = \alpha_{k-1}$ 出发进行迭代。考虑 S_k 上误差

$$\sum_{i \in S_k} (h_{w_{k+1}(\alpha)}(\xi_{i,x}) - \xi_{i,y})^2$$

   若其比原来有下降则将 α_{k-1} 翻倍，持续翻倍直到误差不再下降，取出使误差最小的步长。否则，持续减半，直到误差比原来下降。将最终步长记为 α_k。

6. 对 g_k^n 进行类似操作，从 α_{k-1}^n 找到最大下降步长 α_k^n。

7. 对比 $w_k - \alpha_k g_k$ 与 $w_k^n - \alpha_k^n g_k^n$，取 S_k 上误差更小为 w_{k+1}。

8. 实际使用的 $\tilde{w}_{k+1} = \frac{1}{k+1} \sum_{i=1}^{k+1} w_i$，$k = k + 1$，回到第二步。

通过自适应的步长更新，自然梯度法与普通随机梯度的更新结合，理论来说可以得到很好的收敛效果。实际测试也符合这一结论，以全 1 进行初始化，迭代结果如**图 37**，而高误差初始化的结果如**图 38**。

可以发现，其收敛性质要远好于一般固定的步长的梯度或自然梯度迭代，在 3000 次迭代就达成了较好的收敛——而不进行自适应步长时，若步长取得太高，一般的随机方法则会非常容易发散。三种算法执行 100000 次迭代的时间分别为 251 秒、175 秒、148 秒，而自适

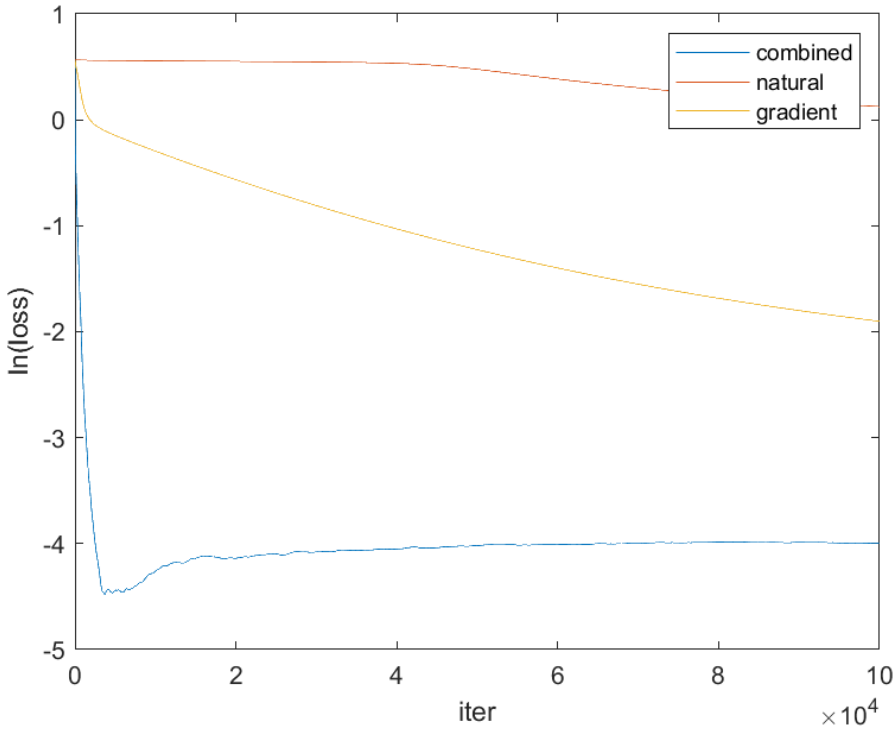

图 37: 全 1 初始化结果

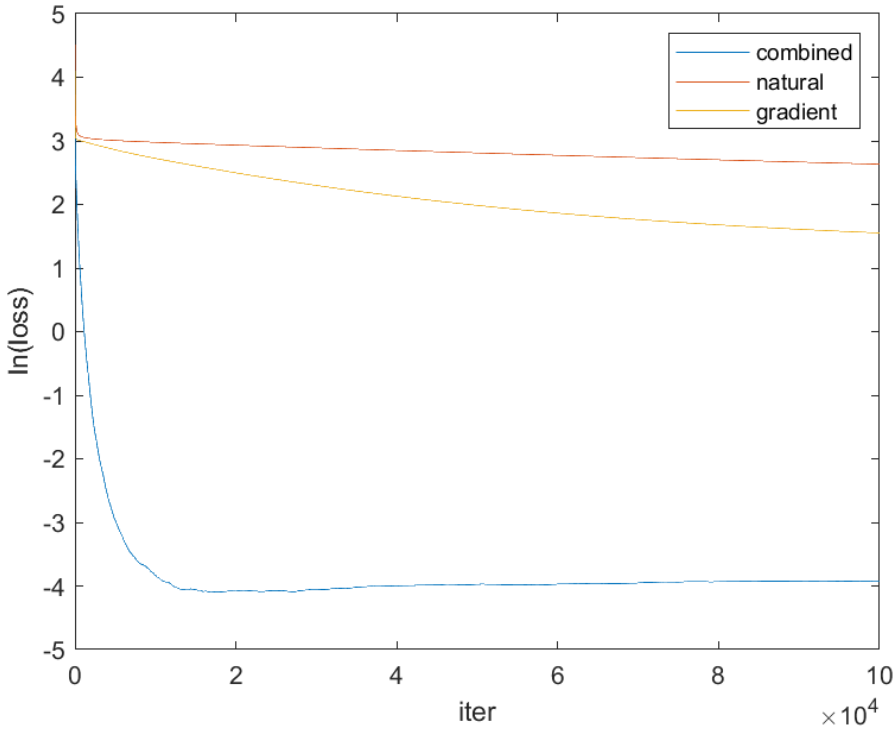

图 38: 高误差初始化结果

应算法在 100000 次迭代中有 22920 次选取了自然梯度的方向，也即直接梯度法事实上是占据主要迭代的。

此方法可以很大程度克服落入局部最优的情况，这并不是通过显式增加干扰实现的，而是通过相对激进的步长迭代策略——直接以 2 的幂进行迭代，有很大概率跨过局部最优点。是误差较大的初始化出发的迭代结果，效果更加明显。

此外，从结果来说，自适应方法单次迭代的比较过程并不会消耗过多时间，其他测试也证明了时间消耗至多为简单自然梯度方法的两倍。事实上，由于取出的批次并不太大，指数量级改变的步长也不可能持续过多次，计算海森阵与其逆才是消耗时间相对最多的部分。

第 6 章　总结与讨论

第 1 节　总结

由于机器学习中一般面对着大量的样本与复杂的模型，随机梯度法成为了常用的优化方案。我们对其进行了相关的理论分析，证明了各种情况的收敛性，并得到了在各种实际情况下的表现。总体来说，随机梯度法可以以较低的迭代代价得到相对良好的结果，但由于随机性，收敛性和收敛速度都无法保证。

接下来，我们尝试了使用动态采样、梯度聚合与迭代平均的方法进行降噪，分析了各个降噪方案的结果。相比之下，迭代平均是成本低且效果良好的降噪方案。

下一章中，我们尝试引入二阶信息以加速收敛。考虑到实际函数未必具有良好的凸性，最终采用的方案是以一阶信息结合正则化给出一个正定的对二阶信息的估计，也即高斯-牛顿法与自然梯度法。实验表明，自然梯度法的总体效果相对较好。

最后，我们结合降噪方案与二阶方法，给出了步长自适应的一种二阶方法。其在保持迭代较稳定的基础上，可以有效达成高收敛速率，且计算代价可控，不容易被局部最优捕获。从此方法出发，也可以推广得到一类综合一阶、二阶信息的自适应迭代模式。

第 2 节　讨论

§6.2.1　存在问题

虽然此方法已经能达到相对较好的结果，仍然有一些问题需要解决：

1. 理论收敛性问题。虽然直接梯度与自然梯度都可以得到期望收敛的结果，但结合自适应步长后，无法简单得到其理论上的收敛性，需要更进一步的分析在各个函数上的情况。

2. 计算代价问题。对一阶与二阶迭代方向逐次对比步长并计算损失是一个代价较高的方案，尤其是模型增大时。如果能有更好的方法确定选择的方向或步长，可以减少损失计算上的复杂度。

3. 收敛速率问题。根据损失对数的下降情况，会在某个位置达到收敛极限，而这是由抽样方差与步长引起的，如果能更好进行调整，或结合动态采样，可以达成更精细的收敛。

§6.2.2 优化方向

基于这些问题与方法的特性，可以给出一些可能的优化方向：

1. 进一步的理论分析。由于自适应步长中实际保证了在批次中损失的下降性，结合期望仍然可以得到此迭代的收敛性，并且一定能够在步长充分小时收敛至极小点。不过，这样的简单分析并不能实际体现自适应后的收敛速率，需要更精细的工具。

2. 计算代价的优化。如上一部分中所述，我们可以试着寻找选择的方向或步长的更好方法，以降低大量比较导致的计算代价。

3. 更多一阶二阶方法的探索。我们此处采用了简单随机梯度作为一阶方法，自然梯度法（以及奇异时的高斯-牛顿法）作为二阶方法，并加以结合。可以考虑采用更丰富的一、二阶方法以达成更好的收敛性，例如以 LBFGS 为基础构建二阶方法，维护拟牛顿阵。

4. 降噪过程的优化。从第一步开始进行迭代平均未必是好的选择，尤其是开始时误差较大，可以考虑在收敛到一定程度后再开始迭代平均。同样，这时开始进行动态采样也可能增加效率。

5. 步长控制的优化。直接以翻倍和减半作为步长控制虽然确实有跨过局部最优的功能，但是面对不少情况可能导致不精确，可以试着结合上下界或其他倍率等要素一起进行控制。

6. 引入估算自适应。如同梯度聚合的思路，事实上可以考虑聚合部分之前迭代的采样结果以得到对真实的梯度、海森阵更好的方向，这可能可以在计算代价相对低的情况下得到更好的收敛性。

§6.2.3 应用展望

虽然此处的自适应二阶方法只是一个简单的例子，但其代表的一类结合一阶、二阶迭代的自适应方法可以在随时间的收敛性与收敛速率方面展现出优势。进一步引入优化后，其的确可以用于解决部分机器学习中的优化问题。

不过，当参数数量极多时，估算海森阵与其逆需要的存储与计算成本几乎是不可接受的，因此仍然需要寻找更简单的计算策略，例如将 LBFGS 作为二阶方法达成时间换空间的效果。

最后，类似有限差分法中通过数值黏度进行一阶与二阶的加权，此方法也可以通过加权而非选择获得更好的迭代方向，并最终应用在模型的学习中。

参考文献

[1] GATYS L A, ECKER A S, BETHGE M. Image style transfer using convolutional neural networks[C/OL]//2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2016: 2414-2423. DOI: 10.1109/CVPR.2016.265.

[2] ZHANG R, ZHU J Y, ISOLA P, et al. Real-time user-guided image colorization with learned deep priors[J]. *ACM Transactions on Graphics (TOG)*, 2017, 9(4).

[3] 刘浩洋, 户将, 李勇锋, 等. 最优化：建模、算法与理论 [M]. 北京: 高等教育出版社, 2020.

[4] PAPA G, BIANCHI P, CLÉMENÇON S. Adaptive sampling for incremental optimization using stochastic gradient descent[C]//CHAUDHURI K, GENTILE C, ZILLES S. Algorithmic Learning Theory. Cham: Springer International Publishing, 2015: 317-331.

[5] JOHNSON R, ZHANG T. Accelerating stochastic gradient descent using predictive variance reduction[C/OL]//BURGES C, BOTTOU L, WELLING M, et al. Advances in Neural Information Processing Systems: volume 26. Curran Associates, Inc., 2013. https://proceedings.neurips.cc/paper_files/paper/2013/file/ac1dd209cbcc5e5d1c6e28598e8cbbe8-Paper.pdf.

[6] DEFAZIO A, BACH F, LACOSTE-JULIEN S. Saga: a fast incremental gradient method with support for non-strongly convex composite objectives[C]//NIPS'14: Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 1. Cambridge, MA, USA: MIT Press, 2014: 1646–1654.

[7] LIU D C, NOCEDAL J. On the limited memory bfgs method for large scale optimization[J]. *Mathematical Programming*, 198, 45: 503-528.

[8] AMARI S I. *Natural Gradient Works Efficiently in Learning*[J/OL]. Neural Computation, 1998, 10(2): 251-276. DOI: 10.1162/089976698300017746.

致谢

在中科大辅修计科的时间虽然不长，但让我学会了很多，也认识了很多可爱的老师、同学们。在此我要对指导过我的老师，帮助过我的同学与照顾过我的朋友们说一声谢谢。

首先感谢我的导师卢建良老师对我的悉心教诲，激发了我对科学研究的兴趣和对科研工作的责任感，也在我的辅修经历中给了我诸多的帮助。导师对学生的关心与对科研的严谨都让我受益良多。

此外，还要感谢杨周旺老师、连德富老师的课程与课件，让我对机器学习中的最优化产生了兴趣，也尝试着进行自己的分析证明与实践检查。

我还要感谢数学学院、计算机科学学院的朋友们，大家在交流问题时给予了我很多帮助，也让我更好地进行了课题的学习、研究。

最后，感谢我的父母与音乐剧社的朋友们，是大家一直以来的支持让这篇论文得以完成。