

程序作业 1

提交日期: 6 月 20 日

问题 1 考虑稀疏二分类的逻辑回归模型

$$\min_x \ell(x) \stackrel{\text{def}}{=} \frac{1}{m} \sum_{i=1}^m \ln(1 + \exp(-b_i a_i^T x)) + \lambda \|x\|_2^2 + \mu \|x\|_1,$$

其中 $\lambda > 0, \mu > 0$ 是给定的参数。 $a_i \in \mathbb{R}^n, b_i \in \{+1, -1\}, m = 32561$ 是给定的 *a9a* 数据集。

按以下要求 1 或 2 中, 选择一条实现。求导可借助 *pytorch* 自动微分实现。

- 令 $\lambda = \frac{1}{2m}, \mu = 10^{-2}$, 对原问题使用近似点梯度法 (lecture 10), 以及 *FISTA* (lecture 12) 算法求解, 对比两个算法的收敛速度, 具体要求:
 - 使用 *backtracking linesearch* 确定步长 t_k , 选择收敛最快的步长进行对比。
 - 作图要求: (1) 画出最优条件 $\|x_k - x_{k+1}\|/t_k$ 与迭代步数的关系, 纵坐标使用 *log scale*, 当 $\|x_k - x_{k+1}\|/t_k \leq 10^{-6}$ 或者迭代步数超过 10000 步时停止算法。(2) 画出函数误差 $\ell(x) - \ell(x^*)$ 与迭代点的关系, x^* 为最优点, 可以用预先求解得到一个较高精度的点代替。
- 令 $\lambda = \frac{1}{2m}, \mu = 10^{-2}$, 采用 *ADMM* 求解该问题, 使用合适的方法求解 *ADMM* 的子问题, 例如牛顿法或者加速梯度法。
 - 要求: 子问题求解使用 *backtracking linesearch* 确定步长 t_k , 选择收敛最快的步长进行对比。
 - 作图要求: (1) 画出 *ADMM* 的两个最优条件与迭代步数的关系, 纵坐标使用 *log scale*, 当二者都小于 10^{-6} 或者迭代步数超过 10000 时停止算法。(2) 画出函数误差 $\ell(x) - \ell(x^*)$ 与迭代点的关系, x^* 为最优点, 可以用预先求解得到一个较高精度的点代替。

通用要求:

- 固定 $\lambda = \frac{1}{2m}$, 调整 μ 的大小, $\{10^{-3}, 10^{-2}, 0.01, 0.05\}$, 列出表格, 形式如下: 稀疏度指的是解中

μ	稀疏度	迭代步数
...

表 1.1: 不同 λ 的求解结果

零元个数占总维度的比例。

- 代码规范要求
 - 注释: 通过注释解释每块代码的意义。分别从做什么、为什么、怎么做来进行注释。结构体和函数一定要写注释, 而且要写得尽可能全面、详细, 而函数内部的注释要相对少一些, 一般都是靠好的命名、提炼函数、解释性变量、总结性注释来提高代码可读性。
 - 可读性: 使用规范的文件名、函数名、变量名增加代码的可读性
 - 格式要求: 注意缩进、对齐等; 一行只做一件事。
- 报告要求:
 - 需要简要描述算法, 所有的参数选取细节
 - 基本知识 (模型与算法的细节), 结果丰富性创新性, 排版要求美观.
- 提交要求: 完整报告、ReadME 和程序等等打包 (文件名为 “name-StudentID-date.zip”) 上传到 bb 上。
 - ReadMe 中指明你程序中所有用到的 python/模块/工具箱的版本, 程序中若使用了非 anaconda/matlab 自带模块/工具箱, 请在 code/README.txt 文件中说明.
 - 其他你想要说明的.