

# 数值优化：程序作业 01

讲师：黄文，厦门大学

截止日期 3 月 23 日 14:20:00（上课前 10 分钟）截止

2022 年 3 月 6 日

## 1 问题描述

在附件提供的代码 `ExampleSDforQuad.m` 中，用一个 Matlab 函数最小化二次凸问题，既最小化

$$f : \mathbb{R}^n \rightarrow \mathbb{R} : x \mapsto \frac{1}{2}x^T A x - b^T x \quad (1.1)$$

这里  $A$  是对称正定矩阵。运行该函数即刻得到一个运行例子。

1. 仔细阅读并理解该代码。该文件中 `SDforQuad` 函数是将最速下降法与待求解问题整合起来。虽然实现了效果，但是并没有起到模块化的作用。也就是如果将优化问题改变，而算法不变，我们需要将算法重新写一遍。不利于纠错、维护与更新。因此可以将此文件实现的功能模块化。分成 3 部分。一部分 `problems` 专门实现与问题相关的函数。一部分 `solvers` 专门负责优化算法。一部分 `tests` 专门使用数据生成模拟测试问题进行测试。模板在附件中已提供。这里要求将 `ExampleSDforQuad` 文件中的功能使用提供的模板中的模块化方式实现。
2. 迭代算法最高可以达到的精度是本底噪声 (noise floor)。Matlab 默认数值表示精度为双精度。研究本次作业的最速下降算法在双精度下可以达到的最高精度。  
可以采用的测试方法：随机生成一个对称正定矩阵  $A$  与向量  $b$ ，使用最速下降法迭代直至算法无法进行或函数值无法降低为止，观察算法达到的精度。这里精度是  $\|x_k - x^*\|$ ，这里  $x^*$  是精确解。我们假设 Matlab 自带函数 `x=linsolve(A, b)` 得到的  $x$  为精确解。做多组测试，汇报你所观察到的结果并给出你的结论。
3. 研究最速下降法收敛速度与矩阵  $A$  条件数的关系。  
可以采用的测试方法：矩阵  $A$  的条件数为其最大特征值与最小特征值的比值。随机生成条件数不同的矩阵  $A$ 。使用最速下降法最小化(1.1)直至算法无法进行或函数值无法降低为止。画半对数图观察  $\|x_k - x^*\|$  与  $k$  直接的关系。判断算法的收敛率。解释不同形状图像所表示的收敛速度的快慢。给出你的结论。

考虑优化问题

$$f : \mathbb{R}^2 \rightarrow \mathbb{R} : (x_1, x_2)^T \mapsto 2x_1^2 + 3x_2^4. \quad (1.2)$$

- 使用模块化编程方法使用最速下降法最小化 (1.2)。
- 画半对数图观察  $\|x_k - x^*\|$  与  $k$  之间的关系以及  $f(x_k) - f(x_*)$  与  $k$  之间的关系。判断算法的收敛率。给出你的结论。

## 2 要求

1. 各参数设置与 ExampleSDforQuad.m 中使用的参数一致。
2. 代码需要合理的缩进。在 function, if, while, switch 等包含的代码需要缩进。
3. 避免重复计算。观察问题(1.1), 在计算该函数和梯度时, 我们都需要计算  $Ax$ , 并且这里的  $x$  是相同的。而且计算这个矩阵向量乘法是整个算法的主要开销。为了避免重复计算, 我们需要在计算梯度时, 能够获取之前计算函数值时取得的信息。为了实现这个目标, 我们使用 2 个技巧
  - $x$  用结构体来表示。将  $x$  本身数据存储在  $x.main$  下。而算法中之后用到的与当前  $x$  相关的数据可以使用其他 field 名存储。如  $x.Ax$  可以用来存储  $Ax$  结果。
  - 将改变的  $x$  返回回算法中, 传给之后需要的代码。如计算函数时, 使用代码  $[fv, x] = f(x)$ 。其中  $fv$  是得到的函数值, 而返回的  $x$  中存储了计算  $f(x)$  时产生的有用信息, 比如  $x.Ax$ 。由于最速下降法中确保梯度的计算一定在函数计算之后, 因此在调用  $\text{Grad}(x)$  时, 梯度的计算可以不需要再计算一次矩阵向量相乘。可以之前从  $x.Ax$  中获取函数计算时计算过的信息。
4. 每隔一定步数后 (隔几步可以自己看情况定), 需要以如下格式输出信息:

Listing 1: 输出信息样本

```
iter:10,f:4.283e-01,|gf|:1.510e+00,s0:-2.1e+00,  
snew:-5.6e-16,t0:1.0e+00,tnew:6.6e-02,lf:20,lgf:10
```

这里换行是为了方便在文档中显示。代码输出不需要换行。 $iter$ 、 $f$ 、 $|gf|$ 、 $s0$ 、 $snew$ 、 $t0$ 、 $tnew$ 、 $lf$  与  $lgf$  分别表示迭代步数、函数值、梯度的模长、线性搜索初始斜率  $p^T \nabla f(x_k)$ 、线性搜索最终斜率  $p^T \nabla f(x_k + tp)$ 、初始步长、最终被接受的步长、目前为止函数计算总次数、目前为止梯度计算总次数。

5. 代码最前端需要通过注释的方式注明学号, 名字, 并将需要回答的问题通过注释的给出。鉴于不同系统对中文支持度不友好, 要求使用英文回答问题。见 Listing 2.

Listing 2: Matlab codes

```
% Numerical Optimization  
% Name: XXXXX, Student number: xxx201xxxxxxxxx  
%  
% This function is the entry of the program  
%  
% Answer the questions in the assignments one by one  
% e.g.,  
% 1, As shown in Figure 1, I found ...  
% As indicated in the output, I concluded that ...  
% 2.1 It can be seen from Figure 2 that ....  
% 2.2 We can observe .... That is because .... My conjecture is ....  
% ...  
  
... Matlab codes ...
```

6. 代码模版已经给出。打开 matlab 进入 Num\_Opt\_Matlab\_template 根目录，运行 “Import-DIR.m” 将子文件夹加入运行目录。tests 文件夹下的文件为代码入口。problems 文件夹下的文件给出优化问题相关信息，如 ProbEucQuadratic.m 需要定义二次凸优化问题 (1.1) 。solvers 文件夹需要下给出优化算法。输入为优化问题信息以及算法参数。输出为算法得到的点以及迭代过程中的你希望返回的信息，如计算时间，每步的迭代点，函数计算次数，梯度计算次数等等。
7. 如需要新建文件，则需要在文件顶端说明该文件用途，解释输入参数以及输出参数。可见代码模版中的例子。本次作业中，你们需要在 problems 文件夹下新建一个针对问题 (1.2) 的问题文件。
8. 编写完代码需要将其发布成 pdf 文件。例子见提供的 Publishcodes.zip
9. 提交作业需要将源代码于发布的 pdf 文件一同提交，源代码根目录使用 “Num\_Opt\_Matlab\_你的学号”。发布出来的 pdf 使用 “Program01\_ 你的学号”。作业统一通过邮件发给助教陈建恒1556921918@qq.com。邮件名称为“数值优化研-你的姓名”。
10. 代码最前端需要通过注释的方式注明学号，名字，并将需要回答的问题通过注释的给出。如果本次作业有与同学、助教、或老师讨论，需要在注释中注明讨论对象，这不影响成绩。
11. 注意：代码作业鼓励讨论，但禁止抄袭，如代码雷同，则该次作业都按照 0 分计算！