

凸优化 homework5 说明

2020 年 11 月 26 日

1 作业要求

你的压缩包命名应该为“gl-学号-日期.zip”，解压你提交的压缩包之后，目录树应当如下：

```
gl-学号-日期/  
├── code/  
│   ├── README.txt  
│   └── your source code(*.m, *.py etc.)  
└── doc/  
    ├── report.pdf  
    └── else(*.tex, *.bib, *.docx etc.)
```

程序语言 请使用 matlab、python 的某一种完成上机作业。matlab 建议使用 2018a 或之后的版本。

上机报告 gl-学号-日期/doc/report.pdf 作为你的上机报告。

README 需要说明的 .

- 后继作业文件和上机报告可以包含之前提交的内容。
- 指明你程序中所有用到的 matlab/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。

2 上机作业批改及评分

程序 60% (由程序性能和程序格式两部分组成).

测试算例:

```
% generate data
seed = 97006855;
ss = RandStream('mt19937ar', 'Seed', seed);
RandStream.setGlobalStream(ss);
n = 512;
m = 256;
A = randn(m,n);
k = round(n*0.1); l = 2;
A = randn(m,n);
p = randperm(n); p = p(1:k);
u = zeros(n,l); u(p,:) = randn(k,l);
b = A*u;
mu = 1e-2;
```

批改方式(批处理在助教机器上运行, 可参照 [Test_group_lasso.m](#)):

```
% function Test_group_lasso
% min 0.5 * ||A * x - b||_2^2 + mu * ||x||_{1,2}
% generate data
seed = 97006855;
ss = RandStream('mt19937ar', 'Seed', seed);
RandStream.setGlobalStream(ss);
n = 512;
m = 256;
A = randn(m,n);
k = round(n*0.1); l = 2;
A = randn(m,n);
```

```
p = randperm(n); p = p(1:k);
u = zeros(n,1); u(p,:) = randn(k,1);
b = A*u;
mu = 1e-2;

x0 = randn(n, 1);

errfun = @(x1, x2) norm(x1 - x2, 'fro') / (1 + norm(x1, 'fro'));
errfun_exact = @(x) norm(x - u, 'fro') / (1 + norm(u, 'fro'));
sparsity = @(x) sum(abs(x(:)) > 1E-6 * max(abs(x(:)))) / (n*1);
opts_mosek = []; %modify options

tic;
[x_mosek, out_mosek] = gl_cvx_mosek(x0, A, b, mu, opts_mosek);
t_mosek = toc;

%calling your solver
opts = [];
%如果opts非空，请在函数内部写默认值，批改的时候传入空集
tic;
[x, iter, out] = your_solver(x0, A, b, mu, opts);
% 只需要提供函数接口。
t = toc;

% 指标包括：
% errfun(x_mosek,x), errfun_exact(x),
% sparsity(x), iter 以及时间 t
```

程序性能 (50%): 得分由以下部分决定:

- 算法解的效果 (比如稀疏度, 恢复效果).
- 相比于 CVX mosek/gurobi 的数值表现情况, 其他不同算法之间的数值效果比较.
- 报告与程序运行结果的一致性.
- 算法在其他种子下的数值表现情况 (请不要根据给定的种子调参).

注: 报告中的绝对时间会作为主要参考. 在运行你的程序时会关心算法解的性质、相对效率以及算法收敛的迭代次数. 为了保证你的结果可以复现, 随机种子请指定.

程序格式 (10%): 算法接口格式:

- `[x, iter, out] = your_method_name(x0, A, b, mu, opts)`, 输入(按顺序) 提供给定的初始解 x_0 , A , b , μ 是给定的数据, $opts$ 是结构体存储了你想提供的其他参数 (如果有, 请在函数内部写默认参数, 批改时是默认传入空集, 并在上机报告及 README.txt 中注明); 前两个输出(按顺序) 为 x , $iter$, x 为你的算法求解出的解, $iter$ 为输出为 x 时所对应的算法迭代次数, 如果迭代次数没法获得, 请传-1, 并在报告中注明; out 是结构体为其他你想输出的指标 (请在上机报告及 README.txt 中注明).
- python 用户的输入和输出格式要求同 matlab, 请指明随机数的种子.
- **统一**的程序命名如下 [题号-函数名] ('_' 为一个下划线, 注意方法名的大小写):
 1. `gl_cvx_mosek.* / gl_cvx_gurobi.*`
 2. `gl_mosek.* / gl_gurobi.*`
 - 3.a. `gl_SGD_primal.m / gl_SGD_primal.py`
 - 3.b. `gl_GD_primal.m / gl_GD_primal.py`
 - 3.c. `gl_FGD_primal.m / gl_FGD_primal.py`
 - 3.d. `gl_ProxGD_primal.m / gl_ProxGD_primal.py`
 - 3.e. `gl_FProxGD_primal.m / gl_FProxGD_primal.py`
 - 3.f. `gl_ALM_dual.m / gl_ALM_dual.py`
 - 3.g. `gl_ADMM_dual.m / gl_ADMM_dual.py`
 - 3.h. `gl_ADMM_primal.m / gl_ADMM_primal.py`

3.i. gl_PPA_dual.m / gl_PPA_dual.py

3.j. gl_BCD_primal.m / gl_BCD_primal.py

实验报告 40% 基本知识 (模型与算法的细节), 结果丰富性创新性, 排版.

- 你的结果请在报告中至少用表格及图片方式呈现.

表格可类似于表1 (仅供参考).

表 1: Compare the efficiency of different solvers.

Solver	Fval	Errfun	Errfun_Exact	Time	Iter	Sparisity
your_solver1	xxx	xx	xx	xx	xx	xx
your_solver2	xxx	xx	xx	xx	xx	xx

Fval 代表着目标函数值. 鼓励呈现其他未列出的统计指标.

- 报告排版请尽可能的整齐美观.