# 大数据中的算法project1数值报告

1.问题描述

min norm(x,1)

s.t. Ax=b

2.解决方法

首先是直接调用cvx，采用solver是gurobi来解决。

然后直接调用gurobi来解决，由于gurobi要采用标准格式输入，故将x改写为x=x+-x-，令x'=[ x+；x-]，A'=[A，-A]，原问题改写为

min



s.t. A'x'=b

x'≥0

即为标准线性规划问题，可以直接调用gurobi求解。

然后自己实现FISTA方法求解，求解方法为



其中L为利普希茨条件的常数，这里取的是ATA的最大特征值。需要说明的是，这里的bregman方法实际求解的是

min norm(x,1)+norm(Ax-b,2)

为了和原问题等价应当求解

min norm(x,1)+t\*norm(Ax-b,2)

前者相当于后者t=1的一个情况。但是根据数值计算的结果，这个解和调用cvx的解相差不多，因此也就默认为t=1是一个合理的近似解。

最后采用ADMM方法求解。首先求出原问题的dual问题

min bTλ

s.t. norm（ATλ，inf）≤1

其增广拉格朗日函数为

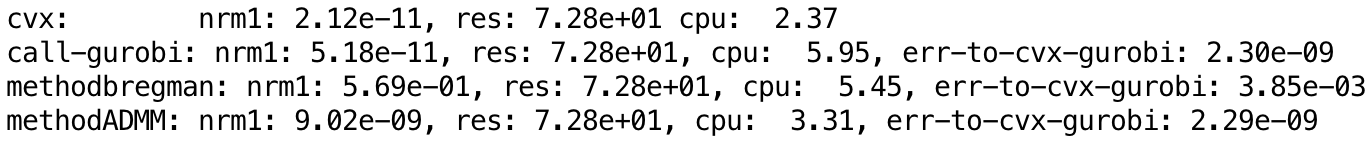


可以得出





迭代解出x。

3.数值结果

可以看出cvx和gurobi的结果是一致的，cvx的运行速度更快，不过可能是由于调用gurobi时要用循环生成cell数组比较慢造成的。bregman方法的结果有一定的误差，ADMM的方法误差很大。考虑到bregman方法相当于只取t=1，因此得到的结果也可以接受；ADMM方法计算速度很快，也非常容易达到比较高的精度，但是需要选择比较合适的μ。

二

1.问题

max logdet(X)-Tr(SX)-ρnorm(X,1) X≥0

2.原问题的dual问题为

min logdet(Y)

s.t. norm(Y-S,inf)≤ρ

3.解法

直接调用cvx进行求解

我实现了如下几个一阶方法，但是效果都不好，3-2题的实现效果也比较一般，可能对于半定规划的方法还不太了解吧。

首先是仿照FISTA方法，将原问题改写为



这里取的是F范数而不是2范数。但是困难的地方在于利普希茨条件的常数L的选取，需要讨论的利普希茨条件。我取的近似解为：假设X的最小特征值为α，则L=1/α2（假设X、Y都是对角阵或者有相同特征向量，就可以得到上述结果），然后按照FISTA的方法



进行求解。但是这个结果并不收敛到最优解，而是收敛到形如X=kI形式的解，k=n/(ρn+Tr(S))。

之后尝试采用了线性ADM的方法。求解



但是这个方法关于μ的性质很差，μ很大时不收敛，μ很小时结果偏差比



较多（这里为了方便比较的是Optimal Value的值）。

这里没有考虑dual问题的ADMM，原因在于dual问题的ADMM需要求解



本身也不好求解。

最后采用了最简单的方法



但是这个算法的结果也不好，最终结果不收敛。

因此综合来看，类FISTA还算是取得了比较好的结果。

对于S取第一种模型，ρ=10， optimal value=-37.47

ρ=0.1， optimal value=-17.45

ρ=0.001， optimal value=-17.06

对于S取第二种模型，ρ=10， optimal value=-38.63

ρ=0.1， optimal value=-25.21

ρ=0.001， optimal value=-24.67

问题3—2

问题：

min norm（X，1）

s.t. norm（SX-I，F）≤σ

X≥0

其dual问题为：

令r=SX-I，拉格朗日函数为



可得dual问题为

Tr（Z）-σλ

s.t. norm（ZS，inf）≤1

norm（Z，F）≤λ

解决方法：直接调用cvx求解

直接调用sdpt3求解

用一阶方法求解

一阶方法中同样有仿照bregman的FISTA方法求解的，但是结果不收敛（对于μ取0.0001到10000），此外还有采用ADMM方法。求解



其中



其中比较麻烦的是第一步的求解,我的做法是先求梯度解出Z，如果Z满足F范数的条件，即可。如果Z不满足，考虑F范数条件取等号，然后用拉格朗日乘子法，对乘子从比较小到比较大的范围内进行搜索，直到找到满足条件的Z。

这个解的结果和μ的取值关系很大，而且第一步中很难得到Z的精确解。但是选取比较合适的μ，可以达到和cvx的结果很接近的程度。因此可以看出ADMM不仅适用于LP问题，也可以用于此类问题。

对于S取第一种模型，

σ=10， optimal value=9.38\*10^（-9）

σ=0.1， optimal value=3.91

σ=0.001， optimal value=4.00

利用ADMM方法达到的结果为（dual问题的结果）

σ=10， μ=10000， optimal value=-0.0041（循环10000次，要等很长时间。。。）

σ=0.1， optimal value=1.85

σ=0.001， optimal value=5.49

对于后两个问题，μ的取值都是10，如果μ取值较大，运算时间显著增长，且optimal value趋于0，理论上也确实如此，μ比较大时相当于约束很松，解就是0，μ比较小时约束变严格（权重增大），0就不是解了。我认为可以取更合适的μ，达到和cvx同样的结果，不过没什么意义，只不过是和cvx算法取相同的权重罢了。

对于S取第二种模型，

σ=10， optimal value=9.14\*10^(-9)

σ=0.1， optimal value=21.63

σ=0.001， optimal value=26.00

利用ADMM算法达到的结果为

μ=1000 σ=10， optimal value=-0.016

μ=0.1 σ=0.1， optimal value=19.17

μ=0.1 σ=0.001， optimal value=20.16

# 程序说明

cvx\_gurobi是用cvx调用gurobi来求解问题1；direct\_gurobi是直接调用gurobi来求解；bregman，shrinkage是自己编程用bregman方法来求解；ADMM和ADMMS是用ADMM方法来求解，test是测试。

cvxsolve3\_1和cvxsolve3\_2是利用cvx来求解3\_1和3\_2。其余分别是用不同的方法求解3\_1和3\_2