CS268 Optimization Homework2

Sihong HE, #ID: 89483859

I use my own optimizer from HW1 on HW2. There are 9 functions on my HW2:

1. optimizer1D() which from my HW1 aim to do linear search.
2. conjugate_gradients() which is base on CG algorithm.
3. func1 $(1-\mathrm{x}_0)^2 + 100(x_0 - x_1^2)^2$ and its gradient function funcg1.
4. func2 $(\mathrm{x}_0 - 1)^2 + (\mathrm{x}_1 - 2x_0)^2 + (x_2 - x_1)^2 + (\mathrm{x}_3 - \mathrm{x}_2)^2 + (2x_4 - x_3)^2 + (x_5 - x_4)^2 + (x_6 - x_6)^2 + (x_7 - 2x_6)^2 + (x_8 - x_7)^2 + (2x_9 - x_8)^2$ and its gradient funcg2. func1 funcg1 func2 funcg2 are used to test conjugate_gradients().
5. alternate_method() which base on steepest descent algorithm.
6. eval() and eval_alt() to print the average error of the output argument vector, function value produced by the algorithm, total number of CG iterations along with statistical error bars on those quantities

My start points are sampled from uniform distribution U (-1,1). The true mathematical optimums for two test functions are [1,1] and [1,2,2,2,1,1,1,2,2,1] respectively. For the stopping criterion, I choose x tolerance = 1e−6 and function tolerance = 1e−8, $\epsilon_G = 1e - 9, \epsilon_A = 1e - 9, \epsilon_R = 1e - 9$. The machine precision is 2.22e−16. I set a seed(123).

Problem 1

Function	Average error	Function value	# of CG iterations
Func1	3.54e-3+-4.56e-3	6.73e-6+-1.96e-5	33.1+-15.13
Func2	1.22+-1.33	0.106+-0.136	45.55+-27.68

Problem 2

Function	Average error	Function value	# of iterations
Func1	2.22e-2+-2.659e-2	2.47e-4+-5.31e-4	639.45+-528.54
Func2	1.90+-2.30	0.113+-0.142	84.4+-62.03

From above tables we can know that:

Conjugate gradient method converges much faster than steepest descent method. Especially in low-dimension optimization problems. And the error terms are smaller when we use conjugate gradient method comparing to steepest descent method.

For conjugate gradient method, the number of CG iterations becomes larger when the dimension of objective function improves. In contrast, for steepest descent method, the number of iterations becomes smaller when dimension becomes larger.

The function value we get by using conjugate gradient method is closer to true value.