Computational Physics III:
Steepest descent, conjugate gradients and SVD

Due on June 08, 2023

Salomon Guinchard

Contents

Problem 1

Steepest descent and conjugate gradient

Say one has the following linear system of equations to solve:

$$\mathbf{A}\mathbf{x} = \mathbf{b}, \tag{1}$$

where $\mathbf{A}$ is square, symmetric and positive definite, and $\mathbf{x}$, $\mathbf{b}$ are column vectors. Then two algorithms can be used to solve the linear system from Eq.(1), namely the steepest descent (SD) and the conjugate gradient (CG) methods. Both methods exploit the fact that solving Eq.(1) for $\mathbf{x}$ is equivalent to extremizing the following quadratic form $\mathbf{f}$:

$$\mathbf{f}(\mathbf{x}) = \mathbf{x}^T\mathbf{A}\mathbf{x} - \mathbf{b}^T\mathbf{x} + \mathbf{c}, \tag{2}$$

for an arbitrary constant $\mathbf{c}$ since the gradient of $\mathbf{f}$ that we shall denote by $\mathbf{f}'$ recovers Eq.(1):

$$\mathbf{f}'(\mathbf{x}) = \mathbf{A}\mathbf{x} - \mathbf{b}. \tag{3}$$

Thus, one notices that finding the values of $\mathbf{x}$ that extremize $\mathbf{f}$ is equivalent to finding the zeros of the gradient and solving Eq.(1).

(1) Algorithms

The **SD** method starts from an arbitrary $\mathbf{x}^{(0)}$ such that each step of the algorithm gives a point $\mathbf{x}^{(i)}$ closer to the solution $\mathbf{x}$. At each step, the direction of the steepest descent of the quadratic form from Eq.(2) is found and the minimum along this direction is determined, leading to the point $\mathbf{x}^{(i+1)}$. Hence the resulting sequence $\{\mathbf{x}^{(i)}\}$ is decreasing and the convergence rate of the error to the solution $\mathbf{e}^{(i)} := \mathbf{x}^{(i)} - \mathbf{x}$ follows:

$$\frac{|\mathbf{e}^{(i)}|_{\mathbf{A}}}{|\mathbf{e}^{(0)}|_{\mathbf{A}}} \leq \left(\frac{\kappa - 1}{\kappa + 1}\right)^i, \tag{4}$$

where $|\cdot|_{\mathbf{A}} := (\mathbf{x}^T\mathbf{A}\mathbf{x})^{1/2}$ and κ is the so called condition number of $\mathbf{A}$, that is the ratio of the largest singular value of $\mathbf{A}$ to the lowest.

The **CG** method shares basically the same base, the only difference is that the vectors from each step are no longer orthogonal to each other, but $\mathbf{A}$-orthogonal, that is $\mathbf{x} \cdot \mathbf{y} = \mathbf{x}^T\mathbf{A}\mathbf{y}$. This condition turns out to be very efficient, in the sense that if one were to consider an ideal case without numerical imprecision, the algorithme should converge in n steps, with n the size of the matrix. The relative error follows Eq.(5).

$$\frac{|\mathbf{e}^{(i)}|_{\mathbf{A}}}{|\mathbf{e}^{(0)}|_{\mathbf{A}}} \leq \left(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)^{i} \tag{5}$$

The two algorithms have been implemented in the following scripts and the tests were passed successfully.

Listing 1: Matlab script for the steepest descent method

```
function [L,U] = lu_decomposition_nopiv(A)
sz=size(A);
L=eye(sz(1));
U=A;
    if sz(1)~=sz(2)
        error('This is not a square matrix.');
    else
        for i=1:sz(1)
            for j=i:sz(1)-1
                L(j+1,i) = U(j+1,i)/U(i,i);
                U(j+1,:) = U(j+1,:)-L(j+1,i)*U(i,:);
            end
        end
    end
end
```

Listing 2: Matlab script for the conjugate gradient method

```
function [L,U] = lu_decomposition_nopiv(A)
sz=size(A);
L=eye(sz(1));
U=A;
    if sz(1)~=sz(2)
        error('This is not a square matrix.');
    else
        for i=1:sz(1)
            for j=i:sz(1)-1
                L(j+1,i) = U(j+1,i)/U(i,i);
                U(j+1,:) = U(j+1,:)-L(j+1,i)*U(i,:);
            end
        end
    end
end
```

(2) System solving

(3) Convergence and machine precision tolerance

Problem 2

Discretization of Poisson Equation

(1) Discretization

(2) Numerical solution

(3) Electric potential

(4) Jacobi preconditioning

Problem 3

Nonlinear conjugate gradient method

(1) Algorithm

(2) $N = 4$

(3) $N = 5$

(4) $N = 6$

Problem 4

SVD: Overdefined system of linear equations

(1)

Problem 5

SVD: Quantum state tomography

The density matrix formalism

(1)

Quantum State Tomography: an example

Problem 6

2D quantum well